

طراحی صفحات وب



استادارامنا: مهندس رحیم رشیدی¹

تالیف: ایوب رشیدی²

¹عضو هیات علمی دانشگاه آزاد

²دانشجوی کارشناسی مهندسی نرم افزار

آموزش نکته به نکته

طراحی صفحات وب



ویژه دانشجویان کاردانی و کارشناسی

تالیف: ایوب رشیدی

انتشارات دانشگاه آزاد واحد بوکان

بسم الله الرحمن الرحيم

مقدمه مولف:

طراحی صفحات و سایتهای اینترنتی و فواید این سایتهها

یک سایت می تواند نقش یک بروشور تبلیغاتی را برای شما بازی کند حتی با امکاناتی بسیار بیشتر از یک بروشور ساده. شما با استفاده از یک سایت اینترنتی می توانید به راحتی محصولات خود را به بازدید کنندگان خود معرفی کنید، آنها را از آخرین تغییرات مطلع کنید، اطلاعات مورد نیاز آنها را در اختیارشان قرار دهید و یا حتی نظرات و پیشنهادات آنها را جمع آوری کنید. حتی می توان با ایجاد یک انجمن گفتگو در سایت به سایرین اجازه داد نظرات خود را اعلام کنند و به نظرات دیگران پاسخ دهند.

از یک سایت اینترنتی می توان برای ایجاد ارتباط مؤثر میان اعضای یک سازمان هم استفاده کرد به صورتی که هر یک از اعضا در هر جا که باشند بتوانند به اطلاعات مورد نیاز خود از طریق سایت سازمان دسترسی پیدا کنند، از آخرین اخبار مطلع شوند و آخرین تغییرات را مشاهده کنند. در حقیقت می توان از یک سایت اینترنتی به عنوان بخش مهمی از سیستم اطلاعاتی داخل سازمان استفاده کرد و هر گروه از افراد به اطلاعات مربوط به سطح سازمانی خود دسترسی داشته باشند. یعنی اطلاعاتی که کارمندان از سایت دریافت می کنند متفاوت با اطلاعات مدیران و همچنین متفاوت با اطلاعاتی باشد که بازدید کنندگان خارج سازمان از سایت دریافت می کنند.

یک وب سایت راهی مناسب برای ارائه اطلاعات مختلف در مورد کالا هم و خدماتی است که شرکت به مشتریان ارائه می دهد. بدون اینکه مشتری مجبور باشد در زمان خاصی به شرکت مراجعه کند و یا اینکه با شرکت تماس بگیرد می تواند در هر کجا که باشد به سایت شرکت مراجعه کند و اطلاعات مورد نظر خود را کسب کند.

طراحی سایتهای اینترنتی تا حد زیادی بستگی به موارد استفاده آن سایت و نیازهای کاربران آن دارد. اگر در طراحی سایت نیاز کاربران در نظر گرفته نشود و یا نیازها به صورت روشن مشخص نشوند نباید انتظار داشت تا سایت کارایی لازم را داشته باشد.

با استفاده از زبانهای برنامه نویسی طرف سرور نظیر PHP و با استفاده از پایگاههای داده ها و بانکهای اطلاعاتی نظیر MySQL و تکنیکهایی مثل Ajax و JavaScript امکان طراحی سایتهای با انواع کاربری مختلف وجود دارد. اما در طراحی سایت ها باید از این امکانات در جای مناسب استفاده شود تا نیازهای کاربران نهایی برآورده شود.

در صورتی که تمایل دارید صاحب یک سایت اینترنتی باشید می توانید با ما تماس بگیرید و هدف سایت و نیازهای کاربران سایت را با ما در میان بگذارید. ما پس از بررسی سایت مورد نظر شما در امکان طراحی سایت شما را بر عهده می گیریم. و سعی می کنیم سایتی مطابق میل شما ایجاد کنیم.

برای ارسال سفارش طراحی سایت یا هر گونه پروژه برنامه نویسی می توانید از صفحه « PC- boukan.blogfa.com » ما استفاده کنید یا با ایمیل Ayoob.rashidi@gmail.com تماس بگیرید. و

کلیه انتقادات و پیشنهادات خود را میتوانید مستقیماً با مولف (09387208853) در میان بگذارید.

فهرست مطالب

صفحه

عنوان

فصل اول

| | |
|-------|----------------------------|
| | آموزش HTML |
| | اصول نوشتن صفحات HTML |
| | کار با متون و کلمات |
| | سر فصلها و پاراگرافها |
| | فونتها و رنگ و اندازه آنها |
| | کاراکترهای خاص HTML |
| | پیوند به دیگر صفحات |
| | افزودن عکس به صفحه |
| | کار با متنها قسمت دوم |
| | کامنت گذاشتن در HTML |
| | افزودن لیست HTML به صفحه |
| | شناسه های تگ body |
| | رنگ متن ها در صفحه |
| | رنگ لینکها در صفحه |
| | انتخاب رنگ زمینه |
| | انتخاب تصویر زمینه |
| | کار با عکسها در صفحه |

..... ایجاد لینک با عکسها

..... تعیین جای تصاویر در صفحه

..... متن جایگزین عکسها

..... لینکهای داخلی در صفحه

..... لینک پست الکترونیک 1

..... لینک پست الکترونیک 2

..... جدول رنگهای هگزادسیمال

..... فرمها: نحوه ایجاد فرمها

..... فرمها: کادرهای متنی

..... فرمها: چک باکس

..... فرمها: دکمه رادیو

..... فرمها: انواع دکمهها

..... کلمه عبور- فیلدهای مخفی

..... فرمها: Text Area

..... فرمها: لیستهای کشوی

..... استفاده از موزیک در صفحه

..... استفاده از متاتگها

..... تغییر مسیر بازدیدکنندگان

..... خط افقی و شناسههای آن

..... تنظیم فونت متن

..... باز شدن لینک در پنجره جدید

..... ایجاد متن متحرک

| | |
|-------|--------------------------------|
| | اصول استفاده از جدولها |
| | کار با جدولها(رنگ زمینه و ...) |
| | ایجاد جدولهای تودرتو |
| | اصول ساختن فریمها |
| | باز کردن لینکها در فریم |
| | فریمهای غیر قابل رویت |
| | دریچهای به دیگر صفحات |
| | رویدادهای پنجره و فرمها |
| | رویدادهای صفحه کلید و ماوس |
| | تنظیم Encoding صفحه |

فصل دوم

| | |
|-------|----------------------------|
| | مقدمه آموزش CSS |
| | اصول نوشتن کدهای CSS |
| | استفاده از شناسه style |
| | CSS در Head صفحه |
| | فایل CSS خارج از صفحه |
| | تنظیم طول و عرض عناصر |
| | ایجاد کادر برای عناصر |
| | حاشیه و تورفتگی عناصر صفحه |
| | کار با متنها 1 |
| | کار با متنها 2 |
| | تنظیم فونت با CSS |

تنظیم رنگ متن با CSS.....

تغییر رنگ و تصویر زمینه.....

تنظیم تکرار تصویر زمینه.....

تنظیم موقعیت تصویر زمینه.....

تغییر شکل نشانگر ماوس.....

تنظیم ظاهر لینکها در صفحه.....

فصل سوم

مقدمه آموزش Java Script.....

اصول نوشتن جاوا اسکریپت.....

عملگرها در جاوا اسکریپت.....

نمایش کادرهای ALERT.....

نوشتن متن با جاوا اسکریپت.....

دستور شرطی if...else.....

دستور شرطی switch.....

رویداد onMouseover.....

دکمه ها و موارد استفاده آنها.....

کار با تاریخچه مرورگر.....

کادر های prompt.....

خاصیت float در جاوا اسکریپت.....

فصل چهارم

آزمایش کدهای HTML.....

پاراگرافها و شکست خط.....

نوشتن انواع سرفصلها

متن ضخیم و مورب و زیر خط دار

فونتها و رنگ و اندازه و قلم

نوشتن علامتهای اختصاری

لینک به دیگر صفحات

استفاده از تصاویر در صفحات

توضیحات اضافی در HTML

لیستهای نامرتب

لینک برای ادرس ایمیل

فرم های HTML

تعبیه موزیک در صفحه

ویژگی های خط افقی

نوشتن متن متحرک

بخش اول جدولها

بخش دوم جدولها

بخش سوم جدولها

رنگ زمینه در سلولها

رنگ زمینه در سطرها

ساخت پرچم ایران با HTML

پروژه فاکتوریل N!

مقدمه آموزش HTML

آنچه که قبل از یادگیری HTML لازم است بدانید:

برای شروع کار با HTML شما به چیزهایی نیاز دارید که در لیست زیر آمده است:

شما به یک ویرایشگر متن نیاز دارید. در ویندوز شما می توانید از برنامه Notepad استفاده

کنید. از هر نرم افزار ویرایش متن دیگری استفاده کنید اما این نکته را در نظر داشته باشید که

نرم افزار WordPad که در ویندوز وجود دارد و نرم افزارهایی مانند Microsoft Office

Word و امثال آن برای نوشتن کدهای HTML مناسب نیستند چون در هنگام ذخیره کردن

متن این نرم افزارها کدهایی را به آن اضافه می کنند.

نرم افزار Notepad را می توانید از طریق منوی استارت در قسمت Accessories پیدا کنید یا

از طریق فرمان Run با تایپ notepad در کادر Open و فشار دادن دکمه Enter.

شما به یک مرورگر وب به انتخاب خود نیاز دارید. می توانید از Internet Microsoft

Explorer استفاده کنید که با ویندوز نصب می شود یا از Firefox Opera, Mozilla و نرم

افزارهای مشابه که اغلب به صورت رایگان برای دانلود در اینترنت وجود دارند.

بهتر است صفحاتی را که می نویسید با چند مرورگر آزمایش کنید. زیرا در HTML قسمتهایی

وجود دارد که مختص به یک مرورگر خاص است و با سایر مرورگرها کار نمی کند.

شما به جایی برای ذخیره کردن متن خود دارید که می تواند جایی روی هارد دیسک شما، یک

فلپی دیسک یا یک سرور باشد. البته شما برای استفاده از HTML نیاز ندارید حتماً به اینترنت

متصل باشید، شما می توانید با استفاده از یک مرورگر وب در کامپیوتر خود از HTML استفاده کنید.

برای ذخیره کردن فایلی که با Notepad نوشته اید کافی است از منوی File گزینه Save As را انتخاب کرده و مسیری را برای ذخیره فایل انتخاب کنید و در کادر File Name در پایین پنجره Save As نام فایل را با پسوند html بنویسید. در آخر هم دکمه Save را کلیک کنید.

اصول نوشتن صفحات HTML

در اینجا می خواهیم به ساختار شناسه ها یا همان تگها (tag) در HTML بپردازیم. یک تگ همیشه با علامت کوچکتر ریاضی یعنی این علامت < آغاز می شود و با علامت بزرگتر > به پایان می رسد. مثلاً تگی که باعث ایجاد زیرخط کلمات می شود اینطور نوشته می شود: <U> این علامت را قبل از متنی که می خواهیم زیر آن خط کشیده شود قرار می دهیم. به این تگ ، تگ ابتدایی (Opening tag) می گویند، که عملیاتی را که شما می خواهید آغاز کنید شروع می کند. به منظور پایان دادن به زیرخط باید از یک تگ پایانی استفاده کنید. یک تگ پایانی با این علامت شروع می شود: / < بقیه تگ مانند تگ ابتدایی می باشد. برای درک بهتر این مطلب به مثال زیر توجه کنید:

Tags and their position

اگر بخواهید متن بالا را با زیرخط در مرورگر خود ببینید باید آنرا مطابق خط روبرو در ویرایشگر متن خود بنویسید:

<u>tags and their position</u>

نتیجه را به این صورت می بینید:

Tags and their position

در HTML همه تگها به یک تگ پایانی نیاز ندارند. برای مثال تگی که برای وارد کردن عکس استفاده می شود از این قبیل تگهاست، که به این صورت می باشد:

``

البته بعداً در مورد این تگ توضیح خواهیم داد در اینجا فقط برای مثال این تگ را آوردیم. موارد دیگری هم با این وضعیت وجود دارند مثلاً `
` که برای رفتن به خط بعدی استفاده می شود، `<hr>` که برای ایجاد یک خط افقی به کار می رود و `<p>` که برای رفتن به پاراگراف بعدی استفاده می شود.

در ضمن نیاز نیست که حتماً تگها با حروف بزرگ نوشته شوند. مثلاً `<P>` همان `<p>` است. البته در نسخه های جدیدتر HTML توصیه می شود کدهای HTML با حروف کوچک نوشته در XHTML هم نوشتن کدها با حروف کوچک الزامی است. فضای خالی هم بین تگها تأثیری ندارد. مثلاً:

`<u>` متن زیرخط دار `</u>`

کد بالا همان نتیجه ای را می دهد که کد زیر دارد:

<U>متن زیرخط دار</U>

به این طریق هم اگر نوشته شود فرقی نمی کند:

<u>

متن زیرخط دار

</u>

قالب پایه ای یک فایل HTML مانند زیر است. ببینید می توانید کاری را که تگهای این کد انجام می دهند حدس بزنید؟ البته همه آنها را بعداً توضیح می دهیم.

<Html>

<head>

<title>html اصلی قالب</title>

</head>

<body>

گیرد در صفحه شما قابل دیدن است هر چیزی که اینجا قرار

</body>

</html>

برای درک بهتر این قسمت سعی کنید تگها را به صورت جفتی در نظر بگیرید (از جایی که باز می شوند تا جایی که بسته می شوند) اولین تگ `<html>` است که علامت شروع یک متن HTML است و تگ پایانی آن `</html>` است که همانطور که ممکن است حدس زده باشید نشانه پایان متن HTML است. هر چیزی در HTML باید بین این دو تگ قرار گیرد. مانند متن، دیگر تگها، عکسها و... چون این دو تگ نشان دهنده آغاز و پایان صفحه شما هستند.

تگ بعدی `<head>` است. این تگ مشخص کننده بخشی است که شما باید عنوان صفحه، کلمات کلیدی، و دیگر توضیحات خود برای این صفحه را در آن قرار دهید. این بخش با `</head>` به پایان می رسد. در مثال بالا تنها قسمت HEAD عنوان صفحه بود که با TITLE مشخص می شود.

تگ `<title>` به شما اجازه می دهد برای صفحه خود یک عنوان برگزینید. این قسمت مورد استفاده موتورهای جستجو قرار می گیرد و همان متنی است که به هنگام مشاهده صفحه در نوار عنوان مرورگر نشان داده می شود. تا هنگامی که این تگ در قسمت BODY قرار نگیرد در متن شما نشان داده نمی شود. برای پایان دادن به تگ عنوان از این تگ استفاده می شود: `</title>` در مثال بالا عنوانی که برای صفحه نشان داده می شود قالب اصلی HTML است.

تگ `<body>` بخشی را آغاز می کند که در صفحه مرورگر شما به نمایش در می آید. این قسمت جایی است که ما باید کار اصلی خود را در آن انجام دهیم. برای پایان دادن به این تگ از `</body>` استفاده می کنیم. در مثال بالا متنی که در صفحه مرورگر دیده می شود به صورت

زیر است (البته در اینجا برای نوشته زیر رنگ و فونت نیز انتخاب شده که در قسمتهای بعد به آنها خواهیم پرداخت) :

هر چیزی که اینجا قرار گیرد در صفحه شما قابل دیدن است

چیزی که در اینجا می بینیم متنی است که به صورت ساده نوشته شده و در طرف چپ صفحه قرار دارد.

در درس بعد در مورد چگونگی کار کردن با متنها بحث می کنیم.

کار با متون و کلمات

نوشتن متن ها به صورت ضخیم، مورب، زیر خط دار و خط خورده

در این درس به کار با متنها می پردازیم. ابتدا با این تگ شروع می کنیم:

1-**** این تگ باعث می شود کلماتی که بین این تگ و تگ پایانی آن قرار دارند به صورت پر

رنگ و بولد (Bold) دیده شوند. به مثال زیر توجه کنید:

****متنی که اینجا قرار بگیرد به صورت بولد نمایش داده می شود****

مثال بالا به صورت زیر دیده می شود:

متنی که اینجا قرار بگیرد به صورت بولد نمایش داده می شود

بیایید با تگهای دیگری ادامه دهیم:

2- زیر خط (Underline):

</u> این متن با زیر خط مشاهده می شود <u>

این هم نتیجه:

این متن با زیر خط مشاهده می شود

3- حروف کج (*Italic*):

</i> این متن به صورت مورب دیده می شود <i>

و نتیجه به این صورت است:

این متن به صورت مورب دیده می شود

4- ~~حروف خط خورده~~:

</strike> این متن به صورت خط خورده دیده می شود <strike>

متن در مرورگر به این صورت دیده می شود:

~~این متن به صورت خط خورده دیده می شود~~

5- وسط چین:

</center> شود این متن در وسط خط نوشته می <Center>

نتیجه کد:

این متن در وسط خط نوشته می شود

شما می توانید بیشتر از یک فرمان را برای یک متن اجرا کنید. مثلاً وقتی که می خواهید متنی به صورت پر رنگ و کج نوشته شود. برای این کار هر دو تگ ابتدایی را برای متن در ابتدای آن و تگهای پایانی را در جایی که می خواهیم پایان پر رنگی و کجی متن باشد قرار می دهیم. با این مثال توجه کنید:

</i> شود این متن به صورت کج و پر رنگ دیده می <i>

نتیجه هم به این صورت دیده می شود:

این متن به صورت کج و پر رنگ دیده می شود

آیا ترتیب قرار گرفتن تگهای پایانی مهم است؟ در پاسخ باید گفت در مثال بالا خیر، اما مواردی وجود دارد که ترتیب قرار گرفتن تگها مهم است مثلاً قرار گرفتن تگ پایانی </HTML> قبل از </BODY> اشتباه است. شما با تمرین و تجربه می توانید این نکته را فرا بگیرید. شما می توانید برای جلوگیری از اشتباه HTML را به صورت زیر بنویسید:

</i>

این متن به صورت پر رنگ و کج دیده می شود

</i>

اما این روش خسته کننده است. در اینجا مهم نیست تگها به چه صورت باز و بسته شوند. اما نکته مهمی که باید در نظر داشته باشید این است که متنها تحت تأثیر تگهایی که قبل از آنها باز شدهاند و بسته نشده اند قرار می گیرند. تأثیر یک تگ تمام می شود وقتی که تگ بسته شود. مثال زیر را در نظر داشته باشید:

<i><u> این متن پر رنگ، کج و دارای زیرخط است </u></i>

این نتیجه را به ما می دهد:

این متن پر رنگ، کج و دارای زیرخط است

اما این یکی:

<u><i> این متن پر رنگ، کج و </i></u>

این نتیجه را به ما می دهد:

این متن پر رنگ، کج و دارای زیرخط است

همان طور که دیدید متن **بولد** و *ایتالیک* زودتر از متن زیرخط دار به پایان رسید و این به این دلیل است که تگهای پایانی **بولد** () و *ایتالیک* (</i>) زودتر نوشته شدند و انتهای متن فقط توسط تگ زیرخط (</U>) تحت تأثیر قرار دارد.

متن‌ها به صورت پیش فرض در سمت چپ صفحه به نمایش در می آیند. اما با استفاده از `<CENTER>` می توان آنها را در وسط صفحه به نمایش در آورد. این تگ را هم می توان با دیگر تگ‌ها ترکیب کرد و نتایج جابی گرفت. به مثال زیر توجه کنید:

`<Center>|></center>` **توجه** `</center>` به این مثال توجه کنید

و این هم نتیجه آن:

به این مثال توجه کنید

شما با تمرین این بخش می توانید آمادگی ورود به درس بعد که در مورد سرفصل، پاراگراف، و خطوط است را به دست آورید.

نوشتن سر فصلها و پاراگراف در HTML

از تگهای سر فصل (HEADING) برای تیترو عنوان مطالب استفاده می شود. به این مثالها توجه کنید:

`<h1>سرفصل بزرگ</h1>`

سرفصل بزرگ

`<h2>سرفصل 2</h2>`

سرفصل 2

`<h3>سرفصل 3</h3>`

سرفصل 3

`<h4>در حال کوچک شدن</h4>`

در حال کوچک شدن

`<h5>کوچکتر</h5>`

کوچکتر

`<h6>بسیار کوچک</h6>`

بسیار کوچک

از دستورات بالا می توانید برای عنوان مطالب استفاده کنید.

وقتی که تگ `
` در متنی وارد می شود بقیه آن خط به خط بعد منتقل می شود. در حقیقت تگ `
` یا همان Line Break کاری مشابه فشردن کلید Enter به همراه Shift را انجام می دهد. قابل ذکر است که این تگ نیازی به تگ پایانی ندارد. به یک مثال توجه کنید:

This is the first line `
` Here is the second line

نتیجه مثال بالا به اینگونه است:

This is the first line

Here is the second line

مرورگر اصولاً تا وقتی که به کمبود فضا برای یک خط بر نخورد به خط بعدی نمی رود. اما تگ

`
` مرورگر را مجبور می کند به خط بعدی برود. مثلاً متن زیر را در ویرایشگر متن خود

بنویسید و به صورت html ذخیره کنید، فقط یک خط را در مرورگر خود خواهید دید:

من،

یک خط

جدید می خواهم

نتیجه به این صورت خواهد بود:

من، یک خط جدید می خواهم

برای اینکه نتیجه ای مشابه متن نوشته شده در ویرایشگر متن داشته باشیم، باید متن را به این

صورت بنویسیم:

من،

یک خط

جدید می خواهم

این هم نتیجه:

من،

یک خط

جدید می خواهم

<p> تگ پاراگراف است. این تگ باعث ایجاد یک فضای خالی عمودی بین دو خط در دو

پاراگراف می شود. درست مثل اینکه شما
 را دو بار تایپ کنید. از این تگ در اول پاراگراف

برای تمایز دو پاراگراف استفاده می شود. به مثال زیر توجه کنید:

خط اول از پاراگراف اول

اول خط دوم از پاراگراف

<p>

این هم پاراگراف جدید

</p>

و نتیجه به این صورت است:

خط اول از پاراگراف اول

خط دوم از پاراگراف اول

این هم پاراگراف جدید

تگ پاراگراف هم نیازی به تگ پایانی ندارد. اما شما می توانید برای فهم بهتر آنرا در متن خود به کار ببرید. مثلاً به این صورت:

<p>

پاراگراف اول

</p>

<p>

دوم پاراگراف

ادامه پاراگراف دوم

</p>

این متن همان نتیجه ای را دارد که متن زیر دارد:

<p> پاراگراف اول </p>

<p>

دوم پاراگراف

ادامه پاراگراف دوم

</p>

این هم نتیجه هر دو متن بالا:

پاراگراف اول

پاراگراف دوم

ادامه پاراگراف دوم

البته اگر می خواهید صفحه ای که طراحی می کنید معتبر باشد بهتر است از تگ پایانی پاراگراف استفاده کنید. استفاده از تگ پایانی در XHTML الزامی است.

تا اینجا شما می توانید یک صفحه وب پر از متن را بنویسید. در درس بعد به اندازه فونت و رنگ متن می پردازیم.

کار کردن با فونتها و رنگ و اندازه آنها

در این قسمت از درس به سراغ کار با فونتها و اندازه آنها می رویم.

اندازه فونت را می توان با دستور زیر تعیین کرد:

کنیم متنی که می خواهیم اندازه آنرا تعیین

به جای «X» عددی با علامت منفی یا مثبت قرار می گیرد. وقتی که می خواهیم یک متن بزرگتر دیده شود از علامت مثبت و زمانی که می خواهیم متن کوچکتر شود از علامت منفی استفاده می کنیم. به مثال زیر توجه کنید:

شود این متن بزرگتر دیده می

این هم نتیجه تغییر سایز فونت:

این متن بزرگتر دیده می شود

برای کوچکتر کردن متن نیز به همین صورت عمل می کنیم:

شود این متن کوچکتر دیده می

این هم متن کوچک شده:

این متن کوچکتر دیده می شود

این هم تعدادی مثال برای درک بهتر مطالب:

است این متن با اندازه 4 نوشته شده

این متن با اندازه 4 نوشته شده است

است این متن با اندازه 3 نوشته شده

این متن با اندازه 3 نوشته شده است

`` این متن با اندازه 2 نوشته شده ``

این متن با اندازه 2 نوشته شده است

`` این متن با اندازه 1 نوشته شده ``

این متن با اندازه 1 نوشته شده است

`` این متن با اندازه 1- نوشته شده ``

این متن با اندازه 1- نوشته شده است

`` این متن با اندازه 2- نوشته شده ``

این متن با اندازه 2- نوشته شده است

`` این متن با اندازه 3- نوشته شده ``

این متن با اندازه 3- نوشته شده است

برای تغییر رنگ متنها نیز می توانیم به روش زیر عمل کنیم، با این فرمان :

``

می توانیم رنگ مورد نظر خود را با نام انگلیسی آن در فرمان بالا قرار دهیم یا از معادل هگزا

دسیمال رنگ به جای اسم رنگ استفاده کنیم.

برای شروع با اسم رنگ شروع می کنیم:

`` متن مورد نظر ``

متن نهایی:

متن مورد نظر

حالا از معادل هگزا دسیمال رنگ قرمز به جای نام رنگ استفاده می کنیم. البته استفاده از اسم رنگها ساده تر است اما شما ممکن است بخواهید از رنگهای پیچیده تری استفاده کنید. در مورد رنگ قرمز باید به این صورت عمل کنیم:

`متن مورد نظر`

در اینجا نتیجه با مثال قبلی یکسان است:

متن مورد نظر

معادل هگزا دسیمال رنگها را باید با علامت # شروع کرد و بعد از این علامت شش حرف یا عدد دیگر قرار می گیرد. برای رنگ قرمز بعد از علامت # دو F و چهار صفر قرار می گیرد. اگر می خواهید معادل هگزادسیمال رنگها را به دست آورید.

فرض کنید می خواهید رنگ و اندازه متنی را همراه با هم تغییر دهید. برای این کار می توانید از دو تگ `` استفاده کنید. اما به خاطر داشته باشید که هر دو را ببندید. مانند زیر:

`متن بزرگ و طلایی`

متن بزرگ و طلایی

همچنین می توانید از شناسه های `size` و `color` در یک تگ `` استفاده کنید. مانند زیر:

`متن بزرگ و طلایی`

متن بزرگ و طلایی

شما می توانید رنگ فونت را در تگ `body` تعریف کنید که بعداً به آن می پردازیم.

این فرمان چهار فضای خالی اضافی بین دو کلمه ایجاد می کند. و در مجموع پنج فضای خالی داریم.

بحث بعدی ما علامت کپی رایت است. شما می توانید به همان روشی که برای فضای خالی اضافی ایجاد کردیم این علامت را نیز به وجود بیاورید. یعنی با استفاده از کد آن. کد علامت کپی رایت این است: `©` به این مثال توجه کنید:

This page Copyright © 2007 by me

متن نتیجه:

This page Copyright © 2007 by me

روش کار برای تمامی کاراکترهای ویژه به همین صورت است فقط کد آنها را در متن HTML قرار دهید.

در جدول زیر می توانید تعدادی از این کاراکترهای ویژه را به همراه کد سازنده آنها ببینید:

| نام کاراکتر | شکل ظاهری کاراکتر | کد سازنده کاراکتر |
|-----------------|-------------------|-------------------|
| امپرسند | & | & |
| علامت کپی رایت | © | © |
| علامت ثبت شده | ® | ® |
| علامت تجاری | ™ | ™ |
| علامت کوچکتر از | < | < |
| علامت بزرگتر از | > | > |
| علامت خنجر | † | † |
| ----- | » | » |

| | | |
|----------|---|-------------------|
| « | « | ----- |
| — | — | خط فاصله ام |
| ° | ° | علامت درجه |
| ¼ | ¼ | یک چهارم |
| ½ | ½ | یک دوم |
| ¾ | ¾ | سه چهارم |
| · | · | نقطه میانی |
| ¡ | ! | علامت تعجب وارونه |
| € | € | علامت یورو |
| £ | £ | علامت پوند انگلیس |
| ÷ | ÷ | علامت تقسیم |
| &radic | √ | علامت رادیکال |

ایجاد پیوند به دیگر صفحات با HTML

در این بخش ایجاد لینک به بقیه صفحات را مورد بررسی قرار می دهیم.

برای ایجاد لینک از تگ زیر استفاده می کنیم:

`متن لینک`

تگ `<a>` نشانه لینک است. شناسه `href` مشخص کننده آدرس لینک می باشد، `` نشانه

بسته شدن تگ است. متنی که بین این دو تگ ابتدایی و پایانی نوشته می شود متنی است که در

صفحه شما به عنوان یک لینک دیده می شود. مثلاً اگر می خواهید به یک صفحه وب مثلاً صفحه

اصلی سایت ما با این آدرس: `http://www.pc-boukan.com/html/index.html`

لینکی را در صفحه خود داشته باشید باید فرمان زیر را در قسمت `body` صفحه خود در جایی که

می خواهید لینک به نمایش درآید قرار دهید:

`نمونه برای لینک متن`

لینک در صفحه شما به این صورت دیده می شود:

[متن نمونه برای لینک](http://www.neopersia.org/html/index.html)

لینکی را که با فرمان بالا ساخته اید به صورت رنگی (بسته به تنظیمات مرورگر رنگ آن

تعیین می شود) و زیرخط دار به وجود می آید. که این خصوصیات نشانه آن است که یک

متن لینک است و وقتی که نشانگر ماوس را بر روی آن قرار دهید به شکل دست در می

آید. البته هر متنی که رنگی بود و دارای زیر خط بود لزوماً نباید یک لینک باشد. شکل

نشانگر ماوس در اینجا به کمک ما می آید.

اگر شما لینکی را که با این روش به وجود آمده کلیک کنید به صفحه مورد نظر می روید.

با این روش شما می توانید به هر صفحه ای لینک بسازید چون آدرس کامل آنرا در لینک

قرار داده اید. برای مثال شما می توانید برای صفحات سایت خود در صفحه اصلی با این روش لینکهایی بسازید.

اگر همه فایل‌های شما مانند صفحه ای که لینک در آن قرار دارد در یک پوشه باشند شما می توانید به جای آدرس کامل صفحه مقصد از روش میانبری استفاده کنید و فقط نام فایل مورد نظر را به جای آدرس کامل آن بنویسید.

در مثال زیر فرض بر این است که هر دو صفحه مبدأ و مقصد لینک در یک پوشه قرار دارند. برای نمونه سرفصل آموزش مقدماتی HTML در همان پوشه ای قرار دارد که این صفحه قرار دارد. نام این صفحه index.php است. برای ایجاد یک لینک در این صفحه می توانیم به اینگونه عمل کنیم:

`لینک نمونه برای لینک`

لینک حاصل شده:

[متن نمونه برای لینک](#)

هر دو روش بالا یک نتیجه را دارند اما در روش دوم زحمت نوشتن کمتر است.

اگر شما شک دارید یا مطمئن نیستید که هر دو صفحه در یک پوشه قرار دارند حتماً از آدرس

کامل استفاده کنید. چون در این روش مهم نیست که صفحات مربوط به چه سایتی باشند.

شناسه دیگری که می توانید در تگ لینک به کار ببرید **title** است. با استفاده از این شناسه می

توانید عنوانی را برای لینک انتخاب کنید و این عنوان در هنگامی که بازدید کننده کاوس خود را

روی لینک قرار می دهد به نمایش در نمی آید. به مثال زیر توجه کنید:

`به زبان فارسی CSS آموزش`

ماوس خود را روی لینک زیر قرار دهید تا نتیجه را مشاهده کنید:

از شناسه های دیگری نیز در تگ لینک می توان استفاده کرد که نحوه نمایش لینک در صفحه را تعیین می کنند. برای اطلاعات بیشتر صفحه [باز کردن لینک در پنجره جدید](#) را مطالعه کنید.

وارد کردن تصاویر در صفحات HTML

آیا شما می خواهید یک عکس را به صفحه خود اضافه کنید؟ برای این کار باید از تگ زیر استفاده کنید:

در این تگ IMG نشانه قرار گرفتن تصویر در صفحه و SRC آدرس محل قرار گرفتن فایل عکس است. حتماً لازم نیست این عکس مربوط به سایت شما باشد یا در سرور شما قرار داشته باشد، شما می توانید هر عکسی که در اینترنت قابل دسترسی باشد استفاده کنید (اگر قصد دارید صفحه خود را در یک سایت اینترنتی قرار دهید). شما می توانید فقط نام فایل را به جای آدرس کامل بنویسید اگر صفحه ای که عکس باید در آن قرار گیرد و خود عکس در یک پوشه قرار داشته باشند.

مثلاً اگر عکس شما در این آدرس قرار دارد

<http://www.example.com/images/image.gif> و صفحه ای که در حال ویرایش

آن هستید در پوشه ای غیر از این قرار دارد شما باید تگ زیر را برای اضافه کردن عکس به صفحه خود به کار ببرید:

``

اگر فایل `html` و عکس هر دو در یک پوشه باشند می توانید برای راحتی در تایپ کردن از این تگ استفاده کنید:

``

پسوند فایل حتماً نباید `gif` باشد. یعنی شما می توانید از فرمتی مثل `jpg` نیز استفاده کنید. این دو فرمت از بقیه پر استفاده تر هستند. اگر شما عکسی با فرمتی غیر از این دو فرمت دارید می توانید از برنامه های ویرایش عکس از قبیل فتوشاپ برای تبدیل کردن آنها استفاده کنید. برای نمونه من می خواهم یک عکس را با نام `learn_html.gif` را که در پوشه همین صفحه قرار دارد را به این صفحه اضافه کنم. در اینجا از این تگ استفاده کرده ام:

``

با نوشتن این تگ، این تصویر را در صفحه نهایی می بینیم:



به این ترتیب یک تصویر را در سمت چپ صفحه خواهیم دید (در صورتی که صفحه ما چپ به راست باشد). اگر می خواهید تصویر در مرکز صفحه (وسط خط) قرار گیرد باید از تگ `<CENTER>` در اطراف تگ تصویر استفاده کنید. مانند زیر:

`<center>`

``

`</center>`

تصویر به دست آمده به صورت زیر است:



روشهای دیگری هم برای نمایش عکس در وسط صفحه وجود دارد که با توجه به کم کاربرد شدن تگ `<center>` در نسخه چهارم HTML از آنها استفاده می شود. مثلاً می توان از تگ `<div>` با شناسه `align="center"` استفاده کرد.

به خاطر داشته باشید که آدرس محل عکس و نام آن به حروف کوچک و بزرگ حساس هستند. پس ممکن است اگر به جای `image.jpg` از `IMAGE.JPG` استفاده کنید ممکن است مرورگر شما عکسی را در صفحه نشان ندهد. پس در نوشتن آنها دقت کنید.

در مورد اضافه کردن عکسها به صفحه بعداً با جزئیات بیشتری توضیح می دهیم. تا اینجا یاد گرفتیم چگونه یک عکس را به صفحه اضافه کنیم و نحوه قرار گیری آنرا در صفحه تعیین کنیم.

تگهای مورد استفاده برای نمایش متن به صورتهای مختلف

تعدادی تگ وجود دارد که در هنگام کار با متن ها می توانید از آنها کمک بگیرید. تگهایی که در این قسمت مورد بحث قرار می گیرند اینها هستند: `<SUB>` , `<SUP>` , `<PRE>` , `<NOBR>` , `<WBR>` , `<BLINK>` ,

-1 `<SUB>` و `<SUP>`

این دو تگ بیشتر در زمانی استفاده می شوند که شما می خواهید در صفحه خود معادلات ریاضی را به نمایش در آورید اما ممکن است شما برای چیزهای دیگری هم از آنها استفاده کنید. از این تگها برای قرار دادن حروف یا اعداد کمی بالاتر یا پایینتر از بقیه متن در صفحه استفاده می شود. مانند این:

استفاده از تگ `<SUB>`:

`y₁`

نتیجه به این صورت دیده می شود. از این تگ می توان برای گذاشتن اندیس برای کلمات استفاده کرد.

`y1`

استفاده از تگ `<SUP>`:

25³

از این روش می توانید برای گذاشتن توان برای اعداد استفاده کنید. نتیجه به این صورت خواهد بود:

25³

-2 <PRE>

از این تگ زمانی استفاده می شود که شما بخواهید همان مقدار فضای خالی که در ویرایشگر متن خود در متن HTML دارید در صفحه نهایی خود نیز داشته باشید. مثلاً وقتی که متنی را در مورد برنامه نویسی در صفحه خود می نویسید.

به این مثال توجه کنید:

```
<PRE>
x=1;
y=2;
if (x==1)
{
    y=2;
}
</PRE>
```


<NORB>

مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

مثال مثال ، ، مثال ، مثال ، مثال ، مثال ،

مثال ، مثال ، مثال ، مثال مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

<NOBR>

این هم نتیجه:

، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ، مثال ،

<BLINK> -4

با استفاده از این تگ می توانید متنی را به صورت چشمک زن در آورید. این تگ فقط در

مرورگرهایی کارایی دارد که بر پایه Netscape ساخته شده باشند.

Opera و Firefox از جمله مرورگرهایی هستند که می توانند این تگ را نمایش دهند.

در Microsoft Internet Explorer فقط متن ساده را می توان مشاهده کرد.

اگر این تگ به جا استفاده شود می تواند توجه بیننده را به نکته خاصی جلب کند اما اگر به جا استفاده نشود می تواند آزار دهنده باشد.

برای کارایی بیشتر می توانید آنرا با اندازه فونتی مثل 10+ استفاده کنید.

به این مثال توجه کنید:

this is <BLINK>VERY VERY</BLINK> important!!!

نتیجه این تگ:

this is VERY VERY important!!!

وارد کردن کامنت در متن اچ تی ام ال

توضیحات (کامنتها) می توانند وقتی که در حال نوشتن صفحه خود هستید به شما کمک کنند.

مرورگرها نمی توانند کامنتها را نشان دهند پس در صفحه نهایی آنها به بیننده نشان داده نمی

شوند. تنها راه مشاهده کامنتها مشاهده صفحه **Source code** صفحه است که با زبان **HTML**

نمایش داده شود. با استفاده از این خاصیت شما می توانید در هنگام نوشتن یک صفحه برای خود

یادداشت بگذارید تا وقتی که بعداً به آن نگاه کردید بهتر آرا درک کنید.

در مواردی که پس از گذشت مدتی از نوشتن یک صفحه نیاز به ویرایش آن صفحه باشد یا بیش از یک نفر در نوشتن یک صفحه نقش داشته باشند کامنتها می توانند در فهمیدن صفحه به طراحان آن کمک کنند.

برای نوشتن یک کامنت شما باید ابتدا این علامت < سپس علامت تعجب ! و بعد دو خط فاصله - (دش) بگذارید. بعد از این علائم هر متنی که بنویسید توسط مرورگر نمایش داده نمی شود. تا وقتی که علامت پایان کامنت را در انتهای آن بگذارید.

علامت پایان کامنت به این صورت نوشته می شود: ابتدا دو خط فاصله -- سپس این علامت >. به مثال زیر توجه کنید:

<-- این یک کامنت است --!>

شما می توانید کامنتها را در چندین خط بنویسید فقط به خاطر داشته باشید که علامت انتهایی را در آخر آن قرار دهید. مانند زیر:

<!--

این یک کامنت است

شوند کامنتها در مرورگر نمایش داده نمی

-->

استفاده از لیست های HTML در صفحه

دو نمونه از پر استفاده ترین لیستها در HTML لیستهای مرتب و لیستهای نامرتب هستند. در لیستهای مرتب ترتیب بخشهای مختلف لیست مشخص می شود ولی در لیستهای نامرتب این ترتیب مشخص نمی شود.

لیستهای نامرتب

در لیستهای نامرتب برای عناصر لیست از شماره و عدد یا حروف الفبا استفاده نمی شود. عناصر این معمولاً با یک دایره تو پر مشخص می شوند. تگ ابتدای آن `` است و تگ پایانی آن `` است. برای جدا کردن هر بخش لیست در ابتدای آن از تگ `` استفاده می شود. بهتر است هر تگ `` که باز می شود با `` بسته شود. در زیر یک مثال برای این نوع لیست آورده شده است:

``

`بخش اول لیست`

`دوم لیست بخش`

`بخش سوم لیست`

``

مثال بالا یک لیست بولت دار با سه بخش را به ما می دهد:

بخش اول لیست

بخش دوم لیست

بخش سوم لیست

به این نکته هم توجه کنید که `` نیازی به تگ پایانی ندارد ولی بهتر است از تگ پایانی آن یعنی `` استفاده شود.. فاصله متن لیست شده از ابتدای خط هم مقداری از بقیه متن بیشتر می شود. شما می توانید این فاصله را با افزودن تگ `` بیشتر کنید فقط به خاطر داشته باشید تگهایی را که باز کرده اید در آخر لیست ببندید. در زیر یک مثال برای این نمونه لیست آورده شده است:

``

``

``

``لیست خط اول``

``خط دوم لیست``

``لیست خط سوم``

``

``

``

نتیجه این فرمان به شکل زیر خواهد بود:

خط اول لیست

خط دوم لیست

خط سوم لیست

از شناسه `type` می توان در تگ `` استفاده کرد و ظاهر لیست را تغییر داد.

شناسه type

از این شناسه در لیستهای نامرتب برای مشخص کردن شکل علامت مشخص کننده هر بخش استفاده می شود. این شناسه می تواند مقادیر زیر را داشته باشد:

circle

با قرار دادن این مقدار برای لیست علامت ابتدایی هر بخش به شکل یک دایره تو خالی در می آید.

disc

این مقدار شکل علامت ابتدای هر قسمت از لیست را به یک دایره تو پر تغییر می دهد.

square

این مقدار شکل علامت لیست را به صورت یک مربع تو پر در می آورد.

به یاد داشته باشید که از شناسه `type` هم می توان در تگ `` استفاده کرد و هم در تگ ``. در صورتی که برای تگ `` از این شناسه استفاده شود علامت مربوط به همه قسمت‌های لیست تغییر می کند اما اگر برای تگ `` از این شناسه استفاده شود فقط علامت مربوط به همان بخش از لیست تغییر می کند.

در مثال زیر می توانید نحوه نمایش لیست با استفاده از شناسه `type` را در حالات مختلف مشاهده کنید. ما از کد زیر برای ساختن این لیست آزمایشی استفاده می کنیم:

<li type="circle">لیست قسمت اول

<li type="disk">قسمت دوم لیست

<li type="square">قسمت سوم لیست

می توانید لیست به دست آمده را در اینجا مشاهده کنید. به خاطر داشته باشید که در مرورگرهای مختلف ممکن است نحوه نمایش لیستها متفاوت باشد.

قسمت اول لیست

قسمت دوم لیست

قسمت سوم لیست

اگر می خواهید از تگ <center> برای لیست خود استفاده کنید در این کار دقت کنید. چون ممکن است تمام سطرهای متن شما به یک اندازه نباشند. در این صورت به جای اینکه لیستی در یک خط و مستقیم داشته باشید ممکن است لیستی در هم ریخته و شلوغ به وجود آید. مانند زیر:

<center>

از بقیه سطرهای لیست است بخش اول لیست که بلندتر

لیست بخش دوم

آخر بخش سوم لیست در سطر

<center>

لیست نهایی به این صورت در صفحه دیده می شود.

بخش اول لیست که بلندتر از بقیه سطرهای لیست است

بخش دوم لیست

بخش سوم لیست در سطر آخر

اگر شما در یک صفحه بیشتر به لیستها نیاز دارید می توانید از روش بالا استفاده کنید اما باید بدانید که ممکن است ظاهر لیست در اندازه های مختلف پنجره مرورگر به هم بخورد. راه دیگری که برای جلوگیری از این به هم خوردگی وجود دارد استفاده از یک Table است که بعداً در مورد آن توضیح خواهیم داد.

لیستهای مرتب

شما می توانید در مورد لیستهای مرتب هم مانند لیستهای نامرتب عمل کنید فقط کافیست به جای استفاده از `` و `` از تگهای `` و `` استفاده کنید. به یک مثال در این زمینه توجه کنید:

``

`` بخش اول لیست

`` دوم لیست بخش

`` بخش سوم لیست

``

نتیجه به صورت لیستی شماره گذاری شده ظاهر می شود:

بخش اول لیست

بخش دوم لیست

بخش سوم لیست

برای کنترل تگ `` از چند شناسه می توان استفاده کرد

شناسه start

در مورد لیستهای مرتب با استفاده از شناسه `start` در تگ `` می توانیم تعیین کنیم که شماره های لیست از چه عددی شروع شود. به مثال زیر توجه کنید:

```
<ol start="15">
```

```
<li>.شماره آن 15 است این سطر اول لیست است ولی</li>
```

```
</ol>
```

نتیجه به صورت زیر مشاهده می شود:

این سطر اول لیست است ولی شماره آن 15 است.

شناسه type

این شناسه در لیستهای مرتب هم مانند لیستهای نامرتب برای تغییر علامت ابتدایی هر بخش از لیست استفاده می شود که در اینجا شماره ای است که برای هر قسمت از لیست استفاده می شود. از این شناسه می توانید به صورت زیر استفاده کنید:

```
type="1"
```

با این شناسه بخشهای لیست با اعداد ریاضی مشخص می شوند.

type="A"

با استفاده از این شناسه بخشهای لیست با حروف بزرگ انگلیسی مشخص می شوند.

type="a"

با استفاده از این شناسه بخشهای لیست با حروف کوچک انگلیسی مشخص می شوند.

type="I"

با استفاده از این شناسه بخشهای لیست با حروف بزرگ یونانی مشخص می شوند.

type="i"

با استفاده از این شناسه بخشهای لیست با حروف کوچک یونانی مشخص می شوند.

این شناسه هم می تواند در تگ `` استفاده شود و هم در تگ `` و در این مورد مانند لیستهای نامرتب عمل می کند.

در مثال زیر می توانید نحوه نمایش حالات مختلف لیست را با استفاده از شناسه `type` مشاهده کنید:

``

`</li type="A">` لیست قسمت اول ``

<li type="a">قسمت دوم لیست

<li type="1">قسمت سوم لیست

<li type="۱">لیست قسمت چهارم

<li type="i">قسمت پنجم لیست

نتیجه به صورت زیر خواهد بود:

قسمت اول لیست

قسمت دوم لیست

قسمت سوم لیست

قسمت چهارم لیست

قسمت پنجم لیست

شناسه های مورد استفاده در تگ body

وقتی که شما یک صفحه HTML می نویسید می توانید چیزهای زیادی را در آن با اضافه کردن

شناسه هایی به تگ <BODY> تغییر داد. تگ زیر را در نظر بگیرید:

```
<body bgcolor="green" text="blue" link="yellow" alink="red" vlink="white" background="image.gif">
```

شما می توانید به هر تعداد از این شناسه ها که خواستید به این تگ اضافه کنید. اگر از شناسه ای

استفاده نکنید این گزینه مطابق پیش فرض مرورگر تنظیم می شود. در زیر توضیحی برای هر

کدام از شناسه های تگ BODY آورده شده است. البته در مورد هر کدام از آنها به طور کامل در

درسهای بعد توضیح خواهیم داد:

bgcolor="color"

این شناسه رنگ زمینه صفحه شما را تغییر می دهد. شما می توانید آنرا به هر رنگی که دوست

دارید تغییر دهید. فقط به جای « رنگ » در فرمان بالا اسم رنگی را که می خواهید یا معادل هگزا

دسیمال آنرا قرار دهید. برای دیدن لیست رنگها به صورت هگزا دسیمال به سر فصل این بحث

بروید. گزینه پیش فرض این ویژگی معمولاً سفید یا خاکستری است.

text="color"

این شناسه رنگ متنی را که در پنجره مرورگر به نمایش در می آید تعیین می کند. شما می

توانید آنرا به هر رنگی که دوست دارید تغییر دهید. فقط به جای « رنگ » در فرمان بالا اسم

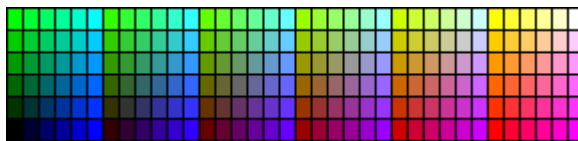
رنگی را که می خواهید یا معادل هگزا دسیمال آنرا قرار دهید. برای دیدن لیست رنگها به صورت

هگزا دسیمال گزینه پیش فرض این ویژگی مشکی است.

link="color"

معادل هگزادسیمال رنگهای HTML

جدول هگزادسیمال رنگها



نشان دادن معادل هگزادسیمال هر رنگ بر روی آن کلیک کنید برای

شود مورد نظر خود در جدول بالا، مقدار هگزا دسیمال آن در کادر زیر نشان داده می پس از کلیک کردن رنگ

این شناسه رنگ تمام لینکهای باز نشده در صفحه را تغییر می دهد. شما می توانید آنرا به هر

رنگی که دوست دارید تغییر دهید. فقط به جای « رنگ » در فرمان بالا اسم رنگی را که می

خواهید یا معادل هگزا دسیمال آنرا قرار دهید. برای دیدن لیست رنگها به صورت هگزا دسیمال

گزینه پیش فرض برای لینکهای باز نشده معمولاً آبی است.

a link="color"

این شناسه رنگ لینکهای فعال را در صفحه شما تعیین می کند. این لینکها آنهایی هستند که به

وسیله کاربر بر روی آنها کلیک شده است. شما می توانید آنرا به هر رنگی که دوست دارید تغییر

دهید. فقط به جای « رنگ » در فرمان بالا اسم رنگی را که می خواهید یا معادل هگزا دسیمال

آنرا قرار دهید. برای دیدن لیست رنگها به صورت هگزا دسیمال.

vlink="color"

این شناسه رنگ لینکهای باز شده را در صفحه تعیین می کند. شما می توانید آنرا به هر رنگی که دوست دارید تغییر دهید. فقط به جای « رنگ » در فرمان بالا اسم رنگی را که می خواهید یا معادل هگزا دسیمال آنرا قرار دهید. برای دیدن لیست رنگها به صورت هگزا دسیمال. گزینه پیش فرض این ویژگی بنفش است.

`background="image.gif"`

این شناسه یک عکس را به عنوان زمینه برای صفحه شما قرار می دهد. اگر شما از این شناسه استفاده کنید تصویر زمینه ای که انتخاب کرده اید جایگزین هر رنگی که به عنوان زمینه انتخاب کرده بودید می شود. اگر از این شناسه استفاده نکنید مرورگر از رنگی که به عنوان زمینه انتخاب کرده اید استفاده می کند و در صورت عدم انتخاب رنگ از رنگ پیش فرض خود استفاده می کند.

تغییر رنگ پیش فرض متن در صفحه

برای تغییر رنگ پیش فرض متن در صفحه باید از تگ `<BODY>` استفاده کنید. شما باید دنبال `text=" "` بعد از تگ `BODY` بگردید. مانند این:

`<BODY text="black">`

ممکن است به جای رنگ از ترکیبی از حروف و اعداد استفاده شده باشد. مانند زیر:

`<BODY text="#000000">`

البته ممکن است هیچکدام از موارد بالا وجود نداشته باشد، یا شما بخواهید یک صفحه تازه بنویسید. در این موارد باید این فرمان اضافی را به تگ `BODY` خود اضافه کنید. درست مثل موارد بالا. اگر فرمانهای دیگری هم دارید می توانید این فرمان را در انتهای آنها اضافه کنید البته با قرار دادن یک فضای خالی بین فرمانها. برای نمونه به سطر زیر توجه کنید، من برای تعیین رنگ زمینه و رنگ متن این صفحه از این فرمان استفاده کرده ام:

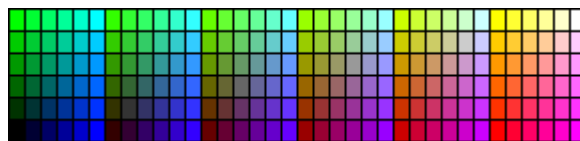
<BODY bgcolor="#66CCFF" text="black">

حالا اگر می خواهید رنگ متن را تغییر دهید می توانید نام انگلیسی رنگ مورد نظر خود را به جای black بنویسید. روش دیگری که برای این کار وجود دارد استفاده از معادل هگزادسیمال رنگها است که در این روش می توانید از بین رنگهای بیشتری رنگ خود را انتخاب کنید چون ما برای تمام رنگهای مورد استفاده نمی توانیم اسم واحدی را به کار ببریم.

جدول رنگهای هگزادسیمال

معادل هگزادسیمال رنگهای HTML

جدول هگزادسیمال رنگها



نشان دادن معادل هگزادسیمال هر رنگ بر روی آن کلیک کنید برای

شود مورد نظر خود در جدول بالا، مقدار هگزا دسیمال آن در کادر زیر نشان داده می پس از کلیک کردن رنگ

برای مثال اگر می خواهید رنگ متن شما زرد باشد از دو طریق می توانید به این خواسته خود برسید:

استفاده از نام رنگ به صورت زیر :

<BODY text="yellow">

استفاده از معادل هگزادسیمال رنگها :

<BODY text="#FFFF00">

بعد از انجام این کار هر متنی که در صفحه مرورگر شما دیده شود به رنگ زرد خواهد بود.

تغییر رنگ لینکهای صفحه در حالات مختلف

اگر بخواهید رنگ لینکها را در صفحه خود تغییر دهید مانند تغییر رنگ پیش فرض متن باید تگ BODY را پیدا کنید. در تگ BODY به دنبال شناسه link بگردید. مثلاً ممکن است تگ BODY به این صورت باشد:

```
<body link="blue">
```

در تگ BODY ممکن است شناسه های دیگری هم وجود داشته باشند، مثلاً شناسه های مربوط به رنگ متن و یا رنگ زمینه. در این صورت می توانید شناسه رنگ لینک را با یک فاصله خالی با آخرین فرمان موجود در تگ BODY به این تگ اضافه کنید.

حالا برای تغییر رنگ لینکها در صفحه نام رنگ مورد نظر خود را به عنوان مقدار شناسه link وارد کنید یا از معادل هگزادسیمال رنگ به جای نام رنگ استفاده کنید. مثلاً برای تغییر رنگ لینکها به زرد می توانیم مانند زیر عمل کنیم:

1- استفاده از نام رنگ:

```
<body link="yellow">
```

2- استفاده از معادل هگزادسیمال رنگ:

```
<body link="#ffff00">
```

حالا همه لینکهای موجود در صفحه شما به جای آبی به رنگ زرد در مرورگر ظاهر می شوند. همین کار را می توانید برای لینکهای فعال و لینکهای بازدید شده تکرار کنید. مانند زیر:

```
<body alink="blue" vlink="violet">
```

مانند مثال بالا شناسه **alink** برای لینکهای فعال و **vlink** برای لینکهای بازدید شده استفاده می شود.

برای تغییر دیگر رنگها هم به همین صورت عمل می شود تنها نکتهای که باید مد نظر قرار گیرد این است که حتماً قبل از وارد کردن فرمان یک فضای خالی با فرمان قبلی بگذارید. روش دیگری که برای تغییر رنگ لینکها وجود دارد استفاده از استایل و **CSS** است. در صورتی که برای تعیین مشخصات لینکها از استایل استفاده شود امکان افزودن ویژگیهای دیگری نظیر کادر و رنگ زمینه هم به لینکها وجود دارد. در صورت تمایل می توانید بخش مربوط به [تغییر ویژگیهای لینکها با استفاده از استایل](#) را مطالعه کنید.

اعمال تغییر بر روی لینکها در صفحه با استفاده از استایل

با استفاده از **CSS** می توانیم نحوه نمایش دادن لینکها را در صفحه تنظیم کنیم. به صورتی که استفاده از جاوا اسکریپت در صفحه تا حد زیادی کم می شود. از طریق استفاده از استایل می توانیم رنگ لینکها، اندازه فونت آنها، زیر خط لینکها، رنگ زمینه هر لینک و بسیاری دیگر از خواص لینکها را تغییر دهیم.

برای اینکه بتوانیم از همه خواصی که برای لینکها در **CSS** وجود دارد استفاده کنیم بهتر است تا استایل را در بخش **HEAD** متن خود یا در فایل های **CSS** خارجی که به صفحه پیوند می شوند استفاده کنیم. در هر دو صورت مراحل انجام کار شبیه هم است. ما در اینجا به استایل را در بخش **HEAD** متن تعریف می کنیم.

برای کار با لینکها در CSS می توانیم از مشخصات یک لینک را به چهار بخش تقسیم کنیم که در زیر می توانید آنها را مشاهده کنید:

a و **a:link**

این مشخصه ها نحوه نمایش لینکهایی را که در صفحه وجود دارند و هنوز روی آنها کلیک نشده است تنظیم می کنند.

a:hover

این مشخصه وضعیت لینک را وقتی که نشانگر ماوس روی آن قرار دارن تنظیم می کند.

a:active

این قسمت مشخص کننده وضعیت لینکی است که بر روی آن کلیک می شود.

a:visited

این مشخصه برای لینکهایی که در بازدیدهای قبلی مورد استفاده قرار گرفته اند استفاده می شود.

برای کار کردن با لینکها ما در قسمت HEAD متن از مشخصه های بالا استفاده می کنیم. یعنی

کد زیر را در بخش HEAD متن وارد می کنیم اگر می خواهید از استایل خارجی استفاده کنید

می توانید [نحوه فرا خوانی استایل خارجی به صفحه](#) را مطالعه کنید.

```
<style type="text/css">
```

```
<!--
```

```
a:link { مشخصات مورد نظر }
```

```
a:hover { مشخصات مورد نظر }
```

```
a:active { مشخصات مورد نظر }
```

```
a:visited { مشخصات مورد نظر }
```

```
-->
```

```
</style>
```

برای هر ویژگی بالا می توانید مشخصات خاص خود را تعریف کنید برای مثال می تواند رنگ از مشخصه `background-color` برای تعیین رنگ زمینه لینک وقتی که ماوس روی آن قرار می گیرد استفاده کنید یا از `color` برای تعیین رنگ متن لینکها. به طور کلی می توانید از بیشتر مشخصه هایی که برای متن ها کارایی دارند در مورد لینکها هم استفاده کنید برای مثال `font-family` و `font-size` و `border` و ...

در اینجا یک مثال مطرح می کنیم که در آن لینکهای صفحه به رنگ سبز نمایش داده می شوند با اندازه فونت `pt12` و بدون زیرخط، وقتی که ماوس روی آنها قرار می گیرد رنگ آنها قرمز می شود و رنگ زمینه متفاوتی پیدا می کنند همچنین یک کادر نیز اطراف آنها به وجود می آید. در موقع کلیک کردن رنگ لینک سفید می شود و لینکهایی که در دفعات قبلی کلیک شده اند با زیر خط نمایش داده می شوند. برای ساختن لینکهایی با مشخصاتی که بیان شد می توانیم از کد زیر در قسمت `HEAD` متن یا استایل خارجی استفاده کنیم:

```
<style type="text/css">
<!--
a:link { color:green; font-size:12pt; text-decoration:none }
a:hover { color:red; background-color:#6699CC; border:#006699 2px
solid }
a:active { color: white }
a:visited { text-decoration: underline }
-->
</style>
```

در قسمت زیر می توانید لینکهایی را که برای آنها از کد بالا استفاده شده است مشاهده کنید:

این لینکها با استفاده از کد بالا تنظیم شده اند.

ماوس خود را روی آنها قرار دهید.

می توانید بر روی آنها کلیک کنید.

همانطور که می بینید کدهایی که در قسمت HEAD این صفحه قرار دارند بر روی لینکهای بالا

تأثیر گذاشته اند اما لینکها در سایر نقاط صفحه ویژگیهای متفاوتی دارند. دلیل این تفاوت این

است که ما لینکهای بالا را در یک DIV قرار دادیم و برای آن یک ID در بخش HEAD صفحه

تعریف کردیم و ویژگیهای مربوط به لینکهای بالا در قالب ID مشخص کردیم. در اینجا می توانید

کدی را که ما مورد استفاده قرار داده ایم ببینید:

```
#pre a:link,#pre a { color:green; font-size:10pt; text-decoration:none }
```

```
#pre a:hover { color:red; background-color:#6699CC; border:#006699  
2px solid; text-decoration:none }
```

```
#pre a:active { color: White }
```

```
#pre a:visited { text-decoration: underline; color:#003399 }
```

کد مربوط به DIV را نیز به صورت زیر نوشتیم:

```
<div align="center" id="pre">
```

لینکها کدهای مربوط به

```
</div>
```

وارد کردن تصاویر در صفحات HTML

اگر می خواهید برای زمینه صفحه خود یک رنگ را انتخاب کنید، باید این رنگ را در تگ BODY تعریف کنید. وقتی که این تگ را پیدا کردید در آن به دنبال شناسه bgcolor بگردید ، که ممکن است مانند زیر باشد:

```
<body bgcolor="gray">
```

ممکن است به جای نام رنگ از معادل هگزادسیمال رنگ استفاده شده باشد. مانند زیر (این فرمانی است که من برای زمینه این صفحه از آن استفاده کرده ام):

```
<body bgcolor="#66ccff">
```

البته ممکن است این شناسه را در تگ BODY پیدا نکنید، در این صورت یک فاصله بعد از BODY قرار داده شناسه bgcolor را وارد کنید.

برای عوض کردن رنگ زمینه نام رنگ مورد نظر خود را به جای نام رنگ در شناسه بالا قرار دهید. به جای نام رنگ می توانید از معادل هگزا دسیمال رنگ استفاده کنید. برای مثال اگر می خواهید رنگ زمینه به رنگ سبز درآید باید مانند زیر عمل کنید:

```
<body bgcolor="green">
```

و یا به این صورت:

```
<body bgcolor="008000">
```

روش دیگری که برای تغییر رنگ زمینه صفحه وجود دارد استفاده از CSS است که با ورود نسخه چهارم HTML استفاده از آن رو به افزایش است. اگر می خواهید مطالب بیشتری در این مورد بدانید به این صفحه مراجعه کنید:

آموزش قرار دادن یک تصویر در زمینه صفحه

برای قرار دادن یک تصویر به عنوان زمینه صفحه باید از تگ BODY استفاده کرد. وقتی که تگ BODY را پیدا کردید ممکن است به صورت ساده باشد یا با فرمانهایی که به آن اضافه شده است. مانند زیر:

```
<body bgcolor="#66ccff" text="#000000">
```

کاری که شما باید انجام دهید، اضافه کردن یک شناسه به تگ BODY است. بنابراین یک فاصله بعد از آخرین فرمان بگذارید و سپس این فرمان را وارد کنید:

```
background=" "
```

اکنون تگ BODY شما باید به این صورت باشد (اگر فرمانهای دیگری به آن اضافه نکرده باشید):

```
<body background=" ">
```

حالا باید آدرس عکس را بین دو دابل کوت قرار دهیم. برای این کار دو راه وجود دارد: برای این کار اگر صفحه و عکس در یک مسیر هستند می توانیم فقط نام کامل عکس با پسوند آنرا بنویسیم. مثلاً:

```
<body background="image.gif">
```

اگر از قرار داشتن صفحه و عکس در یک مسیر مطمئن نیستید یا عکس و صفحه در یک مسیر نیستند می توانید از آدرس کامل عکس استفاده کنید.

```
<body background="http://www.example.com/image.gif">
```

تصاویر برای زمینه باید با یکی از دو فرمت GIF یا JPG باشند. اگر عکس به این صورت نبود باید آن را با برنامه ای مثل Paint Shop Pro یا Photoshop به یکی از این فرمتها تبدیل کرد.

برای مثال فرض کنید یک عکس با نام background.jpg داریم که در این مسیر قرار دارد:

http://www.mysite.com/img . حال اگر بخواهیم صفحه ای را در پوشه ای به غیر از

پوشه img در این سایت طراحی کنیم باید از آدرس کامل عکس یعنی

http:www.mysite.com/img/background.jpg استفاده کنیم. و تگ BODY ما به

این صورت در می آید:

```
<body background="http:www.mysite.com/img/background.jpg">
```

ما برای تعیین تصویر زمینه این صفحه قبلاً از این تگ استفاده می کرده ایم:

```
<body background="http://www.neopersia.org/images/index.jpg">
```

اما اگر بخواهیم صفحه را به پوشه img منتقل کرده یا صفحه جدیدی را در پوشه img طراحی

کنیم می توانیم از نام عکس یعنی: background.jpg استفاده کنیم و تگ BODY ما به این

صورت در می آید:

```
<body background="background.jpg">
```

برای تغییر دادن تصویر زمینه روش دیگر هم وجود دارد که استفاده از CSS است. در مورد این

روش می تونید مطالب بیشتری را در این صفحات مطالعه کنید:

تنظیم نحوه نمایش عکس در صفحه

اگر شما یک عکس دارید که می خواهید از آن استفاده کنید اما اندازه آن عکس اندازه مورد نظر شما نیست. مکن است عکس به قدری بزرگ باشد که تمام فضای صفحه را بگیرد یا شما اندازه ای بزرگتر از اندازه فعلی عکس را بخواهید.

شما می توانید اندازه عکس را تغییر دهید. این کار را می توانید با افزودن شناسه های `width` (برای عرض عکس) و `height` (برای ارتفاع عکس) به تگ `img` انجام دهید. به یک مثال در این مورد توجه کنید:

فرض کنیم یک عکس با نام `learn_html.gif` داریم و می خواهیم آنرا کوچکتر از اندازه اصلی آن نشان دهیم. تنها چیزی که باید بدانیم اندازه اصلی عکس است. اندازه عرض و ارتفاع عکس معمولاً با مقیاس پیکسل سنجیده میشود. عکس ما `137` پیکسل عرض و `77` پیکسل ارتفاع دارد. در زیر می توانید عکس را با اندازه های واقعی ببینید.



حالا می خواهیم شناسه های `width` و `height` را در تگ عکس وارد کنیم. تگ ما به این صورت در می آید:

```

```

حالا کافی است اندازه مورد نظر را در بین دو دابل کوت (" ") وارد کنید. برای مثال من اندازه 100 را برای عرض و 50 را برای ارتفاع وارد می کنم . به صورت زیر:

```

```

در زیر می توانید عکس را پس از تغییر اندازه مشاهده کنید:



برای بزرگتر کردن عکس هم کاری مشابه بالا را انجام می دهیم:
مثلاً با عرض 200 و ارتفاع 100:

```

```



همانطور که می بینید اندازه عکس بزرگتر از اندازه واقعی شد:

همانطور که ممکن است متوجه شده باشید در تصاویری که با این روش تغییر اندازه پیدا می کنند

اگر به نسبت بین عرض و ارتفاع عکس رعایت نشود ممکن است تناسب عکس به هم بخورد. مثلاً

تصویر بیش از حد کشیده به نظر برسد و...

اگر بخواهید تصاویر با همان نسبت قبلی تغییر اندازه پیدا کنند باید نسبت بین عرض تصویر و

ارتفاع آن را به دست آورید مثلاً وقتی من مو خواهم تصویری با عرض 100 پیکسل داشته باشم

می توانم به این صورت ارتفاعی را که در آن تناسب عکس به هم نمی خورد پیدا کنم:

$$77 \div 137 = 0.562$$

$$\text{height} \div 100 = 0.562 \quad \text{--->} \quad \text{height} = 100 \times 0.562 \quad \text{--->} \quad \text{height} = 56.2$$

حال اگر من اندازه تصویر را با عرض 100 پیکسل و ارتفاع 2/56 پیکسل تعریف کنم تصویری با همان تناسب تصویر اصلی به دست خواهیم آورد مانند زیر:

```

```



همانطور که در تصویر روبرو می بینید تناسب تصویر تا حد زیادی رعایت شده

است. البته ما از 0.2 پیکسل در ارتفاع تصویر چشم پوشی کردیم.

برای راحتی کار می توانید تصاویر را با استفاده از برنامه های گرافیکی مثل فتوشاپ، کارل دراو و...

اندازه تصاویر را تغییر داده و بعداً در صفحه خود استفاده کنید. در این صورت کار شما راحت تر

است و گذشته از راحتی وقتی عکسی را با یک برنامه گرافیکی کوچک می کنید سائز فایل عکس

هم کوچکتر می شود و بارگذاری صفحه شما را طولانی نمی کند.

ساختن لینک با استفاده از تصاویر

برای اینکه از یک عکس به عنوان یک لینک استفاده کنید با ید از دو نکته که قبلاً آموختیم استفاده کنیم.

[چگونه یک لینک بسازیم.](#)

[چگونه یک عکس را به صفحه اضافه کنیم.](#)

برای اینکه بخواهیم به یک صفحه لینکی را ایجاد کنیم باید از این تگ استفاده کنیم:

```
<a href="url">متن لینک</a>
```

اگر بخواهیم یک عکس را به صفحه اضافه کنیم باید از این تگ استفاده کنیم:

``

حال اگر بخواهیم از یک عکس به عنوان یک لینک استفاده کنیم باید تگ عکس را به جای متن لینک استفاده کنیم. یعنی به صورت زیر:

``

تنها نکته ای که در اینجا باید به آن دقت کنید این است که نباید تگ لینک را قبل از ورود تصویر ببندید. یعنی تگ عکس باید بین تگ ابتدایی و انتهایی لینک قرار بگیرد تا به عنوان یک لینک عمل کند. به این مثال توجه کنید:

می خواهیم از عکسی با نام `learn_html.gif` که در پوشه همین صفحه قرار دارد به عنوان لینک به مقدمه آموزش HTML که در همین پوشه است استفاده کنیم (نام صفحه مورد نظر `index.html` است). به این صورت عمل می کنیم:

``

لینک به دست آمده مانند سطر زیر است:



اگر ماوس خود را بر روی عکس بالا قرار دهید خواهید دید که نشانگر ماوس به شکل دست در می آید. اگر بر روی عکس کلیک کنید به صفحه سرفصل آموزش HTML خواهید رفت.

اغلب مرورگرها به تصاویری که به عنوان لینک استفاده می شوند یک خط حاشیه (`border`) در لبه های تصویر اضافه می کنند. برای اینکه این خط دیده نشود شناسه زیر را به تگ عکس خود اضافه کنید:

border="0"

مثال قبل را با اضافه کردن شناسه بالا تکرار می کنیم:

```
<a href="index.html"></a>
```

نتیجه به این صورت در می آید:



شما می توانید خط حاشیه را کلفت تر نیز نشان دهید. فقط کافی است عدد بزرگتری را در فرمان

border قرار دهید. مانند زیر:

```
<a href="index.html"></a>
```

نتیجه را در زیر مشاهده می کنید:



منظور از قرار دادن خط حاشیه مشخص کننده محدوده لینک است.

تنظیم نحوه قرار گرفتن تصاویر در صفحه

برای اینکه بتوانیم یک متن را در کنار یک عکس بنویسیم باید در تگ عکس از یکی این فرمانها استفاده کنیم (در صورتی که از این فرمانها استفاده نشود متن و عکس در دو سطر نمایش داده می شوند):

`align="left"`

`align="right"`

تگ عکس با استفاده از این فرمانها به صورت زیر در می آید:

``

عکس مورد نظر در صفحه به این صورت نمایش داده می شود:

اکنون می توانید متن را تایپ کنید و نگران قرار گرفتن متن زیر عکس هم نباشید چون متن در اطراف عکس به نمایش در خواهد آمد.

به مثال زیر توجه کنید:

`
` این سطر قبل از وارد کردن عکس است

``

`
` درآید این سطر باید در کنار عکس به نمایش

`
` همینطور تا وقتی که

`
` برسد متن به پایان عکس

`
` متن باید در کنار عکس قرار گیرد

`
` برسد تا به پایان عکس

نتیجه هم به صورت زیر است:

این سطر قبل از وارد کردن عکس است

باید در کنار عکس به نمایش درآید این سطر



همینطور تا وقتی که

برسد متن به پایان عکس

متن باید در کنار عکس قرار گیرد

تا به پایان عکس برسد

البته ممکن است شما احتیاجی به استفاده از تگ سطر جدید یعنی `
` نداشته باشید. من از

این تگ برای جلوگیری از طولانی شدن متن استفاده کردم. اگر از این تگها استفاده نکنید تا جایی

که عرض صفحه اجازه می دهد سطر ادامه پیدا می کند و بعد به سطر بعد خواهید رفت.

شما می توانید از فرمان `align="right"` برای قرار دادن عکس در سمت راست صفحه استفاده

کنید. در این صورت عکس در سمت راست صفحه نمایش داده می شود و متن در سمت چپ

کس دیده می شود.

استفاده از متن جایگزین (کامنت) برای تصاویر

از شناسه **alt** برای وارد کردن متن جایگزین برای عکس در تگ `` استفاده می شود. موقعی که مرورگر به هر دلیلی نتواند عکس موجود در صفحه شما را نمایش دهد به جای آن این کامنت را نشان می دهد. و می تواند به بیننده بفهماند که چه چیزی را از دست داده است. از این شناسه مانند مثال زیر استفاده می شود :

alt="متن کامنت"

این فرمان را در تگ تصویر (IMG) قرار دهید، فاصله بین دو فرمان را فراموش نکنید. مانند زیر:

ما این کامنت را به عنوان جانشین عکس در تگ IMG وارد می کنیم. تا هر وقت عکس به نمایش در نیامد این کامنت به نمایش درآید.

در سطر زیر نشان می توانید یک کامنت جایگزین را ببینید چون عکس اصلی به نمایش در نمی آید (در اینجا عکس به نمایش در نمی آید چون در مسیر مربوطه چنین عکسی وجود ندارد):

توصیه می کنیم تا حد امکان برای عکسهای صفحه خود از این کامنت استفاده کنید (البته برای نه برای عکسهای تزئینی). در این صورت صفحه شما می تواند برای تمام کاربران مفید باشد. چون ممکن است به هر دلیلی برخی از کاربران امکان مشاهده تصاویر صفحه شما را نداشته باشند.

یکی از این دلایل که در کشور ما هم مصداق دارد عدم دسترسی به اینترنت با سرعت بالا و استفاده از خطوط تلفن برای دسترسی به اینترنت است. در این رابطه تقریباً تمامی مرورگرها، این امکان را در اختیار کاربر قرار می دهند که از بارگذاری تصاویر در صفحات جلوگیری کنند، تا بتوانند از متن صفحه را سریعتر مشاهده کنند، یا فایل‌های خود را با سرعت بیشتری داتلود کنند. موتورهای جستجو هم وقتی که صفحه ای را ایندکس می کنند نمی توانند تصاویر آنرا ببینند و فقط نام تصویر و کامنت جایگزین آنرا می خوانند پس برای تصاویری که به موضوع صفحه شما مرتبط هستند و یا می خواهید بازدید کنندگان بتوانند در موتورهای جستجو آنها را پیدا کنند حتماً از این کامنت استفاده کنید.

ساختن لینک به نقاط مختلف یک صفحه

بیا یاد تصور کنیم که یک صفحه بلند داریم. ممکن است شما بخواهید که بیننده بتواند با یک کلیک از نقطه معینی دوباره به بالای صفحه باز گردد. احتمالاً این لینک را زیاد دیده اید:

[Back to the Top](#)

ممکن است شما بخواهید صفحه را به چندین قسمت تقسیم کنید و در بالای صفحه فهرستی برای آن قرار دهید که بیننده با کلیک کردن بر روی عنوان هر قسمت به آن قسمت در صفحه برود.

در این موارد شما می توانید از پیوندهای نامگذاری شده که در قسمت‌های مشخصی از صفحه شما قرار دارند که شما می خواهید به آن نقاط لینک بسازید استفاده کنید. شما می توانید به هر نقطه از صفحه لینک بسازید اما فعلاً در مورد لینک به بالای صفحه توضیح می دهیم.

برای این کار به بالای قسمت BODY در متن HTML خود بروید (درست بعد از تگ BODY)
اکنون این تگ را تایپ کنید:

```
<a name="top"></a>
```

شما می توانید هر نامی را بین دو دابل کوت بنویسید. نوشتن top کار ما را برای درک بهتر آسان می کند.

حالا به جایی در قسمت BODY بروید که می خواهید از آنجا لینکی به بالای صفحه ایجاد کنید.
این فرمان را تایپ کنید:

```
<a href="#top">بازگشت به بالا</a>
```

اکنون وقتی لینک بازگشت به بالا را کلیک کنید به بالای صفحه می روید.

علامت # به مرورگر می فهماند که مقصد پیوندی نامگذاری شده در همین صفحه است. اگر شما یک پیوند با نام "UP" ساخته بودید اکنون باید از "#up" به جای "#top" استفاده می کردید.

شما می توانید در هر کجای صفحه یک پیوند نامگذاری شده بسازید و به وسیله یک لینک از قسمت‌های دیگر به آن قسمت بازگردید.

لینک زیر را برای نمونه به بالای صفحه ساخته ام. می توانید آنرا امتحان کنید:

بازگشت به بالای صفحه

با استفاده از این روش نه تنها می توانید لینکی بسازید که بازدید کننده با کلیک بر روی آن به قسمت دیگری از همان صفحه برود، بلکه می توانید از این روش در لینکهایی که در دیگر صفحات قرار دارند استفاده کنید و بازدید کننده را به قسمت مشخصی از یک صفحه دیگر بفرستید. تنها نکته ر اینجا این است که در صفحه مقصد باید لینکهای نامگذاری شده وجود داشته باشند و از این نامگذاری در لینکی که رد صفحه مبدأ قرار دارد استفاده شود.

برای مثال اگر شما لینک زیر را در صفحه خود قرار دهید با کلیک بر روی آن می توانید بخش پایینی این صفحه را مشاهده کنید و باز دید خود را از پایین این صفحه شروع کنید. (ما یک لینک نام گذاری شده با نام #bottom در پایین این صفحه قرار داده ایم) :

 ساختن

 بازگشت به بالای صفحه لینک

ساختن لینک برای آدرس ایمیل، بخش اول

برای ایجاد یک لینک که بیننده با آن بتواند به شما ایمیل بفرستد باید ز تگ استاندارد لینک استفاده کنیم. اما تفاوت این لینک با لینک به یک صفحه در آدرسی است که به عنوان مقصد لینک نوشته می شود. برای اینکه مرورگر را مجبور کنیم لینک را به عنوان یک لینک ایمیل بشناسد، باید از این عبارت : `mailto` به جای `http://` در آغاز آدرس استفاده کنیم و بعد از آن آدرس ایمیل را بنویسیم. به یک مثال توجه کنید:

`لینک متن`

لینکی که در صفحه مشاهده خواهید کرد به صورت زیر است که برای فرستادن ایمیل به آدرس `Ayoob.rashidi@example.com` ساخته شده است. آنرا امتحان کنید:

متن لینک

تنها کاری که شما باید انجام دهید جایگزین کردن `Ayoob.rashidi@example.com` با آدرس ایمیل خودتان است.

اگر روی این نوع لینک کلیک کنید مرورگر شما پنجره ای تازه برای ارسال ایمیل باز می کند که آدرس در یافت کننده در آن تایپ شده است.

شما میتوانید همچنین یک موضوع هم برای ایمیل در لینک قرار دهید تا بیننده مجبور نباشد چیزی غیر از متن اصلی بنویسد.

این کار را می توانید با افزودن علامت سوال به انتهای آدرس ایمیل و نوشتن `subject` بعد از آن انجام دهید. مانند زیر:

متن لینک

با کلیک کردن بر روی این لینک صفحه ای برای فرستادن ایمیل باز می شود که قسمت آدرس (to) و قسمت موضوع (subject) در آن کامل شده است و آماده ارسال است. می توانید آنرا امتحان کنید:

[متن لینک](#)

ساختن لینک پست الکترونیک، قسمت دوم

در بخش اخیر ما در مورد اینکه چگونه یک Subject را برای پیام خود از طریق یک لینک E-mail مشخص کنیم بحث کردیم. ما این کار را با استفاده از این عبارت subject=? و نوشتن موضوعی که می خواهیم در بخش subject در ایمیل نوشته شود بعد از عبارت مذکور انجام می دهیم. برای مثال لینک زیر برای ارسال پیام با موضوع Learn HTML به کار می رود:

متن لینک

این لینک در صفحه مانند زیر به نظر می رسد می توانید آنرا کلیک کرده و امتحان کنید:

[متن لینک](#)

به جز اضافه کردن موضوع شما می توانید چیزهای دیگری را هم اضافه کنید.

برای مثال شما میتوانید یک رونوشت (Carbon Copy) از پیام را به همراه آن برای کس دیگری بفرستید. یا یک نسخه مخفی را برای دیگری بفرستید. یا هر دوی اینها را به کار ببرید و باهم ترکیب کنید و...

برای فرستادن یک رونوشت از ایمیل باید بعد از علامت سوال از فرمان = CC استفاده کرده و بعد آدرس ایمیل کسانی که قرار است رونوشت را دریافت کنند بنویسید. در سطر زیر یک نمونه از این نوع لینک آورده شده است:

لینک متن

می توانید آنرا امتحان کنید:

[متن لینک](#)

برای فرستادن یک نسخه مخفی از ایمیل نیز مانند بالا عمل می کنیم فقط به جای CC در فرمان بالا از bcc استفاده می کنیم. مانند زیر:

لینک متن

لینک به دست آمده مانند زیر است. شما می توانید آنرا امتحان کنید:

[متن لینک](#)

اگر خواستید یک ایمیل را به چند نفر بفرستید باید بین آدرسهای مختلف از کاما استفاده کنید.

مانند زیر:

```
<a href="mailto:main@example.com?bcc=other1@example.com, other2@example.com">متن لینک</a>
```

لینک زیر با دستور بالا ساخته شده است< آنرا امتحان کنید:

[متن لینک](#)

اگر می خواهید بیش از یکی از این فرمانها را در یک لینک استفاده کنید باید برای فرمان اول از علامت سوال استفاده کنید و برای فرمانهای بعد از علامت & استفاده می شود (در انتهای فرمان اول). برای مثال اگر می خواهید با یک لینک هم موضوع پیام را مشخص کنید هم یک رونوشت از آن برای فرد دیگری فرستاده شود باید به اینگونه عمل کنید:

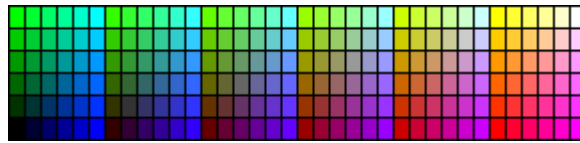
```
<a href="mailto:main@example.com?subject=learn html&cc=other@example.com">متن لینک</a>
```

برای امتحان کردن لینک زیر را کلیک کنید:

[متن لینک](#)

معادل هگزادسیمال رنگهای HTML

جدول هگزادسیمال رنگها



نشان دادن معادل هگزادسیمال هر رنگ بر روی آن کلیک کنید برای
شود مورد نظر خود در جدول بالا، مقدار هگزا دسیمال آن در کادر زیر نشان داده می‌شود از کلیک کردن رنگ

معرفی فرم در HTML و شناسه های مهم آن

اگر در صفحه بخواهیم از بازدید کننده اطلاعاتی را دریافت کنیم باید از فرم برای این کار استفاده کنیم. فرم برای انتقال اطلاعاتی که توسط کاربر وارد شده اند به یک آدرس مشخص استفاده می شود. هر فرم می تواند شامل اجزایی مانند کادر ورودی متن (`Field Text`)، جعبه مربوط به ورود متن (`Text Area`)، دکمه های رادیویی (`Radio Button`)، منوهای کشویی، چک باکس و ... باشد.

در بخشهای بعدی به توضیح اجزایی که می توانند در یک فرم به کار روند خواهیم پرداخت. در این قسمت به تگ

`<form>` و شناسه هایی که در آن به کار می روند می پردازیم. شکل کلی این تگ معمولاً به صورت زیر است :

```
<form action="formProcessor.php" method="get"
name="formName" target="_blank">
```

..... اجزای مورد استفاده برای فرم.....

```
</form>
```

از میان شناسه های مورد استفاده در فرم استفاده از شناسه `action` ضروری است. این شناسه مشخص کننده صفحه ای است که اطلاعات فرم باید به آن ارسال شود. آدرس صفحه مورد نظر باید به عنوان مقدار این شناسه قرار گیرد. لازم به یادآوری است که پس از اینکه کاربر اطلاعات مورد نظر را وارد کرد و دکمه ارسال را کلیک کرد صفحه `action` برای پردازش اطلاعات باز

خواهد شد.

البته استفاده از شناسه **method** هم اهمیت زیادی دارد ولی در صورتی که از این شناسه

استفاده نشود از مقدار پیش فرض آن یعنی **get** استفاده می شود.

در اینجا لیستی از شناسه های پر استفاده در تگ فرم آورده شده است :

:Action

همانطور که گفته شد از این شناسه برای مشخص کردن صفحه پردازش کننده اطلاعات ورودی

کاربر استفاده می شود.

: method

این شناسه طریقه ارسال اطلاعات را به فایل **action** مشخص می کند و شامل دو مقدار زیر

است:

get : در این روش اطلاعات از طریق آدرس صفحه (نوار آدرس) ارسال می شوند. به این صورت

که اطلاعات وارد شده در فرم پس از یک علامت سؤال بعد از آدرس صفحه **action** قرار می

گیرند و ارسال می شوند. از این روش نمی توان برای اطلاعات طولانی یا کاراکترهای غیر ASCII

استفاده کرد. همچنین بهتر است برای اطلاعات مهم از قبیل نام کاربری و کلمه عبور هم از این

روش استفاده نشود. چون در این صورت این اطلاعات در نوار آدرس مرورگر نمایش داده می شوند.

post : این روش برای ارسال اطلاعات از آدرس صفحه استفاده نمی کند. مزیت این روش نسبت

به روش قبیل امنیت بیشتر به دلیل عدم نمایش اطلاعات ارسالی به کاربر و همچنین امکان ارسال

اطلاعات طولانی تر است.

:Name

از این شناسه برای تعریف کردن یک نام منحصر به فرد برای فرم استفاده می شود.

: target

این شناسه روش باز شدن صفحه پردازشگر اطلاعات را مشخص میکند. می توان برای مقدار این شناسه از نام یک فریم برای باز شدن صفحه **action** در آن فریم استفاده کرد یا از مقادیر از پیش تعریف شده زیر:

_blank : صفحه **action** در یک پنجره جدید باز خواهد شد.

_self : صفحه **action** در فریمی که فرم در آن قرار دارد باز خواهد شد.

_parent : صفحه **action** در **frameset** محتوی فرم باز خواهد شد.

_top : صفحه **action** در تما پنجره باز خواهد شد.

علاوه بر شناسه های بالا که فقط مربوط به فرم هستند از شناسه های استاندارد زیر نیز می توان در تگ فرم استفاده کرد:

lang, xml:lang ,id, class, title, style, dir

رویدادهای زیر هم در تگ فرم قابل استفاده هستند.

onclick, ondblclick, onmousedown, onmouseup, ,onsubmit, onreset

onmouseout, onkeypress, onkeydown, onkeyup ,onmouseover, onmousemove

ساختن کادرهای ورودی متن برای استفاده در فرمهای HTML

ما با استفاده از تگ `<input>` می توانیم بسیاری از اطلاعاتی را که در فرمها نیاز داریم از طریق کاربران به دست آوریم. در حقیقت این تگ برای وارد کردن اطلاعات توسط کاربر استفاده می شود. این تگ در HTML4.01 نیازی به تگ پایانی ندارد.

از تگ `<input>` می توانیم برای ساختن کادرهای مربوط به وارد کردن متن، کادر کلمه عبور، فیلدهای مخفی، دکمه های رادیو، چک باکس، دکمه ارسال فرم، دکمه پاک کردن فرم و انواع دیگر دکمه ها و چند نوع ورودی دیگر استفاده کنیم. در اینجا هر کدام از این ورودی ها را به صورت جداگانه مورد بررسی قرار می دهیم.

جعبه ورودی متن (Text Box)

از این نوع ورودی برای ایجاد یک کادر یک سطری برای ورود متن استفاده می شود. برای ساختن این کادر باید از شناسه `type="text"` در تگ `<input>` استفاده کنیم. در صورت تعریف نشدن شناسه `type` در مرورگرهای جدید این شناسه به صورت پیش فرض به کار می رود و یک Text Box را نمایش می دهد اما برخی از مرورگرهای قدیمی در صورتی که از ایسن شناسه استفاده نشود این کادر را نمایش نمی دهند.

یک نمونه از کد مربوط به ایجاد این کادر را می توانید در اینجا مشاهده کنید:

```
<input type="text" name="testField" maxlength="50" size="30" value="کادر ورود متن">
```

کادر متنی بالا به صورت زیر نمایش داده می شود:

کادر ورود متن

در اینجا به توضیح برخی از شناسه‌هایی که با این نوع ورودی به کار می‌روند می‌پردازیم. به خاطر داشته باشید که همه این شناسه‌ها باید در تگ `<input>` به کار روند.

: type

برای اینکه یک کادر متنی ایجاد کنیم باید مقدار این شناسه را **text** قرار دهیم.

: name

این شناسه برای اختصاص یک نام برای ورودی مربوطه استفاده می‌شود.

: maxlength

این شناسه برای مشخص کردن حداکثر تعداد کاراکترهای ورودی استفاده می‌شود. این شناسه در مورد ورودی متنی یک سطری استفاده می‌شود و مقدار آن باید عددی مثبت باشد.

: size

این شناسه در مورد این نوع ورودی طول کادر را بر حسب تعداد کاراکتر مشخص می‌کند. البته مرورگرهای مختلف ممکن است بر حسب نوع فونت مورد استفاده اندازه این کادر را مانند هم نشان ندهند. مقدار این شناسه باید عددی مثبت باشد. مقدار پیش فرض این شناسه 20 است.

: value

این شناسه در مورد کادرهای متنی، متنی را که به صورت پیش فرض در کادر متنی نمایش داده می‌شود معین می‌کند.

: readonly="readonly"

این شناسه فقط برای کادرهای متنی مورد استفاده قرار می گیرد و باعث می شود تا کاربر نتواند تغییری در متن نوشته شده در این کادر ایجاد کند اما کاربر می تواند متن را مشاهده کند و آنرا انتخاب کند.

مقادیر مربوط به فیلدهایی که به صورت readonly هستند اگرچه توسط کاربر تغییر نمی کنند ولی با ارسال فرم به سرور ارسال می شوند.

: disabled="disabled"

از این شناسه برای غیر فعال کردن کادر متنی استفاده می شود. به صورتی که کاربر نمی تواند متنی در این کادر بنویسد و یا متن آنرا انتخاب کند. البته این رفتار ممکن است در مرورگرهای مختلف تفاوت داشته باشد.

مقادیر مربوط به فیلدهایی که به صورت disabled هستند با ارسال فرم به سرور ارسال نمی شوند.

برای مشاهده کادر در حالات فقط خواندنی (readonly) و غیر فعال (disabled) حالت مورد نظر خود را در مثال زیر انتخاب کنید و نتیجه را در کادر متنی زیر مشاهده کنید:
حالت مورد نظر شما :

فقط خواندنی غیر فعال

کادر متنی در حالت فقط خواندنی

کادر متنی نمونه :

علاوه بر شناسه های بالا که فقط مربوط به فرم هستند از شناسه های استاندارد زیر نیز می توان در تگ فرم استفاده کرد:

lang, xml:lang, id, class, title, style, dir

رویدادهای زیر هم در تگ فرم قابل استفاده هستند.

onselect, onchange, onclick, ondblclick, onfocus, onblur
onmousemove, onmousedown, onmouseup, onmouseover
onmouseout, onkeypress, onkeydown, onkeyup

آموزش نحوه ساختن چک باکس برای استفاده در فرمهای html

چک باکس (checkbox)

برای ساختن چک باکس هم باید از تگ **<input>** استفاده کنیم. برای این کار باید از

checkbox به عنوان مقدار شناسه **type** استفاده کنیم. در اینجا یک نمونه از کدی که برای

ساختن چک باکس آورده شده است:

<input name="someNeme" type="checkbox" value="نظر مقدار مورد">

در اینجا می توانید چک باکس ایجاد شده را مشاهده کنید:



در اینجا به توضیح برخی از شناسه هایی که برای چک باکس به کار می روند می پردازیم. به

خاطر داشته باشید که همه این شناسه ها باید در تگ **<input>** به کار روند.

: type

برای اینکه چک باکس متنی ایجاد کنیم باید مقدار این شناسه را **checkbox** قرار دهیم.

: name

این شناسه برای اختصاص یک نام برای ورودی مربوطه استفاده می شود.

: value

این شناسه در مورد در مورد چک باکس مقداری است که در صورت انتخاب شدن این دکمه باید به صفحه **action** انتقال داده شود. استفاده از این شناسه در مورد چک باکس و دکمه های رادیو ضروری است.

: checked="checked"

این شناسه فقط برای چک باکس و دکمه های رادیو مورد استفاده قرار می گیرد و باعث می شود تا در هنگام باز شدن صفحه این دکمه ها به صورت انتخاب شده نمایش داده شوند.

: disabled="disabled"

از این شناسه برای غیر فعال چک باکس استفاده می شود. به صورتی که کاربر نمی تواند دکمه مربوطه را انتخاب کند یا آنرا به صورت انتخاب نشده در آورد. مقادیر مربوط به فیلدهایی که به صورت **disabled** هستند با ارسال فرم به سرور ارسال نمی شوند. برای مشاهده چک باکس در حالت غیر فعال (**disabled**) حالت مورد نظر خود را در مثال زیر انتخاب کنید و نتیجه را در چک باکس زیر مشاهده کنید:

حالت مورد نظر شما : فعال غیر فعال

چک باکس نمونه :

آموزش نحوه ساختن دکمه های رادیو در فرمهای html

دکمه های رادیو (Radio Button)

برای ساختن دکمه های رادیو هم باید از تگ **<input>** استفاده کنیم. برای این کار باید از

radio به عنوان مقدار شناسه **type** استفاده کنیم. در اینجا یک نمونه از کدی که برای

ساختن دکمه رادیو آورده شده است:

```
<input name="someNeme" type="radio" value="نظر مقدار مورد">
```

در اینجا می توانید دکمه رادیو ایجاد شده را مشاهده کنید:



حتماً تا به حال دکمه های رادیویی را که در یک زمان فقط یکی از آنها می تواند انتخاب شده

باشد مشاهده کرده اید. برای ساختن این نوع دکمه ها از کد بالا استفاده می شود همه دکمه هایی

که باید به این گونه رفتار کنند دارای یک نام هستند اما مقادیر قرار گرفته شده در شناسه **value**

در آنها متفاوت است. در زیر می توانید کد مربوط به یک دسته دکمه رادیو را ببینید:

```
<input name="Neme" type="radio" value="اول مقدار دکمه">Radio Button
```

1


```
<input name="Neme" type="radio" value="مقدار دکمه دوم">Radio Button
```

2


```
<input name="Neme" type="radio" value="مقدار دکمه سوم">Radio Button
```

3

در اینجا هم می توانید این دکمه ها را مشاهده کرده و آنها را آزمایش کنید:



Radio Button 1



Radio Button 2



Radio Button 3

در اینجا به توضیح برخی از شناسه‌هایی که برای دکمه‌های رادیو به کار می‌روند می‌پردازیم. به خاطر داشته باشید که همه این شناسه‌ها باید در تگ `<input>` به کار روند.

: type

برای اینکه یک دکمه رادیو ایجاد کنیم باید مقدار این شناسه را **radio** قرار دهیم.

: name

این شناسه برای اختصاص یک نام برای ورودی مربوطه استفاده می‌شود.

: value

این شناسه در مورد در مورد دکمه رادیو مقداری است که در صورت انتخاب شدن این دکمه باید به صفحه **action** انتقال داده شود. استفاده از این شناسه در مورد چک باکس و دکمه‌های رادیو ضروری است.

: checked="checked"

این شناسه فقط برای چک باکس و دکمه‌های رادیو مورد استفاده قرار می‌گیرد و باعث می‌شود تا در هنگام باز شدن صفحه این دکمه‌ها به صورت انتخاب شده نمایش داده شوند. توجه داشته باشید که در یک دسته دکمه رادیو که در بالا یک نمونه از آن آورده شده است نباید بیش از یک دکمه دارای این شناسه باشد.

disabled="disabled" :

از این شناسه برای غیر فعال دکمه رادیو استفاده می شود. به صورتی که کاربر نمی تواند دکمه مربوطه را انتخاب کند. مقادیر مربوط به فیلدهایی که به صورت disabled هستند با ارسال فرم به سرور ارسال نمی شوند.

برای مشاهده دکمه رادیو در حالت غیر فعال (disabled) حالت مورد نظر خود را در مثال زیر انتخاب کنید و نتیجه را در دکمه رادیوی زیر مشاهده کنید:

حالت مورد نظر شما : فعال غیر فعال

دکمه رادیو نمونه :

ساختن دکمه ها با استفاده از تگ input در html

دکمه ارسال فرم (Submit)

برای ساختن دکمه ارسال فرم باید از تگ `<input>` استفاده کنیم. برای این کار باید از `submit` به عنوان مقدار شناسه `type` استفاده کنیم. در اینجا یک نمونه از کدی که برای ساختن دکمه ارسال فرم آورده شده است:

```
<input name="buttonName" type="submit">
```

در اینجا می توانید دکمه ارسال فرم ایجاد شده را مشاهده کنید البته چون این دکمه در تگ `<form>` قرار ندارد با کلیک کردن بر روی آن اتفاقی رخ نمی دهد اما اگر این دکمه در تگ فرم قرار داشته باشد با کلیک کردن بر روی آن اطلاعات وارد شده در فرم به صفحه `action` ارسال می شوند.

Submit Query

دکمه پاک کردن فرم (Reset)

برای ساختن این دکمه هم از تگ `<input>` استفاده می شود به این صورت که برای شناسه `type` از مقدار `reset` استفاده می کنیم. در اینجا می توانید یک نمونه از کدی را که برای ساختن این دکمه استفاده می شود ملاحظه کنید:

```
<input name="buttonName" type="reset">
```

در اینجا می توانید دکمه ای را که با استفاده از تگ بالا ساخته شده است ملاحظه کنید. این دکمه هم مثل دکمه ارسال فرم باید در تگ فرم قرار داشته باشد تا با کلیک بر روی آن اطلاعاتی که کاربر وارد کرده است حذف شوند و فرم به حالت اولیه بازگردد.



ساختن دکمه (Button)

به جز دکمه های ارسال فرم و پاک کردن فرم دکمه هایی هم وجود دارد که بیشتر برای استفاده در جاوا اسکریپت استفاده می شوند. این دکمه ها به صورت پیش فرض وظیفه خاصی را انجام نمی دهند و وظیفه آنها بستگی به موقعیت و مورد استفاده آنها از نظر برنامه نویسی دارد. در اینجا به توضیح برخی از شناسه هایی که برای دکمه های رادیو به کار می روند می پردازیم. به خاطر داشته باشید که همه این شناسه ها باید در تگ `<input>` به کار روند. برای مثال دکمه هایی که در پایین همین صفحه قرار دارند و برای رفتن به بخش قبلی یا بعدی استفاده می شوند از همین نوع دکمه ها هستند.

از کد زیر برای ساختن این دکمه ها استفاده می شود:

```
<input name="buttonName" type="button">
```

در اینجا به معرفی برخی از شناسه هایی که برای دکمه ها در تگ `<input>` استفاده می شوند می پردازیم :

type :

برای دکمه ارسال فرم باید از submit برای مقدار این شناسه استفاده شود.

برای دکمه پاک کردن فرم باید از `reset` برای مقدار این شناسه استفاده شود.

برای سایر دکمه ها باید از `button` برای مقدار این شناسه استفاده شود.

: name

این شناسه برای اختصاص یک نام به هر دکمه استفاده می شود.

: value

در مورد دکمه ها این شناسه مشخص کننده متنی است که در روی دکمه نمایش داده می شود.

: disabled="disabled"

از این شناسه برای غیر فعال دکمه ها استفاده می شود. به صورتی که کاربر نمی تواند دکمه مربوطه را کلیک کند. مقادیر مربوط به فیلدهایی که به صورت `disabled` هستند با ارسال فرم به سرور ارسال نمی شوند.

برای مشاهده دکمه زیر در حالت غیر فعال (`disabled`) حالت مورد نظر خود را در مثال زیر

انتخاب کنید و نتیجه را در دکمه زیر مشاهده کنید:

حالت مورد نظر شما : فعال غیر فعال

دکمه نمونه :

کادر کلمه عبور و فیلدهای مخفی در طراحی فرمهای HTML

کادر کلمه عبور (Password)

احتمالاً تا به حال به سایتی برخورد کرده اید که برای ورود به آن نیاز به کلمه عبور داشته باشید. شما مسلماً تمایل ندارید وقتی کلمه عبور خود را وارد می کنید کلمه عبور شما نمایش داده شود و دیگران قادر به مشاهده آن باشند. به همین دلیل معمولاً سایتها برای اینکه کلمه عبور کاربران در هنگام پر کردن فرم نمایش داده نشود از نوعی کادر ورودی استفاده می کنند که کاملاً شبیه کادری است که برای وارد کردن متن استفاده می شود (Text Box) با این تفاوت که به جای نمایش کاراکترهای کلمه عبور به جای هر کاراکتر از یک نقطه یا ستاره (بستگی به مرورگر دارد) نمایش می دهد.

برای ساختن کادر کلمه عبور می توانید از کد زیر استفاده کنید:

```
<input type="password" name="someName" size="30">
```

کادر به دست آمده مانند زیر است. می توانید متنی را در آن بنویسید و نتیجه را مشاهده کنید:

همانطور که در کد بالا مشاهده می کنید ما از شناسه **size="30"** برای این کادر استفاده

کردیم. تقریباً همه شناسه هایی که برای کادرهای متنی استفاده می شوند در مورد کادر کلمه

عبور هم قابل استفاده هستند. برای مشاهده این شناسه ها می توانید لینک زیر را کلیک کنید:

فیلدهای مخفی (Hidden Fields)

ممکن است برای شما پیش آمده باشد که بخواهید قسمتی از اطلاعات به همراه فرم ارسال شوند ولی کاربر آنها را مشاهده نکند و نیازی نباشد آنها را وارد کند. برای این کار می توانید از فیلدهای مخفی استفاده کنید. کد زیر مربوط به یک فیلد مخفی است:

```
<input type="hidden" name="FieldName" value=" شما اطلاعات مورد نظر" >
```

در کد بالا شما می توانید مقدار مورد نظر خود را در قالب شناسه **value** به صفحه **action** ارسال کنید بدون اینکه در صفحه نمایش داده شود.

کادر متنی با بیش از یک سطر (Text Area)

کادر های متنی با بیش از یک سطر (Text Area)

از یک **Text Area** کاربران می توانند برای وارد کردن تعداد نامحدودی کاراکتر در بیش از یک سطر وارد کنند. بهتر است تا برای متنهای طولانی به جای **Text Box** از **Text Area** استفاده کنید تا کاربران راحتتر بتوانند متن خود را مشاهده کنند و یا آنرا ویرایش کنند.

برای ایجاد **Text Area** باید از تگ **<textarea>** استفاده کنید. این تگ بر خلاف تگ

<input> یک تگ پایانی دارد و باید به وسیله تگ **</textarea>** بسته شود. در مثال زیر

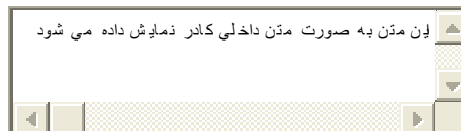
می توانید کد مربوط به یک **Text Area** را مشاهده کنید:

`<textarea name="someName" cols="40" rows="3">`

به صورت متن داخلی کادر نمایش داده می شود این متن

`</textarea>`

می توانید نتیجه کد بالا را در اینجا مشاهده کنید:



همانطور که مشاهده می کنید متنی که بین تگ ابتدایی و تگ پایانی یک `Text Area` نوشته

شود به صورت پیش فرض در کادر `Text Area` نمایش داده می شود.

در اینجا می توانید برخی از شناسه های قابل استفاده در تگ `<textarea>` را مشاهده کنید:

: cols

این شناسه عرض کادر را بر حسب تعداد کاراکتر تعیین می کند. مقدار این شناسه باید عددی

مثبت باشد. البته این عرض بستگی به اندازه فونت تعریف شده برای صفحه و مرورگر دارد.

استفاده از این شناسه برای تگ `<textarea>` الزامی است.

: rows

این شناسه مشخص کننده تعداد سطرهای قابل مشاهده در کادر است. این مقدار هم باید عددی

مثبت باشد. استفاده از شناسه `rows` برای تگ `<textarea>` الزامی است.

: name

از این شناسه می توان برای اختصاص دادن یک نام مشخص به textarea استفاده کرد.

: readonly="readonly"

این شناسه کادر را به صورت فقط خواندنی در می آورد و کاربر نمی تواند متن موجود در کادر را تغییر دهد.

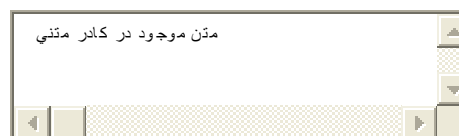
: disabled="disabled"

این شناسه کادر را به صورت غیر فعال در می آورد و بازدید کننده نمی تواند متن موجود در آن را تغییر دهد یا آنرا انتخاب کند. (این رفتار بسته به مرورگر متفاوت است)

برای مشاهده کادر در حالات فقط خواندنی (readonly) و غیر فعال (disabled) حالت مورد نظر خود را در مثال زیر انتخاب کنید و نتیجه را در Text Area زیر مشاهده کنید:

حالت مورد نظر شما : فقط خواندنی غیر فعال

کادر متنی نمونه :

A screenshot of a text area control. The text area contains the Persian text "متن موجود در کادر متنی". The text area has a light gray background and is surrounded by a thin border. On the right side, there are vertical scrollbars, and on the bottom side, there are horizontal scrollbars.

در تگ `<textarea>` علاوه بر شناسه های بالا که مربوط به همین تگ هستند می توان از

شناسه های استاندارد زیر هم استفاده کرد:

`style, dir, lang, xml:lang, tabindex, accesskey, id, class, title`

همچنین از رویدادهای زیر هم می توان در تگ `<textarea>` استفاده کرد :

`,onfocus, onblur, onselect, onchange, onclick, ondblclick`

`onmousedown, onmouseup, onmouseover, onmousemove,`

`onkeydown, onkeyup ,onmouseout, onkeypress`

اصول نوشتن لیستهای کشویی برای استفاده در فرمهای HTML

لیستهای کشویی (Select)

برای ساختن لیستهای کشویی که به کاربران اجازه می دهد یک یا چند گزینه موجود در لیستی

را در یک فرم انتخاب کند باید از تگ `<select>` استفاده کنیم. گزینه های موجود در این تگ

بین `<option>` و `</option>` قرار می گردند و پس از گزینه ها تگ پایانی لیست به

صورت `</select>` نوشته می شود.

در اینجا می توانید یک نمونه از کدی را که با آن یک لیست کشویی ساخته می شود مشاهده

کنید:

```
<select name="someName">
```

```
<option value="value 1" >گزینه اول</option>
```

```
<option value="value 2">گزینه دوم</option>
```

```
<option value="value 3">گزینه سوم</option>  
</select>
```

در این قسمت می توانید لیست مربوط به کد بالا را ملاحظه کنید:

با انتخاب هر یک از گزینه های موجود در لیست و کلیک بر روی دکمه ارسال فرم، مقدار مربوط به آن گزینه به صفحه **action** ارسال می شود.

از شناسه های زیر می توان در تگ **<select>** استفاده کرد:

: name

از این شناسه برای تخصیص یک نام به لیست استفاده می شود.

: size

این شناسه مشخص می کند چه تعدادی از گزینه ها در یک زمان قابل مشاهده باشند. در واقع ارتفاع لیست را بر حسب تعداد گزینه ها مشخص می کند. مقدار این شناسه باید عددی بزرگتر از صفر باشد.

: multiple="multiple"

در صورتی که این شناسه به تگ **<select>** اضافه شود بازدید کننده می تواند بیش از یک گزینه از گزینه های لیست را انتخاب کند.

: disabled="disabled"

این شناسه لیست را برای کاربر غیر فعال می کند و در نتیجه گزینه های آن امکان انتخاب شدن ندارند.

برای مشاهده لیست در حالت غیر فعال (disabled) حالت مورد نظر خود را در مثال زیر انتخاب کنید و نتیجه را مشاهده کنید:

حالت مورد نظر شما : فعال غیر فعال

کادر متنی نمونه :

از رویداد های زیر هم می توانیم در تگ **<select>** استفاده کنیم:

onfocus, onblur, onchange

وارد کردن گزینه ها به لیست با تگ **<option>**

برای ساختن گزینه هایی که در لیست نمایش داده می شود باید از تگ **<option>** استفاده کنیم. با استفاده از مقداری که برای این تگ با استفاده از شناسه **value** تعریف می کنیم می توانیم در اطلاعات ارسال شده فرم گزینه انتخاب شده توسط بازدید کننده را تشخیص دهیم. برای تگ **<option>** هم می توانید از شناسه هایی استفاده کنید که در این جا مشاهده می کنید:

: value

این گزینه مشخص کننده متنی است که در صورت انتخاب شدن گزینه مربوطه به عنوان مقدار لیست کشویی به صفحه **action** ارسال می شود.

: selected="selected"

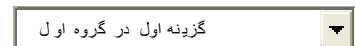
از این گزینه برای مشخص کردن گزینه ای که به هنگام باز کردن صفحه به صورت پیش فرض انتخاب شده است استفاده می شود. در صورتی که برای هیچکدام از گزینه های موجود در لیست از این شناسه استفاده نشده باشد گزینه اول به عنوان گزینه پیش فرض استفاده می شود. استفاده از رویداد های زیر در تگ **<option>** مجاز است:

,onclick, ondblclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress, onkeydown, onkeyup

دسته بندی گزینه های لیست های کشویی با استفاده از **<optgroup>** اگر یک لیست کشویی طولانی دارید توانید برای سهولت کار برای بازدید کننده گزینه های مرتبط با هم را در یک دسته قرار دهید به صورتی که از سایر گزینه ها جدا باشند. برای انجام این کار می توانید از تگ **<optgroup>** استفاده کنید. با استفاده از این تگ می توانید گزینه های مورد نظر خود برای یک دسته را بین **<optgroup>** و **</optgroup>** قرار دهید. کد زیر یک نمونه از لیست کشویی را نشان می دهد که گزینه ها در آن دسته بندی شده اند:

```
<select>
<optgroup label="اول عنوان گروه">
<option value ="some value "> گزینه اول در گروه اول </option>
<option value ="saab"> گزینه دوم در گروه اول </option>
</optgroup>
<optgroup label="دوم عنوان گروه">
<option value ="mercedes"> گزینه اول در گروه دوم </option>
<option value ="audi"> گزینه دوم در گروه دوم </option>
</optgroup>
</select>
```

در اینجا می توانید لیست ساخته شده با کد بالا را مشاهده کنید:



استفاده از شناسه زیر در تگ **<optgroup>** الزامی است :

label :

این شناسه برای هر گروه از گزینه ها یک عنوان یا برچسب مشخص می کند که در لیست در بالای هر گروه نمایش داده می شود و آن گروه را از سایر گزینه ها جدا می کند.

از رویدادهای زیر می توانید در تگ **<optgroup>** استفاده کنید:

onmouseover, ,tabindex, onclick, ondblclick, onmousedown, onmouseup
onmousemove, onmouseout, onkeypress, onkeydown, onkeyup

استفاده از موزیک در صفحه

برای قرار دادن صدا و موزیک در صفحه دو راه وجود دارد: راه اول این است که شما می توانید یک لینک به فایل موزیک ایجاد کنید تا افراد بتوانند آنرا دانلود کنند و سپس به آن گوش دهند. راه دوم جاسازی کردن موزیک در صفحه است.

در ابتدا به اولین روش می پردازیم:

فرض کنید می خواهیم یک فایل صوتی با نام `example.mid` را در صفحه قرار دهیم. در صورتی که فایل در همان پوشه صفحه قرار داشته باشد می توانیم به این صورت عمل کنیم:

```
<a href="example.mid">متن لینک موزیک</a>
```

به این طریق می توان یک لینک به یک فایل موزیک ایجاد کرد. در این مثال ما از یک فایل MIDI استفاده کردیم، همین کار را می توان با فایل‌هایی با پسوند، `wav` و `aud` و اغلب فرمت‌های صوتی انجام داد. کاربر پس از باز شدن صفحه لینکی مشابه زیر را خواهد دید که با کلیک کردن بر روی آن دانلود فایل موزیک شروع می شود.

متن لینک موزیک

احتمالاً پس از اتمام داتلود فایل پنجره جدیدی باز می شود و شروع به پخش فایل می کند. کاربر بسته به `plug-in` هایی که دارد ممکن است توانایی جلو و عقب بردن موزیک و متوقف کردن آنرا نیز داشته باشد. در این حالت کاربر توانایی ذخیره فایل صوتی را مانند یک تصویر دارد. کاربر می تواند پنجره جدید را ببندد و به ادامه بازدیدش از سایت بدون صدای موزیک بپردازد. مشکل این روش این است که کاربر باید لینک موزیک را کلیک کند و صبر کند تا آن دانلود شود.

البته این مسئله در مواقعی که شما آهنگهای زیادی در صفحه خود دارید می تواند به کاربر کمک کند تا آهنگ مورد علاقه خود را پیدا کند.

جاسازی موزیک در صفحه:

این روش برای پخش یک آهنگ در صفحه روش مناسبی است چون به کاربر اختیارات بیشتری نسبت به یک لینک می دهد. در این روش می تان رابط صدا در صفحه را طوری تنظیم کرد که به صورت خودکار شروع به پخش موزیک کند و یا موزیک را تکرار کند یا آنرا فقط یک بار پخش کند. برای اجرای این روش باید از تگ **EMBED** استفاده کنید. این تگ مانند تگ عکس مورد استفاده قرار می گیرد. ابتدا باید منبع صدا را مشخص کنیم و بعد در صورت نیاز به تگ فرمانهایی را بافزاییم.

برای نمونه به تگ زیر نگاه کنید:

```
<"embed src="example.mid">
```

این تگ به مرورگر می گوید تا فایل صوتی را همان جایی که شما تگ را قرار داده اید در صفحه قرار دهد. در این روش نیز شما مجبور نیستید صرفاً از فایلهای **MIDI** استفاده کنید و می توانید از فرمتهای دیگر استفاده کنید. همانطور که می بینید در این تگ فرمان " " **src=** مشخص کننده محل قرار گیری فایل صوتی در سایت شما است. شما می توانید از آدرس کامل فایل در اینجا استفاده کنید یا اگر فایل و صفحه ای که می خواهید فایل در آن شنیده شود در یک پوشه هستند می توانید از نام فایل استفاده کنید.

برای مثال در این روش شما چیزی شبیه این را خواهید دید:

شما می توانید به این تگ فرمانهایی اضافه کنید تا بهتر بتوانید صدای خروجی همچنین ظاهر صفحه را کنترل کنید. در اینجا تعدادی از این فرمانها آورده شده است که می توانید از آنها استفاده کنید:

: width

این فرمان تعیین کننده عرض فایل صوتی در صفحه است. شما با قرار دادن یک عدد در مقیاس پیکسل بین دو دابل کوت این فرمان می توانید تعیین کنید فایل شما با چه میزان عرض در صفحه به نمایش درآید.

: height

این فرمان تعیین کننده ارتفاع فایل صوتی در صفحه است. شما با قرار دادن یک عدد در مقیاس پیکسل بین دو دابل کوت این فرمان می توانید تعیین کنید فایل شما با چه میزان ارتفاع در صفحه به نمایش درآید. معمولاً برای این فرمان از ارتفاع 60 پیکسل استفاده می شود.

: autostart="true"

مرورگر را وادار می کند تا بلافاصله پس از بارگذاری صفحه شروع به پخش فایل صوتی کند. شما می توانید این گزینه را با قرار دادن مقدار false به جای true غیر فعال کنید. معمولاً این گزینه به طور پیش فرض غیر فعال است.

: loop="true"

این فرمان مرورگر را وادار می کند پس از به اتمام رسیدن فایل دوباره از اول آنرا اجرا کند، و این عمل تا هنگامی که صفحه باز است یا کاربر دکمه توقف را کلیک نکرده است ادامه پیدا می کند. شما می توانید با قرار دادن **false** به جای **true** این گزینه را غیر فعال کنید. البته معمولاً این گزینه به صورت پیش فرض غیر فعال است.

: hidden="true"

این فرمان به مرورگر دستور مخفی کردن رابط گرافیکی پخش صدا را می دهد. در این صورت کاربران نمی توانند برنامه ای را که صدا را پخش می کند و همچنین دکمه های کنترل آنرا ببینند. در این صورت صدا تا زمانی که به پایان نرسیده یا تا زمانی که تنظیم شده باشد پخش می شود و کاربر نمی تواند از پخش آن جلوگیری کند. این گزینه با قرار دادن **false** به جای **true** غیر فعال می شود. به صورت پیش فرض هم این گزینه غیر فعال است. اگر می خواهید نمونه ای از یک فایل صوتی را که با استفاده تمام فرمانهای بالا در صفحه قرار داده شده ببینید لینک زیر را کلیک کنید:

[ورود به صفحه جدید برای پخش موزیک](#)

این هم تگ سازنده مثال بالا:

```
<embed src="example.mid" width="300" height="60"  
autostart="true" loop="true" hidden="true">
```

آشنایی با متا تگها و موارد استفاده آنها در HTML

تگهای META مورد استفاده موتورهای جستجو برای بررسی و شاخص گذاری صفحات قرار می گیرند. مخصوصاً اگر صفحه شما فریم (قاب) داشته باشد. این تگها باید در قسمت HEAD صفحه شما قرار گیرند.

در سطر زیر یکی از معمول ترین تگهای META را می بینید:

```
<head>
```

```
<title>عنوان صفحه</title>
```

```
<" "=meta name=" " content>
```

```
</head>
```

شناسه **name** :

مشخص کننده نوع متا تگی است که شما قصد استفاده از آن را دارید.

شناسه **content** :

تعیین کننده محتویات متا تگ برای موتورهای جستجو است.

در این بخش به توضیح دو متا تگ که برای موتورهای جستجو اهمیت بیشتری دارند می پردازیم.

META Keywords

```
<" "=meta name="keywords" content>
```

name="keywords"

این فرمان برنامه اسپایدر موتورهای جستجو را از وجود یک دسته از کلمات کلیدی این صفحه در این تگ آگاه می کند.

content="موتا تگ، موتور جستجو"

این قسمت لیستی از کلمات کلیدی موجود در این صفحه است. در این قسمت برای جدا کردن کلید واژه ها از یکدیگر از علامت کاما استفاده می شود. توجه داشته باشید که نباید یک کلید واژه را چندین بار تکرار کنید و کلی واژه ها باید با متن شما در صفحه مربوطه ارتباط باشند. موتورهای جستجو معمولاً رتبه بندی صفحاتی را که در آنها با تکرار بیش از حد یک کلید واژه یا ذکر کلید واژه ای که با متن صفحه ارتباطی ندارد سعی می شود رتبه بالایی را به دست بیاورند، متوقف می کنند.

META Description

این تگ محتویات صفحه را توصیف می کند.

<meta name="description" content="افزودن متا تگ به صفحه">

name="description"

این بخش توضیحی در باره صفحه برای برنامه اسپایدر موتورهای جستجو است.

"افزودن متا تگ به صفحه"=content

این بخش توصیفی است که شما از صفحه خود می نویسید. در این قسمت هر چیزی را که می خواهید درباره صفحه خود توضیح دهید، همچنین می توانید از یک دو تا از کلید واژه ها هم استفاده کنید. بسیاری از موتورهای جستجو این قسمت را در لیست نتایج جستجو به عنوان توضیح صفحه نشان می دهند.

دو متا تگ ذکر شده در بالا از بقیه متا تگها پر کاربرد تر هستند. متا تگ های دیگری هم وجود دارند که در زیر به برخی از آنها اشاره می شود:

<meta name="author" content="نام نویسنده صفحه">

این تگ به شما اجازه می دهد که نویسنده صفحه را مشخص کنید. فقط نام نویسنده را در این تگ بنویسید تا هر کسی که کدهای HTML این صفحه را می بیند، بداند چه کسی این صفحه را نوشته است. موتور های جستجو معمولاً از این تگ برای رتبه بندی صفحات استفاده نمی کنند.

<meta name="generator" content="نام برنامه مورد استفاده شما">

این تگ به شما اجازه می دهد که به افرادی که کدهای این صفحه را می بینند بگویید که شما از چه برنامه ای برای نوشتن این کدها استفاده کرده اید. این تگ نیز برای رتبه بندی صفحات در موتورهای جستجو استفاده نمی شود.

تغییر مسیر بازدید کنندگان پس از مدت مشخص با HTML

اگر صفحه شما جایجا شد یا به هر دلیلی خواستید مسیر بازدید کنندگان خود را تغییر دهید می توانید از تگ `meta refresh` برای بردن بازدید کنندگان به صفحه مورد نظرتان استفاده کنید. البته در استفاده از این روش برای تغییر مسیر بازدید کنندگان، مخصوصاً در صفحه اصلی سایت خود باید به این نکته توجه کنید که برخی از موتورهای جستجو به دنبال تگهای `meta refresh` می گردند که بازدید کنندگان را از یک صفحه که با کلمات کلیدی پر شده است، به صفحه واقعی می برند که ممکن است در آن از کلمات کلیدی استفاده نشده باشد و در نتیجه با موضوع مورد نظر بازدید کننده مرتبط نباشد. بنابراین تغییر مسیر بازدید کننده از یک صفحه با سرعت زیاد (معمولاً کمتر از 10 ثانیه) ممکن است باعث شود این صفحه در لیست موتور جستجو قرار نگیرد. البته اگر این صفحه صفحه اصلی سایت شما نباشد ممکن است نخواهید در لیست موتور جستجو قرار گیرد.

بسیار خوب، کد زیر را می توان برای تغییر مسیر بازدید کننده استفاده کرد:

```
<html>
```

```
<head>
```

```
"http-equiv="refresh meta>
```

```
<"content="3;url=http://www.yoursite.com/destinationpage.htm
```

```
<title/>این صفحه جا به جا شده است</title>
```

```
</head>
```

<body>

متن مورد نظر شما

</body>

</html>

برای مشاهده نتیجه، لینک زیر را کلیک کنید:

[ورود به صفحه آزمایشی](#)

این نکته را مد نظر داشته باشید که تگ **Meta refresh** قبل از عنوان صفحه (**TITLE**) نوشته می شود.

فرمان `content=""` دو کار را انجام می دهد. اول مشخص می کند که مرورگر قیل از تغییر مسیر چند ثانیه باید صبر کند. دوم آدرس صفحه ای که باید مسیر به آن آدرس تغییر یابد. به این نکته توجه کنید که هر دو قسمت زمان و آدرس صفحه در یک جفت دابل کوت قرار دارند نه دو جفت. برای جدا کردن این دو قسمت باید از نقطه ویرگول (;) استفاده کرد.

در بخش **BODY** هم ر چیزی که بخواهید می توانید بنویسید. همچنین می توانید برای کاربرانی که از مرورگرهای قدیمی استفاده می کنند یک لینک را قرار دهید تا به صورت دستی به صفحه جدید وارد شوند.

```
<html>
<head>
<meta http-equiv="refresh"
content="3;url=http://www.yoursite.com/destinationpage.htm">
<title>صفحه جا به جا شده است این</title>
</head>
<body>
<a
جا به جا شده است. اگر مرورگر شما به صورت اتوماتیک به صفحه جدید نرفت این صفحه
href="http://www.yoursite.com/destinationpage.htm">اینجا</a>
را
شوید کلیک کنید تا به صورت دستی به صفحه جدید وارد
</body>
</html>
```

شناسه های مختلف قابل استفاده برای خط افقی

فرض کنید می خواهیم از یک خط افقی در صفحه خود استفاده کنیم، اما می خواهیم طول آن برابر یک تصویر یا قسمتی از یک متن باشد. یا شاید به یک خط افقی کوتاه که در سمت چپ یا راست صفحه قرار بگیرد، نیاز داشته باشیم. اینها خواصی هستند که ما می توانیم برای تگ **HR** یعنی همان خط افقی استفاده کنیم:

:Width

این فرمان مقدار عرض خط افقی را مشخص می کند. ر این فرمان شما می توانید از مقیاس پیکسل یا درصد استفاده کنید. گزینه پیش فرض برای این فرمان 100٪ است.

: align="left"

این فرمان سمت قرار گرفتن خط افقی را مشخص می کند. شما می توانید خط را به صورت چپ چین، راست چین و در مرکز صفحه قرار دهید. گزینه پیش فرض این فرمان center است که خط را در مرکز صفحه قرار می دهد. البته زمانی این وضعیت مشخص می شود که عرض خط افقی کمتر از عرض صفحه باشد.

: size="3"

این فرمان اندازه خط را مشخص می کند. اندازه پیش فرض برای خط 2 است.

: noshade="noshade"

این فرمان خط را به صورت تو پر نشان می دهد.

در اینجا هر کدام از فرمانهای بالا را با یک مثال بیشتر توضیح می دهیم.

<hr width="250">

با این فرمان خطی افقی مانند زیر خواهید داشت:

`<hr width="250" align="right">`

با این تگ یک خط افقی با عرض 250 پیکسل و در سمت راست صفحه خواهید داشت. مانند

زیر:

`<hr width="250" size="10">`

با این تگ یک خط افقی با عرض 250 پیکسل و پهنای 10 خواهید داشت:



حالا یک اندازه بزرگ را برای پهنای خط انتخاب می کنیم، مثلاً 80 ، نتیجه به این صورت

خواهید بود:



همانطور که ملاحظه می کنید، خط به صورت تو خالی و مانند یک حباب به نظر می رسد.

حالا اگر از فرمان noshade استفاده کنید، خط به صورت تو پر نمایش داده می شود:

```
<hr width="250" noshade="noshade">
```



این بار با اندازه 80 این تگ را امتحان می کنیم:

```
<hr width="250" size="80" noshade>
```



در برخی مرورگرها مانند اینترنت 4.0 Internet Explorer به بعد و Mozilla Firefox

امکان تغییر رنگ خط نیز وجود دارد. اما ممکن است این رنگ در همه مرورگرها مثلاً Opera

9.0 قابل مشاهده نباشد و خط به رنگ پیش فرض مشاهده شود.

در این قسمت اگر با مرورگری مثل اینترنت اکسپلورر صفحه را مشاهده می کنید، می توانید

خطی را به رنگ زرد مشاهده کنید اما اگر با مرورگری مثل Opera این صفحه را مشاهده می

کنید، خط زیر را به رنگ خاکستری (رنگ پیش فرض) خواهید دید.

```
<hr width="250" size="20" color="yellow" noshade>
```

تنظیم قلم مورد استفاده برای متن در HTML

اگر تمایل دارید شکل فونت متن خود را در یک صفحه تغییر دهید، می توانید این کار را با استفاده از تگ `font` انجام دهید. (این روش تأثیری بر تصاویر موجود در صفحه شما ندارد). با افزودن این شناسه: `face=" "` به تگ فونت می توانید شکل آنرا تغییر دهید. به یک مثال توجه کنید:

```
<font face="Times New Roman">This line is Written by Times  
New Roman font face</font>
```

این هم متن نوشته شده با فونت Times New Roman:

This line is Writen by Times New Roman font face

البته نکاتی هست که باید به آنها توجه کنید:

افرادی که صفحه شما را تماشا می کنند برای اینکه بتوانند صفحه را با فونتی که شما برای آن تنظیم کرده اید ببینند باید حداقل مرورگر اینترنت اکسپلورر یا نت اسکپ 3 به بالا داشته باشند. (زیاد نگران این قسمت نباشید، اکنون تقریباً تمامی کاربران مرورگرهای جدیدتری دارند) افراد بیننده صفحه شما باید فونت مورد استفاده شما را قبلاً نصب کرده باشند تا بتوانند صفحه شما را به خوبی مشاهده کنند.

وقتی شما در صفحه خود می خواهید از یک فونت خاص استفاده کنید مراقب باشید و فونتی را استفاده کنید که بیشتر افراد به صورت پیش فرض داشته باشند. برای مثال وقتی ویندوز XP را

نصب می کنید چندین فونت به صورت خودکار بر روی کامپیوتر شما نصب می شوند. مثلاً

Tahoma, Arial, Times New Roman و چندین فونت دیگر.

قابل ذکر است که اگر بیننده شما فونت مورد استفاده شما را نداشته باشد مرورگر به طور خودکار

از فونت پیش فرض خود که به طور معمول Times New Roman است استفاده می کند.

در اینجا لیستی از فونتهایی که زیاد مورد استفاده قرار می گیرد آورده شده است که می توانید

با شناسه " " face= استفاده کنید:

| | | |
|-----------------|--------------------|---------------------------|
| Arial | Arial | متن نمونه برای زبان فارسی |
| Tahoma | Tahoma | متن نمونه برای زبان فارسی |
| Times New Roman | Roman Times New | متن نمونه برای زبان فارسی |
| Arial Black | Arial Black | متن نمونه برای زبان فارسی |
| Courier | Courier | متن نمونه برای زبان فارسی |
| Courier New | Courier New | متن نمونه برای زبان فارسی |
| Modern | Modern | متن نمونه برای زبان فارسی |
| Terminal | Terminal | متن نمونه برای زبان فارسی |
| Symbol | Σψμβολ | متن نمونه برای زبان فارسی |

کارایی دارد ولی HTML4.01 < font باید توجه داشته باشید که اگرچه تگ > استفاده کنید. CSS استفاده از این تگ توصیه نمی شود. بهتر است برای تنظیم فونت متن خود از لینک زیر را کلیک کنید (لینک در یک CSS برای یادگیری نحوه تنظیم فونت متن با استفاده از پنجره جدید باز می شود):

مشاهده صفحات لینک شده در پنجره جدید

برای اینکه یک لینک را وادار کنیم که در یک صفحه جدید باز شود، باید این شناسه را به تگ

لینک اضافه کنیم: target="_blank"

تگ لینک به این صورت به نظر می رسد:

[آموزش](http://www.neopersia.org/)

[طراحی صفحات وب](#)

لینک شما در صفحه نهایی به صورت زیر مشاهده می شود. می توانید روی آن کلیک کنید و

مطمئن این صفحه را از دست نخواهید داد، چون لینک در یک پنجره جدید باز خواهد شد.

[آموزش CSS](#)

شما می توانید از این روش برای باز کردن تصاویر در یک پنجره جدید استفاده کنید.

به این منظور می توانید به صورت زیر عمل کنید:

```
<a href="http://learn_html.gif" target="_blank">
```

```

```

```
</a>
```

در اینجا یک تصویر داریم که با کلیک کردن بر روی آن تصویر در پنجره جدید باز می شود.

می توانید آنرا امتحان کنید:



شناسه **target** مقادیر دیگری هم می تواند داشته باشد. برای مثال از **_top** برای باز کردن

لینک در تمام پنجره استفاده می شود. البته در صورتی که صفحه در فریم استفاده نشده باشد این

_top تأثیر و کاربردی ندارد بیشتر کارایی آن هنگامی است که صفحه در یک فرم قرار دارد و

شما نمی خواهید لینک در همان فرم باز شود.

البته وقتی صفحه در فریم قرار دارد، در صورتی که سایر فریمهای پنجره نام گذاری شده باشند

می توان از نام آن فریمها در شناسه **target** استفاده کرد تا لینک در آن فریمی که نام آنرا به کار

برده اید باز شود نه در فریمی که خود آن قرار دارد.

ایجاد متن های متحرک در صفحه HTML

تگ `<marquee>` به شما اجازه می دهد که یک متن را به دلخواه خود در صفحه به حرکت در آورید. برای انجام این کار به جاوا اسکریپت نیاز ندارید. تنها مشکل این است که مرورگرهایی که بر پایه Netscape ساخته شده اند، نمی توانند به طور کامل با این تگ سازگار شوند و در پاره ای از اوقات با آن مشکل پیدا می کنند و تنها متن بدون حرکت را نشان می دهند.

در اینجا می توانید کدی را که برای ساخت متن متحرک ستفاده می شود مشاهده کنید:

```
<marquee loop="infinite">متن مورد نظر خود را اینجا بنویسید</marquee>
```

می خواهید ببینید چگونه کار می کند:

شناسه **loop="infinte"** باعث می شود که متن تا هنگامی که صفحه باز است به حرکت

خود ادامه دهد. شما می توانید به جای این شناسه از **loop="-1"** هم استفاده کنید.

شما می توانید تغییراتی را در این تگ ایجاد کنید به جای `infinite` در شناسه `loop` می توانید

از اعداد استفاده کنید. برای مثال `loop="2"` به مرورگر می گوید که حرکت متن را پس از دو

بار چرخش متوقف کند.

می توانید با وارد کردن شناسه `bgcolor=" "` به شما امکان این را می دهد که رنگ زمینه

محلّی را که متن در آن در حال چرخش است تغییر دهید. می توانید برای تغییر رنگ زمینه از نام

رنگ یا معادل هگزادسیمال آنها استفاده کنید. به یک مثال توجه کنید:

```
<marquee loop="infinite" bgcolor="yellow">
```

متن مورد نظر خود را اینجا بنویسید

```
</marquee>
```

و این هم نتیجه این تگ:

با اضافه کردن شناسه **width=" "** می توانید عرض متن متحرک خود را تعیین کنید. در این

شناسه می توانید از مقیاس پیکسل و یا در صد استفاده کنید.

```
<marquee loop="infinite" bgcolor="yellow" width="400">
```

متن مورد نظر خود را اینجا بنویسید

```
</marquee>
```

با تگ بالا می توانید یک متن متحرک با عرض 400 پیکسل و رنگ زمینه زرد ایجاد کنید. مانند زیر:

```
<marquee loop="infinite" bgcolor="yellow" width"%50" = >
```

متن مورد نظر خود را اینجا بنویسید

```
</marquee>
```

با تگ بالا می توانید متنی متحرک با 50 در صد عرض پنجره مرورگر و با رنگ زمینه زرد ایجاد

کنید. مانند زیر:

آموزش ساختن جدول در طراحی صفحه با HTML

آیا شما می خواهید یک جدول (table) در صفحه خود ایجاد کنید؟ برای ایجاد یک جدول باید از این تگ استفاده کنید:

`<table>`

.....جدول محتویات.....

`</table>`

برای ایجاد یک جدول باید تگ ابتدایی آن یعنی `<table>` را در جایی که می خواهید جدول وارد شود، قرار دهید. و در پایان جدول هم از تگ پایانی آن یعنی `</table>` استفاده کنید.

برای افزودن محتویات به جدول باید از تگ `<td>` استفاده کنید. نام این تگ مخفف `table data` است. محتویات جدول چیزهایی هستند که شما بعد از این تگ قرار می دهید. در پایان هم باید پس از محتویات جدول از تگ پایانی `</td>` قبل از تگ پایانی جدول استفاده کنید. مانند

زیر:

`<table>`

`<td>`

.....جدول محتویات.....

`</td>`

`</table>`

جدول به دست آمده مانند زیر به نظر می رسد:

.....محتویات جدول.....

البته این یک جدول ساده است و شاید در ظاهر زیاد تفاوتی با یک نوشته ساده نداشته باشد.
می توانیم برای جدول کادر (border) هم تعریف کنیم. برای این کار باید شناسه مربوطه را در تگ جدول قرار دهیم. مانند زیر:

```
<table border="3">
```

```
<td>
```

.....جدول محتویات.....

```
</td>
```

```
</table>
```

حالا جدول ما کمی مشخص تر شد، به آن نگاه کنید:

| |
|------------------------|
|محتویات جدول..... |
|------------------------|

شما می توانید ضخامت کادر جدول را به دلخواه خود تغییر دهید. برای این کار فقط کافی است تا عددی را که به عنوان مقدار شناسه border به کار بردیم، تغییر دهید. اگر مقدار این عدد را صفر قرار دهید کادر دیده نخواهد شد.

تا اینجا هر چه گفتیم درباره جدولی با یک سلول بود. برای اینکه جدولی با تعداد بیشتری سلول داشته باشیم، مثلاً بخواهیم یک سلول دیگر در همان خط ایجاد کنیم، باید دوباره از تگ td استفاده کنیم. مانند زیر:

<table border="3">

<td>

اول سلول

</td>

<td>

دوم سلول

</td>

</table>

می توانید سلولهای ایجاد شده را در اینجا مشاهده کنید:

| | |
|----------|----------|
| سلول اول | سلول دوم |
|----------|----------|

حالا فرض کنید می خواهیم به خط بعدی یا به اصطلاح به ردیف بعدی برویم. برای این کار باید از

تگ **tr** استفاده کنیم. این تگ با **<tr>** شروع و با **</tr>** پایان می پذیرد. مانند زیر:

<table border="3">

<td>

اول سطر اول و سلول

</td>

<td>

دوم سطر اول و سلول

</td>

<tr>

<td>

اول سطر دوم و سلول

</td>

<td>

دوم سطر دوم و سلول

</td>

</tr>

</table>

نتیجه به این صورت مشاهده می شود:

| | |
|--------------------|--------------------|
| سطر اول و سلول اول | سطر اول و سلول دوم |
| سطر دوم و سلول اول | سطر دوم و سلول دوم |

شما می توانید بنا به نیاز خود فرمانهایی را به تگ جدول اضافه کنید. در اینجا برخی از این

فرمانها آورده شده است:

:Cellspacing

برای ایجاد فاصله بین سلولها عدد مورد نظر خود را در این فرمان وارد کنید.

:Ellpadding

از این فرمان برای افزودن فضا به داخل یک سلول استفاده کنید. مقدار فضای مورد نظر خود را در این فرمان وارد کنید.

فرض کنید می خواهیم در مثال قبل بین سلولها یک فاصله به اندازه 10 ایجاد کنیم. تگ جدول را مطابق زیر تغییر می دهیم:

```
<table border="3" cellspacing="10">
```

جدول به دست آمده به شکل زیر خواهد بود:

| | |
|--------------------|--------------------|
| سطر اول و سلول اول | سطر اول و سلول دوم |
| سطر دوم و سلول اول | سطر دوم و سلول دوم |

حالا فرض کنید می خواهیم یک فاصله به اندازه 10 را به داخل سلولهای جدول مثال قبل اضافه کنیم. تگ جدول را به این صورت تغییر می دهیم:

```
<table border="3" cellpadding="10">
```

جدول به دست آمده به صورت زیر خواهد بود:

| | |
|--------------------|--------------------|
| سطر اول و سلول اول | سطر اول و سلول دوم |
|--------------------|--------------------|

| | |
|--------------------|--------------------|
| سطر دوم و سلول دوم | سطر دوم و سلول اول |
|--------------------|--------------------|

حالا می خواهیم هر دو مثال قبل را با هم امتحان کنیم، یعنی تگ جدول را مطابق زیر تغییر دهیم:

```
<table border="3" cellspacing="10" cellpadding="10">
```

نتیجه را ملاحظه کنید:

| | |
|--------------------|--------------------|
| سطر اول و سلول دوم | سطر اول و سلول اول |
| سطر دوم و سلول دوم | سطر دوم و سلول اول |

شما می توانید هر چیزی را که بخواهید داخل سلولها وارد کنید. مانند عکس، لینک، پاراگراف و... برای این کار فقط کد مورد نظر خود را بین تگ `<td>` و `</td>` وارد کنید. برای مثال فرض کنید می خواهیم یک لینک در جدول ایجاد کنیم. مانند زیر عمل می کنیم:

```
<table border="2">
```

```
<td>
```

```
<a href=" ../javascript/index.html">اسکرپت آموزش جاوا</a>
```

```
</td>
```

```
</table>
```

این هم جدولی که بیننده نهایی در صفحه مشاهده می کند:

[آموزش جاوا اسکریپت](#)

برای وارد کردن عکس نیز می توانیم به این صورت عمل کنیم:

```
<table border="2">  
<td>  
  
</td>  
</table>
```

این هم نتیجه کد:



جداول HTML : رنگ زمینه، نحوه قرارگیری افقی و عمودی ...

اگر بخواهیم عرض یک جدول را تغییر دهیم می توانیم از شناسه `width` استفاده کنیم. با اضافه کردن این شناسه به تگ جدول می توانیم مقدار عرض جدول را مشخص کنیم. این مقدار می تواند در مقیاس پیکسل باشد یا به صورت درصد. برای تغییر دادن ارتفاع جدول نیز میتوانیم از شناسه `height` استفاده کنیم. برای مثال اگر بخواهیم یک جدول با عرض 500 پیکسل و ارتفاع 50 پیکسل داشته باشیم می توانم به صورت زیر عمل کنیم:

```
<table border="3" width="500" height="50">
```

```
<td>
```

محتویات جدول

```
</td>
```

```
</table>
```

در اینجا می بینید که برخلاف متن درون جدول که متن کوتاهی است عرض جدول نسبتاً زیاد است (همان 500 پیکسلی که برای آن تعریف کردیم):

| |
|--------------|
| محتویات جدول |
|--------------|

اگر بخواهیم نحوه قرار گرفتن محتویات درون جدول را تعیین کنیم می توانیم از شناسه های زیر را در تگ `<td>` استفاده کنیم:

: align="left"

این شناسه محتویات جدول شما را در سمت چپ آن قرار می دهد.

: align="right"

با این شناسه می توانید محتویات جدول را به صورت راست چین نمایش دهید.

: align="center"

با این شناسه می توانید محتویات جدول را متمایل به مرکز جدول قرار دهید.

حالا این شناسه ها را در یک جدول به کار می بریم:

```
<table width="600" border="3">
```

```
<td align="left">
```

متمایل به چپ

```
</td>
```

```
<td align="center">
```

متمایل به مرکز

```
</td>
```

```
<td align="right">
```

متمایل به راست

```
</td>
```

```
</table>
```

می توانید نحوه قرارگیری محتویات سلولها را در جدول زیر ملاحظه کنید:

| | | |
|----------------|----------------|--------------|
| متمايل به راست | متمايل به مركز | متمايل به چپ |
|----------------|----------------|--------------|

اگر بخواهيم نحوه قرارگيري محتويات سلولها را به صورت عمودي تنظيم كنيم، مي توانيم از شناسه `valign` استفاده كنيم:

`: valign="top"`

با اين شناسه محتويات سلول در بالاي آن قرار مي گيرند.

`: valign="middle"`

به وسيله اين شناسه محتويات سلول در وسط آن به نمايش در مي آيند.

`: valign="bottom"`

با اين شناسه محتويات سلول در پايين آن نمايش داده مي شوند.

در مثال زير مي توانيد نحوه استفاده از شناسه هاي بالا در يك جدول را ببينيد:

```
<table width="600" height="100" border="3">
```

```
<td align="center" valign="top">
```

متمايل به بالا

```
</td>
```

```
<td align="center" valign="middle">
```

متمايل به وسط

```
</td>
```

<td align="center" valign="bottom">

پایین متمایل به

</td>

</table>

این هم جدول نهایی:

| | | |
|-----------------|---------------|----------------|
| متمایل به پایین | متمایل به وسط | متمایل به بالا |
|-----------------|---------------|----------------|

نحوه قرارگیری عمودی محتویات سلولها زمانی مفید خواهد بود که تعداد سطرهای محتویات سلولهای مختلف در یک ردیف جدول با هم برابر نباشند. در حقیقت زمانی که همه سلولها دارای یک تعداد سطر باشند و نحوه قرار گیری عمودی محتویات آنها به یک صورت است.

شناسه های دیگری نیز وجود دارد که می توانیم در تگ **td** استفاده کنیم:

:Rowspan

این شناسه مشخص کننده این است که ارتفاع سلول معادل چند سلول است

:Colspan

این شناسه مشخص کننده اندازه افقی یک سلول بر اساس تعداد سلول است.

به یک مثال توجه کنید:

```
<table border="2">
<td align="center">
اول سلول
</td>
<td rowspan="2" align="center">

</td>
<tr>
<td align="center">
سلول دوم
</td>
</tr>
</table>
```

این هم جدول به دست آمده. همانطور که می بینید سلولی که محتوی عکس است در کنار دو سلول دیگر قرار گرفته است نه در قسمت بالای سلول دوم:

| | |
|--|----------|
|  | سلول اول |
| | سلول دوم |

حال فرض کنید می خواهیم یک جدول مانند زیر داشته باشیم:

| | | |
|------------|----------|----------|
| سلول اول | | |
| سلول چهارم | سلول سوم | سلول دوم |

برای ایجاد این جدول در تگ `<td>` در ردیف اول از شناسه `colspan="3"` استفاده می کنیم.

با این کار سلول موجود در ردیف اول به اندازه سه سلول در عرض کشیده می شود.

در اینجا می توانید کد مربوط به این جدول را ببینید:

```
<table border="1">
```

```
<tr>
```

```
<td align="center" colspan="3">
```

```
<b>اول سلول</b>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td align="center">
```

```
سلول دوم
```

```
</td>
```

```
<td align="center">
```

```
سلول سوم
```

```
</td>
```

```
<td align="center">
```

سلول چهارم

```
</td>
```

```
</tr>
```

```
</table>
```

شما می توانید در صورت تمایل رنگ زمینه جدول، سطر، ستون یا هر سلول را در جدول که بخواهید تغییر دهید.

برای تعیین کردن رنگ زمینه برای جدول باید این شناسه `bgcolor` را در تگ جدول به کار ببرید و رنگ مورد نظر خود را در این شناسه به کار ببرید.

برای مثال اگر بخواهیم جدولی با زمینه زرد ایجاد کنیم، می توانیم به این صورت عمل کنیم:

```
<table border="3" bgcolor="yellow">
```

```
<td>
```

جدول محتویات

```
</td>
```

```
</table>
```

جدول زیر به دست می آید:

| |
|--------------|
| محتویات جدول |
|--------------|

قابل ذکر است که اگر به جای تگ ابتدایی جدول از تگ زیر نیز استفاده کنیم همان نتیجه را به دست می آوریم:

```
<table border="3" bgcolor="#ffff00">
```

اگر بخواهیم هر سلول رنگ زمینه مربوط به خودش را داشته باشد باید شناسه مربوط به رنگ زمینه را به تگ `<td>` اضافه کنیم. برای نمونه به جدول زیر توجه کنید:

| | |
|------------------|------------------|
| محتویات سلول اول | محتویات سلول دوم |
|------------------|------------------|

آیا می خواهید کد مربوط به این جدول را ببینید:

```
<table border="3">  
<td bgcolor="yellow">  
محتویات سلول اول  
</td>  
<td bgcolor="green">  
محتویات سلول دوم  
</td>  
</table>
```

در آخر می خواهیم رنگ زمینه هر ردیف را تغییر دهیم. برای این کار شناسه مربوط به رنگ زمینه را در تگ `<tr>` قرار می دهیم:

```
<table border="3" width="300">
<tr bgcolor="pink">
<td>صورتی</td>
<td>است این ردیف صورتی</td>
<tr bgcolor="gray">
<td>خاکستری</td>
<td>است این ردیف خاکستری</td>
</table>
```

این هم جدول به دست آمده:

| | |
|---------|----------------------|
| صورتی | این ردیف صورتی است |
| خاکستری | این ردیف خاکستری است |

قرار دادن دو جدول در کنار هم با استفاده از یک جدول بزرگتر

فرض کنید می خواهیم دو جدول را در یک سطر قرار دهیم. اگر بخواهیم دو تگ جدول را پشت سر هم استفاده کنیم، جدول دوم در سطر بعدی و زیر جدول اول قرار می گیرد. برای حل این مشکل می توانید دو جدول مورد نظر را در یک جدول که آنها را در بر می گیرد قرار دهیم. در این صورت می توان دو جدول در یک سطر ایجاد کرد. در اینجا می توانید کدی را که یک جدول را درون یک جدول دیگر ایجاد می کند:

<table width="500" border="4"> <!-- تگ آغازی جدول بیرونی -->

<td align="center"> <!-- سلول اول در جدول بیرونی -->

<table width="400" border="2"> <!-- آغاز جدول درونی -->

<td align="center"> <!-- اولین سلول جدول درونی -->

محتویات جدول درونی

</td> <!-- پایان سلول جدول درونی -->

</table> <!-- پایان جدول درونی -->

</td> <!-- پایان سلول جدول بیرونی -->

</table> <!-- پایان جدول بیرونی -->

این هم جدول به دست آمده:

| |
|--------------------|
| محتویات جدول درونی |
|--------------------|

به یاد داشته باشید که جدولهای درونی را باید در سلولهای جدول بیرونی قرار دهید یعنی بین

<td> و </td> در جدول بیرونی.

حالا فرض کنید می خواهیم بیش از یک جدول درون یک جدول ایجاد کنیم. برای اینکه این

جدولهای درونی، در یک سطر قرار گیرد کافی است به تعداد جدولهای درونی، در جدول بیرونی

سلول ایجاد کنیم و جدولها را در سلولهای جدول بیرونی قرار دهیم.

برای مثال فرض کنید می خواهیم دو جدول را در یک سطر داشته باشیم. می توانیم مانند زیر

عمل کنیم:

```
<table width="600" border="4"> <!-- تگ آغازی جدول بیرونی -->
```

```
<td align="center"> <!-- سلول اول در جدول بیرونی -->
```

```
<table width="250" border="2"> <!-- آغاز جدول درونی -->
```

```
<td align="center"> <!-- اولین سلول جدول درونی -->
```

محتویات اولین جدول درونی

```
</td> <!-- پایان سلول اولین جدول درونی -->
```

```
</table> <!-- درونی پایان اولین جدول -->
```

```
</td> <!-- سلول جدول بیرونی پایان اولین -->
```

```
<td align="center"> <!-- آغاز دومین سلول جدول بیرونی -->
```

```
<table width="250" border="2"> <!-- آغاز دومین جدول درونی -->
```

```
<td align="center"> <!-- آغاز سلول دومین جدول درونی -->
```

محتویات دومین جدول درونی

```
</td> <!-- پایان سلول دومین جدول درونی -->
```

```
</table> <!-- درونی پایان دومین جدول -->
```

```
</td> <!-- سلول جدول بیرونی پایان دومین -->
```

```
</table> <!-- پایان جدول بیرونی -->
```

این هم نتیجه به دست آمده:

| | |
|--------------------------|--------------------------|
| محتویات دومین جدول درونی | محتویات اولین جدول درونی |
|--------------------------|--------------------------|

قابل ذکر است که ما در این مثالها برای بهتر مشخص شدن محل جدولها از شناسه `border`

استفاده کرده ایم. شما می توانید به دلخواه از آن استفاده کنید.

بهتر است در هنگام نوشتن کدها عرض جدولها را مشخص کنید تا در پایان به هم ریخته نباشند.

با این روش می توانید هر تعداد جدول که نیاز داشته باشید در کنار هم قرار دهید.

اصول طراحی صفحات حاوی فریم -- بخش اول

برای اینکه بتوانید از فریمها در صفحه خود استفاده کنید، ابتدا باید بدانید این فریمها چگونه کار می کنند. یک صفحه که در آن از یک فریم استفاده شده است در حقیقت به دو یا چند قسمت تقسیم شده است که هر قسمت متن `html` مربوط به خود را دارد. لینک زیر را کلیک کنید تا بتوانید یک صفحه ساده که در آن از فریم استفاده شده است را ببینید.

[ایجاد صفحات با استفاده از فریمها](#)

همانطور که در این مثال می توانید ملاحظه کنید در واقع با استفاده از فریمها، دو صفحه وب را در قالب یک صفحه نمایش می دهیم. هر کدام از صفحات در این مجموعه (یعنی صفحاتی که در فریمها قرار گرفته اند و صفحه ای که فریمها در آن قرار دارند، کد `html` مربوط به خود را دارند. برای ایجاد فریم در یک صفحه به جای تگ `body` از تگ `frameset` استفاده می کنیم. می توانیم برای ساخت صفحه ای که فریمها در آن قرار می گیرند از کد زیر استفاده کنیم:

```
<html>
<head>
<title>نمایش یک فریم ساده</title>
</head>
<"cols="50%,50% frameset>
<"frame src="page1.htm>
<"src="page2.htm frame>
</frameset>
</html>
```

در اینجا به وظیفه تگهای مورد استفاده در اینجا می پردازیم:

:<FRAMESET>

این تگ به مرورگر می گوید که به جای یک صفحه معمولی منتظر یک دسته فرم باشد.

cols="50%,50%"

این فرمان به مرورگر می گوید که صفحه را به دو قسمت عمودی تقسیم کند و پهنای هر کدام از این قسمتها نیمی از پهنای پنجره باشد. شما می توانید این نسبت را تغییر دهید یا به جای درصد از مقیاس پیکسل استفاده کنید. اما اگر می خواهید از مقیاس درصد استفاده کنید فراموش نکنید که از نشانه درصد یعنی % در بعد از هر عدد استفاده کنید. اگر فقط برای یکی از اعداد علامت درصد استفاده کنید مرورگر عدد دیگر را در مقیاس پیکسل فرض می کند.

:<FRAME SRC="frame_1.htm">

این فرمان آدرس صفحهء اولین فریم از سمت چپ را برای مرورگر مشخص می کند.

:<FRAME SRC="test2.htm">

این فرمان آدرس صفحه بعدی را برای فریم بعدی از سمت چپ مشخص می کند.

حالا فرض کنید ما به سه فریم نیاز داشته باشیم. می توانیم به صورت زیر عمل کنیم:

```
<html>
<head>
<title/> نمایش سه فریم در یک صفحه
</head>
<"cols="33%,33%,33% frameset>
<"frame src="page1.htm>
<"src="page2.htm frame>
<"frame src="page3.htm>
</frameset>
</html>
```

با این کد می توانیم سه فریم در یک صفحه داشته باشیم که هر کدام از آنها 33 درصد پهنای صفحه را در بر می گیرند. یک درصد باقیمانده را هم مرورگر تخصیص می دهد.

لینک زیر را کلیک کنید تا بتوانید نتیجه را ملاحظه کنید:

[سه فریم در یک صفحه](#)

فرض کنید می خواهیم فریمها را به صورت عمودی و زیر یکدیگر در صفحه قرار دهیم. در این صورت باید به جای شناسه cols در تگ frameset از شناسه rows استفاده کنیم.

به کد زیر توجه کنید:

```
<html>
<head>
<title>عمودی نمایش فریمها به صورت افقی و</title>
</head>
<frameset cols="50%,50%">
<frame src="page1.htm">
  <frameset rows="50%,50%">
<frame src="page2.htm">
<frame src="page3.htm">
</frameset>
</frameset>
</html>
```

لینک زیر را برای مشاهده نتیجه کلیک کنید:

[فریمها به صورت افقی و عمودی](#)

ما با اولین تگ `frameset` به مرورگر می گوئیم که صفحه را به دو قسمت (دو ستون) تقسیم کند.

`<FRAME SRC="page1.htm">` به مرورگر می گوید که ستون اول از سمت چپ باید با

صفحه `page1.htm` پر شود.

تگ `frameset` دوم درون تگ `frameset` اول قرار دارد. این تگ به مرورگر می گوید که ستون دوم را به دو ردیف افقی تقسیم کند. (فریم دوم از سمت چپ خود به دو فریم تبدیل می شود)

تگ `<FRAME SRC="page2.htm">` به مرورگر می گوید که بالاترین فریم قرار گرفته در ستون دوم صفحه را با صفحه `page2.htm` پر کند.

تگ `<FRAME SRC="page3.htm">` به مرورگر می گوید که دومین فریم از بالا را در ستون دوم صفحه با صفحه `page3.htm` پر کند.

در پایان فراموش نکنید که همه تگهای `frameset` باز شده را در پایان ببندید.

فریمها، باز کردن لینکها در یک فریم مشخص

حالا شما می توانید یک صفحه با استفاده از فریمها بسازید. اما وقتی که شما یک لینک را در یک فریم قرار می دهید پس از کلیک کردن لینک در همان فریم باز می شود. این مسأله به این دلیل است که هر فریم یک صفحه HTML است و مانند یک صفحه عمل می کند. پس اگر برای لینک شناسه ای را تعریف نکنیم در همان صفحه (در حقیقت در همان فریم) باز می شود. شما می توانید صفحه را طوری طراحی کنید که لینکهای یک فریم در فریم دیگری باز شوند یا در یک صفحه جدید باز شوند که فریمی ندارد.

برای این کار باید ابتدا فریمها را نامگذاری کنیم و سپس از شناسه **target** در تگ لینک استفاده کنیم. به یک مثال در این رابطه توجه کنید:

```
<html>
<head>
<title>لینکها تعیین فریم مقصد برای</title>
</head>
<frameset cols="25%,75%">
<frame src="page4.htm" name="left">
<frame src="page2.htm" name="right">
</frameset>
</html>
```

ما در این مثال از دو فریم با نامهای **left** و **right** استفاده کردیم. شما می توانید هر نامی را که مناسب می دانید استفاده کنید.(این نامی است که در تگ لینک مورد استفاده قرار خواهد گرفت و لزوماً نباید همان نام فایل **HTML** قرار گرفته شده در فریم باشد.)

حالا که فریمها را نامگذاری کردیم می توانیم از این نامها به عنوان مقادیری برای شناسه **target** در تگ لینکها در هر کدام از فریمها استفاده کنیم. فرض کنید ما در فریم سمت چپ لیستی از صفحات را داریم که می خواهیم با کلیک کردن بر روی آنها در فریم سمت راست باز شوند. در این حالت باید این شناسه را به تگ لینکهای موجود در فریم سمت چپ اضافه کنید:

target="right" . برای مثال به لینک زیر توجه کنید:

[HTML آموزش](../../html/index.htm)

حالا پس از کلیک کردن لینک، فریم سمت چپ دست نخورده باقی می ماند و لینک در فریم سمت راست باز می شود. شما می توانید این کار را در صفحاتی که بیش از یک فریم هم دارند انجام دهید. فقط فراموش نکنید که فریمها را نامگذاری کنید.

در اینجا می توانید یک مثال برای این نوع صفحات را ببینید. لینک زیر را کلیک کنید و به صفحه آزمایشی وارد شوید تا لینکهای قرار گرفته در فریمها را امتحان کنید:

[ورود به صفحه آزمایشی](#)

میتوان لینکها را به جای اینکه در یک فریم باز کنیم، ر تمام پنجره و بدون فریم باز کنیم. برای این کار باید شناسه `target="_top"` را به تگ لینک اضافه کنیم. مثلاً می توانیم تگ لینکی مانند زیر بنویسیم:

[این لینک در تمام صفحه باز می شود](#)

در اینجا تعدادی از شناسه هایی را که می توان به تگ `frame` اضافه کرد آورده ایم. با این شناسه ها می توانیم رفتار این فریمها را بهتر کنترل کنیم:

`: scrolling="no"`

این فرمان به شما اجازه می دهد که اسکرول بار (`scroll bar`) را از سمت راست صفحه حذف کنید، تا کاربران نتوانند با استفاده از آن به بالا و پایین صفحه بروند. اگر از این شناسه استفاده نکنید، مرورگر تصمیم می گیرد که آیا اسکرول بار را در صفحه ای که درون فریم قرار دارد نشان

دهد یا نه.

اگر مقدار این شناسه **yes** باشد، اسکرول بار همیشه نشان داده خواهد شد. اگر **no** باشد اسکرول بار هرگز نمایش داده نخواهد شد. در صورتی که نیاز داشته باشید در این صفحه به بالا یا پایین و چپ و راست بروید می توانید با کلیدهای جهت صفحه کلید این کار را انجام دهید.

: border="3"

این شناسه به شما اجازه می دهد تا مقدار کلفتی حاشیه های فریم را تعیین کنید.

resize="no"

این فرمان ب شما امکان می دهد تعیین کنید که آیا کاربر می توان اندازه فریم را تغییر دهد یا نه.

اگر مقدار این شناسه **no** باشد کاربر نمی تواند با قرار دادن نشانگر ماوس خود در لبه فریمها، اندازه آنها را تغییر دهد. اگر مقدار آن **yes** باشد کاربر می تواند اندازه فریمها را تغییر دهد. گزینه پیش فرض این شناسه **yes** است.

noresize="no"

شما می توانید از این شناسه به جای **resize="no"** استفاده کنید، چون همه مرورگرها

شناسه قبلی را نمی شناسند. در هر حال می توانید برای اطمینان از هر دو شناسه با هم استفاده کنید.

: marginheight="2" و marginwidth="2"

این شناسه ها به شما امکان می دهد میزان حاشیه محتویات فریم را از لبه فریم تعیین کنید.

برخی از مرورگرهای قدیمی ممکن است نتوانند فریمها را نشان دهند. شاید شما بخواهید برای این دسته از کاربران با مرورگرهای قدیمی پیغامی را بنویسید یا لینک صفحه ای را که در آن از فریم استفاده نکرده اید به آنها نشان دهید. برای این کار می توانید از فرمان زیر استفاده کنید:

<noframes>

مشاهده می شود متن پیغامی که در صورت نمایش ندادن فریمها

</noframes>

حالا کاربرانی که نمی توانند فریمها را مشاهده کنند پیغام شما را به جای یک صفحه خالی می بینند.

استفاده از فریمهای نامرئی در صفحه

اگر شما می خواهید در صفحه خود از فریمهای نامرئی استفاده کنید، یعنی نمی خواهید خط حاشیه فریمها قابل دیدن باشد باید چند فرمان را به تگ **<frameset>** اضافه کنید و تگهای **<frame>** را آنطور که دوست دارید تنظیم کنید. در اینجا می توانید کدی را که یک صفحه را با فریمهای نامرئی می سازد می بینید:

```
<html>
<head>
<title>صفحه مخفی کردن فریم در</title>
</head>
<frameset cols="25%,75%" border="0" framespacing="0"
frameborder="0">
<frame src="page1.htm" name="left" scrolling="no">
<frame src="page2.htm" name="right">
</frameset>
</html>
```

در تگ `<frameset>` از شناسه های مختلفی می توان استفاده کرد که در اینجا لیست برخی از آنها را مشاهده می کنید:

`border="0"`

این شناسه میزان ضخامت کادر اطراف فریم را تعیین می کند که ما در اینجا آنرا صفر قرار داده ایم.

`frameborder="0"`

مقدار حاشیه فریمها را صفر قرار می دهیم تا نمایش داده نشوند.

framespacing="0"

فاصله بین فریمها را صفر قرار می دهیم تا بین آنها فاصله ای دیده نشود.

scrolling="no"

این شناسه باعث می شود در صورتی که محتویات فریمها از اندازه پنجره بزرگتر بود نوار اسکرول بار نمایش داده نشود.

آموزش ایجاد فریم های داخلی

به لطف HTML 4.0 ما توانایی ایجاد فریمها را در بدنه صفحات پیدا کردیم. با استفاده از این نوع فریمها (inline frame) می توانیم صفحه خود را به صورت عادی طراحی کنیم و در هر کجای آن که مایل بودیم دریچه ای به یک صفحه دیگر بسازیم.

Your browser does not support inline frames or is currently
.inline frames configured not to display

در قسمت‌های قبل با طرز ساختن فریمها و خواص آنها آشنا شدیم. در این قسمت می خواهیم به ساختن فریمهای درونی بپردازیم. یک نمونه از این فریمها را می توانید در سمت راست این نوشته ببینید.

احتمالاً در بسیاری از صفحات نمونه هایی از این نوع فریم را دیده اید. اگر به ساختن آنها علاقه مند شده اید به سراغ تگ سازنده این فریمها می رویم:

با قرار دادن این تگ در کد HTML صفحه در جایی که می خواهید فریم قرار بگیرد می توانید یک فریم اینلاین بسازید:

```
<iframe src=" ../examples/html/page1.htm" width="400"  
height="250" frameborder="1"></iframe>
```

با این تگ شما می توانید یک فریم در صفحه خود بسازید. همانطور که می بینید ما در صفحه اصلی از تگ **body** استفاده کردیم و از تگ **frameset** هم استفاده نشد.

در اینجا به توضیح شناسه های قابل استفاده در این تگ می پردازیم:

:<iframe>

این تگ آغازی یک فریم اینلاین است.

: src="frame_1.htm"

این قسمت مشخص کننده آدرس صفحه ای است که می خواهیم در فریم نمایش داده شود.

: width="350"

مشخص کننده مقدار عرض فریم در مقیاس پیکسل است. البته می توانیم از مقیاس درصد هم استفاده کنیم.

: height="250"

مشخص کننده ارتفاع فریم در صفحه است. مقیاس آن می تواند پیکسل یا درصد باشد.

: frameborder="1"

مشخص کننده ضخامت لبه های فریم است.

در مورد لینکها هم رفتار این فریمها مانند بقیه فریمها است. می توانیم به آنها یک نام بدهیم و از

شناسه **target** در لینکها استفاده کنیم. به یک مثال توجه کنید:

در ابتدا به تگ فریم خود یک نام نسبت می دهیم. مانند زیر:

```
<iframe src="../examples/html/page1.htm" width="400"  
height="250" frameborder="1" name="inlineframe1"></iframe>
```

در مرحله بعد به تگ لینک مورد نظر شناسه **target** را اضافه می کنیم. مانند زیر:

```
<a href="index.html" target="inlineframe1">فریم نمایش این صفحه در  
</a>. داده خواهد شد
```

حالا می توانید لینک زیر را کلیک کنید. خواهید دید که لینک در فریم همین صفحه باز خواهد

شد:

[این صفحه در فریم نمایش داده خواهد شد.](#)

مانند بقیه فریمها می توانید از شناسه **scrolling="no"** در تگ فریم استفاده کنید تا در فریم

اسکرول بار نداشته باشید.

رویدادهای مربوط به پنجره مرورگر و فرمها در HTML

یکی از امکاناتی که در HTML 4.0 به وجود آمد استفاده از event در عناصر HTML صفحه بود. یک نمونه از این رویدادها (event) اجرا شدن یک فرمان جاوا اسکریپت به هنگام کلیک کردن روی عناصر HTML در صفحه است.

در اینجا لیستی از رویدادهایی که در تگهای HTML استفاده می شوند آورده شده است. بهتر است برای اینکه بتوانید به خوبی از این رویدادها استفاده کنید اطلاعاتی در مورد [جاوا اسکریپت](#) داشته باشید.

برای استفاده از این رویدادها می توانید اسکریپتهای نوشته شده با جاوا اسکریپت را به عنوان مقادیر این رویدادها قرار دهید تا بسته به نوع رویداد مورد نظر اجرا شوند.

رویدادهای مربوط به پنجره مرورگر

این رویدادها باید فقط در تگهای **frameset** و **body** استفاده شوند. دو رویداد در این دسته قرار می گیرند که در زیر آنها را مشاهده می کنید:

:Onload

این رویداد باعث می شود تا اسکریپتی که به عنوان مقدار آن قرار داده شده است در هنگام باز شدن صفحه اجرا شود.

:Onunload

اسکرپتی که به عنوان مقدار این شناسه قرار می گیرد در هنگامی که بازدید کننده از صفحه خارج می شود اجرا می شود.

:OnResize

این رویداد هنگامی که اندازه پنجره مرورگر را تغییر می دهید اجرا می شود. برای مشاهده یک مثال برای این رویدادها لینک زیر را کلیک کنید. صفحه ای که به آن وارد می شوید حاوی دو رویداد بالا در تگ `body` خود است.

[ورود به صفحه آزمایشی برای رویدادهای مربوط به پنجره مرورگر](#)

می توانید تگ `body` مورد استفاده در مثال بالا را در اینجا مشاهده کنید:

```
"('body onload="alert">
```

```
onunload="alert"('onresize="alert" شما اندازه
```

```
پنجره را تغییر دادید!<
```

رویدادهای مربوط به فرمها

رویدادهای زیر در فرمها استفاده می شوند:

:OnChange

اسکرپت موجود در این شناسه به هنگام ایجاد تغییر در عناصر موجود در فرمها اجرا می شود.

: onsubmit

وقتی که فرم ارسال می شود اسکرپت موجود در این شناسه اجرا می شود.

: onreset

وقتی که دکمه **reset** (پاک کردن فرم) فشار داده شود اسکریپت موجود در این شناسه اجرا می شود.

: onselect

وقتی که قسمتی از فرم که این شناسه را در خود دارد انتخاب شود اسکریپت موجود در این شناسه اجرا می شود.

: onfocus

وقتی که تگی که حاوی این شناسه است مورد استفاده کاربر قرار گیرد اسکریپت موجود در این رویداد اجرا می شود.

: onblur

این شناسه بر عکس شناسه بالا عمل می کند یعنی وقتی کار کاربر با یک تگ تمام شد و به سراغ قسمت دیگری از صفحه رفت اجرا می شود.

در اینجا می توانید کدی را که در مثال بالا استفاده شده ببینید البته فقط قسمتهای ضروری آن آورده شده است:

```
onsubmit="msg('onSubmit')" "form action="#" method="post">
<"('onreset="msg('onReset
onfocus="msg('onFocus')" "input type="text">
</"('onblur="msg('onBlur
<"('name="select" onchange="msg('onChange select">
<option/>گزینه اول<option>
<option/>گزینه دوم<option>
<option/>گزینه سوم<option>
<option/>گزینه چهارم<option>
<select/>
<"('onSelect="msg('onSelect textarea">
<textarea/>onselect قرار دارد انتخاب کنید اسکریپت موجود در این رویداد اجرا می شود.
</"input type="reset" name="reset" value="reset">
</"input type="submit" name="Submit" value="Submit">
<form/>
```

رویدادهای مربوط به صفحه کلید و ماوس در HTML

در این قسمت به توضیح بخش دوم رویدادهای HTML که مربوط به صفحه کلید و ماوس می شود می پردازیم.

رویدادهای مربوط به صفحه کلید

سه رویداد در این دسته قرار می گیرد:

: onkeydown

در هنگام فشار دادن دکمه صفحه کلید اسکریپت تعریف شده در مقدار این شناسه اجرا می شود.

: onkeypress

وقتی که یک دکمه در صفحه کلید فشرده می شود و رها می شود اسکریپت موجود در این شناسه اجرا می شود.

: onkeyup

در هنگام رها شدن یک کلید این شناسه فعال می شود.

رویدادهای مربوط به صفحه کلید نباید در تگهای زیر استفاده شود:

`<base>`, `
`, `<bdo>`, `<frame>`, `<frameset>`, `<iframe>`,
`<html>`, `<head>`, `<meta>`, `<param>`, `<script>`, `<style>`, `<title>`

کدی را که در نوشتن این مثال از آن استفاده شده است می توانید در اینجا مشاهده کنید:

```
<input value="onkeypress" type="text"
onKeyPress="msg('onkeypress')" />
<input value="onkeypress" type="text"
onKeyUp="msg('onkeyup')" />
<input value="onkeypress" type="text"
onKeyDown="msg('onkeydown')" />
```

رویدادهای مربوط به ماوس

هفت رویداد در این دسته قرار دارند:

: onclick

در هنگام کلیک کردن بر روی قسمتی که این شناسه را دارد اجرا می شود.

: ondblclick

مقدار موجود در این شناسه به هنگام دابل کلیک بر روی قسمتی که این شناسه در آن به کار رفته اجرا می شود.

: onmousedown

وقتی که کلید ماوس فشرده می شود این شناسه عمل می کند.

: onmousemove

وقتی که ماوس را در صفحه حرکت می دهید این رویداد اجرا می شود.

: onmouseover

وقتی ماوس روی قسمتی که حاوی این شناسه است قرار می گیرد این رویداد اجرا می شود.

: onmouseout

این رویداد وقتی که ماوس از روی قسمتی که حاوی این رویداد است خارج می شود اجرا می شود.

: onmouseup

وقتی که کلید ماوس رها می شود این رویداد اجرا می شود.

رویدادهای مربوط به ماوس هم مانند رویدادهای صفحه کلید نباید در تگهای زیر استفاده شوند:

`<base>`, `
`, `<bdo>`, `<frame>`, `<frameset>`, `<iframe>`,
`<html>`, `<head>`, `<meta>`, `<param>`, `<script>`, `<style>`, `<title>`

می توانید کدی را که در نوشتن این مثال از آن در سلولهایی که حاوی رویدادها هستند استفاده شده است در اینجا مشاهده کنید:

اجرا `onClick` رویداد در این قسمت کلیک کنید تا `<td onClick="msg('onClick')">`
`</td>` شود.

در این قسمت دابل کلیک کنید تا رویداد `<td onDbIcIck="msg('onDbIcIck')">`
`onDbIcIck` اجرا شود `</td>`

`<td onMouseDown="msg('onMouseDown')">` در این قسمت کلید ماوس را
`</td>` اجرا شود `onMouseDown` رویداد پایین نگهدارید تا

`<td onMouseMove="msg('onMouseMove')">` روی این قسمت ماوس خود را
`</td>` اجرا شود `onMouseMove` حرکت دهید تا رویداد

`<td onMouseOver="msg('onMouseOver')">` ماوس خود را روی این قسمت
`</td>` اجرا شود `onMouseOver` قرار دهید تا رویداد

`<td onMouseOut="msg('onMouseOut')">` وقتی ماوس از روی این قسمت کنار
`</td>` اجرا می شود `onMouseOut` رود رویداد

`<td onMouseUp="msg('onMouseUp')">` وقتی کلید ماوس روی این قسمت رها
`</td>` اجرا می شود `onMouseUp` شود رویداد

در این مثال `'msg(متن پیام)'` تابع جاوا اسکریپتی است که در صفحه مثال تعریف شده است.
به جای آن می توانید از کدهای دیگری استفاده کنید.

تنظیم پیج کد مورد استفاده در صفحه

حتماً تا به حال با صفحاتی مواجه شده اید که به زبان فارسی نوشته شده اند، اما وقتی از آنها بازدید می کنید با کلماتی به هم ریخته و غیر قابل فهم مواجه می شوید. دلیل این مشکل ناسازگار بودن Encoding انتخاب شده در مرورگر با زبان فارسی است.

این ناسازگاری ممکن است به دلیل کامل نشدن بارگزاری صفحه ایجاد شده باشد یا به دلیل اینکه در هنگام طراحی Encoding متناسب با زبان فارسی انتخاب نشده باشد یا اصلاً در طراحی صفحه هیچگزینه ای برای Encoding صفحه در نظر گرفته نشده باشد.

در اکثر مواقع می توانید صفحه مورد نظر را به روش زیر به صورت فارسی مشاهده کنید:

مرورگر اینترنت اکسپلورر در ویندوز (Microsoft Internet Explorer) :

از منوی **view** گزینه **Encoding** را انتخاب کنید و گزینه **utf-8** را کلیک کنید. در صورتی که در لیست اولیه کد پیجهای مختلف نتوانستید **utf-8** را مشاهده کنید نشانگر ماوس خود را روی گزینه **More** نگهدارید و **utf-8** را از بین کد پیجهای مختلف پیدا کرده و روی آن کلیک کنید.

در سایر مرورگر ها هم به همین روش باید عمل کنید فقط مسیر منو ها کمی تفاوت دارد. می توانید مسیر های زیر را دنبال کنید.

مرورگر فایرفوکس (Firefox) :

view --> Character Encoding --> unicode (utf-8)

یا

view --> Character Encoding --> More Encodings --> Unicode -->
Unicode (utf-8)

مرورگر اپرا (Opera) :

view <-- Encoding <-- Unicode <-- UTF-8

البته اگر به این روش مشکل صفحه مورد نظر شما حل نشد به دلیل این است که صفحه با پیچ
کد دیگری نوشته شده است. می توانید گزینه هایی مثل (Arabic(Windows رانیز امتحان
کنید.

برای اینکه یک صفحه فارسی طراحی کنیم دو چیز را باید مد نظر قرار دهیم:

راست به چپ بودن صفحه

برای اینکه یک صفحه راست به چپ طراحی کنیم می توانیم به چند طریق عمل کنیم.

استفاده از شناسه **dir="rtl"** در تگ **<html>**

از این شناسه در بسیاری از تگهای HTML می توان استفاده نمود. در صورتی که صفحه ای

راست به چپ باشد و بخواهیم از قسمتهای چپ به راست نیز در آن صفحه استفاده کنیم می

توانیم از شناسه **dir="ltr"** استفاده کنیم.

استفاده از **CSS** برای تعیین **direction** صفحه

مثلاً استفاده از یک استایل خارجی یا در بخش **HEAD** متن یا استفاده از شناسه

style="direction: rtl" در تگهایی که می خواهیم به صورت راست به چپ نمایش داده شود.

تنظیم Encoding در هنگام طراحی صفحه

برای تنظیم Encoding صفحه باید از یک متاتگ در بخش HEAD متن استفاده کنیم. این متاتگ برای کد پیچ utf-8 که بیشترین سازگاری را با صفحات فارسی دارد به صورت زیر است:

```
<meta http-equiv="Content-Type" content="text/html"; charset=utf-8>
```

اگر در طراحی صفحه ای از این متاتگ استفاده شود کد پیچ utf-8 در مرورگر به صورت خودکار برای آن صفحه اعمال می شود.

فصل دوم

آموزش کار با استایل (CSS)

CSS یا همان Sheet Style Cascading روشی است برای قالب بندی و طراحی اجزای صفحه از قبیل متن، تصویر، زمینه صفحه، کادرها و تقریباً هر چیزی که در طراحی صفحه استفاده می شود بدون استفاده از کدهای HTML. از طریق این استایل شیتها می توانیم از دوباره نویسی کدهای HTML که باعث مشکل شدن کد نویسی و همچنین کند شدن سرعت بارگذاری صفحه در اینترنت می شود جلوگیری کنیم. برای مثال می توانیم یک بار نوع فونت مورد استفاده در یک صفحه را تعیین کنیم و دیگر مجبور نباشیم کد مربوط به فونت را در صفحه به کار ببریم. همچنین می توانیم از این ابزار قدرتمند در مواردی به جای جاوا اسکریپت نیز استفاده کنیم و چون مشکلات جاوا اسکریپت مثلاً غیر فعال بودن جاوا اسکریپت در مرورگر را ندارد با خیالی آسوده تر می توانیم از آن استفاده کنیم.

CSS هم مانند جاوا اسکریپت این امکان را به ما می دهد که از یک فایل استایل خارجی برای تعیین خواص بخشهای مختلف صفحه های یک سایت استفاده کنیم. در این صورت می توانیم ظاهر همه صفحه ها را با ایجاد یک تغییر کوچک در فایل استایل خارجی تغییر دهیم. مثلاً فرض کنید که رنگ متن همچنین نوع فونت متن صفحه هایمان را در یک فایل استایل تعیین کرده ایم و از آن در طراحی چندین صفحه استفاده کرده ایم، حالا اگر بخواهیم رنگ متن همه این صفحات را تغییر دهیم تنها کافی است تا رنگ متن را در همان فایل استایل تغییر دهیم و نیازی به تغییر دادن صفحات نداریم.

در این قسمت به معرفی قابلیت های مختلف CSS خواهیم پرداخت. چون این قسمت در سایت ما به تازگی راه اندازی شده است هنوز کامل نشده و به مرور زمان سعی می کنیم آنرا کاملتر کنیم. اگر انتقاد یا پیشنهادی برای بهتر شدن این قسمت دارید می توانید از طریق «فرم تماس با ما» ما را مطلع کنیم. ما نیز خوشحال می شویم از نظرات شما در جهت بهبود این بخش استفاده کنیم.

استفاده از کلاس و id در طراحی صفحات با CSS

سلکتور کلاس

با استفاده از کلاس می توانیم برای یک تگ HTML استایلهای متفاوتی را تعریف کنیم. برای نمونه فرض کنیم می خواهیم در صفحه دو نوع پاراگراف داشته باشیم: یکی با متن مشکی و دیگری با متن قرمز. برای مشخص کردن این دو نوع پاراگراف می توانیم از سلکتور کلاس استفاده کنیم. ابتدا استایل را به صورت زیر می نویسیم:

```
p.black {color: black}
```

```
p.red {color:red}
```

در مرحله بعد باید برای هر پاراگرافی که می خواهیم به رنگ مشکی باشد از شناسه

`class="black"` در تگ `<p>` استفاده کنیم و برای پاراگراف قرمز از شناسه

`class="red"`. البته نام کلاسها اختیاری است ولی بهتر است آنها را طوری انتخاب کنید که

مفهوم داشته باشند تا وقتی که حجم استایل شما بیشتر شد به گنگ نباشد. در مثال زیر می

توانید کد مربوط به نوشتن دو پاراگراف بالا را در متن HTML ملاحظه کنید:

```
<p class="black">شود این متنی است که به رنگ مشکی نمایش داده می</p>
```

```
<p class="red">شود این متن به رنگ قرمز نمایش داده می</p>
```

نکته: شما می توانید بیش از یک کلاس برای یک تگ HTML تعریف کنید اما برای این کار

نباید دو بار از شناسه `class` استفاده کرد. بلکه باید در یک شناسه `class` نام دو کلاس مورد

نظر را با یک فاصله بین آنها وارد کرد. برای مثال برای پاراگراف زیر از دو کلاس فرضی `red` و `center` استفاده شده است:

`<p class="red center">` و `red` دو کلاس این متن پاراگرافی است که تحت تأثیر `</p>` قرار دارد `center`

در نوشتن استایل می توان از نوشتن نام تگ در سلکتور کلاس چشم پوشی کرد. در این صورت ویژگیهای تعریف شده در آن کلاس بر هر تگ HTML که از کلاس مربوطه در شناسه `class` آن استفاده شده باشد اثر می گذارد. برای مثال در کلاس زیر ویژگی وسط چین را تعریف می کنیم و از نام تگ در سلکتور کلاس استفاده نمی کنیم:

`.center {text-align: center}`

اکنون برای هر تگی که احتیاج به وسط چین داشته باشد می توانیم از کلاس بالا استفاده کنیم. به چند نمونه در زیر توجه کنید:

داده می متن این پاراگراف به صورت وسط چین نمایش `<p class="center">` شود `</p>`

نمایش داده می متن این قسمت به صورت وسط چین `<div class="center">` شود `</div>`

وسط چین نمایش داده می متن این سرفصل به صورت `<h2 class="center">` شود `</h2>`

تذکر : هیچ وقت نام کلاس را با یک عدد شروع نکنید چون در این صورت مرورگرهای **Firefox و Mozilla** نمی توانند از آن کلاس استفاده کنند.

ما می توانیم از سلکتور کلاس برای تگی که حاوی شناسه **class** است استفاده کنیم که در قسمتهای قبلی در این مورد توضیح داده شد. علاوه بر این می توانیم از ویژگیهای تعریف شده در سلکتور کلاس برای تگهایی که در یک تگ محتوی شناسه **class** قرار دارند نیز استفاده کنیم. مثلا برای لینکهایی که در یک **DIV** قرار دارند و برای **DIV** کلاس تعریف شده است. برای این کار در نوشتن استایل مربوطه باید پس از نقطه که مشخص کننده سلکتور کلاس است نام کلاس را وارد کنیم و پس از وارد کردن یک فضای خالی نام تگی که می خواهیم ویژگیهای بر روی آن اعمال شود. به مثال زیر توجه کنید:

در این مثال از استایل زیر در استفاده می کنیم:

```
.test { border: #F00 1px double }  
.test a { color: green }  
.test a:hover { border: yellow 2px solid }
```

کد **HTML** زیر برای این مثال مورد استفاده قرار می گیرد:

```
<div class="test">  
<a href="#">1 لینک آزمایشی شماره  
</div>  
<a href="#">2 لینک آزمایشی شماره
```

می توانید نتیجه را در اینجا مشاهده کنید:

[لینک آزمایشی شماره 1](#)

[لینک آزمایشی شماره 2](#)

همانطور که در مثال بالا می بینید خواص لینک آزمایشی شماره 1 تحت تأثیر استایل مورد استفاده برای تگ DIV قرار دارد ولی لینک آزمایشی شماره 2 مانند سایر لینکهای خارج از DIV نمایش داده می شود.

افزودن استایل به نوع مشخصی از عناصر

همانطور که می دانید با وجود اینکه برخی از عناصر در صفحه با هم تفاوت دارند، از یک تگ HTML برای ایجاد آنها استفاده می شود. برای مثال برخی از این عناصر را که از تگ `<input>` در به وجود آوردن آنها استفاده می شود می توانید در زیر مشاهده کنید:

`<input type="text">`

کادرهای متنی که در فرمها استفاده می شود.

`<input type="checkbox">`

چک باکس هم با استفاده از تگ `<input>` به وجود می آید.

`<input type="radio">`

در ساختن دکمه های رادیو هم از همان تگ استفاده می شود.

`<input type="button">`

این تگ در ساختن هم استفاده می شود.

<input type="submit">

برای ساختن دکمه ارسال فرم هم از تگ <input> استفاده می شود.

حالا فرض کنید ما در یک صفحه از چند نمونه از عناصر بالا استفاده کرده ایم ولی فقط می خواهیم ویژگیهای یکی از آنها را تغییر دهیم. دو روش برای این کار وجود دارد: یکی استفاده از کلاس است که در این روش اگر تعداد زیادی از عنصر مورد نظر در صفحه وجود داشته باشد باید به همان تعداد از شناسه **class** در تگهای آنها استفاده کنیم و ممکن است این کار باعث افزایش حجم صفحه یا موجب سختی در نوشتن کد شود. روش دوم تعریف کردن سلکتور **input** برای عناصر مورد نظر است که به صورت زیر انجام می گیرد:

فرض کنید می خواهیم فقط عرض کادرهای متنی را در صفحه تغییر دهیم. در صورتی که در استایل از سلکتور **input** به تنهایی استفاده کنیم عرض همه چک باکسها ، دکمه ارسال ، دکمه های رادیو و سایر عناصری که از تگ <input> استفاده کرده اند نیز تغییر می کند. برای انجام این کار می توانید سلکتور را به صورت زیر تعریف کنیم:

```
input[type="text"] {  
width: 200px  
}
```

کد استایلی که در بالا نوشتیم فقط بر روی کادر های متنی که در آنها از شناسه

type="text" استفاده شده است اثر می گذارد.

البته این روش ممکن است در مرورگر **Internet Explorer** پشتیبانی نشود اما مرورگرهای **Opera** و **Firefox** از آن پشتیبانی می کنند.

سلکتور id

روش دیگری که برای تعریف استایل وجود دارد استفاده از **id** است. شناسه **id** احتمالاً در آینده جایگزین شناسه **name** در تگهای HTML می شود بنا بر این در هر صفحه نباید بیش از یک بار از هر **ID** استفاده شود. اگر بیش از یک بار از یک **id** استفاده شود کارایی صفحه تغییری نمی کند ولی صفحه اعتبار خود را از دست می دهد. سلکتور **id** با علامت **#** مشخص می شود. با استایل زیر همه عناصر که شناسه **id="border"** دارند با کادری سبز رنگ نمایش داده می شوند:

```
#border { border: green thin solid }
```

برای مثال ما از استایل بالا در این صفحه استفاده کرده ایم. حالا کد زیر را وارد صفحه می کنیم:

```
<h3 id="border">3</h3>
```

```
<div id="border">یک</div>
```

در قسمت زیر می توانید نتیجه کد بالا را مشاهده کنید:

این یک سرفصل با اندازه 3 است

این قسمت یک **div** است

همانطور که کلاس را می توانستیم برای یک عنصر مشخص تعریف کنیم در مورد **id** نیز می

توانیم این کار را انجام دهیم. به استایل زیر توجه کنید:

```
p#border { border: green thin solid }
```

استایل بالا فقط روی پاراگرافی که شناسه **id="green"** در آن به کار رفته اثر می گذارد.

نکته : هرگز نام یک **id** را با یک عدد آغاز نکنید چون در این صورت مرورگرهای **Firefox و Mozilla** نمی توانند از آن استفاده کنند.

وارد کردن CSS در تگهای HTML با استفاده از شناسه Style

ساده ترین راه برای افزودن استایل به یک صفحه استفاده از شناسه **style** در تگهای HTML است.

برای افزودن استایل به یک تگ باید عبارت `style=" "` را وارد تگ مربوطه کنیم و ویژگیهای مورد نیاز خود را بین دو دابل کوت (") وارد می کنیم. مثلاً فرض کنید می خواهیم به یک تگ **DIV** استایل اضافه کنیم. در این صورت تگ **DIV** به صورت زیر در می آید.

`<div style="property:value"`

برای نمونه وقتی بخواهیم متن داخل یک **DIV** به رنگ سبز نمایش داده شود می توانیم به صورت زیر تگ **DIV** را بنویسیم:

`<div style="color:green"> متن داخل DIV </div>`

توجه کنید که بین ویژگی **color** و رنگ سبز (**green**) از علامت دو نقطه (:) استفاده شده نه از علامت مساوی که در تگهای **HTML** استفاده می شود. همچنین لازم نیست مقادیر مورد نظر را بین علامت های نقل قول (") قرار دهیم. می توانید نتیجه به کارگیری کد بالا را در سطر زیر مشاهده کنید:

متن داخل **DIV**

به خاطر داشته باشید که برای اینکه خاصیت این استایل از بین برود باید تگی را که استایل در آن به کار رفته است ببندید. در اینجا تا زمانی که تگ **DIV** بسته نشده باشد متن ما به رنگ قرمز نمایش داده خواهد شد.

این امکان نیز وجود دارد که به یک تگ **HTML** بیش از یک ویژگی افزوده شود. تنها کافی است بین ویژگیهای مختلف از یک نقطه ویرگول (;) استفاده کنیم. برای مثال اگر بخواهیم متنی که در یک **DIV** قرار دارد به رنگ قرمز و به صورت *ایتالیک* باشد می توانیم به صورت زیر کد **DIV** را بنویسیم:

```
<div style="color:red; font-style:italic">قرمز رنگ و ایتالیک این متن</div>  
</div>.
```

نتیجه به صورت زیر خواهد بود:

این متن قرمز رنگ و ایتالیک است.

در این روش می توانیم به هر تعداد که لازم باشد از ویژگیهای مختلف درون یک تگ **HTML** استفاده کنیم. به مثال زیر توجه کنید:

```
<div style="color:blue; font-style:italic; font-weight:bold; font-size:12pt; text-align:center">می گیرد متن مورد نظر اینجا قرار</div>
```

در این مثال ما متنی را با رنگ آبی، ایتالیک، ضخیم، اندازه فونت **pt12** و در وسط صفحه خواهیم داشت:

متن مورد نظر اینجا قرار می گیرد.

در قسمتهای بعدی لیستی را از ویژگیهای مختلفی که با استایل می توانیم آنها را کنترل کنیم خواهیم آورد. در اینجا فقط قصد آموزش نحوه قرارگیری استایل در تگهای HTML را داشتیم.

تعریف کردن استایل در بخش head صفحه HTML

استفاده از استایلهای می تواند به ما کمک کند تا از دوباره نویسی فرمانها جلوگیری کنیم. برای انجام این کار ما می توانیم یک `Style` را در بخش `HEAD` در متن `HTML` خود تعریف کنیم. یک استایل در `HEAD` با تگ `<STYLE>` آغاز می شود و با `</STYLE>` به پایان می رسد. در قسمت زیر می توانید یک نمونه از استایل را که مشخصاتی را برای تگ `` تعریف می کند مشاهده کنید که در بخش `HEAD` در متن قرار گرفته است.

```
<head>
```

```
<style>
```

```
<!--
```

```
span { color:red; font-style:italic }
```

```
-->
```

```
</style>
```

```
</head>
```

همانطور که می بینید استایل با تگ `<style>` آغاز می شود. سپس یک کامنت را شروع می کنیم تا مرورگرهایی که از CSS پشتیبانی نمی کنند استایل را نادیده بگیرند و در صفحه به نمایش در نیاورند. در قسمت بعد این خط را مشاهده می کنید:

`span { color:red; font-style:italic }`

این خط مشخص می کند که هر بار که در صفحه از تگهای `<STYLE>` `</STYLE>` استفاده می شود. متن بین آنها به رنگ قرمز و به صورت ایتالیک نمایش داده می شود. در نوشتن استایل باید در نظر داشته باشید که نباید از علامتهای کوچکتر و بزرگتر (`<` و `>`) که در اطراف تگهای HTML استفاده می شود، استفاده کنید. همچنین به جای استفاده از علامتهای نقل قول (`" "`) که در متن HTML استفاده می شود از آکولاد (`{ }`) در اطراف خواص استایل استفاده می کنیم. خواص مختلف یک استایل هم به وسیله یک نقطه ویرگول (`;`) از هم جدا می شوند. وقتی ما استایلی را در بخش HEAD متن خود برای یک تگ خاص تعریف می کنیم برای محتویات همه تگهایی که استایل برای آن تعریف شده در صفحه مورد استفاده قرار می گیرد. برای مثال وقتی که ما کد بالا را در HEAD متن خود قرار دادیم، هر تگ `` که در صفحه استفاده شود دارای متنی قرمز و ایتالیک است و نیاز ندارید هر بار دوباره مشخصات متن مورد نظر را تعریف شود.

برای مثال کد زیر را وارد متن HTML خود می کنیم:

 این متن خارج تگ اما این متن قرمز و ایتالیک است،

. ندارد است و مشخصات آنرا

. این متن نیز قرمز و ایتالیک است

نتیجه را به صورت زیر مشاهده خواهید کرد:

این متن قرمز و ایتالیک است، اما این متن خارج تگ است و مشخصات آنرا ندارد.

این متن نیز قرمز و ایتالیک است.

با این روش ما می توانیم تقریباً خواص همه تگهای HTML را کنترل کنیم، اما این کافی نیست. در قسمتهای بعدی روشهایی را معرفی می کنیم که یک استایل مشخص به وسیله تگهای مختلفی مورد استفاده قرار بگیرد. این کار با استفاده از CLASS و ID انجام می دهیم که به آن خواهیم پرداخت.

استفاده از استایل خارجی برای طراحی صفحات HTML

فرض کنید می خواهیم برای چندین صفحه از یک سری استایل استفاده کنیم که در همه صفحات مشابه هستند. در این مورد می توانیم از یک فایل استایل خارجی استفاده کنیم تا نیازی به چندین بار نوشتن استایلها نباشد.

برای شروع با استفاده از یک ویرایشگر متن مانند Notepad در ویندوز یک فایل متنی ایجاد کنید. سپس اطلاعات زیر را در این فایل وارد کنید:

```
div { font-family:Arial }
```

```
.redtext { color:red }
```

حالا این متن را با نام `example.css` ذخیره کنید (شما می توانید از هر نامی که می خواهید برای ذخیره کردن فایل استفاده کنید، فقط به خاطر داشته باشید که پسوند فایل باید `CSS` باشد.)

برای پیوند دادن یک فایل استایل خارجی به صفحه `HTML` باید از تگ `<LINK>` استفاده کنیم که در بخش `HEAD` در متن `HTML` قرار می گیرد. یک نمونه از این تگ را می توانید در اینجا ببینید:

```
<link rel="stylesheet" type="text/css" href="URL">
```

شناسه `rel="stylesheet"` مشخص می کند که فایل مورد نظر شما یک استایل است.

شناسه `type="text/css"` مشخص کننده نوع متن فایل برای مرورگر است.

شناسه `href=" "` مشخص کننده محل قرار گرفتن فایل استایل شماست. شما می توانید از

آدرس کامل فایل استفاده کنید و یا اگر فایل شما در همان پوشه ای است که متن `HTML` شما قرار دارد می توانید تنها از نام فایل استفاده کنید.

حالا متن `HTML` مورد نظر که می خواهید استایل به آن افزوده شود را باز کنید . خط زیر را بین تگ `<HEAD>` و `</HEAD>` وارد کنید. مطمئن شوید که آدرس فایل `CSS` را درست وارد کرده اید:

```
<link rel="stylesheet" type="text/css"
```

```
href="http://www.neopersia.org/css/example.css">
```

شما باید این کد را در همه صفحاتی که می خواهید از استایل تعریف شده در فایل قبل استفاده کنید وارد کنید وگرنه صفحات آنگونه که طراحی شده اند نمایش داده نمی شوند.

حالا هر تگ **DIV** که در صفحات مورد نظر شما وجود داشته باشد محتویاتی با فونت **Arial** خواهد داشت. برای مثال فایل **example.css** به همین صفحه پیوند شده است و در صورتی که در این صفحه از **DIV** استفاده شود محتویات آن با فونت **Arial** نمایش داده می شوند:

<div> شد نمایش داده خواهد **Arial** این متن با فونت **</div>**

این هم نتیجه:

این متن با فونت **Arial** نمایش داده خواهد شد.

همچنین وقتی که از کلاس **redtext** در تگها استفاده شود متن به رنگ قرمز نمایش داده می شود:

<p class="redtext"> شد این متن به رنگ قرمز نمایش داده خواهد **</p>**

نتیجه به این صورت خواهد بود:

این متن به رنگ قرمز نمایش داده خواهد شد.

تنظیم عرض، ارتفاع، و overflow با CSS

عرض و ارتفاع

width

این مشخصه به ما امکان می دهد تا عرض یک بخش از صفحه را تعیین کنیم. برای مثال می توانیم با آن عرض یک DIV را تعیین کنیم. در مثال زیر برای DIV عرضی معادل 200 پیکسل تعیین می کنیم:

```
<div style="width:200px"> از 200 پیکسل بیشتر این جمله پس از اینکه عرض آن </div> کند بعدی ادامه پیدا می سطرهای شد در
```

نتیجه به صورت زیر خواهد بود:

این جمله پس از اینکه عرض آن از 200 پیکسل بیشتر شد در سطرهای بعدی ادامه پیدا می کند.

مقادیری که برای پارامتر width قرار می دهید می تواند بر اساس پیکسل باشد که در این صورت باید پس از مقدار مربوطه حروف px که نشانه پیکسل است را بنویسید. مقادیر ممکن است بر اساس درصد باشند که در آن صورت باید علامت درصد (%) را پس از مقدار قرار دهید.

height

این پارامتر به ما اجازه می دهد تا ارتفاع بخش مشخصی را تعیین کنیم. برای مثال وقتی که ما این پارامتر را برای یک DIV استفاده می کنیم و مقدار آن را 100 پیکسل قرار می دهیم تا وقتی که متن موجود در DIV از 100 پیکسل کمتر ارتفاع دارد ارتفاع DIV 100 پیکسل

خواهد بود. ولی وقتی که متن داخل DIV زیاد باشد و از 100 پیکسل بیشتر شود در نتیجه ارتفاع DIV هم از 100 تجاوز خواهد کرد تا همه متن موجود قابل خواندن باشد. به مثال زیر توجه کنید:

ابتدا هنگامی که ارتفاع متن از ارتفاع DIV کمتر است:

`<div style="height:100px">100</div>` پیکسل کمتر است ارتفاع این متن از 100

ارتفاع این متن از 100 پیکسل کمتر است.

حالا هنگامی که ارتفاع متن از ارتفاع DIV بیشتر است:

`<div style="height:100px">
ارتفاع
` متن `
` این `
` ارتفاع

`</div>` است `
` کمتر `
` پیکسل `
` 100 `
` از

ارتفاع

این

متن از

100

پیکسل

کمتر

است.

اگر بخواهیم ارتفاع DIV همان 100 پیکسل بماند و تغییر نکند باید از پارامتر **overflow** در استایل استفاده کنیم. این پارامتر به ما این امکان را می دهد که تعیین کنیم وقتی محتویات DIV از عرض یا ارتفاع DIV بزرگتر می شود چگونه نمایش داده شود. در اینجا برای پارامتر **overflow** از مقدار **hidden** استفاده می کنیم که تعیین می کند هر چیزی بزرگتر از عرض یا ارتفاع DIV نمایش داده نشود. به مثال زیر توجه کنید:

```
<div style="height:100px;overflow:hidden">ارتفاع<br>این<br>متن<br>کمتر<br>است.</div>
```

ارتفاع

این

متن از

100

پیکسل

کمتر

است.

مقادیری که برای پارامتر **height** قرار می دهید می تواند بر اساس پیکسل باشد که در این صورت باید پس از مقدار مربوطه حروف **px** که نشانه پیکسل است را بنویسید. مقادیر ممکن است بر اساس درصد باشند که در آن صورت باید علامت درصد (%) را پس از مقدار قرار دهید.

overflow

وقتی محتویات قسمتی از صفحه طول یا عرضی بزرگتر از آن قسمت در بر گیرنده داشته باشد، با پارامتر **overflow** می توانیم مشخص کنیم که آن قسمت زیادی نمایش داده شود یا مخفی شود. به صورت پیش فرض قسمتهای اضافی نمایش داده می شوند ولی با این پارامتر ما می توانیم آنها را مخفی کنیم. به مثال زیر توجه کنید:

<div style="width:200px; overflow:hidden"><nobr> طول این سطر

دیده DIV قسمتهای بیشتر از عرض **overflow** از است ولی با استفاده **DIV** بیشتر از عرض **</nobr></div>** شوند نمی

طول این سطر بیشتر از عرض **DIV** است ولی با استفاده از **overflow** قسمتهای بیشتر از عرض **DIV** دیده نمی شوند.

به طور کلی چند مقدار را می توانیم برای پارامتر **overflow** استفاده کنیم:

visible

یا قرار دادن این مقدار برای **overflow** همه محتویاتی که طول یا عرض بیش از اندازه داشته اند نمایش داده خواهند شد.

hidden

این مقدار باعث خواهد شد تا محتویات با طول یا عرض بیش از اندازه نمایش داده نشوند.

scroll

این مقدار باعث خواهد شد تا در صورتی که محتویات اندازه ای بیش از طول و عرض لازم دارند با استفاده از اسکرول بار قابل مشاهده باشند.

به یک مثال توجه کنید:

```
<div style="width:200px; overflow:scroll; border:dashed 1px  
#3399FF; white-space:nowrap">
```

قسمتهای بیشتر از **overflow** استفاده از است ولی با **DIV** طول این سطر بیشتر از عرض

شوند دیده نمی **DIV** عرض

```
</div>
```

این هم نتیجه:

طول این سطر بیشتر از عرض **DIV** است

نحوه ایجاد کادر برای عناصر صفحه با استفاده از استایل

کادر (**Boder**)

border-style

با استفاده از استایل ما می توانیم برای اجزای یک صفحه کادر ایجاد کنیم. با استفاده از پارامتر

border-style می توانیم مشخص کنیم که آیا می خواهیم کادر داشته باشیم یا نه، و اگر می

خواهیم نوع کادری را که می خواهیم ایجاد کنیم مشخص کنیم. در اینجا می توانید برخی از

مقادیری را که می توانیم برای **border-style** استفاده کنیم را ببینید:

none

با استفاده از این مقدار تعیین می کنیم که نمی خواهیم کادر داشته باشیم. در صورتی که مقداری

غير از none تعيين شود كادر خواهيم داشت.

solid

DIV have a solid border This

dashed

DIV have a dashed border This

dotted

DIV have a dotted border This

double

DIV have a double border This

groove

DIV have a groove border This

hidden

DIV have a hidden border This

inset

DIV have a inset border This

outset

DIV have a outset border This

ridge

DIV have a ridge border This

همانطور که می بینید عرض این کادرها بیش از آن چیزی است که ما نیاز داریم چون ما برای این

کادرها عرضی مشخص نکردیم. اگر بخواهیم عرض این کادرها به اندازه مشخصی باشد باید از

پارامتر **width** که قبلاً با آن آشنا شدیم استفاده کنیم. به مثال زیر توجه کنید:

این کادر عرض 300 برای `<div style="width:300px; border-style:solid">`

`</div>` پیکسل تعیین شده است

نتیجه به این صورت خواهد بود:

برای این کادر عرض 300 پیکسل تعیین شده است.

border-width

این پارامتر به ما اجازه می دهد تا میزان ضخامت کادرها را تعیین کنیم. این پارامتر می تواند مقادیر زیر را داشته باشد:

مقدار عددی بر حسب پیکسل

در این روش می توانیم مقدار ضخامت کادر را بر حسب پیکسل تعیین کنیم. به مثال زیر توجه کنید:

این کادر 10 `<div style="border-style:solid; border-width:10px">`

`</div>` پیکسل ضخامت دارد

این هم کادر به دست آمده:

این کادر 10 پیکسل ضخامت دارد

thin

این مقدار باعث خواهد شد تا کادر ما باریک باشد. مانند زیر:

ضخامت این کادر `<div style="border-style:solid; border-width:thin">`

`</div>` است در نظر گرفته شده `thin`

این هم کادر به دست آمده:

ضخامت این کادر `thin` در نظر گرفته شده است.

thick

این مقدار به ما یک کادر ضخیم خواهد داد. به مثال زیر توجه کنید:

ضخامت این کادر `<div style="border-style:solid; border-width:thick">`

`</div>` است در نظر گرفته شده `thick`

نتیجه به صورت زیر خواهد بود:

ضخامت این کادر `thick` در نظر گرفته شده است.

border-color

شما می توانید در صورت تمایل رنگ کادرهای خود را تغییر دهید. برای این کار باید از پارامتر

`border-color` استفاده کنید. برای مقدار این پارامتر می توانید از نام رنگها استفاده کنید. در

صورتی که از نام رنگها استفاده کنید فقط می توانید از تعداد کمی از رنگها استفاده کنید. راه

دیگری که برای انتخاب رنگ وجود دارد استفاده از معادل هگزادسیمال رنگها است که نسبت به

نام رنگها انعطاف پذیری بیشتری دارد. اگر جدول معادل هگزا دسیمال رنگها را می خواهید

[اینجا را کلیک کنید.](#)

به یک نمونه برای تغییر رنگ کادر توجه کنید:

استفاده از نام رنگها:

`<div style="border-style:solid; border-color:green">` این کادر به رنگ

`</div>`.سبز خواهد بود

این کادر به رنگ سبز خواهد بود.

استفاده از معادل هگزا دسیمال رنگها:

`<div style="border-style:solid; border-color:#008000">` این کادر به

`</div>`.رنگ سبز خواهد بود

این کادر به رنگ سبز خواهد بود.

تنظیم حاشیه و فاصله از لبه در عناصر صفحه

حاشیه ها

margin-left

این گزینه به ما اجازه می دهد تا برای هر یک از اجزای صفحه حاشیه چپ مشخصی را تعیین کنیم. در حقیقت این گزینه مقدار فاصله اجزا را از سمت چپ صفحه مشخص می کند. به یک مثال در این مورد توجه کنید:

دویست پیکسل این `<div style="margin-left:200px; border:double">` `</div>` از سمت چپ صفحه فاصله دارد

همانطور که می بینید کد بالا یک `DIV` را با کادر مضاعف و `200` پیکسل حاشیه از چپ نشان می دهد:

این `DIV` دویست پیکسل از سمت چپ صفحه فاصله دارد.

`margin-right`

این گزینه هم مانند گزینه بالا عمل می کند با این تفاوت که مقدار حاشیه را از سمت راست تعیین می کند:

دویست `DIV` این `<div style="margin-right:200px; border:double">` `</div>` پیکسل از سمت راست صفحه فاصله دارد

همانطور که می بینید کد بالا یک `DIV` را با کادر مضاعف و `200` پیکسل حاشیه از راست نشان می دهد:

این `DIV` دویست پیکسل از سمت راست صفحه فاصله دارد.

margin-top

این گزینه هم مانند گزینه های بالا کار می کند با این تفاوت که میزان حاشیه را از بالای صفحه

تعیین می کند:

`<div style="margin-top:100px; border:double">` این DIV صد پیکسل از

`</div>` بالای صفحه فاصله دارد

این DIV صد پیکسل از بالای صفحه فاصله دارد.

margin-bottom

این گزینه هم مانند گزینه های بالا کار می کند با این تفاوت که میزان حاشیه را از پایین صفحه

تعیین می کند:

`<div style="margin-bottom:100px; border:double">` این DIV صد

`</div>` پیکسل از پایین صفحه فاصله دارد

این DIV صد پیکسل از پایین صفحه فاصله دارد.

Padding

padding-right

پارامتر padding در CSS مشابه شناسه cellpadding در جدولهاست که با این تفاوت که

می توان آنرا برای بالا، پایین، چپ و راست به صورت جداگانه تعریف کرد به جای اینکه به صورت

یکجا تعیین شود. در اینجا می توانید یک نمونه از **padding** را که برای راست تعریف شده است را ببینید:

این برای `<div style="padding-right:100px; border-style:double">`
`</div>` تعریف شده است **padding** معادل 100 پیکسل **DIV**

می توانید نتیجه را در اینجا مشاهده کنید:

برای این **DIV** معادل 100 پیکسل **padding** تعریف شده است.

padding-top

مانند قسمت قبل است با این تفاوت که به جای راست برای بالا تعریف می شود:

این برای `<div style="padding-top:30px; border-style:double">`
`</div>` تعریف شده است **padding** معادل 30 پیکسل **DIV**

می توانید نتیجه را در اینجا مشاهده کنید:

برای این **DIV** معادل 30 پیکسل **padding** تعریف شده است.

padding-left

مانند **padding-right** است با این تفاوت که برای چپ تعیین می شود.

padding-bottom

مانند padding-top است با این تفاوت که برای پایین تعیین می شود.

به یاد داشته باشید که در صورتی که پارامتر padding بدون right, left, top یا bottom به کار رود مقدار وارد شده برای هر چهار جهت مورد استفاده قرار می گیرد، البته در مورد margin هم به همین صورت است:

در `<div style="padding:35px; width:250px; border-style:solid ">`

`</div>` میگذارد بر روی هر چهار جهت تأثیر padding پارامتر DIV این

این هم نتیجه:

در این DIV پارامتر padding بر روی هر چهار جهت تأثیر میگذارد.

تغییر موقعیت عناصر صفحه با استفاده از CSS

position

ویژگی position به ما اجازه می دهد تا تعیین کنیم که آیا می خواهیم موقعیت یک عنصر در صفحه مشخص شود یا نه. اگر می خواهیم آن عنصر موقعیت مشخصی داشته باشد موقعیت عنصر را در صفحه تعیین می کند. چندین مقدار را می توان برای این ویژگی قرار داد. در اینجا می توانید این مقادیر را با کمی توضیح در مورد هر کدام ببینید:

: static

این مقدار پیش فرضی است که در صورت تعریف نشدن **position** در صفحه استفاده می شود.
با این مقدار اجزا در همان جایی که کد آنها را می نویسید دیده می شوند.

: absolute

این مقدار به ما اجازه می دهد تا یک عنصر از صفحه را در هر موقعیتی نسبت به بالا، راست، پایین یا چپ صفحه که بخواهیم نمایش دهیم.

: fixed

این گزینه به ما اجازه می دهد تا موقعیت یک عنصر صفحه را در پنجره مرورگر مشخص کنیم. در صورتی که برای یک عنصر از این ویژگی استفاده کنیم، موقعیت آن عنصر حتی در صورتی که به بالا و پایین صفحه برویم یا اندازه پنجره مرورگر را تغییر دهیم به همان صورت قبلی باقی می ماند. البته اینترنت اکسپلورر در ویندوز از این ویژگی پشتیبانی نمی کند ولی اینترنت اکسپلورر در مکینتاش همچنین مرورگرهای نت اسکایپ از این ویژگی پشتیبانی می کنند.

static

این ویژگی در حقیقت مشخص می کند که عنصر مربوطه باید در محلی که کد آن نوشته شده نمایش داده شود. وقتی که از این ویژگی استفاده می کنیم ویژگیهای **top, left, bottom, right** کاربردی ندارند.

این ویژگی به صورت پیش فرض از طرف مرورگر برای صفحه مورد استفاده قرار می گیرد.

fixed

چون اینترنت اکسپلورر بیشتر از سایر مرورگرها مورد استفاده قرار می گیرد و از ویژگی **fixed** در CSS پشتیبانی نمی کند زیاد به این ویژگی نمی پردازیم. اما در هر صورت ویژگی جالبی است. در اینجا فقط به ذکر یک مثال بسنده می کنیم:

در این مثال از ویژگی **position:fixed** برای تعیین موقعیت یک **DIV** استفاده می کنیم:

اگر با مرورگر `<div style="position:fixed; top:50%; float:right">`

صفحه بازدید کنید محتویات این از این **opera** و **firefox** یا مرورگرهایی مثل **netscape**

`</div>` آیند در میانه صفحه و سمت راست آن به نمایش در می **div**

اگر می خواهید ببینید این ویژگی چگونه کار می کند لینک زیر را با یکی از مرورگرهای **Opera, Netscape** یا **Firefox** مشاهده کنید و صفحه را با استفاده از اسکرول بار به بالا و پایین ببرید:

[استفاده از ویژگی **position:fixed**](#)

absolute

ویژگی **position:absolute** مشخص می کند که عنصر مربوطه باید در محل تعیین شده به نمایش در آید نه محلی که کد آن نوشته شده است.

در اینجا می توانید یک مثال را در مورد استفاده از ویژگی **position:absolute** در صفحه ببینید:

```
<div style="position:absolute; top:65px; right:20px; border:double #0033cc; width:350px; color:#ffff00; background-color:#33cccc; height: 40px;"> این div در موقعیت تعیین شده در کد آن نمایش می شود داده می </div>
```

همانطور که در کد بالا می بینید برای DIV موقعیتی با فاصله 65 پیکسل از بالای صفحه و 20 پیکسل از سمت راست صفحه تعیین کرده ایم. با کلیک کردن بر روی لینک زیر می توانید نتیجه را مشاهده کنید:

[استفاده از absolute برای ویژگی position](#)

top

این ویژگی موقعیت یک عنصر را از بالای صفحه تعیین می کند. (ویژگیهای top, left, bottom, right بیشتر زمانی مورد استفاده قرار می گیرند که برای یک عنصر ویژگی position تعیین شده باشد.)

در مثال قبل همانطور که ملاحظه می کنید از ویژگی top:65px استفاده شده است که این ویژگی معین می کند که DIV با فاصله 65 پیکسل از بالای پنجره مرورگر به نمایش در می آید. مقداری که برای ویژگیهای top, bottom, left, right در نظر گرفته می شود به صورت می تواند باشد. یا به صورت عددی در واحد پیکسل که در این صورت باید پس از عدد مربوطه از حروف اختصاری px استفاده شود یا به صورت درصد که پس از عدد با id از علامت درصد (/.) استفاده شود.

bottom

کاربردی همانند ویژگی بالا دارد با این تفاوت که میزان فاصله عناصر را از پایین پنجره مرورگر مشخص می کند. در مثال بعدی از این ویژگی استفاده خواهیم کرد.

right

این ویژگی فاصله عناصر را از سمت راست صفحه تعیین می کند. همانطور که در مثال قبل می بینید، از این ویژگی به صورت `right:20px` استفاده شده است یعنی `DIV` مربوطه با فاصله 20 پیکسل از سمت راست پنجره مرورگر قرار خواهد گرفت.

left

این ویژگی موقعیت عنصر را از سمت چپ پنجره مرورگر تعیین می کند.
در مثال زیر از ویژگیهای `bottom` و `left` برای همان `DIV` مثال قبل استفاده می کنیم:

```
<div style="position:absolute; bottom:65px; left:20px; border:double #0033cc; width:350px; color:#ffff00; background-color:#33cccc; height:40px;"> این div </div>
```

برای اینکه نتیجه را مشاهده کنید لینک زیر را کلیک کنید:

[استفاده از ویژگیهای `left` و `bottom`](#)

در مثال ذکر شده `DIV` به اندازه 65 پیکسل از پایین و 20 پیکسل از سمت چپ پنجره مرورگر فاصله دارد.

ویژگیهای متن در CSS

در این قسمت برخی از ویژگیهای متن را که در CSS استفاده می شوند به شما معرفی می کنیم. با کمک این ویژگیها می توانیم نحوه قرار گرفتن متن در صفحه، فاصله بین حروف و بسیاری از خواص یک متن را تغییر دهیم.

letter-spacing

این ویژگی به ما اجازه می دهد تا فاصله میان حروف یک کلمه را در یک متن تعیین کنیم. این ویژگی می تواند مقادیری را در مقیاس پیکسل به خود بگیرد یا از مقدار `normal` استفاده کند. گزینه پیش فرض برای این ویژگی همان `normal` است. در اینجا به ذکر یک مثال در مورد این ویژگی می پردازیم:

فرض کنید می خواهیم از این ویژگی برای متن موجود در یک تگ `DIV` استفاده کنیم:

```
<div style="letter-spacing:5px">
```

فاصله 5 پیکسل از یکدیگر نمایش داده خواهند شد حروف این متن با

```
</div>
```

نتیجه را به صورت زیر مشاهده خواهید کرد:

حروف این متن با فاصله 5 پیکسل از یکدیگر نمایش داده

خواهند شد.

text-align

این ویژگی به ما اجازه می دهد تا نحوه قرارگیری قسمتی از متن را از لحاظ راست چین، چپ چین یا وسط چین بودن تنظیم کنیم.

این ویژگی می تواند مقادیر زیر را به مورد استفاده قرار دهد:

: left

با استفاده از این مقدار می توانیم محتویات یک تگ را در سمت چپ صفحه نمایش دهیم.

: right

این مقدار معین می کند که محتویات یک تگ باید در سمت راست صفحه نمایش داده شوند.

: center

این مقدار محتویات یک تگ را به صورت وسط چین تعریف میکند.

برای مثال اگر می خواهید متنی که در یک DIV قرار دارد به صورت چپ چین نمایش داده شود می توانیم از این ویژگی به مانند زیر استفاده کنیم:

محتویات `<div style="text-align:left; border:thin dashed #00CCFF">`

`</div>` در سمت چپ صفحه نمایش داده می شوند DIV این

نتیجه به صورت زیر خواهد بود:

محتویات این DIV در سمت چپ صفحه نمایش داده می شوند.

text-decoration

این ویژگی به ما اجازه می دهد تا برای قسمت خاصی از متن خاصیت‌های از قبیل زیرخط تعریف کنیم. مثلاً فرض کنید می خواهیم یک متن را با خطی بالای آن به وجود بیاوریم. در اینجا می توانیم از این ویژگی به صورت زیر استفاده متین:

<div style="text-decoration: overline">توجه کنید به این متن</div>

این هم نتیجه:

به این متن توجه کنید!!

این ویژگی چندین مقدار دارد که در اینجا به آنها می پردازیم:

none

این گزینه به صورت پیش فرض بر روی متن اعمال می شود (البته برای لینکها به صورت پیش فرض **underline** اعمال می شود) و در نتیجه آن متنها به صورت ساده و معمولی نمایش داده می شوند. در صورتی که بخواهیم لینکهایی در متن ما بدون زیر خط نشان داده شوند می توانیم از این گزینه برای تگ لینک استفاده کنیم.

: underline

از این گزینه می توان برای زیر خط دار کردن متنها استفاده کرد.

: overline

این گزینه برای نمایش خط بالای متن استفاده می شود.

: line-through

این گزینه یک خط را در میانه متن ایجاد می کند و آنرا به صورت خط خورده نشان می دهد.

: blink

از این گزینه می توانیم برای چشمک زن کردن متن استفاده کنیم. البته این گزینه توسط مرورگر

اینترنت اکسپلورر حمایت نمی شود و فقط در مرورگرهای Netscape نمایش داده می شود.

در اینجا یک لینک را می بینید که با استفاده از ویژگی بالا بدون زیر خط نمایش داده می شود:

مورد استفاده در `` متن

`` لینک

این هم لینک بدون زیر خط:

متن مورد استفاده در لینک

ویژگیهای مربوط به متن در CSS

در این قسمت به ادامه ویژگیهایی که بر روی متن ها اثر می گذارند می پردازیم.

text-transform

این ویژگی مشخص می کند که متن با استفاده از حروف بزرگ نوشته شود یا حروف کوچک و یا

به صورت معمولی البته بدون آنکه در حروفی که تایپ شده است تغییری به وجود آید. با این

روش اگر خواستید قسمتی از متن مثلاً برای تأکید بیشتر با حروف بزرگ مشاهده شود نیازی

نیست که دوباره آن قسمت را از اول تایپ کنید. البته این ویژگی در زبان فارسی چندان استفاده ای ندارد ولی در زبانهایی مثل انگلیسی می توانید آنرا به کار ببرید.
به یک مثال در مورد این ویژگی توجه کنید:

<div style="text-transform: uppercase">This is an example of uppercase in text-transform property.</div>

این هم نتیجه:

THIS IS AN EXAMPLE OF UPPERCASE IN TEXT-TRANSFORM PROPERTY.

همانطور که مشاهده می کنید بدون استفاده از حروف بزرگ در نوشتن کد، متن ما با حروف بزرگ نمایش داده می شود.

این ویژگی چند مقدار دارد که در اینجا مشاهده می کنید:

: none

با انتخاب این گزینه متن به همان صورتی که تایپ شده است نمایش داده می شود.

: lowercase

با انتخاب این گزینه متن با حروف کوچک نمایش داده می شود.

: uppercase

این گزینه متن را با حروف بزرگ نمایش می دهد.

: capitalize

این گزینه حروف اول کلماتی را که با حروف کوچک نوشته شده اند با حروف بزرگ و بقیه حروف را با حروف کوچک نمایش می دهد. به مثال زیر توجه کنید:

```
<div style="text-transform: capitalize">this is an example of  
capitalize value for text-transform propety</div>
```

متن به صورت زیر نمایش داده می شود:

this is an example of capitalize value for text-transform propety

line-height

این ویژگی به ما امکان می دهد تا ارتفاع سطرها را مشخص کنیم. این ارتفاع می تواند به صورت درصد یا در مقیاس پیکسل باشد و فاصله بین سطرها را مشخص می کند.

برای مثال اگر بخواهیم سطرها از هم 50 پیکسل فاصله داشته باشند می توانیم به صورت زیر عمل کنیم:

```
<div style="line-height: 50px">   
<br>20 پیکسل بعدی بین این سطر و سطر  
</div>. فاصله وجود دارد
```

نتیجه به صورت زیر خواهد بود:

بین این سطر و سطر بعدی

20 پیکسل فاصله وجود دارد.

text-indent

این ویژگی مقدار تو رفتگی متن را در سطر اول هر قسمت (مثلاً سطر اول هر پاراگراف) مشخص می کند. این مقدار می تواند بر حسب پیکسل یا در صد باشد. در مثال زیر این مقدار تو رفتگی را 30 پیکسل در نظر می گیریم:

```
<div style="text-indent: 30px">
```

```
<br>با 30 پیکسل تو رفتگی نمایش داده می شود سطر اول این قسمت
```

```
<br>صورت عادی و در حالی که بقیه سطرها به
```

```
بدون تو رفتگی نمایش داده می شوند.
```

```
</div>
```

این هم نتیجه:

سطر اول این قسمت با 30 پیکسل تو رفتگی نمایش داده می شود.

در حالی که بقیه سطرها به صورت عادی و

بدون تو رفتگی نمایش داده می شوند.

تنظیم فونت صفحه با استفاده از استایل

در این قسمت به روش تنظیم فونت متن با استفاده از استایل می پردازیم. بسیاری از اعمالی را که قبلاً با تگ `` انجام می دادیم می توانیم با استایل انجام دهیم.

font-family

ویژگی font-family مشخص می کند که متن با چه فونتی باید نشان داده شود. این ویژگی در حقیقت جانشین شناسه face که در تگ font استفاده می شود است. گزینه پیش فرض برای

فونت ممکن است در مرورگرهای مختلف تفاوت داشته باشد، اما معمولاً Times New Roman است.

در مثال زیر از ویژگی font-family برای تغییر فونت متن داخل یک div به Arial استفاده می کنیم:

`<div style="font-family: Arial"> نمایش داده می Arial این متن با فونت </div>` شود.

نتیجه را به صورت زیر مشاهده می کنید:

این متن با فونت Arial نمایش داده می شود.

البته باید به خاطر داشته باشید که اگر بازدید کننده فونت مورد نظر شما را نداشته باشد، مرورگر به جای آن از فونت پیش فرض خود استفاده می کند. برای جلوگیری از بروز این مشکل باید از فونتهایی استفاده کنید که معمولاً به همراه سیستم عامل نصب می شوند و در اکثر سیستمها موجود هستند.

font-size

این ویژگی مشخص کننده اندازه فونت در صفحه می باشد. گزینه پیش فرض برای این ویژگی medium است.

فرض کنید می خواهیم متن درون یک DIV با اندازه 30 پیکسل نمایش داده شود:

```
<div style="font-size: 30px">30 پیکسل نمایش داده می این متن با اندازه 30  
</div> .شود
```

نتیجه را ملاحظه کنید:

این متن با اندازه 30 پیکسل نمایش داده می شود.

font-style

کار این ویژگی تقریباً شبیه تگ <I> در HTML است، اما چون استفاده از تگهایی نظیر , , <U> و <I> در حال کمرنگ شدن است بهتر است از استایل برای مورب نشان دادن متن استفاده شود. این ویژگی می تواند مقادیر زیر را داشته باشد:

normal

italic

oblique

به یک مثال در مورد این ویژگی توجه کنید:

```
<div style="font-style: oblique">مشاهده می این متن به صورت مورب  
</div> .شود
```

نتیجه به این صورت خواهد بود:

این متن به صورت مورب مشاهده می شود.

font-weight

این ویژگی به ما امکان می دهد تا میزان ضخامت متن را تعیین کنیم. می توانیم از مقادیر زیر

برای این ویژگی استفاده کنیم:

normal

bold

bolder

lighter

100

200

300

400

500

600

700

800

900

در اینجا به ذکر یک مثال می پردازیم:

مشاهده می این متن تقریباً ضخیم `<div style="font-weight: 700">`

`</div>` شود.

نتیجه به این صورت خواهد بود:

این متن تقریباً ضخیم مشاهده می شود.

نحو تغییر رنگ متن با استفاده از CSS

color

از ویژگی **color** برای تغییر رنگ متن قسمت‌های مختلف صفحه استفاده می شود. برای مثال می

توانید رنگ متن یک جدول را تغییر دهیم و یا قسمتی از متن صفحه را که با تگ **SPAN**

مشخص کرده ایم با رنگ متفاوتی با بقیه متن نمایش دهیم. این تگ از آن جهت که کاربرد تگ

`` و در نتیجه شناسه **color** که در تگ فونت استفاده می شود، در **HTML4** در حال

محدود شدن است اهمیت بیشتری پیدا می کند.

البته در قسمت‌های قبلی هم از این ویژگی استفاده کرده ایم اما در باره آن توضیحی داده نشده

است.

مانند **COLOR** در تگ **FONT** رنگها در این ویژگی به دو صورت تعیین می شوند. شکل کلی

این ویژگی به صورت زیر است:

نام رنگ مورد نظر: **color:**

معادل هگزادسیمال رنگ: **color:#**

اگر می خواهید از رنگهای معمولی که با نام شناخته می شوند استفاده کنید کافی است تا نام رنگ مورد نظر خود را در این ویژگی استفاده کنید. این روش ساده تر است ولی انعطاف پذیری کمتری هم دارد چون تنها می توانید رنگ مورد نظر خود را از بین تعداد اندکی نام رنگ انتخاب کنید. در صورتی که می خواهید از رنگهای بیشتری استفاده کنید و بهتر بتوانید رنگ مورد نظر خود را انتخاب کنید می توانید از معادل هگزادسیمال رنگها استفاده کنید. در این صورت می توانید از **216** رنگ مورد استفاده در وب رنگ مورد نظر خود را انتخاب کنید.

اگر جدول مربوط به معادل هگزادسیمال رنگها را می خواهید.

در اینجا به ذکر چند مثال برای استفاده از ویژگی **color** می پردازیم. ما در اینجا از شناسه **style** برای تعریف کردن این ویژگی استفاده می کنیم اما به خاطر داشته باشید که می توانیم این ویژگی را برای یک تگ مشخص یا به صورت **class** و **id** در بخش **HEAD** متن یا استایل خارجی نیز به کار ببریم.

برای مثال در استفاده از **color** برای تغییر رنگ متن یک جدول می توانیم کد جدول را به

صورت زیر بنویسیم:

```
<table border="1" style="color: red">
```

```
<tr>
```

```
<td>
```

قرمز نمایش داده می شود این متن به رنگ

```
</td>
</tr>
</table>
```

نتیجه کد بالا با کدی که در پایین آمده است یکسان است:

```
<table border="1" style="color: #FF0000">
<tr>
<td>
    رنگ قرمز نمایش داده می شود این متن به
</td>
</tr>
</table>
```

نتیجه هر دو کد بالا به صورت زیر است:

این متن به رنگ قرمز نمایش داده می شود.

به یاد داشته باشید که می توانید از ویژگی **color** برای هر کدام از سطرها یا سلولهای یک جدول به صورت جداگانه استفاده کنید. فقط کافیست تا **style="color: XXX"** را در هر قسمت قرار دهید و رنگ مورد نظر خود را به جای **XXX** قرار دهید.

کار با تصویر زمینه در طراحی صفحات با استایل

ما با استفاده از شناسه **background** در تگ **BODY** می توانستیم برای صفحه تصویر زمینه مشخص کنیم. اما با استفاده از استایل می توانیم کنترل بیشتری بر این تصویر زمینه داشته باشیم چه این تصویر زمینه برای کل صفحه استفاده شود چه برای قسمت مشخصی از صفحه. برای مثال می توانیم تصویر زمینه را به گونه ای تعریف کنیم که تکرار نشود، ثابت باشد، یا برای آن نقطه شروع تعیین کنیم. برخی از این خواص را می توانید در جدول زیر ملاحظه کنید:

| ویژگی | مقادیر قابل استفاده | توضیحات |
|---|--|---|
| background-image: url(url) | آدرس محل قرار گرفتن تصویر | مشخص کننده محل قرار گرفتن تصویر در اینترنت است |
| background-repeat: مقدار ویژگی | repeat, no-repeat, repeat-x, repeat-y | نوع تکرار تصویر را مشخص می کند |
| background-attachment: مقدار ویژگی | scroll, fixed | مشخص می کند که آیا تصویر زمینه با اسکرول کردن صفحه حرکت می یا در جای خود ثابت می ماند کند |
| background-position: مقدار مقدار اول دوم | مقدار اول: top, center, bottom, پیکسل، عددی در عددی در مقیاس مقیاس درصد مقدار دوم: left, center, right, عددی در مقیاس پیکسل، عددی در مقیاس درصد | موقعیت تصویر زمینه در صفحه را مشخص می کند |

برای استفاده از این ویژگیها و کاربرد آنها برای تگ **BODY** می توانید یک استایل در قسمت

HEAD متن تعریف کنید یا از استایل خارجی استفاده کنید. مثلاً برای تصویر زمینه صفحات

سایت ما می توان در قسمت **HEAD** متن یک استایل به صورت زیر تعریف کرد:

```
<style type="text/css">
<!--
body {
background-
image:url(http://www.neopersia.org/images/index.jpg);
background-repeat:repeat-y }
-->
</style>
```

در این صورت تصویر زمینه ای مشابه سایت ما خواهید داشت. می توانید آنرا امتحان کنید.

background-repeat

همانطور که در درس قبل ملاحظه کردید در مورد **background-image** توضیحاتی

داده شد. در اینجا به سایر ویژگیهای مربوط به تصویر زمینه می پردازیم.

این ویژگی تعیین کننده نوع تکرار شدن تصویر زمینه است. در اینجا مقادیری را که می توان به

این ویژگی اختصاص داد بیان می کنیم:

repeat

این مقدار تعیین می کند که تصویر زمینه در صورتی که از اندازه پنجره کوچکتر بود هم به صورت

افقی و هم عمودی تکرار شود تا همه زمینه پنجره را بپوشاند.

no-repeat

این گزینه مشخص می کند که تصویر زمینه نباید تکرار شود حتی اگر از اندازه پنجره کوچکتر بود.

repeat-x

این گزینه مشخص می کند که تصویر باید به صورت افقی تکرار شود نه عمودی.

repeat-y

این گزینه مشخص می کند که تصویر زمینه باید به صورت عمودی تکرار شود.

background-attachment

از این ویژگی برای تعیین موقعیت تصویر زمینه در هنگام بالا و پایین بردن صفحه به وسیله اسکرول بار در پنجره مرورگر است. مقادیر زیر را می توان برای این ویژگی استفاده کرد:

scroll

اگر این گزینه را انتخاب کنید تصویر زمینه به همراه محتویات صفحه به بالا و پایین می رود.

fixed

با انتخاب این گزینه مشخص می کنید که تصویر زمینه در هنگام حرکت کردن محتویات صفحه باید ثابت بماند و حرکت نکند.

کد زیر را می توان برای مشخص کردن تصویر زمینه ثابت در بخش **HEAD** متن استفاده کرد. در

این مثال لوگوی سایت ما به عنوان زمینه ثابت در نظر گرفته شده است. برای مشاهده یک نمونه

از این تصویر زمینه

```
<style type="text/css">
<!--
body {
background-
image:url(http://www.neopersia.org/images/cssBGtest.jpg);
background-attachment:fixed;
background-repeat:no-repeat;
background-position:center center }
-->
</style>
```

این مثال را به یاد داشته باشید. دوباره از آن استفاده خواهیم کرد. در درس بعدی به سایر ویژگیهای مرتبط با تصویر زمینه می پردازیم.

تنظیمات موقعیت عکس زمینه عناصر صفحه با CSS

background-position

این ویژگی جهت تعیین موقعیت تصویر زمینه در صفحه استفاده می شود. این ویژگی بیشتر زمانی

مورد استفاده قرار می گیرد که برای تصویر زمینه از **no-repeat**:

repeat استفاده شده باشد. این ویژگی می تواند دو مقدار را همزمان به خود اختصاص دهد. با

این دو مقدار می توانیم موقعیت عمودی و افقی تصویر در صفحه را تعیین کنیم. توجه داشته

باشید که این دو مقدار باید با یک فاصله خالی از هم جدا شوند نه کاما یا نقطه ویرگول.

دو مقداری که می توان به این ویژگی اختصاص داد از مقادیر زیر انتخاب می شوند:

top, center, or bottom

left, center, or right

برای این ویژگی باید از هر دسته از مقادیر بالا فقط یکی انتخاب شود. در اینجا می توانید برخی از موارد قابل استفاده را مشاهده کنید:

background-position:top **right**

این ویژگی تصویر زمینه را در بالا و سمت راست صفحه نمایش می دهد.

background-position:center **right**

این ویژگی تصویر را در سمت راست وسط صفحه نمایش می دهد.

background-position:bottom **center**

این ویژگی تصویر زمینه را در پایین مرکز صفحه نمایش می دهد.

می توانیم به جای گزینه های بالا از مقادیری در مقیاس پیکسل یا درصد قرار دهیم تا تصویر

زمینه را در هر فاصله ای از بالا و سمت چپ صفحه که بخواهیم قرار دهیم. در این روش فقط

کافیست مقدار فاصله از سمت چپ صفحه را نوشته و مقیاس آن را مشخص کنیم و با یک فاصله

مقدار فاصله از بالای صفحه همراه با مقیاس آنرا بنویسیم. به مثال زیر توجه کنید:

در این مثال از لوگوی سایت ما به عنوان زمینه صفحه استفاده شده است که در فاصله 20

background- پیکسل از بالا و 40 پیکسل از چپ صفحه قرار می گیرد. در ضمن از ویژگی **background-**

fixed attachment استفاده شده است تا در صورت حرکت دادن صفحه زمینه به صورت

ثابت قرار بگیرد. کد زیر را به منظور تعیین تصویر زمینه صفحه در بخش HEAD صفحه قرار می دهیم:

```
<style type="text/css">  
<!--  
body {  
background-  
image:url(http://www.neopersia.org/images/cssBGtest.jpg);  
background-attachment:fixed;  
background-position:40px 20px }  
-->  
</style>
```

برای مشاهده نتیجه

با ترکیب کردن ویژگی‌هایی که توضیح داده شد می توانیم انواع تصویر زمینه مختلف و جالبی را به وجود بیاوریم. می توانید این کار را انجام دهید و تنوع آنرا ملاحظه کنید.

تغییر شکل نشانگر ماوس با استفاده از استایل

با استفاده از استایل این امکان وجود دارد که شکل نشانگر ماوس را وقتی روی قسمت مشخصی از صفحه قرار می‌گیرد تغییر دهیم. به این منظور می‌توانیم از ویژگی زیر در استایل مربوط به تگ مورد نظر استفاده کنیم:

نوع نشانگری که مورد نظر ماست: **cursor:**

برای مثال فرض کنید می‌خواهیم وقتی نشانگر ماوس روی یک قسمت متن قرار می‌گیرد به شکل ساعت شنی در آید. می‌توانیم به صورت زیر عمل کنیم (ما برای مشخص شدن متن از رنگ قرمز در تگ **span** استفاده کرده ایم):

روی این متن قرار ماوس خود را ****
**** دهید.

نتیجه را می‌توانید اینجا مشاهده کنید:

ماوس خود را روی این متن قرار دهید.

اگر بخواهید وقتی ماوس کاربر روی یک قسمت مشخص از صفحه مثلاً جدولها قرار می‌گیرد شکل خاصی داشته باشد می‌توانید در بخش **HEAD** صفحه این ویژگی را برای جدول تعریف کنید تا در همه جدولهای صفحه به کار گرفته شود. از **class** و **id** هم می‌توانید استفاده کنید. روش ساده تر هم استفاده از شناسه **style** در تگهای **HTML** است.

در جدول زیر برخی از مقادیر مختلف این ویژگی که شکل نشانگر ماوس را عوض می کنند آورده شده است. می توانید ماوس خود را بر روی توضیحات این گزینه ها نگه دارید تا ببینید به چه شکلی در می آید.

| مقدار قابل استفاده | توضیحات |
|--------------------|---|
| Auto | این گزینه نشانگر ماوس را با توجه به تنظیمات مرورگر کاربر تغییر می دهد. |
| Crosshair | نشانگر ماوس به صورت به علاوه در می آید. |
| Default | نشانگر ماوس به صورت همان فلش همیشگی در می آید. |
| Move | نشانگر به شکلی در می آید که در هنگام جا به جا کردن چیزی به خود می گیرد. |
| Pointer | نشانگر به شکل دست در می آید. |
| Help | یک علامت سؤال کنار پیکان نشانگر ماوس قرار می گیرد. |
| Text | به شکلی در می آید که بر روی متنها به خود می گیرد. |
| Wait | به شکل ساعت شنی در می آید. |
| n-resize | یک فلش به سمت شمال |
| s-resize | یک فلش به سمت جنوب |
| e-resize | یک فلش به سمت مشرق |
| w-resize | یک فلش به سمت مغرب |
| ne-resize | یک فلش به سمت شمال شرقی |
| nw-resize | یک فلش به سمت شمال غربی |
| se-resize | یک فلش به سمت جنوب شرقی |
| sw-resize | یک فلش به سمت جنوب غربی |

اعمال تغییر بر روی لینکها در صفحه با استفاده از استایل

با استفاده از CSS می توانیم نحوه نمایش دادن لینکها را در صفحه تنظیم کنیم. به صورتی که استفاده از جاوا اسکریپت در صفحه تا حد زیادی کم می شود. از طریق استفاده از استایل می توانیم رنگ لینکها، اندازه فونت آنها، زیر خط لینکها، رنگ زمینه هر لینک و بسیاری دیگر از خواص لینکها را تغییر دهیم.

برای اینکه بتوانیم از همه خواصی که برای لینکها در CSS وجود دارد استفاده کنیم بهتر است تا استایل را در بخش HEAD متن خود یا در فایل های CSS خارجی که به صفحه پیوند می شوند استفاده کنیم. در هر دو صورت مراحل انجام کار شبیه هم است. ما در اینجا به استایل را در بخش HEAD متن تعریف می کنیم.

برای کار با لینکها در CSS می توانیم از مشخصات یک لینک را به چهار بخش تقسیم کنیم که در زیر می توانید آنها را مشاهده کنید:

a و a:link

این مشخصه ها نحوه نمایش لینکهایی را که در صفحه وجود دارند و هنوز روی آنها کلیک نشده است تنظیم می کنند.

a:hover

این مشخصه وضعیت لینک را وقتی که نشانگر ماوس روی آن قرار دارن تنظیم می کند.

a:active

این قسمت مشخص کننده وضعیت لینکی است که بر روی آن کلیک می شود.

a:visited

این مشخصه برای لینکهایی که در بازدیدهای قبلی مورد استفاده قرار گرفته اند استفاده می شود. برای کار کردن با لینکها ما در قسمت HEAD متن از مشخصه های بالا استفاده می کنیم. یعنی کد زیر را در بخش HEAD متن وارد می کنیم اگر می خواهید از استایل خارجی استفاده کنید می توانید [نحوه فرا خوانی استایل خارجی به صفحه](#) را مطالعه کنید.

```
<style type="text/css">
<!--
a:link { مشخصات مورد نظر }
a:hover { مشخصات مورد نظر }
a:active { مشخصات مورد نظر }
a:visited { مشخصات مورد نظر }
-->
</style>
```

برای هر ویژگی بالا می توانید مشخصات خاص خود را تعریف کنید برای مثال می تواند رنگ از مشخصه **background-color** برای تعیین رنگ زمینه لینک وقتی که ماوس روی آن قرار می گیرد استفاده کنید یا از **color** برای تعیین رنگ متن لینکها. به طور کلی می توانید از بیشتر مشخصه هایی که برای متن ها کارایی دارند در مورد لینکها هم استفاده کنید برای مثال **font-family** و **font-size** و **border** و ...

در اینجا یک مثال مطرح می کنیم که در آن لینکهای صفحه به رنگ سبز نمایش داده می شوند با اندازه فونت pt12 و بدون زیرخط، وقتی که ماوس روی آنها قرار می گیرد رنگ آنها قرمز می شود و رنگ زمینه متفاوتی پیدا می کنند همچنین یک کادر نیز اطراف آنها به وجود می آید. در موقع کلیک کردن رنگ لینک سفید می شود و لینکهایی که در دفعات قبلی کلیک شده اند با زیر خط نمایش داده می شوند. برای ساختن لینکهایی با مشخصاتی که بیان شد می توانیم از کد زیر در قسمت HEAD متن یا استایل خارجی استفاده کنیم:

```
<style type="text/css">
<!--
a:link { color:green; font-size:12pt; text-decoration:none }
a:hover { color:red; background-color:#6699CC; border:#006699
2px solid }
a:active { color: white }
a:visited { text-decoration: underline }
-->
</style>
```

در قسمت زیر می توانید لینکهایی را که برای آنها از کد بالا استفاده شده است مشاهده کنید:

این لینکها با استفاده از کد بالا تنظیم شده اند.

ماوس خود را روی آنها قرار دهید.

می توانید بر روی آنها کلیک کنید.

همانطور که می بینید کدهایی که در قسمت HEAD این صفحه قرار دارند بر روی لینکهای بالا تأثیر گذاشته اند اما لینکها در سایر نقاط صفحه ویژگیهای متفاوتی دارند. دلیل این تفاوت این است که ما لینکهای بالا را در یک DIV قرار دادیم و برای آن یک ID در بخش HEAD صفحه تعریف کردیم و ویژگیهای مربوط به لینکهای بالا در قالب ID مشخص کردیم. در اینجا می توانید کدی را که ما مورد استفاده قرار داده ایم ببینید:

```
#pre a:link,#pre a { color:green; font-size:10pt; text-decoration:none }  
#pre a:hover { color:red; background-color:#6699CC;  
border:#006699 2px solid; text-decoration:none }  
#pre a:active { color: White }  
#pre a:visited { text-decoration: underline; color:#003399 }
```

کد مربوط به DIV را نیز به صورت زیر نوشتیم:

```
<div align="center" id="pre">  
لینکها کدهای مربوط به  
</div>
```

آموزش جاوا اسکریپت

جاوا اسکریپت یک زبان اسکریپت نویسی تحت وب است که برای افزایش پویایی صفحات HTML استفاده می شود. از این زبان می توان برای ارزیابی اعتبار فرمهای HTML قبل از ارسال به سرور، شناسایی مرورگر کاربران، ایجاد انواع منوها و بسیاری کارهای دیگر استفاده کرد.

نکته : باید توجه داشت که جاوا اسکریپت یک زبان طرف کاربر (Client Side) است توسط کامپیوتر شخصی کاربر تفسیر می شود. کدهای مربوط به جاوا اسکریپت توسط کاربر نهایی قابل مشاهده هستند پس توصیه می شود از آن برای مقاصد مهم نظیر اتصال به دیتابیس استفاده نشود.

می توان کدهای جاوا اسکریپت را به صورت تعبیه شده در صفحات HTML استفاده کرد و یا کدهای جاوا اسکریپت را در یک فایل جداگانه با پسوند JS ذخیره کرد ولی در هر صورت یان فایلها هم باید توسط یک صفحه HTML فراخوانی شده و نتیجه آنها در صفحات HTML نمایان خواهد شد.

بهتر است قبل از شروع به یادگیری جاوا اسکریپت با HTML و CSS آشنایی داشته باشد تا بتوانید به خوبی از قابلیت های جاوا اسکریپت استفاده کنید. برای یادگیری زبانهای ذکر شده می توانید از پیوندهای زیر استفاده کنید :

[آموزش HTML به زبان فارسی](#)

[آموزش سی اس اس \(CSS\)](#)

برای آزمایش کردن اسکریپتهایی که می نویسد باید از مرورگری استفاده کنید که از جاوا اسکریپت پشتیبانی می کند. قابل ذکر است که تقریباً همه مرورگرهای جدید از جاوا اسکریپت پشتیبانی می کنند از جمله این مرورگرها می توان به اینترنت اکسپلورر (IE) که در ویندوز به صورت پیش فرض نصب می شود، فایرفوکس (Firefox) و اپرا (Opera) اشاره کرد. به خاطر داشته باشید که نسخه های قبل از نسخه 4.0 اینترنت اکسپلورر و Netscape از جاوا اسکریپت پشتیبانی نمی کنند.

نحوه استفاده از کدهای جاوا اسکریپت در صفحه

به طور کلی به سه طریق می توانیم از جاوا اسکریپت در یک صفحه استفاده کنیم:

وارد کردن اسکریپت در بخش **<head>** صفحه

وارد کردن جاوا اسکریپت در بخش **<body>** صفحه

استفاده از فایل اسکریپت خارجی

به یاد داشته باشید که می توانید از همه روشهای بالا به طور همزمان برای وارد کردن جاوا

اسکریپت در صفحه استفاده کنید.

در دو روش اول کدهای جاوا اسکریپت به طور مستقیم وارد صفحه HTML می شوند. برای وارد

کردن جاوا اسکریپت در این دو روش از تگ **script** استفاده می شود. تمامی کدهای جاوا

اسکریپتی که می خواهید به طور مستقیم وارد صفحه کنید باید در تگ **script** قرار داشته

باشند وگرنه مرورگر به جای به کار بردن آنها را به صورت متن در صفحه نمایش می دهد. معمولاً کدهای جاوا اسکریپت با ساختار زیر وارد صفحه می شوند :

```
<script language="javascript" type="text/javascript">
```

دارید در این قسمت وارد می شوند دستورات جاوا اسکریپتی که قصد استفاده از آنها را

```
</script>
```

در تگ بالا برای مشخص کردن نوع موتور اسکریپتی که مرورگر باید برای تفسیر کردن کد استفاده کند از دو شناسه **language** و **type** استفاده می کنیم.

وارد کردن اسکریپت در بخش <head>

در هنگام باز شدن یک صفحه وب عناصر موجود در بخش <head> قبل از سایر اجزای صفحه لود می شوند. معمولاً هنگامی کدهای جاوا اسکریپت در بخش <head> متن استفاده می شود که بخواهیم مطمئن شویم به هنگام نیاز کدهای ما لود شده اند.

در اینجا یک مثال از استفاده از جاوا اسکریپت در <head> را مشاهده می کنید:

```
<html dir="rtl">
```

```
<head>
```

```
<script language="javascript" type="text/javascript">
```

```
function message() {
```

```
document.write('<div align="center"> کدهای جاوا اسکریپت موجود در
```

```
<head>  
<script>  
به وسیله این متن <br> صفحه قبل از سایر اجزای صفحه لود می شوند  
</div>');  
}  
</script>  
</head>  
  
<body>  
<p align="center"><a href="javascript:message();" > روی این متن  
</a></p>  
</body>  
  
</html>
```

[مشاهده صفحه ساخته شده با کد بالا](#)

صفحه قبل از سایر اجزای صفحه لود می شوند <head> کدهای جاوا اسکریپت موجود در این متن به وسیله همین نوع اسکریپت به وجود آمده است

وارد کردن جاوا اسکریپت در قسمت بدنه متن HTML

این امکان وجود دارد که کدهای جاوا اسکریپت را در قسمت <body> صفحه وارد کنیم. در این صورت لود شدن آنها تا هنگامی که آن بخش از صفحه لود شود به عقب می افتد و اگر صفحه به

صورت کامل باز نشود ممکن است اسکریپت ما کار نکند. مثال زیر یک نمونه از وارد کردن جاوا

اسکریپت در `<body>` صفحه HTML را نشان می دهد:

```
<html dir="rtl">
<head>
<title>صفحه جاوا اسکریپت در بدنه</title>
</head>
<body>
<script language="javascript" type="text/javascript">
document.write('<br><h1 align="center"> این متن توسط جاوا اسکریپت
</h1>');
</script>
</body>
</html>
```

[مشاهده صفحه مربوط به کد بالا](#)

این متن توسط جاوا اسکریپت نوشته شده است.

فرا خوانی اسکریپت خارجی به صفحه

گاهی اوقات ممکن است بخواهید از یک سری از کدهای جاوا اسکریپت برای چندین صفحه استفاده کنید و نمی خواهید همه آنها را به صورت جداگانه در هر صفحه بنویسید. در این مواقع می توانید کدهای مورد نظر خود را در یک فایل متنی وارد کنید و آنرا با پسوند **.js** ذخیره کنید. در هر صفحه ای که بخواهید از این فایل استفاده کنید کافی است آنرا در آن صفحه فرا خوانی کنید. (به یاد داشته باشید که در این فایلهای خارجی نباید از تگهای **<script>** که مخصوص HTML هستند استفاده کنید)

برای فرا خوانی این فایلها به صفحه از همان تگ **script** که در موارد قبل استفاده می کردیم کمک می گیریم با این تفاوت که در اینجا به آن یک شناسه **src** که آدرس اسکریپت ما به عنوان مقدار آن قرار گرفته است اضافه می کنیم. برای مثال کد جاوا اسکریپت زیر را در نظر بگیرید :

```
var text = '<br><h1 align="center" dir="rtl" >متن با استفاده از جاوا این </h1>';
```

```
document.write(text);
```

ما این کد را در فایلی با نام **external.js** در همان پوشه ای که صفحه ما قرار دارد ذخیره کرده ایم. حالا با وارد کردن کد زیر در صفحه می توانیم فایل بالا را فراخوانی کنیم:

```
<script language="javascript" type="text/javascript" src="external.js"></script>
```

معرفی عملگرهای مورد استفاده در JavaScript

در جاوا اسکریپت برای جمع، تفریق، مقایسه و بسیاری از عملیات دیگر از عملگرها استفاده می کنیم. عملگرها به چند دسته تقسیم می شوند: عملگرهای محاسباتی، مقایسه ای، جایگزینی، منطقی، رشته ای، و شرطی. در ادامه به ارائه توضیحاتی برای هر نوع از این عملگرها خواهیم پرداخت.

عملگرهای محاسباتی

از این نوع عملگرها برای انجام عملیات ریاضی نظیر جمع و ضرب استفاده می شود. می توانید این عملگرها را در اینجا مشاهده کنید (در مثال زیر متغیر x را برابر 8 و متغیر y را برابر 4 در نظر بگیرید):

| نام | عملگر | مثال | نتیجه |
|-----------|-------|----------|-------|
| جمع | + | $x + y$ | 12 |
| تفریق | - | $x - y$ | 4 |
| ضرب | * | $x * y$ | 32 |
| تقسیم | / | x / y | 2 |
| باقیمانده | % | $9 \% y$ | 1 |
| افزایش | ++ | $x++$ | 9 |
| کاهش | -- | $x--$ | 7 |

از عملگرهای افزایش و کاهش به ترتیب برای افزودن 1 به مقدار قبلی و کم کردن 1 از مقدار قبلی استفاده می شود.

عملگرهای جایگزینی

این عملگرها ترکیبی از عملگرهای دیگر هستند و به همین دلیل آنها را عملگرهای ترکیبی هم می نامند. می توانید این عملگرها را در جدول زیر مشاهده کنید (در مثال زیر متغیر X را برابر 8 و متغیر Y را برابر 4 در نظر بگیرید) :

| عبارت معادل | مثال | عملگر | نام |
|--------------|------------|--------|------------------|
| $x = 8$ | $x = 8$ | $=$ | انتساب |
| $x = x + y$ | $x += y$ | $+=$ | انتساب جمع |
| $x = x - y$ | $x -= y$ | $-=$ | انتساب تفریق |
| $x = x * y$ | $x *= y$ | $*=$ | انتساب ضرب |
| $x = x / y$ | $x /= y$ | $/=$ | انتساب تقسیم |
| $x = x \% y$ | $x \% = y$ | $\% =$ | انتساب باقیمانده |

عملگرهای مقایسه ای

از این عملگرها برای مقایسه دو مقدار استفاده می شود که شامل عملگرهای زیر هستند.

در جدول زیر مقادیر X و Y و Z را مطابق زیر در نظر بگیرید و توجه داشته باشید که مقدار متغیرهای X و Y عددی و مقدار متغیر Z از نوع رشته ای است :

x = 4;

y = 8;

z = "4";

| نتیجه | مثال | عملگر | نام |
|-------|---------|-------|-----------------|
| True | x == z | == | تساوی |
| False | x === z | === | همانی |
| True | x != y | != | نامساوی |
| False | x > y | > | بزرگتر از |
| True | x < y | < | کوچکتر از |
| True | x >= z | >= | بزرگتر یا مساوی |
| True | x <= y | <= | کوچکتر یا مساوی |

تفاوت عملگر تساوی با عملگر همانی در این است که در عملگر تساوی نوع متغیرها در نظر گرفته

نمی شود و اگر مقدار متغیرها با هم برابر باشد حتی اگر از یک نوع هم نباشند نتیجه True

خواهد بود. ولی در عملگر همانی باید متغیرها مقداری مساوی داشته باشند و از یک نوع هم باشند

تا نتیجه True باشد.

عملگرهای منطقی

این عملگرها برای انجام عملیات منطقی بر روی دو عبارت به کار می روند (در مثال زیر متغیر X را برابر 8 و متغیر y را برابر 4 در نظر بگیرید) :

| نتیجه | مثال | عملگر | نام |
|-------|----------------------|-------|------|
| True | $!(x==y)$ | ! | نقیض |
| False | $!(y==3)$ | | |
| True | $(x=4 \ \&\& \ y=8)$ | && | و |
| False | $(x>4 \ \&\& \ y=8)$ | | |
| True | $(x>4 \ \ y=8)$ | | یا |
| False | $(x>4 \ \ y>8)$ | | |

عملگر رشته ای

از این عملگر برای چسباندن دو رشته به هم استفاده می شود. مثلاً برای اتصال دو متغیر از نوع رشته از این عملگر استفاده می شود.

| مثال | عملگر | نام |
|---|-------|----------------|
| <pre>a = 'طراحی وب'; b = 'با جاوا اسکرپت'; c = a+b;</pre> | + | عملوند رشته ای |

در مثال بالا مقدار متغیر C این است: 'طراحی وب با جاوا اسکرپت'

برای ایجا فاصله بین کلمه وب و کلمه با می توانیم یک فاصله خالی به یکی از متغیرهای a یا b اضافه کنیم یا با استفاده از عملوند رشته ای یک فضای خالی به متغیر C اضافه کنیم:

```
a = 'طراحی وب';
```

```
b = 'با جاوا اسکرپت';
```

```
c = a+b;
```

```
a = 'طراحی وب';
```

```
b = 'اسکرپت با جاوا';
```

```
c = a+' '+b;
```

در نتیجه کد بالا مقدار متغیر C به این صورت خواهد بود: 'طراحی وب با جاوا اسکرپت'

عملگر شرطی

در جاوا اسکریپت نوعی عملگر وجود دارد که بسته به شرایط مقداری را به یک متغیر تخصیص می دهد به این عملگر عملگر شرطی می گویند. ساختار این عملگر به صورت زیر است :

VAR_NAME=(شرط)?value1:value2

این عملگر در صورتی که شرط برقرار باشد مقدار اول را به متغیر می دهد و در صورت برقرار نبودن شرط مقدار دوم را به متغیر تخصیص می دهد. به مثال زیر توجه کنید :

age_group=(age>18)?'بزرگسال':'زیر سن بلوغ'

در این مثال اگر متغیر age بزرگتر از 18 باشد مقدار 'بزرگسال' به متغیر age_group تخصیص داده می شود و در صورتی که متغیر age کوچکتر از 18 باشد مقدار 'زیر سن بلوغ' به عنوان مقدار متغیر age_group قرار می گیرد.

نمایش پیام هشدار در صفحه با استفاده از جاوا اسکریپت

آیا مایلید یکی از آن کادر هایی که ناگهان ظاهر می شوند و معلوم نیست از کجا آمده اند را به صفحه خود اضافه کنید؟ بسیار خوب بیایید با کادر هشدار شروع کنیم که به بیننده چیزهایی را که شما می خواهید می گوید.

این فرمان هشدار است:

alert('متنی که می خواهید به بازدید کننده نشان داده شود')

حالا برای استفاده از آن ما باید آنرا در یک فرمان دیگر قرار دهیم. بیایید آنرا در یک فرمان آشنا مثل فرمان `onmouseover` استفاده کنیم. اسکریپت مقابل باعث ظاهر شدن یک کادر هشدار وقتی که ما نشانگر ماوس خود را روی لینک به وجود آمده قرار می دهیم می شود. مثلاً در لینکی که نمی خواهیم بیننده کلیک کند:

```
<a href="alert.php" onmouseover="alert('Insert your message here')">
```

ماوس خود را روی این لینک قرار دهید

```
</a>
```

آنرا امتحان کنید. فقط ماوس خود را روی آن قرار دهید:

[ماوس خود را روی این لینک قرار دهید](#)

در اینجا به ارائه توضیحاتی در مورد دستورات جاوا اسکریپت استفاده شده در کد بالا می پردازیم :

`onmouseover`

این رویداد مسئولیت اجرای کد جاوا اسکریپت موجود در خود را وقتی که ماوس روی آن قرار می گیرد بر عهده دارد.

`alert('')`

این تابع متنی را که به عنوان مقدار آن قرار گیرد در یک کادر محاوره ای با یک دکمه OK نمایش می دهد.

بسیار خوب، یک روش دیگر هم وجود دارد که بیشتر باعث گیج شده بیننده می شود. آن هم کادر هشدار به محض باز شدن پنجره است. شما با این روش می توانید مطالبی را که می خواهید به بیننده خود بگویید قبل از باز شدن صفحه به او نشان دهید.

برای ساختن این نوع کادر هشدار فرمان هشدار را بین `<head>` و `</head>` قرار دهید. برای این کار باید از تگهای `<script>` که در اول فصل با آن آشنا شدیم استفاده کنیم تا کدهای جاوا اسکریپت بین بقیه کدها تشخیص داده شوند:

```
<head>  
<script language="javascript">  
<!-- hide from old browsers  
alert('به این صفحه خوش آمدید');  
//-->  
</script>  
</head>
```

فرمان بالا یک کادر محاوره ای را وقتی که بیننده صفحه شما را باز می کند نشان می دهد. وقتی که OK را کلیک کنید بارگذاری صفحه ادامه پیدا می کند.

آیا می خواهید آنرا امتحان کنید؟ لینک زیر را کلیک کنید و به صفحه آزمایشی که ما برای این فرمان ساخته ایم بروید:

[!پیام هشدار به محض باز شدن صفحه!](#)

آیا می خواهید چند گام جلو تر برویم؟ چندین فرمان هشدار را در تگ **SCRIPT** قرار دهیم تا چندین کادر هشدار پشت سر هم به محض باز شدن صفحه نمایش داده شوند و البته برای هر پیام باید دکمه **OK** را کلیک کنیم تا به پیام بعدی برویم. خودتان این را امتحان کنید :

[چندین پیام هشدار به محض باز شدن صفحه!](#)

این هم کدی که مسئول این پیامها است:

```
<head>
<title>javascript example 2</title>
<script language="javascript">
<!--
alert('لطفاً سایت ما را به منوی favorite اضافه کنید خود');
alert('از منوی favorite گزینه add to favorite را انتخاب کنید');
alert('میانبر هم انجام دهید می توانید این کار را با کلیدهای');
alert('کلید فقط کلید کنترل را نگهدارید و d را فشار دهید');
//-->
</script>
</head>
```

شما می توانید برای ظاهر شدن کادر هشدار از دکمه ها هم استفاده کنید. فقط کد را در جایی که می خواهید دکمه نشان داده شود قرار دهید. در این حالت شما نیازی به تگهای **<script>** هم ندارید. این هم یک نمونه از این نوع پیام هشدار:

این هم کد این اسکریپت:

```
<input type="button" value="اینجا را کلیک اگر این صفحه را مطالعه کرده اید"
```

```
"کنید"
```

```
onclick="alert('باز هم سری به ما بزنید از اینکه به سایت ما مراجعه کردید متشکریم')">
```

ما در این دکمه از رویداد `onclick` استفاده کردیم. کار این رویداد هم مانند `onmouseover`

است با این تفاوت که کد مربوطه را در هنگام کلیک کردن روی آن اجرا می کند.

نوشتن متن با استفاده از جاوا اسکریپت

در جاوا اسکریپت این امکان وجود دارد که متنی را نوشته و در صفحه نمایش بدهیم. متن مورد نظر را می توانیم قالب بندی کنیم و از تگهای `HTML` نیز در آن استفاده کنیم. برای این کار از فرمان `document.write()` استفاده می کنیم که یک فرمان جاوا اسکریپت مخصوص چاپ کردن خروجی در صفحه است. روش کار به صورت زیر است :

نوشتن متنهای ساده

```
<script language="JavaScript" type="text/javascript">
```

```
document.write('بنویسید متن مورد نظر خود را اینجا');
```

```
</script>
```

البته نقطه ویرگولی که در آخر فرمان نوشته شده تنها در صورتی الزامی است که بیش از یک

فرمان در اسکریپت خود داشته باشیم.

برای نوشتن متن به این صورت کافی است تا فرمان بالا را در جایی از صفحه HTML که می خواهید متن به نمایش در آید بنویسید. در مثال زیر متنی با استفاده از جاوا اسکریپت در صفحه نمایش داده می شود، برای مشاهده مثال لینک زیر را کلیک کنید :

[نوشتن متنهای ساده با استفاده از جاوا اسکریپت](#)

نوشتن متنهای قالب بندی شده

با استفاده از فرمان **document.write** امکان نوشتن متنهای قالب بندی شده با استفاده از تگهای HTML هم وجود دارد. برای اینکار فقط کافیست متن HTML مورد نظر خود را در تابع **document.write()** قرار دهید. برای مثال به کد زیر توجه کنید :

```
<script language="JavaScript" type="text/javascript">  
document.write('<h1>جاوا اسکریپت نوشته شده این متن با استفاده از</h1>');  
</script>
```

برای مشاهده نتیجه این مثال لینک زیر را کلیک کنید :

[نوشتن متنهای قالب بندی شده با HTML توسط جاوا اسکریپت](#)

تابع () document.write با مقدار چند سطری

اگر بخواهیم متنی که با استفاده از جاوا اسکریپت متنی را بنویسیم که در صفحه در چند سطر نمایش داده شود به راحتی می توانیم با وارد کردن `
` در متن مورد نظر یا با سایر تگهای HTML این کار را انجام دهیم. اما اگر بخواهیم مقداری را که به تابع () `document.write` می دهیم در چند سطر وارد کنیم باید مقدار مورد نظر را در هر سطر از هم جدا کرده و با استفاده از عملوند رشته ای + آنها را در چند سطر به هم پیوند بزنیم.

نکته : نباید بدون استفاده از عملگر های مربوطه مقدار تابع بالا را در بیش از یک سطر بنویسید. در صورتی که متن مورد نظر خود را به این گونه در چند سطر بنویسید تابع کار نمی کند و احتمالاً با پیغام خطا مواجه می شوید.

اگر کد خود را به صورت زیر بنویسید با خطا مواجه خواهید شد :

```
<script language="JavaScript" type="text/javascript">  
document.write('First part of text  
Second part of text');  
</script>
```

برای اینکه با خطا مواجه نشوید و کد شما کار خود را انجام دهد باید آنرا به صورت زیر بنویسید :

```
<script language="JavaScript" type="text/javascript">
document.write('First part of text'
+
'Second part of text');
</script>
```

معرفی دستور شرطی **if...else** و کاربردهای مختلف آن

از دستورات شرطی برای انجام دستوراتی به شرط برقراری یک رابطه دیگر استفاده می شود. در طراحی صفحات وب بسیار پیش می آید که می خواهیم در شرایط خاصی صفحه یک رفتار مشخص داشته باشد و در موارد دیگر رفتار صفحه تفاوت داشته باشد. در این مواقع می توانیم از دستور شرطی **if** استفاده کنیم. با این دستور مشخص می کنیم اگر شرط خاصی برقرار بود یک عمل خاص انجام شود. ساختار این دستور به شکل زیر است :

دستور شرطی **if**

if (condition)

{

در اینجا قرار می گیرد دستوری که در صورت درستی شرط اجرا می شود

}

به یک مثال برای دستور **if** توجه کنید :

```
var test = confirm('می شود اگر این پیام را تأیید کنید دستور شرطی اجرا');
```

```
if (test == true) {
```

```
alert('و این پیام به نمایش در آمد شما دکمه مورد نظر را کلیک کردید');
```

```
}
```

برای اینکه کد بالا را آزمایش کنید لینک زیر را کلی کنید و در پیامی که نمایش داده می شود دکمه OK را کلیک کنید. اگر دکمه Cancel را کلیک کنید شرط بالا برقرار نمی شود و بدون اجرا شدن دستور شرطی صفحه مثال باز می شود.

[مشاهده مثال بالا در عمل](#)



دستور شرطی if...else

برخی مواقع ممکن است بخواهیم دستور شرطی یک شرط را چک کند تا در صورت برقرار بودن آن شرط یک فرمان خاص را اجرا کند و در صورتی که شرط برقرار نبود یک فرمان دیگر را به اجرا

در آورد. در این موارد می توانیم از دستور **if** به همراه **else** استفاده کنیم. ساختار این دستور به شکل زیر است :

```
if (condition) {
```

شوند دستوراتی که در صورت برقراری شرط اجرا می

```
}
```

```
else {
```

دستوراتی که در صورت عدم برقراری شرط اجرا می شوند

```
}
```

در کد بالا ابتدا شرط بررسی می شود و در صورت برقراری آن دستورات مربوطه اجرا می شوند ولی در صورتی که شرط برقرار نباشد دستورات مربوط به قسمت **else** اجرا می شود.

```
var yourmark = prompt('یک شماره از 0 تا 20 وارد کنید');
```

```
if (yourmark > 10) {
```

```
alert('است شماره ای که وارد کرده ای از 10 بزرگتر');
```

```
}
```

```
else {
```

```
alert('!نیست عبارتی که شما وارد کرده اید از 10 بزرگتر');
```

```
}
```

برای مشاهده نتیجه مثال بالا لینک زیر را کلیک کنید و در کادری که نمایش داده می شود یک

عدد از 0 تا 20 وارد کنید و دکمه OK را کلیک کنید.

مثال استفاده از دستور if...else



استفاده از دستور if...else if...else (دستورات شرطی تو در تو)

ممکن است در نظر داشته باشیم تا در صورت برقراری یک شرط یک دستور خاص اجرا شود و در

صورتی که شرط برقرار نبود شرط دیگری بررسی شود و دستور مربوط به آن اجرا شود و در

صورتی که هیچ یک از شرطهای قبل برقرار نبود دستور دیگری اجرا شود. در این مواقع می توانیم

از دستور else if در دستور if...else قبلی استفاده کنید. به صورت زیر :

```
if (condition 1) {
```

```
    شوند دستوراتی که در صورت درستی شرط اول اجرا می
```

```
}
```

```
else if (condition 2) {
```

```
    می شوند دستوراتی که در صورت درستی شرط دوم اجرا
```

```
}
```

```
else {
```

```
    می شوند دستوراتی که زمانی که هیچکدام از شرطها صحیح نباشد اجرا
```

```
}
```

در این روش محدودیتی از نظر تعداد شرطها وجود ندارد و می توانیم به تعداد نامحدود از دستور `else if` استفاده کنیم ولی در مواردی که تعداد این شرطها خیلی زیاد می شود بهتر است به جای این روش از دستور `switch` که بعداً توضیح داده خواهد شد استفاده کنیم.
در اینجا مثال قبل را با کمی تغییر برای این روش استفاده می کنیم :

```
var yourmark = prompt('یک شماره به دلخواه وارد کنید','');  
  
if (yourmark > 0 && yourmark < 100 ) {  
    alert('وارد شده بین 0 تا 100 است شماره');  
}  
else if (yourmark > 100) {  
    alert('شماره وارد شده از 100 بزرگتر است');  
}  
else {  
    alert('وارد شده یا 0 است یا به جای عدد از حروف استفاده کرده اید عبارت');  
}
```

برای مشاهده نتیجه مثال دستور [if...else if...else](#) اینجا را کلیک کنید.



دستور شرطی switch در جاوا اسکریپت

با استفاده از دستور if امکان این وجود دارد که برنامه را حداکثر به یکی از دو جهت ممکن هدایت کنیم. یعنی اگر شرط مورد نظر برقرار بود اسکریپت یک سری کد را اجرا می کند و در صورت عدم برقراری شرط یک سری کد دیگر اجرا می شود. اگر بخواهیم برای اسکریپت خود مسیره های بیشتری در نظر بگیریم یک راه موجود استفاده از دستورات if تو در تو است. این روش ممکن است در موارد جزئی کارایی داشته باشد اما اگر تعداد شرطها زیاد شود ما را مجبور می کند از تعداد زیادی if و else در اسکریپت خود استفاده کنیم که این مسئله هم باعث طولانی شدن برنامه و هم باعث گیج کننده شدن آن می شود از طرفی امکان اشتباه را هم بالا می برد. راه حل دیگر استفاده از دستور شرطی switch است. این دستور را می توان در هنگامی که می خواهیم یک متغیر را با چندین مقدار مقایسه کنیم و مقدار متناسب با آنرا پیدا کنیم کار برد دارد. ساختار این دستور به صورت زیر است :

switch(x)

{

case مقدار اول:

با مقدار اول اجرا می شودX این کد در صورت برابری متغیر

break

case مقدار دوم:

با مقدار دوم اجرا می شودX این کد در صورت برابری متغیر

break

default:

با هیچکدام از مقادیر ذکر شده برابر نباشد این کد در صورتی اجرا می شود که مقدار متغیر

}

در این دستور مقدار متغیر معرفی شده در قسمت `switch` با مقادیر آورده شده در هر `case` مقایسه می شود. در صورت مساوی بودن مقادیر کد همان `case` اجرا می شود و با استفاده از دستور `break` آورده شده در هر قسمت برنامه پس از اجرای کد آن قسمت به خارج از دستور `switch` هدایت می شود. در غیر این صورت برنامه به سراغ `case` بعدی می رود و در صورتی که مقدار متغیر مورد نظر با هیچکدام از `case` ها برابر نبود کد قسمت `default` اجرا خواهد شد.

قابل ذکر است که استفاده از `default` اختیاری است و شما بنا به نیاز خود می توانید از آن استفاده کرده یا آنرا حذف کنید. در دستور `switch` محدودیتی در تعداد `case` ها وجود ندارد و می توانید به هر تعدادی که نیاز داری از این مورد استفاده کنید.

در این قسمت به ذکر یک نمونه برای دستور `switch` می پردازیم :

به کدی که در اینجا آورده شده است توجه کنید. پس از اجرای این کد کادری باز می شود که از شما میخواهد عددی را وارد کنید. با وارد کردن عدد در این کادر و تأیید آن پیامی متناسب با روز معادل آن عدد در هفته ظاهر می شود. البته ما این کار را فقط برای اعداد 1 تا 3 انجام دادیم، شما در صورت تمایل می توانید آنرا تکمیل کنید :

```
var x=prompt('لطفاً یک عدد بین 1 تا 7 وارد کنید');
```

```
switch (x) {
```

```
case '1':
```

```
alert('شماره وارد شده مربوط به روز « شنبه » است');
```

```
break
```

```
case '2':
```

```
alert('شماره وارد شده مربوط به روز « یکشنبه » است');
```

```
break
```

```
case '3':
```

```
alert('شماره وارد شده مربوط به روز « دوشنبه » است');
```

```
break
```

```
case '4':
```

```
alert('شماره وارد شده مربوط به روز « سه شنبه » است');
```

```
break
```

```
case '5':
```

```
alert('شماره وارد شده مربوط به روز « چهارشنبه » است');
```

```
break
```

```
case '6':
```

```
alert('شماره وارد شده مربوط به روز « پنجشنبه » است');
```

```
break
case '7':
alert('شماره وارد شده مربوط به روز « جمعه » است');
break
default:
alert('روز مربوط به شماره وارد شده در این برنامه وجود ندارد');
}
```

[مشاهده مثال بالا](#)



رویداد onmouseover

وقت اون رسیده که نوشتن اولین اسکریپت خود را شروع کنید. این یکی راحت‌تره چون مجبور نیستیم درگیر اضافه کردن تگهای SCRIPT شویم. این اسکریپت کوچک به دلخواه شما وقتی نشانگر ماوس روی یک لینک قرار می‌گیرد متنی را در نوار وضعیت مرورگر نشان می‌دهد. به

مثال نگاه کنید:

```
<a href = "JavaScript_2.htm"  
onmouseover = "window.status = '« آموزش جاوا اسکریپت »'; return true" >
```

!!! دهید ماوس خود را روی این لینک قرار

```
</a>
```

شما هم امتحان کنید و ببینید چگونه کار می کند. نشانگر ماوس خود را روی لینک زیر قرار دهید ولی کلیک نکنید (اگر کلیک کنید، چون در کد بالا آدرس لینک، آدرس همین صفحه است پس از کلیک کردن دوباره همین صفحه باز می شود). حالا پایین پنجره مرورگر خود به نوار وضعیت نگاه کنید باید این جمله را ببینید: « آموزش جاوا اسکریپت »

[ماوس خود را روی این لینک قرار دهید!!!](#)

بسیار خوب، این چیزی است که با فرمان `onmouseover` اتفاق می افتد:

```
onmouseover = " "
```

این شکل فرمان `onmouseover` است. مرورگر انتظار فرمان دیگری بین دو علامت دابل کوت (") را دارد.

```
window.status = '« آموزش جاوا اسکریپت »'
```

این فرمان به مرورگر میگوید که جمله ای را در نوار وضعیت بنویسد. شما می توانید جمله بین دو علامت (') را با آنچه که می خواهید در نوار وضعیت به نمایش در آید جابجا کنید.

return true

عبارت را تأیید کرده و مرورگر آن را نمایش می دهد.

دلیل استفاده از علامت سینگل کوت (') در فرمان `window.status` این است که `window.status` خود قسمتی از فرمان `onmouseover` است که از (") استفاده کرده است. اگر دوباره از علامت دابل کوت (") استفاده کرده بودیم ممکن بود مرورگر در مورد اینکه چه کاری باید انجام دهد گیج می شد. چون ممکن بود علامت دابل کوت دوم را به منزله پایان فرمان `onMouseovr` تلقی کند و فرمان جدیدی را با `window.status` شروع کند:

```
onmouseover=""window.statuse
```

مشکلی که ممکن است در اینجا به وجود آید پاک نشدن متنی است که می خواستید در نوار وضعیت نمایش داده شود. دو راه برای از بین بردن این مشکل وجود دارد. یکی استفاده از فرمان `onmouseout` است، و دیگری فرا خواندن تابعی که متن را پس از مدتی پاک می کند. راه دوم نیازمند استفاده از توابع و اسکرپت تگها است. در اینجا آسانترین راه را به شما نشان می دهیم:

استفاده از فرمان `onmouseout` :

```
<a href ="JavaScript_2.htm"
```

```
onmouseover="window.status='متن خود را اینجا قرار دهید'; return true"
```

```
onmouseout="window.status=' ' ; return true">
```

قرار دهید لینک خود را اینجا

```
</a>
```

این فرمان متن نوار وضعیت را وقتی که نشانگر ماوس روی آن نیست ناپدید می کند. مانند زیر:

لینک خود را اینجا قرار دهید

در حقیقت ما همان کار قبل را تکرار کردیم ولی به جای متن از فضای خالی در فرمان `window.status` استفاده کردیم.

این اسکریپت جالب و مفید است و می تواند به بازدید کنندگان صفحه شما کمک کند تا بهتر بدانند لینک آنها را به چه صفحه ای می برد. و می توانند توضیحات شما را در مورد لینک ببینند.

دکمه ها و موارد استفاده آنها

ما برای نوشتن اسکریپت با استفاده از دکمه ها، ابتدا باید بدانیم چگونه می توان یک دکمه را در یک صفحه قرار داد. برای این کار شما باید از تگهای `<FORM>` در اطراف تگ دکمه استفاده کنید. در مثال زیر نمونه یک HTML که یک دکمه را در یک صفحه قرار می دهد آورده شده است:

`<FORM>`

`<INPUT type="button" value="اینجا کلیک کنید" name="button1">`

`</FORM>`

این فرمان یک دکمه را در صفحه شما قرار می دهد، اما اگر روی آن کلیک کنید اتفاقی نمی افتد...

در زیر کار هر قسمت از فرمان بالا را می بینیم:

<FORM>

این قسمت یک فرم را ایجاد می کند بنابراین ما می توانیم یک دکمه بسازیم.

<INPUT>

این تگ به ما اجازه می دهد نوعی ناحیه ورودی بسازیم.

type="button"

این فرمان ناحیه ورودی ما را به عنوان یک دکمه تعریف می کند.

value="اینجا کلیک کنید"

این متنی است که افراد روی دکمه می بینند. شما می توانید هر چیزی را که می خواهید

بینندگان روی دکمه ببینند اینجا بنویسید.

name="button1"

شما می توانید برای مراجعات بعدی یا احتمالاً استفاده در یک اسکریپت به دکمه خود یک نام

بدهید.

احتمالاً شما نمی خواهید یک دکمه بسازید که کاری انجام نمی دهد. در زیر یک فرمان جاوا

اسکریپت آورده شده است که مرورگر را مجبور می کند وقتی دکمه را کلیک کردید عکس العمل

نشان دهد:

onClick="javascript command"

فقط این فرمان را در تگ INPUT قرار دهید که دکمه را با آن ساختید، مانند زیر:

<INPUT type="button" value="اینجا کلیک کنید" name="button1"

onClick=" " ">

برای مثال ما می توانیم مانند فرمان `onMouseover` در این دکمه نیز استفاده کنیم و متنی را در نوار وضعیت پنجره چاپ کنیم!

```
<FORM>
```

```
<INPUT type="button" value="اینجا کلیک کنید" name="Button2"
```

```
onClick="window.status='متن خود را اینجا بنویسید'; return true">
```

```
</FORM>
```

حالا دکمه را فشار دهید و به متن نوار وضعیت نگاه کنید:

شما می توانید با استفاده از دکمه ها به بینندگان خود اجازه دهید رنگ زمینه صفحه را برای خودشان تغییر دهند. فقط از فرمان زیر به جای فرمان `window.status` استفاده کنید:

```
document.backgroundColor='رنگ مورد نظر به زبان انگلیسی نوشته شود'
```

فرمان زیر را به عنوان تمرین فرمان `onClick` وارد کنید:

```
<FORM>
```

```
<INPUT type="button" value="Change Background Color"
```

```
name="button3" onClick="document.backgroundColor='yellow'">
```

```
</FORM>
```

حالا شما دکمه ای مانند زیر دارید که با کلیک کردن بر روی آن زمینه صفحه به رنگ زرد در می آید!

شما می توانید به هر تعداد که می خواهید از این دکمه ها اضافه کنید، فقط مطمئن شوید که آنها گزینه بازگشت به حالت اولیه را نیز داشته باشند. اسکریپت زیر به شما چهار گزینه می دهد :

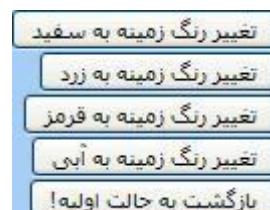
سفید، زرد، قرمز و آبی . می توانید از دکمه «بازگشت به رنگ اولیه!» برای بازگرداندن تصویر زمینه این صفحه استفاده کنید. کدی که در این صفحه از آن استفاده شده است به دلیل وجود تصویر زمینه کمی با کد زیر متفاوت است ولی کارایی آن تفاوتی ندارد.

<FORM>

```
<INPUT type="button" value="به سفید تغییر رنگ زمینه"
name="button6" onClick="document.backgroundColor='white'"><br>
<INPUT type="button" value="تغییر رنگ زمینه به زرد"
name="button3" onClick="document.backgroundColor='yellow'"><br>
<INPUT type="button" value="تغییر رنگ زمینه به قرمز"
name="button4" onClick="document.backgroundColor='red'"><br>
<INPUT type="button" value="تغییر رنگ زمینه به آبی"
name="button5" onClick="document.backgroundColor='#66CCFF'">
</FORM>
```

برای دکمه چهارم به جای نام رنگ از معادل هگزادسیمال رنگها استفاده شده است. برای یافتن معادل هگزادسیمال رنگها و توضیح درباره آنها می توانید به [این صفحه](#) مراجعه کنید.

حالا شما سه دکمه دارید، آنها را امتحان کنید!



در اسکریپت آخری که در این قسمت بررسی می کنیم، اجازه می دهیم تا یک دکمه به عنوان یک لینک عمل کند. به این منظور فقط فرمان زیر را برای فرمان `onClick` استفاده کنید:

```
window.location='URL'
```

برای مثال این هم اسکریپتی که شما را به صفحه آموزش جاوا اسکریپت ما می برد:

```
<FORM>
```

```
<INPUT type="button" value="اسکریپت آموزش جاوا" name="button6"
```

```
onClick="window.location='./'">
```

```
</FORM>
```

این هم دکمه، آن را امتحان کنید تا به سرفصل آموزش جاوا اسکریپت بروید:

آموزش جاوا اسکریپت

آموزش کار با تاریخچه مرورگر با استفاده از جاوا اسکریپت

مرورگر ها به صورت پیش فرض آدرس و حتی خود صفحاتی را که کاربر مشاهده می کند ذخیره می کنند. این آدرسها به ترتیبی که کاربر آنها را مشاهده کرده است در تاریخچه (**history**) مرورگر ثبت می شوند و دکمه های **Back** و **Forward** پنجره مرورگر بر اساس این تاریخچه کار می کنند. با استفاده از جاوا اسکریپت می توانیم به این تاریخچه دسترسی داشته باشیم و با کمک آن مسیر کاربران را هدایت کنیم.

ما برای کار با تاریخچه مرورگر در اینجا از دکمه های **HTML** استفاده می کنیم که با کمک رویداد **onclick** در هنگام کلیک کردن روی این دکمه ها دستورات مورد نظر ما اجرا خواهند شد. در اینجا می توانید کد **HTML** مربوط به ساخت این دکمه ها را مشاهده کنید (البته مقدار مربوط به شناسه **onclick** مربوط به دستورات جاوا اسکریپت است که بعداً وارد خواهد شد) :

```
<input type="button" value="متن روی دکمه" onclick="">
```

برای آشنایی بیشتر با رویداد های **HTML** می توانید از صفحه [رویدادهای مربوط به صفحه کلید و ماوس در HTML](#) استفاده کنید. (لینک در صفحه جدید باز خواهد شد)

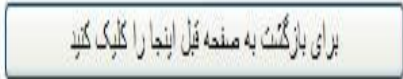
دستورات مختلفی برای کار با تاریخچه مرورگر وجود دارد که همه آنها از شیء **history** استفاده می کنند. در ادامه می توانید تعدادی از این دستورات را که با بیشتر مرورگرها کار می کنند مشاهده کنید

history.back()

از این فرمان می‌توانید جهت هدایت کاربران به صفحه‌ای که قبل از صفحه کنونی در پنجره مرورگر مشاهده کرده‌اند استفاده کنید. این دستور دقیقاً مانند دکمه **Back** در پنجره مرورگر عمل می‌کند و در صورتی که صفحه کنونی اولین صفحه‌ای باشد که از زمان باز شدن پنجره نمایش داده شده باشد این دستور کاربردی ندارد. برای مشاهده یک نمونه از کاربرد این دستور می‌توانید لینک را کلیک کنید :

[history.back\(\)](#) مشاهده کاربرد دستور

در صورتی که جابجا اسکرین در مرورگر شما فعال باشد می‌توانید از دکمه زیر برای بازگشت به صفحه قبلی استفاده کنید. در غیر این صورت می‌توانید از دکمه Back در پنجره مرورگر خود استفاده کنید.



برای بازگشت به صفحه قبل اینجا را کلیک کنید

در اینجا می‌توانید کدی مورد استفاده در مثال بالا را مشاهده کنید :

```
<input type="button" value="کلیک کنید برای بازگشت به صفحه قبل اینجا را"
onclick="history.back()">
```

history.forward()

در صورتی که بازدید کننده قبلاً از دکمه **Back** در مرورگر خود استفاده کرده باشد یا با استفاده از دستور

`history.back()` به صفحه قبلی بازگشته باشد می‌توانیم از دستور

history.forward() برای هدایت بازدید کننده به صفحه ای که پس از صفحه کنونی باز کرده است استفاده کنیم. این دستور کاربردی مشابه دکمه **Forward** در پنجره مرورگر دارد و کاربرد آن عکس کاربرد **history.back()** است.

دکمه زیر با استفاده از دستور **history.forward()** شما را به صفحه ای که برای مثال

مربوط به بازگشت به صفحه قبلی مشاهده کرده اید هدایت می کند :

می توانید کد استفاده شده در ساخت دکمه بالا را در اینجا مشاهده کنید :

```
<input type="button" value="نمایش صفحه بعدی"  
onclick="history.forward()">
```

history.go()

با استفاده از دستورهای **history.back()** و **history.forward()** فقط امکان این وجود دارد

که کاربران را یک صفحه به عقب یا جلو هدایت کنیم. اگر بخواهیم بازدید کننده را به چندین

صفحه عقب یا جلو هدایت کنیم باید از دستور **history.go()** استفاده کنیم.

1- استفاده از مقداری مثبت برای **history.go()**

اگر برای دستور **history.go()** از مقداری مثبت استفاده شود کاربر به همان تعداد صفحه در

تاریخچه مرورگر به جلو هدایت می شود. به کد زیر توجه کنید :

```
<input type="button" value="متن روی دکمه"
```

```
onclick="history.go(3)">
```

کد بالا دکمه ای را می سازد که کلیک کردن بر روی آن همان کاری را انجام می دهد که سه بار فشار دادن دکمه Forward در پنجره مرورگر انجام می دهد.

در صورتی که دستور به صورت (history.go(1 نوشته شود مانند دستور history.forward() عمل می کند.

2- استفاده از مقدار صفر برای history.go()

در صورتی که این دستور به صورت history.go(0 نوشته شود باعث باز بارگذاری مجدد همان صفحه خواهد شد. برای مثال به کد زیر و دکمه ساخته شده با آن توجه کنید :

```
<input type="button" value="بارگذاری مجدد صفحه"
```

```
onclick="history.go(0)">
```

این هم دکمه ساخته شده :

3- استفاده از مقادیر منفی برای history.go()

اگر بخواهیم کاربر را چند صفحه در تاریخچه مرورگر به عقب هدایت کنیم می توانیم برای دستور history.go() از مقادیر منفی استفاده کنیم. برای مثال به کد زیر توجه کنید :

```
<input type="button" value="متن روی دکمه" onclick="history.go(-3)">
```

این کد دکمه ای را می سازد که وقتی کاربر آنرا کلیک کند به سومین صفحه ای که قبل از صفحه کنونی در پنجره مرورگر بازدید کرده است هدایت می شود. کاربرد این دکمه در اینجا مشابه سه بار فشار دادن دکمه Back در پنجره مرورگر است.

history.length

با استفاده از دستور history.length می توانید تعداد صفحاتی را که در تاریخچه مرورگر وجود دارد پیدا کنید. یعنی با استفاده از این دستور می توانیم به تعداد صفحاتی که از زمان باز شدن پنجره مرورگر در آن نمایش داده شده است پی ببریم. برای مثال می توانید با کلیک کردن بر روی لینک زیر تعداد صفحات ذخیره شده در تاریخچه پنجره مرورگر خود را پیدا کنید :

در اینجا می توانید کد مورد استفاده در ساخت لینک بالا را مشاهده کنید :

```
<a href="javascript:alert(history.length)">
```

صفحات موجود در تاریخچه مرورگر نمایش تعداد

```
</a>
```

درخواست اطلاعات از کاربر

بیاپید فرض کنیم می خواهیم نام فردی را قبل از ورود به صفحه بپرسیم و سپس نام آنها را در صفحه برای خودشان نمایش دهیم. این کار را با استفاده از جاوااسکریپت می توان انجام داد. فرمانی که برای این کار استفاده می شود به این صورت است:

`prompt('سؤال شما', '');`

این فرمان یک پنجره را قبل از باز شدن صفحه نمایش می دهد که سؤال شما در آن پرسیده می شود و بیننده می تواند جواب خود را در کادری که برای او در نظر گرفته شده بنویسد. دو سینگل کوت بعدی جواب پیش فرض شما برای سؤال است. اگر شما چیزی بین این دو سینگل کوت ننویسید، بیننده با یک لیست خالی، آماده برای نوشتن مواجه می شود. این پیغام قبل از بارگذاری صفحه نمایش داده می شود پس شما می توانید پاسخ بیننده را در صفحه نمایش دهید. اگر مثالی در این مورد می خواهید لینک زیر را کلیک کنید. در صفحه مثال نام شما پرسیده می شود و پس از وارد کرده آن، نام شما در صفحه جدید نمایش داده می شود:

[لینک صفحه آزمایشی](#)

حالا به سراغ اسکریپتی که این کار را انجام می دهد می رویم. توجه داشته باشید که دستورات

`prompt` و `if/else` در بخش `HEAD` صفحه وارد می شوند ولی نام کاربر در بخش `BODY`

صفحه مشاهده می شود.

```
<html dir="rtl" >
<head>
<script language="javascript">
<!--
var yourname=prompt('لطفاً نام خود را وارد کنید', '');
if ( (yourname=="") || (yourname==null) )
{
yourname="ناشناس";
}
//-->
</script>
</head>
<body>
<script language="javascript">
<!--
document.write("<center><h1>سلام " +yourname+ " به سایت ما،
خوش آمدید </h1></center>");
//-->
</script>
</body>
</html>
```

اولین چیزی که اتفاق می افتد این است که متغیر `yourname` مقداری را که کاربر برای آن وارد می کند به خود می گیرد. پس مقدار متغیر `yourname` معادل استرینگی است که نام کاربر را می سازد.

دستور `if/else` نام « ناشناس » را در صورتی که کاربر نامی را وارد نکند به عنوان مقدار متغیر `yourname` قرار می دهد. این دستور چک می کند که اگر نامی وارد نشود (`null`) و یا اگر فضای خالی وارد شود، مقدار « ناشناس » را به متغیر `yourname` می دهد.

در بخش `body` متن دوباره از تگ `script` استفاده می کنیم تا کدهای جاوا اسکریپت از بقیه متن `html` متمایز شود. در این قسمت ما از یک فرمان جدید یعنی `document.write(" ")` استفاده کرده ایم. این همان چیزی است که به متغیر `yourname` اجازه نوشته شدن در صفحه `html` را می دهد. ما در این پیغام دو استرینگ را به علاوه متغیر `yourname` در صفحه می نویسیم. متغیر `yourname` در این فرمان بین دو دابل کوت قرار نمی گیرد چون یک مقدار است نه یک استرینگ، (البته مقدار متغیر `yourname` یک استرینگ است). به دلیل اینکه متغیر `yourname` یک استرینگ نیست ما از دو علامت بعلاوه در ابتدا و انتهای آن استفاده می کنیم. این سری نوشته ها مرورگر را وادار می کند ابتدا یک استرینگ سپس مقدار متغیر و سپس استرینگ دوم را بنویسد. چون ما در حال نوشتن یک کد جاوا اسکریپت هستیم برای اینکه بتوانیم متنی را در صفحه بنویسیم نمی توانیم مانند نوشتن متن در `html` عمل کنیم، و باید متن مورد نظر را بین دابل کوت و در قالب استرینگ در صفحه بنویسیم.

شاید متوجه شده باشید تگهای پایانی `h1` و `center` به گونه ای متفاوت با قبل نوشته شده اند (`</h1>`). این تفاوت بدین دلیل است که اسلش در جاوا اسکریپت به معنای تقسیم است و برای جلوگیری از اشتباه شدن آن با علامت تقسیم از یک بک اسلش قبل از آن استفاده می کنیم. در نتیجه استفاده از یک بک اسلش (\) و بلافاصله بعد از آن یک اسلش (/)

تنظیم خاصیت float برای عناصر HTML از طریق جاوا اسکریپت

امروز با در خواست یکی از بازدید کنندگان در مورد خاصیت `float` و جاوا اسکریپت صحبت میکنیم. احتمالاً با خاصیت `float` در `CSS` آشنایی دارید. این خاصیت به ویژه وقتی مورد استفاده قرار میگیرد که طراح قصد داشته باشد بدون استفاده از جدولها چهارچوب صفحه را طراحی کند.

این خاصیت باعث میشود تا یک بلوک `HTML` بسته به مقداری که به این ویژگی داده میشود تا حد امکان در سمت چپ یا راست المنت والد آن نمایش داده شود، به طوری که سایر المنتها و متن موجود در المنت والد در اطراف بلوک مورد نظر قرار گیرند. توضیح بیشتر در مورد این خاصیت مربوط به بخش آموزش `CSS` هست بنابراین به همین مقدار توضیح بسنده میکنیم. برای اختصاص دادن یک ویژگی `CSS` به المنتهای `HTML` از شیء `style` استفاده می شود. اما این نکته را باید در نظر داشته باشیم که `float` از کلمات کلیدی رزرو شده در جاوا اسکریپت است. به همین دلیل نمیتوانیم از متد `style.float` برای این منظور استفاده کنیم.

برای جلوگیری از تداخل بین float مورد استفاده در CSS و کلمه کلیدی float در جاوا اسکریپت به منظور اختصاص این ویژگی به المنتهای HTML از متد cssFloat استفاده میکنیم. برای درک بهتر میتوانید به مثال زیر در دو حالت قبل و بعد از اختصاص این ویژگی با به المنت توجه کنید:

```
<div style="border:1px #666 dashed; width:50%; margin:10px auto">
```

```
<div style="width:100px; height:100px; background: #fc0;"> بلوک
```

```
</div>
```

```
<p> از اختصاص ویژگی به المنت در زیر بلوک نمایش داده میشود این متن قبل
```

```
</div>
```

قبل از اختصاص ویژگی float:



کد مربوط به اعمال ویژگی با استفاده از جاوا اسکریپت:

```
<div style="border:1px #666 dashed; width:50%; margin:10px auto">
```

```
<div id="my-block-2" style="width:100px; height:100px; background: #fc0;">ویژگی بلوک مورد نظر قبل از اختصاص</div>
```

```
<p>نمایش داده میشود این متن قبل از اختصاص ویژگی به المنت در زیر بلوک<br clear="all"></div>
```

```
<script type="text/javascript"> document.getElementById("my-block-2").style.cssFloat = "right"; </script>
```

و در انتها این چیزی هست که در مرورگر نمایش داده خواهد شد:

بلوک مورد نظر پس از اختصاص ویژگی

این متن پس از اختصاص ویژگی به المنت در کنار بلوک نمایش داده میشود

مقادیر زیر را میتوان برای متد `cssFloat` به کار برد:

: `noen`

این مقدار پیش فرض متد هست و تأثیر `float` را بر روی المنت خنثی میکند.

: right

این مقدار باعث می شود تا بلوک مورد نظر تا حد امکان در سمت راست المنت والد خود قرار گیرد.

:Left

این مقدار باعث میشود تا بلوک مورد نظر تا حد امکان در فضای سمت چپ المنت والد خود قرار گیرد.

در پایان این مقاله باید به این نکته اشاره کنم که مقادیر متد `cssFloat` از نوع رشته ای (String) هستند پس باید در کد جاوا اسکریپت در کوتیشن نوشته شوند.

نمونه کد مابھی

»

HTML

کد

```
<div dir="rtl">  
<p>پاراگراف اول</p>  
<p>  
پاراگراف دوم  
<br>  
ادامه پاراگراف دوم  
</p>  
</div>
```

نتیجه

پاراگراف اول

پاراگراف دوم

ادامه پاراگراف دوم

```
<div dir="rtl">  
<h1>سرفصل شماره 1</h1>  
<h2>سرفصل شماره 2</h2>  
<h3>سرفصل شماره 3</h3>  
<h4>سرفصل شماره 4</h4>  
<h5>سرفصل شماره 5</h5>  
<h6>سرفصل شماره 6</h6>  
</div>
```

نتیجه

سرفصل شماره 1

سرفصل شماره 2

سرفصل شماره 3

سرفصل شماره 4

سرفصل شماره 5

سرفصل شماره 6

```
<div dir="rtl">
```

```
<strong>متن ضخیم</strong>
```

```
<br>
```

```
<b>متن ضخیم</b>
```

```
<br><br>
```

```
<i>متن مورب</i>
```

```
<br>
```

```
<em>متن مورب</em>
```

```
<br><br>
```

```
<u>متن زیر خط دار</u>
```

```
</div>
```

نتیجه

متن ضخیم

متن ضخیم

متن مورب

متن مورب

متن زیر خط دار

کد

```
<"div dir="rtl>  
<font/>متن با اندازه فونت 3<"font size="3>  
<br><br>  
<font/>متن با رنگ آبی<"font color="blue>  
<br><br>  
<Times New Roman</font متن با قلم<"font face="Times New Roman>  
  
<div/>
```

نتیجه

متن با اندازه فونت 3

متن با رنگ آبی

Times New Roman متن با قلم

&

**
**

©

**
**

®

**
**

™

**
**

†

**
**

»

**
**

«

**
**

—

**
**

°

**
**

¼

**
**

½

**
**

¾

**
**

·

**
**

¡

**
**

€

**
**

£

**
**

÷

**
**

√

نتیجه

&
©
®
™

†
»
«
—

°
¼
½

```
<div dir="rtl" align="center">
```

: ماوس خود را روی لینک زیر قرار دهید تا عنوان آنرا مشاهده کنید

```
<br>
```

```
<a href="http://www.neopersia.org/"
```

```
title="طراحی وب - آموزش طراحی صفحات وب"
```

```
target="_blank">
```

ساختن لینکی که در صفحه جدید باز می شود

```
</a>
```

```
</div>
```

نتیجه

ماوس خود را روی لینک زیر قرار دهید تا عنوان آنرا مشاهده کنید :

[ساختن لینکی که در صفحه جدید باز می شود](http://www.neopersia.org/)

```

```

نتیجه



کد

این قسمت خارج از کامنت نوشته شده است

<!--

این یک کامنت است

این بخش در داخل کامنت نوشته شده است

کامنتها در مرورگر نمایش داده نمی شوند

-->

این متن خارج از کامنت و بعد از آن نوشته شده است

نتیجه

این قسمت خارج از کامنت نوشته شده است این متن خارج از کامنت و بعد از آن نوشته شده است

```
<div dir="rtl">  
<ul>  
<li type="circle">قسمت اول لیست</li>  
<li type="disk">قسمت دوم لیست</li>  
<li type="square">قسمت سوم لیست</li>  
<li>قسمت چهارم لیست</li>  
</ul>  
</div>
```

نتیجه

- قسمت اول لیست
- قسمت دوم لیست
- قسمت سوم لیست
- قسمت چهارم لیست

```

<div dir="rtl">
<ol start="8">
<li type="1">قسمت اول لیست</li>
<li type="A">قسمت دوم لیست</li>
<li type="a">قسمت سوم لیست</li>
<li type="I">قسمت چهارم لیست</li>
<li type="i">قسمت پنجم لیست</li>
</ol>
</div>

```

نتیجه

8. قسمت اول لیست
- . قسمت دوم لیست
- a. قسمت سوم لیست
- .II قسمت چهارم لیست
- .iii قسمت پنجم لیست

<div dir="rtl">

لینک زیر برای ارسال ایمیل به همراه یک کپی و یک کپی مخفی با مشخص کردن
موضوع و متن بدنه ایمیل استفاده می شود:

<a href="mailto:Ayoob.rashidi@gmail.com?cc=info@pc-
boukan.org&bcc=admin@pc-boukan.org&subject=HTML In
Practice&body=Learn HTML with Exams">

تماس با ما از طریق ایمیل

</div>

نتیجه

لینک زیر برای ارسال ایمیل به همراه یک کپی و یک کپی مخفی با مشخص کردن موضوع و متن
بدنه ایمیل استفاده می شود:

تماس با ما از طریق ایمیل

```
<div dir="rtl">
```

```
<form name="testForm" method="post"
action="http://www.neopersia.org/samples/myForm.php?testForGet=this
section uses GET method" target="_blank">
```

```
<br>: متن مربوط به تکست باکس
```

```
<input name="frm_textbox" type="text" size="30" maxlength="30">
```

```
<br>
```

```
<br>: متن مربوط به باکس کلمه عبور
```

```
<input name="frm_pass" type="password" size="30" maxlength="30">
```

```
<br>
```

```
TextArea :<br>
```

```
<textarea name="frm_textarea" cols="25" rows="3"></textarea>
```

```
<br>
```

```
<input name="frm_checkbox" type="checkbox" value="چک باکس انتخاب شده
است">
```

```
<br>: متن مربوط به چک باکس
```

```
<label>
```

```
<input type="radio" name="RadioGroup" value="value1">
```

```
</label> دکمه 1
```

```
<br>
```

<label>

<input type="radio" name="RadioGroup" value="value2">

2 دکمه</label>

<select name="frm_list" size="1">

<option value="مقدار اول در لیست">گزینه اول</option>

<option value="مقدار دوم در لیست">گزینه دوم</option>

<option value="مقدار سوم در لیست">گزینه سوم</option>

</select>

<input value="پاک کردن فرم" type="reset">

<input value="ارسال فرم" type="submit">

</form>

</div>

نتیجه

متن مربوط به تکست باکس :

متن مربوط به باکس کلمه عبور :

: TextArea

متن مربوط به چک باکس

دکمه ۱

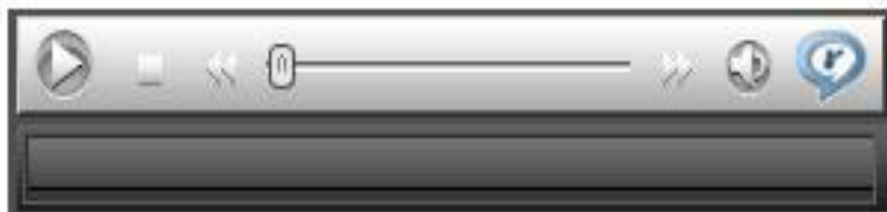
دکمه ۲

گزینه اول

کد تعبیه موزیک در صفحه

```
<embed src="http://www.neopersia.org/samples/example.mid" width="300" height="60" autostart="true" loop="true">
```

نتیجه



کد

```
<hr  
align="left"  
width="250"  
size="20"  
noshade="noshade">
```

نتیجه



کد متن متحرک

```
<marquee loop="infinite" bgcolor="yellow" width="250"> متن مورد نظر خود را اینجا  
</marquee> بنویسید
```

نتیجه

متن مورد نظر خود را اینجا بنویسید

```
<div dir="rtl">  
<table border="3" cellpadding="5" cellspacing="5">  
<tr>  
<td>سطر اول و سلول اول</td>  
<td>سطر اول و سلول دوم</td>  
</tr>  
<tr>  
<td>سطر دوم و سلول اول</td>  
<td>سطر دوم و سلول دوم</td>  
</tr>  
</table>  
</div>
```

نتیجه

| | |
|--------------------|--------------------|
| سطر اول و سلول اول | سطر اول و سلول دوم |
| سطر دوم و سلول اول | سطر دوم و سلول دوم |

```
<div dir="rtl">
```

```
<table width="350" height="100" border="3">
```

```
<td align="center" valign="top">
```

متمايل به بالا

```
</td>
```

```
<td align="center" valign="middle">
```

متمايل به وسط

```
</td>
```

```
<td align="center" valign="bottom">
```

متمايل به پايين

```
</td>
```

```
</table>
```

```
</div>
```

نتيجه

| | |
|-----------------|----------------|
| | متمايل به بالا |
| متمايل به پايين | متمايل به وسط |

```
<div dir="rtl">
<table border="2">
<td align="center">
سلول اول
</td>
<td rowspan="2" align="center">

</td>
<tr>
<td align="center">
سلول دوم
</td>
</tr>
</table>
</div>
```

نتیجه



```
<div dir="rtl">  
<table border="3">  
<td bgcolor="yellow">  
محتویات سلول اول  
</td>  
<td bgcolor="green">  
محتویات سلول دوم  
</td>  
</table>  
</div>
```

نتیجه

| | |
|------------------|------------------|
| محتویات سلول اول | محتویات سلول دوم |
|------------------|------------------|

```
<div dir="rtl">  
<table border="3" width="300">  
<tr bgcolor="pink">  
<td>صورتی</td>  
<td>این ردیف صورتی است</td>  
<tr bgcolor="gray">  
<td>خاکستری</td>  
<td>این ردیف خاکستری است</td>  
</table>  
</div>
```

نتیجه

| | |
|---------|----------------------|
| صورتی | این ردیف صورتی است |
| خاکستری | این ردیف خاکستری است |

```
<html>
<head>
<title>نمایش پرچم ایران با استفاده از کدهای اچ تی ام ال و سی اس اس</title>
<style type="text/css">
body { background-color:#ddd }
.top { background-color:green }
.middle { background-color:white; text-align:center }
.bottom { background-color:red }
</style>
</head>
```

نتیجه



پروژه‌ای با استفاده از تکنیک

JavaScript

محاسبه فاکتوریل اعداد با استفاده از جاوا اسکریپت

مطرح شده توسط: ایوب رشیدی

بدست آوردن فاکتوریل با دو تا جعبه ساده و یک `button` که عدد را در جعبه اول نوشته و در دومی فاکتوریل را محاسبه کند با استفاده از فرم در صفحه اینترنت موضوع طراحی سایت یعنی بر روی `button` که کلیک کنی فاکتوریل بگیرد.

انجام عملیات ریاضی با استفاده از HTML امکان ندارد چون در حقیقت HTML را نمیتوان یک زبان برنامه نویسی محسوب کرد. HTML یک زبان علامت گذاری ابر متن است. برای انجام عملیات ریاضی باید از زبانهای برنامه نویسی نظیر PHP, ASP و JavaScript استفاده کرد. در این مورد استفاده از جاوا اسکریپت به دلیل اینکه برای اجرا شدن به امکانات اضافی در سرور احتیاج ندارد مناسب تر است.

فاکتوریل یک عدد برابر با حاصلضرب همه اعداد صحیح مثبت کوچکتر از عدد مورد نظر است.

برای مثال فاکتوریل 5 به صورت زیر محاسبه می شود:

$$5! = 5 \times 4 \times 3 \times 2 \times 1 \Rightarrow$$

$$5! = 120$$

استثنایی که در فاکتوریل وجود دارد عدد صفر است. فاکتوریل عدد صفر برابر با 1 است.

با استفاده از تابع زیر میتوانیم فاکتوریل یک عدد را به دست آوریم:

```
function factorial (aNumber) {  
    // این قسمت در صورتی که عدد وارد شده به صورت اعشاری باشد آنرا به اولین عدد صحیح  
    بزرگتر از آن گرد میکند  
    aNumber = Math.floor(aNumber);  
  
    // این قسمت در صورتی که مقدار وارد شده منفی باشد یا عدد نباشد تابع را متوقف میکند  
    if(!isFinite(aNumber) || aNumber < 0)  
        return false;  
  
    // این قسمت در صورتی که عدد وارد شده صفر باشد مقدار 1 را باز میگرداند  
    if(aNumber == 0)  
        return 1;  
  
    var output = 1;  
    for (i = 1; i <= aNumber; i++) {  
        output *= i;  
    }  
  
    return output;  
}
```

حالا نوبت به طراحی قسمت مربوط به وارد کردن عدد و نمایش نتیجه در صفحه می رسد. برای

این کار میتوانیم از کد زیر استفاده کنیم:

عدد مورد نظر را وارد کنید:

```
<input type="text" name="number" id="number" size="10">
```

```
<input type="button" value="محاسبه" onclick="CountResult()">
```

```
<input type="text" name="result" id="result">
```

ما برای راحتی کار از تابع زیر برای خواندن عدد مورد نظر و نمایش نتیجه آن استفاده میکنیم:

```
function CountResult() {  
    // خواندن عدد ورودی  
    var inputNumber = document.getElementById("number").value;  
    // اجرای تابع محاسبه فاکتوریل  
    var result = factorial(inputNumber);  
  
    if(!result) {  
        alert("مقدار وارد شده معتبر نیست");  
    }  
    else {  
        document.getElementById("result").value = result;  
    }  
}
```

در اینجا می‌توانید کد صفحه‌ای را که برای پیدا کردن فاکتوریل اعداد طراحی کردیم مشاهده کنید:

```
<html>
<head>
<script type="text/javascript">
function factorial(aNumber) {
    aNumber = Math.floor(aNumber);
    if(!isFinite(aNumber) || aNumber < 0)
        return false;
    if(aNumber == 0)
        return 1;
    var output = 1;
    for (i = 1; i <= aNumber; i++) {
        output *= i;
    }
    return output;
}

function CountResult() {
    var inputNumber = document.getElementById("number").value;
    var result = factorial(inputNumber);
    if(!result) {
        alert("مقدار وارد شده معتبر نیست");
    }
    else {
        document.getElementById("result").value = result;
    }
}
```

```
    }  
}  
</script>  
</head>  
<body dir="rtl">  
<p>  
عدد مورد نظر را وارد کنید:  
  
<input type="text" name="number" id="number" size="10">  
<input type="button" value="محاسبه" onclick="CountResult()">  
  
<input type="text" name="result" id="result" size="10">  
</p>  
</body>  
</html>
```

نتیجه

یک نمونه از فرم ایجاد شده با روش بالا را می توانید در اینجا مشاهده کنید:

| | | |
|-----|--------|---|
| ۱۲۰ | محاسبه | ۵ |
|-----|--------|---|

تماس با ما

- برای تماس با ما می توانید از فرم زیر استفاده کنید.
- همچنین میتوانید به صورت مستقیم با آدرس ایمیل Ayooob.rashidi@gmail.com مکاتبه نمایید.
- در صورت تمایل برای تماس تلفنی میتوانید با شماره 09387208853 تماس حاصل فرمایید. در صورت امکان پیام خود را به صورت SMS ارسال فرمایید.
- در صورتی که می خواهید ما با شما تماس بگیریم
- وارد وبلاگ pc-boukan.blogfa.com شوید و حتماً نام و ایمیل خود را وارد نمایید.