

# بسمه تعالی

## فهرست

### بخش اول :

فصل اول ( شروع کار با Html )

فصل دوم ( متن ها در Html )

فصل سوم ( لینک ها و تصاویر در Html )

فصل چهارم ( کار با فرم ها در Html )

فصل پنجم ( قابها و جداول Html )

فصل ششم ( نکات تکمیلی در مورد Html )

### بخش دوم :

فصل هفتم ( شروع کار با CSS )

فصل هشتم ( ویژگی های متن در CSS )

# بخش اول

## فصل اول

### - Html چیست ؟

برنامه های تحت وب را می توان به دو دسته تقسیم کرد :

#### 1- برنامه های Static

#### 2- برنامه های Dynamic

برنامه های Static : برنامه های هستند که صفحات آنها از تصاویر، صوت، متن و غیره تشکیل شده است و احتمالاً یک یا چند لینک به سایر آدرس ها در آنها قرار گرفته است و در نتیجه می توان گفت این گونه برنامه ها در تامل با کاربر قرار ندارد ، کاربران تنها می توانند از صفحات از پیش طراحی شده استفاده کنند.

برنامه Dynamic : برنامه های هستند که داده ای را از کاربر دریافت می کنند ، مورد پردازش قرار می دهند و نتیجه آن را در اختیار کاربر قرار می دهند پس می توان گفت خصیصه اصلی آنها این می باشد که با کاربر در تامل بوده و با گرفتن ورودی ها از کاربر برای او تحلیل و یا پردازشی انجام داده و با توجه به درخواست کاربر به او سرویس می دهند.

(زبان نشانه گذاری متن) **HTML = Hyper Text Markup Language**

برای نوشتن برنامه ها یا سایت های استاتیک از نرم افزارهای گوناگونی استفاده می شود ولی توجه داشته باشید که در انتها صفحه طراحی شده باید به فرمت فایل های متنی و Html برای نمایش در وب تبدیل شود .

این زبان ، یک زبان مبتنی بر متن یا Text Base می باشد که ساختار آن بر تگ ها بنا شده است .

یک سند Html یک پرونده مبتنی بر متن است که معمولاً با پسوند (.htm یا .html) نام گذاری شده و محتویات آن از برجسب های Html تشکیل می شود .

**- Web Browser : web browser** یک نرم افزار است که به صورت interpreter می مفصر کار می کند interpreter قابلیت تفسیر تگ های Html و همچنین اسکریپت ها رو دارد . فرق interpreter با compiler در این می باشد که کامپایلر کل برنامه رو در صورت صحیح بودن اجرا می کند اما اینترپرتر خط به خط کد را اجرا می کند به عبارت دیگر زماری که یک کامپایلر می خواهد یک برنامه را کامپایل کند باید به تمام کدها برنامه دسترسی داشته باشد و همه آنها را ابتدا خوانده و بررسی کند که آلی در آن خطا وجود دارد یا خیر سپس در صورتی که برنامه خطای نداشته باشد آن را کامپایل نموده و برنامه اجرا می شود اما در اینترپرتر روند دیگری وجود دارد به این ترتیب که برنامه ها در هنگام فعالیت می تفسیر دستورات به صورت خط به خط پیش رفته و در همان لحظه آن خط را اجرا کرده و نتیجه آن را نمایش می دهد . اینترپرتر در صورت برخورد با خطا در آن خط اعلام خطا می کند اما متوقف نمی شود .

### - شروع کار با Html :

برای آغاز کار با Html باید از یک Editor استفاده کنید که در این رابطه میتوان به WordPad ، Notepad و Microsoft Office اشاره کرد ، البته برای نوشتن دستورات Html ویرایشگر های پیشرفته وجود دارد .

### - ساختار تگ ها در Html :

یک تگ یا نشانه همیشه با علامت کوچکتر ریاضی (<) آغاز شده و با علامت بزرگتر (>) به پایان می رسد .  
فرم کلی تگ ها به صورت زیر است :

< Start Tag > Text </End Tag>

تگ ممکن است دارای تعدادی ویژگی باشد این ویژگی خواصی را برای آن متن بیان می کند این ویژگی در تگ آغازین نوشته می شود و شامل موارد متعددی هستند که بسته به نوع تگ ها متفاوت است اکثر تگ ها دارای ویژگی هستند. تگ زیر خاصیت رنگ زمینه را به رنگ قرمز تنظیم می کند .

```
<body bg color ="red"> Text </body>
```

بعضی از تگ ها از یک بخش تشکیل شده اند که آن تگ ها موضوعی را در برنمی گیرند و فقط برای انجام عملی به کار می رود و فرمول آن به شکل زیر است :

```
<br/>
```

این تگ برای شکستن یک سطر و رفتن به سطر بعدی مورد استفاده قرار می گیرد . نکته قابل توجه دیگر در مورد تگ ها این است که نحوه قرار گیری متن در بین دو تگ اهمیتی ندارد .

– سند Html :

سند Html ساختاری است که با استفاده از آن یک صفحه وب را ایجاد می کند این ساختار به صورت کلی به شکل زیر است :

```
<html> شروع سند
```

```
<head>
```

```
<title> عنوان </title>
```

```
</head>
```

```
<body>
```

بدنه اصلی سند

</body>

پایان سند<html>

بعد از کامل شدن یک سند Html مانند سند بالا باید آن را در سطح دیسک ذخیره کرد اما توجه باید داشت که پسوند فایل یا سند شما ( .html ) یا ( .htm ) باشد .

## فصل دوم

– متن ها در Html:

انواع نوشتن متن در Html : تگ های برای کار کردن با متون وجود دارد که در جدول زیر به معرفی تعدادی از آن پرداخته شده است .

توضیحات	تگها
متن را بصورت برجسته نمایش میدهد	<b>
متن را بصورت بزرگ نمایش میدهد	<big>
متن را بصورت مورب نمایش میدهد	<em>
متن را بصورت مورب نمایش میدهد	<i>
متن را بصورت کوچک نمایش میدهد	<small>
متن را بصورت برجسته نمایش میدهد	<strong>
متن را بصورت زیرنویس نمایش میدهد	<sub>
متن را بصورت بالانویس نمایش میدهد	<sup>
قطعه متنی را که در متن اصلی جا داده شده نمایش میدهد	<ins>
قطعه متنی را که از متن اصلی حذف شده نشان می دهد	<del>
متن را بصورت خط خورده نمایش میدهد	<s>
متن را بصورت خط خورده نمایش میدهد	<strike>
متن را بصورت زیرخط دار نمایش میدهد	<u>

- پاراگراف بندی متون : برای پارگراف بندی متون از تگ <p> استفاده میکنیم که متن یک پاراگراف با پاراگراف بعدی به اندازه دو خط خالی می گذارد .

<p> text </p>

<p> text 2 </p>

- تعیین ویژگی های متن : برای تنظیم خصوصیات یک متن از جمله نوع فونت ، اندازه فونت و رنگ آن از تگ <font> استفاده می کنیم .

<font size="+ or – (2)"> This is a test </font>

مقادیر ورودی بر تعیین سایز فونت می تواند مثبت یا منفی به همراه یک عدد باشد، از علامت مثبت برای زمانی که می خواهیم اندازه متن بزرگتر شود و از علامت منفی هنگامی که می خواهیم اندازه متن کوچکتر شود استفاده می کنیم .

شناسه دیگر این تگ **face** می باشد از این شناسه برای تعیین فونت متن استفاده می کنیم.

**<font face="arial" size="+2"> text </font>**

- ویژگی دیگر متن ها که می توانیم از طریق تگ فونت آن را تعیین کنیم رنگ آنها می باشد برای اینکه رنگ یک آیتم را در **Html** را تعیین کنیم دو روش متفاوت وجود دارد اولین روش استفاده از نام رنگ می باشد ، استفاده از این روش یک عیب دارد و آن اینکه با این روش نمی توان همه رنگ ها را تعیین کنیم برای رفع این مشکل از اعداد هگزا دسیمال برای مقدار دهی استفاده می کنیم .

**<font color="red"> This is a Test </font>**

**<font color="#FF0000"> This is a Test </font>**

- **ایجاد خط** : برای جدا کردن قسمت های مختلف یک صفحه می توانیم خط های افقی با پهنای متفاوت ایجاد کنیم برای اینکار از تگ **<hr/>** استفاده کنیم .

**<hr size="10" width="100"/>**

- **کارکتر های خاص** : اگر بخواهیم بعضی از کارکتر های خاص مثل علامت کوچکتر و بزرگتر را تایپ کنیم و هم این طور بخواهیم بین کلمات بیش از یک کارکتر فاصله قرار دهیم از یک سری دستور که در جدول زیر مشخص شده استفاده می کنیم .

Result	Description	Entity Name	Entity Number
	non-breaking space	&nbsp;	&#160;
<	less than	&lt;	&#60;
>	greater than	&gt;	&#62;
&	ampersand	&amp;	&#38;
¢	cent	&cent;	&#162;
£	pound	&pound;	&#163;
¥	yen	&yen;	&#165;
€	euro	&euro;	&#8364;
§	section	&sect;	&#167;
©	copyright	&copy;	&#169;
®	registered trademark	&reg;	&#174;
™	trademark	&trade;	&#8482;

- لیست های تعریف : لیستی از عبارات و توضیحات مربوط به عبارت می باشد ، برای تعریف این لیست از تگ <dl> استفاده می شود . به این صورت که ابتدا و انتها لیست آن تگ قرار می گیرد ، سپس در بدنه این لیست عناوین و عباراتی که می خواهیم تعریف کنیم را در تگ <dt> قرار داده و تعریف مربوط به آنها را در تگ <dd> و بلافاصله پس از آنها قرار می دهیم .

<dl>

<dt></dt>

<dd></dd>

</dl>



## فصل سوم

### - لینک و تصاویر

- آشنایی با تگ **a** : کار تگ **a** ساخت لینک به بقیه صفحات است . برای ایجاد لینک از تگ **<a>** استفاده می شود . این تگ دارای یک صفت (**href**) می باشد که برای آدرس دهی به مقصد مورد نظر استفاده می شود . به عنوان مثال =

**< a href="http://www.Google.com"> Google </a>**

- اضافه کردن تصویر به صفحه : هرگاه بخواهید به صفحه خود به جز متن ، تصویری را اضافه کنید باید از تگ **<img>** استفاده کرد که مخفف **Image** می باشد .

این تگ دارای شناسه به نام **src** می باشد این شناسه آدرس محل فایل عکس را به عنوان مقدار دریافت می کند توجه داشته باشید که تگ **Image** تنها از یک قسمت تشکیل شده و قسمت ابتدائی و انتهای ندارد .

****

تگ **<img>** و شناسه های آن :

1-dir = نحوه نمایش یا محل قرار گیری متن و غیره در صفحه را تعیین می کند .

2-bgcolor = رنگ زمینه صفحه را تعیین می کند .

3-text = رنگ نوشته متن را تعیین می کند .

4-link = رنگ پیش فرض لینک را تعیین می کند .

5-alink = زمانی که کاربر روی لینک کلیک می کند رنگ آن را می توان تعیین کرد .

6-vlink = رنگ لینک های که قبلا کاربر بازدید کرده است را تعیین می کند .

ساخت لینک با استفاده از عکس : برای انجام اینکار باید در داخل تگ <a> تگ <img> را قرار دهیم  
به عنوان مثال :

</a> >"نام عکس همراه با پسوند" <img=" >"نام سایت مورد نظر" a href="

برای ایجاد حاشیه دور عکس می توان از صفت border در تگ <img> استفاده کرد که اگر مقدار برابر با صفر باشد حاشیه برداشته می شود .

- **لینک های E-mail :** در این قسمت می خواهیم لینکی ایجاد کنیم که بیننده بتواند با کلیک برروی آن به آدرس تعیین شده ایمیل بفرسد برای اینکار از تگ <a> استفاده می شود 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 لینک ایمیل بشناسد باید از عبارت mailto به جای Http در آغاز آدرس استفاده کنیم وبعد از آن آدرس ایمیل را بنویسیم.

<a href="mailto:test@test.com"> Email </a>

برای ایجاد موضوع بعد از قسمت ایمیل یک علامت ؟ و بعد از آن کلمه subject را قرار میدهیم .

## فصل چهارم

### آشنایی با با فرم ها

– فرم ها : فرم ها در صفحات Html برای ارسال اطلاعات ورودی توسط کاربر به سمت مقصد استفاده

می شوند . به طوری کلی اگر در صفحه بخواهیم از بازدید کننده اطلاعاتی را دریافت کنیم ، باید از فرم استفاده نماییم . برای ایجاد فرم از تگ `<form>` استفاده می شود . این تگ دارای صفحات زیادی است

که تعدادی از آنها در زیر مشخص شده است :

1- Action : ( مشخص کننده صفحه که فرم باید به آن ارسال شود . وجود ایت شناسه ضروری است )

2- Method (get / post) : ( طریقه ارسال اطلاعات به فایل action را مشخص می کند )

3- Name : ( برای تعریف یک نام منحصر به فرد برای فرم )

4- Target : ( روش باز شدن صفحه پردازشگر اطلاعات را مشخص می کند )

– Textbox ها : برای اینکه بتوانیم امکانی را فراهم کنیم تا کاربران اطلاعات را در صفحه Html وارد نمایند از تگ `<input>` استفاده می شود از تگ `<input>` میتوان برای ساختن فیلدهای مربوط به وارد کردن متن ، کادر کلمه عبور ، کادرهای مخفی ، دکمه های رادیویی ، چک باکس ، دکمه ارسال فرم ، دکمه پاک کردن فرم و انواع دیگر دکمه ها و چند نوع ورودی دیگر استفاده کنیم .

`< input type = " text " >`

صفات تگ `<input>` :

1- Type : نوع فیلد را مشخص میکند .

2- Name : اختصاص یک نام برای ورودی مربوطه

3-Maxlength : تعیین حداکثر تعداد کارکتر

4-Size : طول کادر بر حسب کارکتر

5- Value : متن پیش فرض در فیلد

6- Readonly : با مقدار Readonly دیگر امکان ویرایش متن وجود ندارد

7- Disabled : با مقدار disabled فیلد غیر فعال می شود

- فیلد کلمه عبور : برای ایجاد کادر کلمه عبور باید شناسه type تگ <input> را برابر با password قرار دهیم .

< input type = “ password” >

- **Checkbox ها :** چک باکس ها بیشتر در مواردی که امکان انتخاب شدن یا نشدن برای یک گزینه به صورت مستقل وجود داشته باشد بکار می رود . برای ساختن چک باکس ها باید از تگ استفاده کرده و مقدار شناسه آن را برابر با چک باکس قرار داد .

< input type = “ checkbox “ name = “ test “ >

شناسه های Checkbox :

Type	- 1
Name	- 2
Value	- 3
Checked	- 4
Disabled	- 5

**RadioBotton ها :** این ابزار بسیار شبیه چک باکس ها است و برای انتخاب گزینه ها بکار می رود تفاوت اصلی این ابزار با چک باکس در این می باشد که به وسیله **RadioBotton** ها از بین گزینه های موجود تنها یکی را می توان انتخاب کرد .

```
< input type= “ radio “ name = “radiobotton” >
```

شناسه های **RadioBotton** :

Type- 1

Name- 2

Value- 3

Checked- 4

Disabled- 5

**Botton ها :** **Botton** ها در **Html** انواع مختلفی دارند که برای ساختن آنها مانند موارد قبل از تگ **<input>** استفاده می شود

اولین نوع **Botton** ها دکمه **Submit** است این دکمه برای ارسال اطلاعات وارد شده در فرم به صفحه **Action** استفاده می شود .

```
< input type = “submit ” value = “ عبارت روی دکمه برای نمایش ” >
```

دومین نوع **Botton** ها دکمه **Reset** است این دکمه برای پاک کردن اطلاعات فرم بکار می رود .

```
< input type =”Reset” value = “Reset “ >
```

سومین نوع Botton ها دکمه Botton است این دکمه وظیفه دارد کدهای جاوا اسکریپت را اجرا کند .

شناسه دکمه های Botton :

Type- 1

Name- 2

Value- 3

Disabled- 4

- **فیلد متن چند خطی** : برای ایجاد فیلد چند سطری از تگ <text area> استفاده می شود با استفاده از یک <text area> کاربران می توانند تعداد نامحدودی کارکتر را در بیش از یک سطر وارد کنند .

< text area name =" name " > text < / text area>

شناسه های تگ < Text Area > :

1 - Name : یک نام تعیین میکند .

2- Cols : عرض فیلد را تعیین میکند .

3- Rows : طول فیلد را تعیین میکند .

4 - ReadOnly : با مقدار ReadOnly دیگر امکان ویرایش متن وجود ندارد.

5 - Disabled : با مقدار Disabled فیلد غیر فعال می شود.

## فصل پنجم

### قابها و جداول

– قابهای **Html**: اگر بخواهیم در یک صفحه بیش از یک سند داشته باشیم باید با مفهوم قاب ها در **Html** آشنا شد. برای ایجاد قاب ها از تگ با عنوان `<frame set>` استفاده می شود این تگ تعیین می کند چگونه یک صفحه به قسمت های مختلف تقسیم شود. هر کدام از قسمت نیز با تگ `<frame>` در داخل این تگ مشخص می شوند که دستور آن به شکل زیر است:

```
< frame set cols= “ 25% , 75% “ >
```

```
<frame src = “ “ >
```

```
<frame src = “ “ >
```

```
< / frame set >
```

برای نمایش یک قاب در صفحه از تگ `<iframe>` استفاده می شود و یا اگر بخواهیم در صفحه یک صفحه دیگر را باز کرده و با آن رفتاری مانند تصاویر داشته باشیم از این تگ استفاده می کنیم.

شناسه های تگ `<iframe>`:

Width- 1

Height- 2

Border- 3

- آشنایی با جداول در **Html** : این جداول از سطر ها و ستون ها تشکیل شده اند که با یکدیگر سلول ها را به وجود می آورند یک سلول داده می تواند شامل متن ها ، عکس ها ، لیست ها ، پاراگراف ها ، فرم ها و خط های افقی و همچنین جدول ها باشد . برای ایجاد یک جدول در صفحه از تگ زیر استفاده می شود .

< table >

اطلاعات و محتویات جدول

< /table >

تگ <td> مخفف tabledata است و محتویات مورد نظر هر سلول باید در این تگ نوشته شود با استفاده از صفت border در تگ table کادر ایجاد میشود که جدول واضح دیده شود . برای ایجاد جدول های با چند سطر از تگ <tr> استفاده می شود به وسیله این تگ می توان سطر های یک جدول را از هم جدا کرد .

<table border = " 2 " >

<tr>

<td> 1 </td>

<td> 2 </td>

</tr>

</table>

شناسه های جدول :



- 1- Width : ( عرض جدول را تعیین می کند . )
- 2- Height : ( طول جدول را تعیین می کند . )
- 3- Valign : ( تعیین موقعیت محتویات به صورت عمودی )
- 4- Align : ( تعیین موقعیت محتویات به صورت افقی )
- 5- Border : ( میزان حاشیه یا کادر جدول را تعیین می کند . )
- 6- Border Color : ( رنگ حاشیه را تعیین می کند . )
- 7- Cellspacing : ( ایجاد فاصله بین سلول ها )
- 8- Cell padding : ( برای افزودن فضا به داخل سلول )

شناسه های تگ <td> :

- 1- Width : ( عرض جدول را تعیین می کند . )
- 2- Height : ( طول جدول را تعیین می کند . )
- 3- Valign : ( تعیین موقعیت محتویات به صورت عمودی )
- 4- Align : ( تعیین موقعیت محتویات به صورت افقی )
- 5- Border : ( میزان حاشیه یا کادر جدول را تعیین می کند . )
- 6- Border Color : ( رنگ حاشیه را تعیین می کند . )
- 7- Rowspan : ( مشخص کننده این است که ارتفاع یک سلول معادل چند سلول دیگر باشد )
- 8- Colspan : ( مشخص کننده این است که اندازه افقی یک سلول معادل چند سلول دیگر است )

برای ایجاد عنوان در جدول از تگ <caption> استفاده می شود این تگ بعد از تگ <table> قرار

می گیرد .

## فصل ششم

### نکات تکمیلی در مورد Html :

– اضافه کردن فیلم و صدا به صفحه : برای قرار دادن صدا یا فیلم در صفحه دو روش وجود دارد ، روش اول این است که شما می توانید یک لینک به فایل صدا یا فیلم ایجاد کنید تا کاربران بتوانند آن را دانلود کرده و سپس به آن گوش دهند ، روش دوم قرار دادن صدا در صفحه است این روش برای پخش صدا در صفحه می باشد برای اینکار از تگ <embed> استفاده می شود .

<embed src = “ test.mp3 ” > </ embed>

شناسه های تگ <embed> :

1 – Width : ( تعیین عرض فایل صوتی یا تصویری )

2 – Height : ( تعیین طول فایل صوتی یا تصویری )

3 – Autostart : ( با دو مقدار Ture و False ، در صورت Ture بودن به محض باز شدن فایل پخش میشود )

4 – Loop : ( با دو مقدار Ture و False ، در صورت Ture بودن فایل پس از اتمام دوباره اجرا میشود )

5- Hidden : ( با دو مقدار Ture و False ، در صورت Ture بودن این شناسه به مرورگر دستور مخفی کردن رابط گرافیکی پخش صوت را می دهد و امکان کنترل را از کاربر می گیرد )

- **مِتَا تَگ ها :** مِتَا تَگ ها ، تَگ های هستند که در تَگ head صفحه بکار گرفته می شود این تَگ ها برای دادن اطلاعات راجع به صفحه Html موجود به کار می روند. اطلاعاتی که در تَگ های مِتَا قرار دارند بر حسب نوع این تَگ ها متفاوت هستند و شامل نام نویسنده ، برنامه تولید کننده ، کلمات کلیدی صفحه و دیگر توضیحات راجع به صفحه می باشد اهمیت اصلی این تَگ ها این است که موتور های جستجو برای پیدا کردن صفحات مورد نظر کاربران از این تَگ های استفاده می کنند و محتوایی موجود در این تَگ ها را در هر صفحه مورد بررسی قرار می دهند .

<meta name = “ نوع مِتَا تَگ ” content = “ محتویات مِتَا تَگ توسط موتور جستجو ” >

- **آشنایی با تَگ <div> :** این تَگ برای ایجاد یک لایه در صفحه Html به کار می رود . در نسل جدید Html و همچنین در ظاهر بندی صفحه سیار زیاد از این تَگ استفاده میشود این تَگ از تَگ های اصلی صفحه می باشد و برای مشخص کردن یک بخش یا قسمتی از صفحه به کار می رود از این تَگ در CSS و برای تعیین مشخصه هر قسمت از صفحه به کمک استایل ، استفاده زیادی می کنیم .

## بخش دوم

### فصل هفتم

## شروع کار با CSS (Cascading Style Sheets)

- آشنایی با **CSS : CSS** روشی است برای قالب بندی و طراحی اجزای صفحه از قبیل متن ، تصویر ،

کادر ها و تقریبا هر چیزی که در طراحی صفحه استفاده می شود .

مزایای CSS :

1 -طراحی ظاهر صفحات بدون استفاده از دستورات Html

2 -جلوگیری از تکرار دستورات

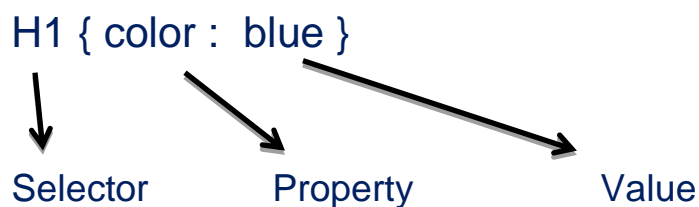
3 -سرعت بارگذاری بیشتر برای صفحات

4 -استفاده از فایل CSS خارجی

- **طریقه نوشتن کدهای CSS :** برای نوشتن دستورات باید با سه مفهوم Selector ، Property ، Value

آشنا بود . هر دستور CSS شامل این سه قسمت می باشد .

دستور زیر ، متن نوشته شده توسط کليه تگ های H1 ، در صفحه را به رنگ آبی نمایش می دهد .



**Class** و **id** شناسه های پرکاربرد آن می باشد . اکثر تگ های **Html** دارای این شناسه ها هستند برای نمونه می خواهیم در صفحه دو نوع پاراگراف داشته باشیم یکی با متن مشکی و دیگری با متن قرمز ، برای مشخص کردن این دو پاراگراف می توانیم از شناسه ی **class** استفاده کنیم .

```
p.black { color : black }
```

```
p.red { color : red }
```

```
<p class = " black " > text </p>
```

```
<p class = " red " > text </p>
```

- **وارد کردن CSS با استفاده از شناسه Style** : ساده ترین راه برای افزودن استایل به یک صفحه استفاده از شناسه ی استایل در تگ های **Html** است . برای اضافه کردن استایل به یک تگ باید ویژگی های مورد نظر را به عنوان مقدار به شناسه ی استایل آن تگ تخصیص دهیم .

```
<div style = "color : green " > text </div>
```

امکان دیگری که با استفاده از شناسه ی استایل وجود دارد این است که به یک تگ بیش از یک ویژگی افزوده شود برای اینکار باید بین ویژگی های مختلف از یک علامت سمیکالون ( ; ) استفاده شود .

- **استفاده از CSS به صورت یک فایل خارجی** : برای قرار دادن یک سری استایل در چندین صفحه که همه صفحات مشابه هستند می توان از یک فایل استایل خارجی استفاده کرد تا نیازی به چندین بار نوشتن استایل ها نباشد . برای انجام اینکار باید ابتدا یک فایل برای تنظیم استایل های مورد نظر ایجاد کرده و استایل ها را در آن قرار می دهیم در آخر این فایل را با پسوند ذخیره می کنیم ، آنگاه در هر فایل که نیاز به آن داشته باشیم با نوشتن یک تگ لینک در حد صفحه این موضوع را مشخص می کنیم .

```
div { font – style : italic ; color : green }
```

شناسه های تگ لینک :

1 – **rel** : ( با مقدار **stylesheet** این شناسه مشخص می کند فایل مورد نظر یک استایل است )

2 - type : ( با مقدار text/css مشخص می شود که نوع متن فایل برای مرورگر می باشد )

3 - href : ( این شناسه آدرس قرار گیری فایل را دریافت می کند )

```
< link rel = " stylesheet " type = " text/css " href = " " />
```

- تنظیم عرض و ارتفاع با استفاده از **CSS** : برای تنظیم عرض و ارتفاع در CSS از شناسه های width و height استفاده می شود .

```
< div style = " width : 100% " > < /div>
```

```
< div style = " height : 100% " > < /div>
```

از شناسه ی overflow برای ثابت نگه داشتن ارتفاع یا عرض <div> استفاده می شود که مقدار hidden ، scroll ، visible را دارا می باشد .

- ایجاد کادر : با استفاده از استایل می توان برای اجزا یک صفحه کادر قرار داد . برای ایجاد کادر از پارامتر

border-style استفاده می شود . مقدار های که برای این پارامتر می توان استفاده کرد :

- solid- 1
- dashed - 2
- dotted- 3
- double- 4
- groove- 5
- hidden- 6
- Inset- 7
- outset- 8
- ridge- 9

برای تعیین عرض کادر از پارامتر **border-width** استفاده می شود که با مقدار **thin** باعث می شود کادر باریک و با مقدار **thick** کادر ضخیم شود و با استفاده از پارامتر **border-color** می توان رنگ کادر را تعیین نمود .

– **تنظیم حاشیه و فاصله از لبه :** برای تنظیم حاشیه در عناصر صفحه از پارامتر **margin** استفاده می شود

```
< div style = “ margin-left : 100px ; border – style : double “ > text < /div>
```

پارامتر **padding** در **css** مشابه شناسه **cellpadding** در جدول **html** است با این تفاوت که می توان آن را برای جهات مختلف به صورت جداگانه تعریف کرد .

padding - left                      برای تعیین فاصله از سمت چپ

padding – right                    برای تعیین فاصله از سمت راست

– **تعیین موقعیت عناصر با استفاده از CSS :** برای تعیین موقعیت عناصر از پارامتر **position** در شناسه استایل استفاده می شود . ویژگی **position** این امکان را به ما می دهد موقعیت یک عنصر در صفحه را مشخص کنیم ، مقدار های متفاوتی برای این شناسه وجود دارد :

1- **static :** ( مقدار پیش فرض است . با این مقدار اجزا در همان جای که کد آنها هستند دیده می شوند )

2- **fixed :** ( موقعیت یک عنصر را در پنجره مرورگر مشخص می کند )

3- **absolute :** ( نمایش یک عنصر از صفحه در هر موقعیت نسبت به جهات مختلف صفحه شامل بالا ، پایین ، چپ ، راست را مشخص می کند )

## فصل هشتم

### ویژگی متن در CSS

– ویژگی **letter-spacing** : این ویژگی به ما اجازه می دهد تا فاصله ی میان حروف یک کلمه را در یک متن تعیین نماییم این ویژگی مقدار اولیه و پیش فرض نرمال را دارد .

```
< div style = “ letter-spacing : 5px “ > text < /div>
```

– ویژگی **text-align** : این ویژگی نحوه قرار گیری قسمتی از متن را از لحاظ راست چین ، چپ چین ، یا وسط چین بودن تعیین می کند .

```
< div style = “ text – align : left or right or center “ > < /div>
```

– ویژگی **text-decoration** : این ویژگی به ما اجازه می دهد تا برای قسمتی خاص از متن خاصیت های از قبیل زیر خط و غیره را تعریف نماییم . این پارامتر برای تعیین ویژگی مورد نظر دارای مقادیر به عنوان ورودی به ترتیب زیر می باشد :



- 1- none : ( پیش فرض )
- 2- under line : ( برای زیر خط کردن متن ها به کار می رود )
- 3- over line : ( برای نمایش خطی در بالای متن به کار می رود )
- 4- line through : ( یک خط را در میانه متن ایجاد می کند و باعث می شود که متن به صورت خط خورده نمایش داده شود )
- 5- blink : ( برای چشمک زدن متن به کار می رود )

- ویژگی **text-transform** : این ویژگی در متن تعیین می کند که نوشته متن در صفحه با حروف بزرگ یا کوچک نوشته شود . مقدار های که این ویژگی به عنوان ورودی دریافت می کند شامل :

None -1

Lowercase -2

Uppercase -3

4- Capitalize : (حروف اول کلمات را بزرگ می کند )

`< div style = “ text-transform : uppercase “ > text < /div>`

- تنظیم رنگ در **CSS** : برای تنظیم رنگ از پارامتر **color** استفاده می شود . از ویژگی **color** برای تغییر رنگ قسمت های مختلف متن صفحه استفاده می شود .

تگ **<span>** در ابتدا و انتهای قسمت مورد نظر از متن قرار می گیرد این تگ به ما این امکان را می دهد که در قسمت انتخاب شده بتوانیم با تعیین استایل خاصی اعمال تغییرات کنیم و ویژگی های را برای آن قسمت تعیین نماییم .

```
<div> text 1 text 2 <span style = " color : red "> text 3 </span>  
text 4 </div>
```

شناسه دیگری به نام **background** برای تعیین رنگ بکار می رود اما تفاوت آن با **color** در این است که به وسیله این شناسه رنگ زمینه تغییر می کند .

**background-color : " red "**

– **تصویر زمینه در CSS** : برای قرار دادن تصویر زمینه از پارامتر **background-image** استفاده می شود به این ترتیب که در قسمت مقدار آن آدرس فایل به همراه نام آن را قرار می دهیم .

```
<div style = " background-image : url (آدرس ونام فایل) "> </div>
```

**background-repeat** : با استفاده از این پارامتر می توان نوع تکرار تصویر را با مقادیر

Repeat-1 ، Repeat-2 ، No Repeat ، Repeat X-3 ، Repeat Y-4 مشخص کرد .

**background-attachment** : این پارامتر با دو مقدار **fixed** ، **scroll** تعیین می کند که آیا تصویر زمینه با اسکرول کردن صفحه حرکت کند یا ثابت بماند .

**background-position** : با استفاده از این پارامتر می توان موقعیت تصویر زمینه را در صفحه مشخص کرد .

- **تغییر نشانگر موس** : با استفاده از استایل این امکان وجود دارد که شکل نشانگر موس را وقتی روی قسمت مشخصی از صفحه قرار می دهیم تغییر کند . برای اینکار از پارامتر **CURSOR** در استایل مربوط به تگ مورد نظر استفاده میشود این پارامتر به عنوان مقدار ورودی عباراتی را دریافت می کند که نشان دهنده شکل مورد نظر برای موس است .

`< span style = " cursor : wait ; color : red ">< /span>`

در جدول زیر برخی از مقادیر مختلف پارامتر **CURSOR** که شکل نشانگر موس را عوض می کند نمایش داده شده است .

مقدار قابل استفاده	توضیحات
 <b>auto</b>	این گزینه نشانگر ماوس را با توجه به تنظیمات مرورگر کاربر تغییر می دهد .
<b>crosshair</b>	نشانگر ماوس به صورت به علاوه در می آید .
<b>default</b>	نشانگر ماوس به صورت همان فلش همیشگی در می آید .
<b>move</b>	نشانگر به شکلی در می آید که در هنگام جا به جا کردن چیزی به خود می گیرد .
<b>pointer</b>	نشانگر به شکل دست در می آید.
<b>help</b>	یک علامت سؤال کنار پیکان نشانگر ماوس قرار می گیرد .
<b>text</b>	به شکلی در می آید که بر روی متنها به خود می گیرد .
<b>wait</b>	به شکل ساعت شنی در می آید .
<b>n-resize</b>	یک فلش به سمت شمال
<b>s-resize</b>	یک فلش به سمت جنوب
<b>e-resize</b>	یک فلش به سمت مشرق
<b>w-resize</b>	یک فلش به سمت مغرب
<b>ne-resize</b>	یک فلش به سمت شمال شرقی
<b>nw-resize</b>	یک فلش به سمت شمال غربی
<b>se-resize</b>	یک فلش به سمت جنوب شرقی
<b>sw-resize</b>	یک فلش به سمت جنوب غربی

پایان

Tag	Description	DTD
<u>&lt;!--...--&gt;</u>	Defines a comment	STF
<u>&lt;!DOCTYPE&gt;</u>	Defines the document type	STF
<u>&lt;a&gt;</u>	Defines an anchor	STF
<u>&lt;abbr&gt;</u>	Defines an abbreviation	STF
<u>&lt;acronym&gt;</u>	Defines an acronym	STF
<u>&lt;address&gt;</u>	Defines contact information for the author/owner of a document	STF
<u>&lt;applet&gt;</u>	<b>Deprecated.</b> Defines an embedded applet	TF
<u>&lt;area /&gt;</u>	Defines an area inside an image-map	STF
<u>&lt;b&gt;</u>	Defines bold text	STF
<u>&lt;base /&gt;</u>	Specifies the base URL/target for all relative URLs in a document	STF
<u>&lt;basefont /&gt;</u>	<b>Deprecated.</b> Specifies a default color, size, or font for all the text in a document	TF
<u>&lt;bdo&gt;</u>	Overrides the current text direction	STF
<u>&lt;big&gt;</u>	Defines big text	STF
<u>&lt;blockquote&gt;</u>	Defines a long quotation	STF
<u>&lt;body&gt;</u>	Defines the document's body	STF
<u>&lt;br /&gt;</u>	Defines a single line break	STF
<u>&lt;button&gt;</u>	Defines a clickable button	STF
<u>&lt;caption&gt;</u>	Defines a table caption	STF
<u>&lt;center&gt;</u>	<b>Deprecated.</b> Defines centered text	TF
<u>&lt;cite&gt;</u>	Defines a citation	STF
<u>&lt;code&gt;</u>	Defines a piece of computer code	STF
<u>&lt;col /&gt;</u>	Defines attribute values for one or more columns in a table	STF
<u>&lt;colgroup&gt;</u>	Specifies a group of one or more columns in a table for formatting	STF
<u>&lt;dd&gt;</u>	Defines a description of an item in a definition list	STF
<u>&lt;del&gt;</u>	Defines text that has been deleted from a document	STF
<u>&lt;dfn&gt;</u>	Defines a definition term	STF
<u>&lt;dir&gt;</u>	<b>Deprecated.</b> Defines a directory list	TF
<u>&lt;div&gt;</u>	Defines a section in a document	STF
<u>&lt;dl&gt;</u>	Defines a definition list	STF
<u>&lt;dt&gt;</u>	Defines a term (an item) in a definition list	STF
<u>&lt;em&gt;</u>	Defines emphasized text	STF

<u>&lt;fieldset&gt;</u>	Groups related elements in a form	STF
<u>&lt;font&gt;</u>	<b>Deprecated.</b> Defines font, color, and size for text	TF
<u>&lt;form&gt;</u>	Defines an HTML form for user input	STF
<u>&lt;frame /&gt;</u>	Defines a window (a frame) in a frameset	F
<u>&lt;frameset&gt;</u>	Defines a set of frames	F
<u>&lt;h1&gt; to &lt;h6&gt;</u>	Defines HTML headings	STF
<u>&lt;head&gt;</u>	Defines information about the document	STF
<u>&lt;hr /&gt;</u>	Defines a horizontal line	STF
<u>&lt;html&gt;</u>	Defines the root of an HTML document	STF
<u>&lt;i&gt;</u>	Defines italic text	STF
<u>&lt;iframe&gt;</u>	Defines an inline frame	TF
<u>&lt;img /&gt;</u>	Defines an image	STF
<u>&lt;input /&gt;</u>	Defines an input control	STF
<u>&lt;ins&gt;</u>	Defines text that has been inserted into a document	STF
<u>&lt;kbd&gt;</u>	Defines keyboard input	STF
<u>&lt;label&gt;</u>	Defines a label for an <input> element	STF
<u>&lt;legend&gt;</u>	Defines a caption for a <fieldset> element	STF
<u>&lt;li&gt;</u>	Defines a list item	STF
<u>&lt;link /&gt;</u>	Defines the relationship between a document and an external resource	STF
<u>&lt;map&gt;</u>	Defines a client-side image-map	STF
<u>&lt;menu&gt;</u>	<b>Deprecated.</b> Defines a menu list	TF
<u>&lt;meta /&gt;</u>	Defines metadata about an HTML document	STF
<u>&lt;noframes&gt;</u>	Defines an alternate content for users that do not support frames	TF
<u>&lt;noscript&gt;</u>	Defines an alternate content for users that do not support client-side scripts	STF
<u>&lt;object&gt;</u>	Defines an embedded object	STF
<u>&lt;ol&gt;</u>	Defines an ordered list	STF
<u>&lt;optgroup&gt;</u>	Defines a group of related options in a drop-down list	STF
<u>&lt;option&gt;</u>	Defines an option in a drop-down list	STF
<u>&lt;p&gt;</u>	Defines a paragraph	STF
<u>&lt;param /&gt;</u>	Defines a parameter for an object	STF
<u>&lt;pre&gt;</u>	Defines preformatted text	STF
<u>&lt;q&gt;</u>	Defines a short quotation	STF