



در این قسمت، ویژگی های تگ های موجود در صفحه HTML را با استفاده از CSS تغییر خواهیم داد تا هریک در موقعیت مناسب خود قرار گرفته و به ظاهر مورد نظرم دست پیدا کنم.

در ادامه مطلب همراه من باشید.

خصوصیات مهمی که در این قسمت با آنها آشنا خواهیم شد و استفاده خواهیم کرد عبارتند از:

display: هر تگ در html به صورت های مختلفی به نمایش در می آید به عنوان مثال: تگ p به صورت block نمایش داده می شود، به این معنی که هنگامی که عنصر p در صفحه قرار می گیرد، عنصر یا تگ دیگر نمی تواند در کنار تگ p قرار بگیرد و به پایین این تگ انتقال پیدا می کند. برای مثال دیگر می توان به تگ span اشاره کرد که به صورت inline نمایش داده می شود بدین معنی که این تگ در یک خط قرار می گیرد، و عناصر هم جوارش قادر خواهند بود تا در یک سطح کنارش قرار بگیرند البته تا جایی که محدودیت عرض به آنها اجازه بدهد. در CSS با استفاده از ویژگی display می توان حالات مختلف نمایش را به تگ ها اعمال کرد باز هم به عنوان مثال: می توان به تگ a، حالت نمایش block را اختصاص داد که به صورت یک تگ block نمایش داده شود یا یک تگ div را با اختصاص دادن مقدار table به این ویژگی به صورت table نمایش داد.

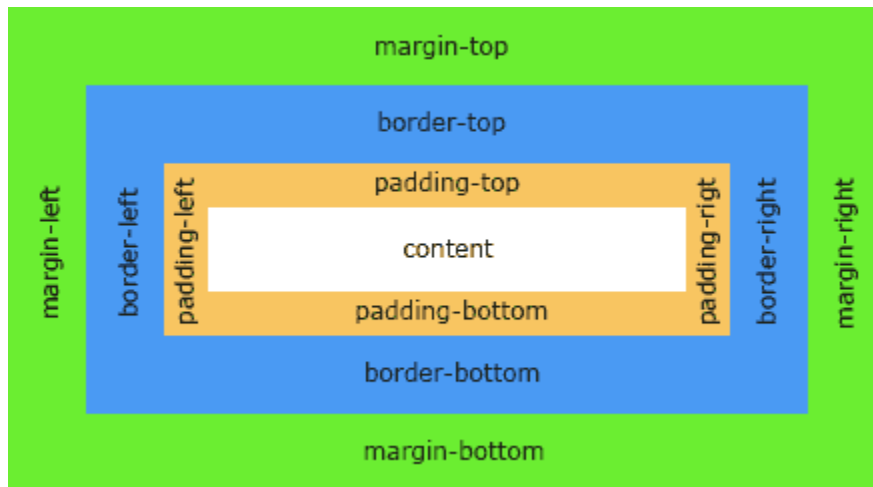
float: از این ویژگی برای کنار هم قرار دادن عناصر استفاده می شود، همانطور که از معنی این کلمه پیداست (شناور) با توجه به مقداری که برای این ویژگی در نظر گرفته شده است عناصر از چپ یا راست شناور می شوند.

clear: این ویژگی در ارتباط با ویژگی float است و باعث لغو ویژگی float می شود، به عنوان مثال هنگامی که به یک تگ ویژگی float را اختصاص می دهید عناصر هم جوار تگ، به صورت شناور در کنارش قرار خواهند گرفت در صورتی که به تگ هم جوار خصوصیت clear با یکی از مقادیر ممکن اختصاص یابد متناسب با مقدار اختصاص یافته ویژگی float لغو می شود.

width: با استفاده از این ویژگی می توان عرض یک تگ یا عنصر را معین کرد.

height: با استفاده از این ویژگی می توان ارتفاع یک تگ یا عنصر را معین کرد.

قبل از توضیح راجع به ویژگی های margin و padding به تصویر زیر دقت فرمایید:



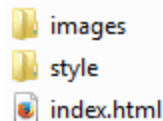
margin: فضای خالی بین عنصر و حاشیه مرورگر یا فضای خالی بین عناصر را مشخص می کند، همانطور که در تصویر مشاهده می کنید اندازه این ویژگی در 4 جهت قابل تغییر می باشد.

border: حاشیه یک تگ یا عنصر را مشخص می کند همانند margin، اندازه این ویژگی در 4 جهت قابل تغییر می باشد.

padding: فضای خالی بین border و محتوای یک عنصر را مشخص می کند، همانند دو ویژگی بالا اندازه ها در 4 جهت قابل تغییر می باشد.

در ادامه ویژگی های هر Selector (گزینشگر) را در کدهای CSS بررسی می کنیم.

در نظر داشته باشید که هر جایی که صفحه index.html را ایجاد کردید همواره در کنارش پوشه های منابع صفحه همچون: style، image یا font را ایجاد کنید و فایل های مورد نیاز صفحه را در جای مناسب قرار دهید. در اینجا من در کنار فایل index.html که کدهای html را در خود جای داده است پوشه style و image را ایجاد کردم و فایل CSS را در پوشه style و تصاویر را در پوشه image قرار دادم.



در پوشه image، برای سایت یک تصویر در نظر گرفتیم که از لینک زیر قابل دانلود می باشد:

[icon.png](#)

در ابتدا آدرس فایل CSS را توسط تگ link در قسمت head به صورت زیر ضمیمه صفحه می کنیم:

```
<head>
  <title>Advertising</title>
  <link rel="stylesheet" href="style/style.css">
</head>
```

همچنین به خاطر استفاده از کاراکترهای فارسی باید نوع کد گذاری مجموعه کاراکترهای صفحه را هم برابر با UTF-8 قرار بدم که توسط تگ meta و خصوصیت charset در قسمت head به صورت زیر قابل انجام می باشد:

```
<head>
  <title>Advertising</title>
  <meta charset="UTF-8">
  <link rel="stylesheet" href="style/style.css">
</head>
```

حالا فایل CSS من ضمیمه صفحه شده و قادر خواهم بود تا تغییرات موردنظرم را بر عناصر تشکیل دهنده صفحه HTML اعمال کنم.

"هر ویژگی جدیدی که در گزینشگرهای CSS آمده را بررسی می کنم و از توضیح ویژگی هایی که در گزینشگرهای قبل راجع به آن توضیح دادم صرف نظر کردم"

در ابتدا ویژگی های body را به صورت زیر اضافه می کنیم:

```
body{
  margin: 0px;
  background-color: #f7deb5;
```

```
font-size:13px;
font-family: tahoma;
}
```

margin: حاشیه بین **body** و کادر مرورگر را مشخص می کند که من برابر 0 قرار دادم و فضای خالی بین **body** و مرورگر از بین می رود.

background-color: رنگ پس زمینه صفحه را مشخص می کند که با کد رنگ، مقادیر **rgb** یا رنگ های از پیش تعریف شده همچون: **black, white, red, blue** قابل تنظیم می باشد که من در اینجا با کد رنگ **f7dbe5**، رنگ پس زمینه را تغییر دادم.

font-size: با استفاده از این ویژگی اندازه فونت صفحه را به **13px** تغییر دادم، واحد های دیگری همچون: **em, rem** قابل استفاده می باشد.

font-family: با استفاده از این ویژگی می توان نوع فونت را معین کرد، انواع فونت هایی همچون: **tahoma, arial** و ... البته در نسخه سوم و جدید **CSS** امکان پیوست فونت به صفحه فراهم شده است، همچنین می توان دو ویژگی بالا را به صورت خلاصه شده به صورت زیر نوشت:

```
font:13px tahoma;
```

قاعده خلاصه نویسی در **CSS** بسیار کارآمد و ساده است به جای اینکه خط های متمادی را برای قسمت های مختلف یک ویژگی اضافه کنیم می توانیم ویژگی را به صورت خلاصه شده در یک خط ویرایش کنیم.

ویژگی های کلاس **top:**

```
.top{
width: 100%;
height: 35px;
border-bottom: 2px solid #B6B6B6;
background-color: #D1D1D1;
}
```

: با استفاده از این ویژگی عرض عنصر دارای کلاس **top** را برابر **100%** در نظر گرفتیم، یکی دیگر از مقادیر اندازه در **CSS** درصد می باشد. عرض این عنصر با استفاده از این واحد، متناسب با مقدار وارد شده و عرض عنصر پدر، که در این جا **body** می باشد معین می شود در واقع با توجه به اینکه مقدار **100%** را برای این ویژگی در نظر گرفتیم تمامی عرض **body** را اشغال می کند، هنگامی که مرورگر تغییر اندازه می یابد متناسب با تغییر، عرض عنصر هم تغییر پیدا می کند.

height: ارتفاع عنصر که با واحد px یا پیکسل تنظیم شده است.

border-bottom: با استفاده از این ویژگی حاشیه پایین عنصر را معین کردم، مقادیر به ترتیب:

1. ضخامت حاشیه
2. نوع حاشیه (خط ممتد، خط غیر ممتد، نقطه چین)
3. رنگ حاشیه

در div با کلاس top، یک تصویر را با استفاده از تگ img به صورت زیر قرار دادم:

```
<div class="top">
  
</div>
```

حال برای دسترسی به عنصر img و تغییر ویژگی ها در CSS، از گزینشگر نام (نام تگ) به صورت زیر استفاده کردم:

```
.top img{
  float: left;
  margin: 5px 16px;
}
```

همانطور که ملاحظه می کنید به صورت سلسله مراتب قادر خواهید بود تا عناصر موجود در صفحه HTML را انتخاب کنید.

float: با استفاده از این ویژگی و مقداری که برای آن در نظر گرفتیم، عنصر img از سمت چپ در عنصر پدر شناور می شود (در سمت چپ عنصر با کلاس top قرار می گیرد)

margin: فضای خالی بین حاشیه بیرونی عنصر، و عنصر پدر را تعیین می کند. همانطور که در بالا ملاحظه می کنید ویژگی خلاصه نویسی شده است، مقدار اول، فضای بالا و پایین عنصر، و مقدار دوم، فضای چپ و راست عنصر را تعیین کرده است.

داخل تگ div با کلاس top، یک تگ ul به همراه دو زیر مجموعه که با تگ li مشخص می گردند به صورت زیر اضافه کردم:

```
<div class="top">
  
```

```

<ul>
  <li><a>درباره ما</a></li>
  <li><a>ورود</a></li>
</ul>
</div>

```

ul که مخفف کلمات unordered lists به معنای لیست های نا مرتب می باشد، یک لیست بدون ترتیب می باشد که در کنار آیتم های آن از علامت هایی همچون:

- یک دیسک (disc)
- دایره تو خالی (circle)
- مربع (square)
- حروف یونانی (upper-roman)

و ... استفاده کرد.

همچون تگ img می توان به صورت زیر به تگ ul دسترسی داشت:

```

.top ul{
  float:right;
  margin:8px 0px;
  list-style-type: none;
}

```

list-style-type با استفاده از این ویژگی می توان نوع لیست را با توجه به مقادیر ممکن تغییر داد.

```

.top ul li{
  float:right;
  margin-right: 30px;
}

```

همانطور که در بالا مشاهده می کنید به صورت سلسله مراتب به تگ های li در ul موجود، در کلاس tag دسترسی پیدا کرده و برخی ویژگی های آن را تغییر دادم، در اینجا قصد دارم که li های موجود به صورت افقی از سمت راست در کنار هم قرار بگیرند که با دو تکه کد پیشین این کار را انجام دادم.

تا اینجا تگ div با کلاس top و محتوای داخلش، با استفاده از CSS به شکل زیر در آمده است:



حال باید ویژگی های `div` با کلاس `content` را تغییر دهیم، کدهای CSS این کلاس به صورت زیر می باشد:

```
.content{
    display: table;
    width: 940px;
    height: auto;
    padding-top: 1px;
    margin: 50px auto;
}
```

`display`: با استفاده از این ویژگی تعیین می کنیم که عنصر مورد نظر به چه صورتی به نمایش گذاشته شود، در اینجا من مقدار `table` را به این ویژگی اختصاص دادم که ارتفاع عنصر مورد نظر به صورت `table` افزایش پیدا کند.

`height`: ارتفاع عنصر را مشخص می کند که من مقدار `auto` را به این ویژگی اختصاص دادم که همانطور که از معنی آن پیداست، تنظیم این مقدار برای این ویژگی باعث افزایش خودکار ارتفاع عنصر می شود.

حال به عنصر `content`، تگ های دیگری به صورت زیر اضافه می کنم:

```
<div class="content">
    <div class="search">
        <table>
            <tr><td><input type="submit" value="جست و
            <td><span>کالا نام</span><input type="radio"
            value="1"><span>قیمت</span><input type="radio"
            value="1"></td><td><input type="text" placeholder="جست و
            <td><span>ورزشی</span></td></tr>
        </table>
    </div>
    <div class="side">
        <span>دسته بندی</span>
        <ul>
            <li><a>ورزشی</a></li>
            <li><a>دیجیتال</a></li>
            <li><a>صنایع دستی</a></li>
        </ul>
    </div>
    <div class="ads">

</div>
</div>
```

تگ `div` با کلاس `search` را برای قرار دادن امکانات دریافت ورودی از کاربر اضافه کردم. در این تگ یک جدول با استفاده از تگ `table` ایجاد کردم و در داخل `table` با استفاده از تگ `tr` و `td` به ترتیب سطرها و ستون ها را اضافه کردم، هر `td` یک سلول در جدول در نظر گرفته می شود.

با استفاده از تگ `input` فیلدهای ورودی را برای دریافت اطلاعات مربوط به جست و جو در تگ `td` در جدول اضافه کردم.

`input type = text`: این تگ `input` یک کادر متنی را ایجاد می کند، یکی از صفت های کاربردی این نوع، `place holder` می باشد که یک متن را به عنوان `hint` یا اشاره ای کوتاه به آنچه که باید در کادر وارد شود به صورت پیش فرض و البته متمایز با متن ورودی (به صورت کم رنگ تر) در نظر می گیرد.

`input type = radio`: این تگ `input` یک دکمه رادیویی را ایجاد می کند.

`input type = submit`: این تگ `input` یک دکمه برای ارسال اطلاعات ورودی ها در فرم به سمت سرور استفاده می شود.

در زیر ویژگی های مربوط به `div` با کلاس `search` را تغییر دادم:

```
.search{
    width: 911px;
    height: 35px;
    margin: 14px 15px;
    border-radius: 2px;
    background-color: #326A95;
}
```

border-radius: از این ویژگی برای گرد کردن گوشه های تگ یا عنصر مورد نظر استفاده می شود، که در اینجا به اندازه `2px` چهار گوشه عنصر گرد شده است.

با اسفاده از کد `CSS` زیر موقعیت جدول را معین کردم:

```
.search table{
    float: right;
    margin: 2px 24px;
}
```


همچنین ویژگی های `margin-left` و `color` تمامی `span` های موجود در جدول را با استفاده از کد زیر ویرایش کردم:

```
.search table span{
    margin-left:15px;
    color:#FFF;
}
```

حال برخی از ویژگی های تگ های `input` را به صورت زیر تغییر می دهیم:

```
.search table input[type=text]{
    width: 400px;
    padding-right: 4px;
    direction: rtl;
}
.search table input[type=radio]{
    margin:0px 25px 0px 8px;
}
.search table input[type=submit]{
    margin-right: 25px;
}
```

با استفاده از `type` و مقدار مشخص شده برای آن می توان به انواع تگ های `input` دسترسی داشت.

direction: با استفاده از این ویژگی می توان جهت نوشتن را مشخص کرد که در اینجا برابر با `rtl` یا `right-to-left` قرار دادم که به معنی از راست به چپ می باشد.

پس از تگ `div` با کلاس `search`، تگ `div` با کلاس `side` قرار گرفته است، این تگ را برای نمایش دسته بندی های مختلف تبلیغات در سایت اضافه کردم، هر دسته بندی در یک تگ `li` در `ul` قرار گرفته است.

در زیر کدهای CSS قسمت `side` را مشاهده می کنید:

```
.side{
    float: right;
    width: 230px;
    height: auto;
    margin: 5px 14px 15px;
    border: 1px solid #BDBDBD;
    border-radius: 2px;
    background-color:#FEFEFE;
}
```

نکته ای که در کدهای بالا حائز اهمیت است، ویژگی **border** است که به صورت یکتا آمده است، بدین صورت حاشیه برای 4 جهت عنصر با مقادیر وارد شده تنظیم می شود.

یک تگ **span** با محتوای "دسته بندی" در قسمت **side** قرار دادم که عنوانی است برای قسمت **side**، در زیر کدهای **CSS** این تگ را مشاهده می کنید:

```
.side span{
    display: block;
    width: 70px;
    font-weight: bold;
    margin: 3px auto;
}
```

با استفاده از **display: block** نحوه نمایش تگ **span** را که به صورت **inline** می باشد را به **block** تغییر دادم، و یک عرض معین را برای این تگ تنظیم کردم و با اختصاص دادن مقدار **auto** به قسمت دوم مقدار ویژگی **margin** تگ **span** را به وسط عنصر پدر انتقال دادم.

در داخل تگ **side** یک تگ **ul** قرار دادم که کدهای **CSS** بدین صورت است:

```
.side ul{
    float: right;
    margin: 15px 15px;
    list-style-type: none;
}
.side ul li{
    margin-bottom: 25px;
    text-align: right;
}
```

text-align: تراز افقی محتوای تگ را معین می کند که در اینجا **right** قرار گرفته است و متن موجود در تگ **li** که محتوای این تگ می باشد، از سمت راست تراز شده است.

margin-bottom یا فضای خالی پایین تگ های **li** در **ul** موجود، همانطور که در کد بالا مشاهده می کنید برابر با 25px قرار گرفته است. این اندازه حاشیه پایین بین عناصر **li** را مشخص می کند اما در مورد تگ آخر، قصد دارم **margin-bottom** را تغییر بدهم، در **CSS** با استفاده از **last-child** به صورت زیر می توانم به فرزند آخر یک عنصر دسترسی پیدا کنم:

```
.side ul li:last-child{
    margin-bottom: 10px;
```

```
}
```

در اینجا آخرین li موجود در ul که در قسمت side قرار گرفته است انتخاب شده است و مقدار margin-bottom آن را تغییر دادم.

تگ div با کلاس ads در سمت چپ عنصر side قرار گرفته است و در آینده تبلیغات در این تگ div نمایش داده می شود، ویژگی های CSS آن به شرح زیر است:

```
.ads{
    float: left;
    width: 650px;
    height: 500px;
    margin: 5px 14px 15px;
    border: 1px solid #BDBDBD;
    border-radius: 2px;
    background-color: #FEFEFE;
}
```

همانطور که در بالا می بینید با استفاده از ویژگی float و مقدار left این تگ از سمت چپ تگ content شناور می شود.

در نهایت قسمت پایین سایت را با استفاده از یک تگ div و کلاس footer که به آن اختصاص دادم ایجاد کردم، در زیر کدهای HTML این تگ div و محتوایش را مشاهده می کنید:

```
<div class="footer">
    <span>سایت تبلیغات</span>
</div>
```

یک تگ span را با محتوای "سایت تبلیغات" به قسمت footer اضافه کردم.

در زیر برخی ویژگی های مربوط به این دو تگ را تغییر دادم:

```
.footer{
    clear: both;
    width: 100%;
    height: 35px;
    border-top: 2px solid #B6B6B6;
    background-color: #D1D1D1;
}
.footer span{
    display: block;
```

```
width: 77px;
margin: 9px auto 0px;
}
```

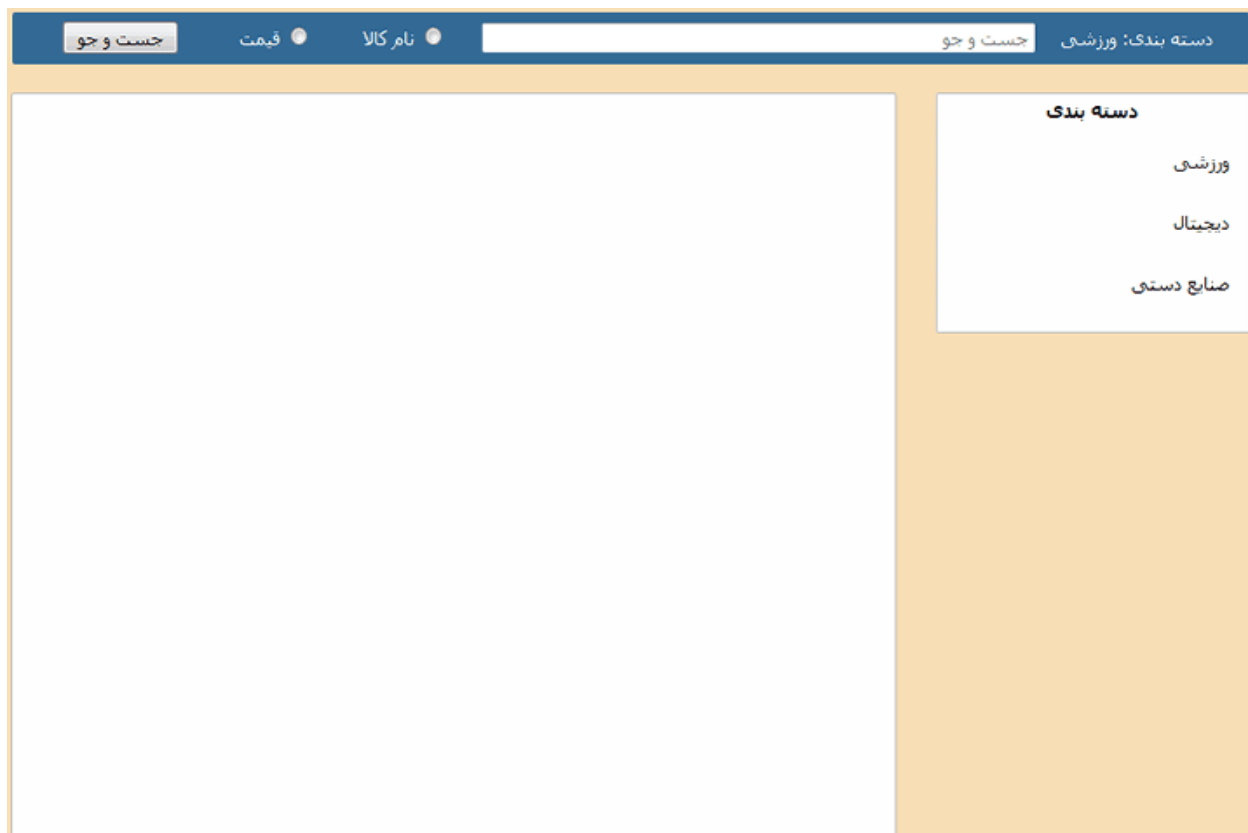
clear: این ویژگی باعث لغو ویژگی float می شود، در اینجا مقدار both را برای این ویژگی قرار دادیم که باعث لغو ویژگی float در هر دو طرف left و right می شود.

نکته ای که در کد CSS گزینشگر footer span حائز اهمیت است خلاصه نویسی ویژگی margin به صورت 3 تایی است، مقادیر به ترتیب:

1. margin-top
2. margin-left + margin-right
3. margin-bottom

با توجه به مقدار auto، تگ span در وسط footer قرار می گیرد.

در نهایت قسمت content به شکل زیر در می آید:



DiNL

SOFTWARE DEVELOPMENT