

ژنرال وب دیزاین جلسه نوزدهم



مرور جلسه قبل (ایجاد انیمیشن در دستورات CSS)

۱- معرفی دستور **animate**
دستور **animate** اجرای دستورات **CSS** را با نرمی خاصی اجرا می کند و در واقع بهشون افکت میده مشابه دستور **transition**

فرم کلی دستور **animate** در **jquery** به شکل زیر هست:

```
$(selector).animate({params}, speed, callback);
```

دستور بالا دارای سه پارامتر هست:
اولی **{param}**: ویژگیهایی که باید در مورد تگ باید تغییر بدیم مثلا عرض یا ارتفاع یا هردو
دومی **speed**: سرعت اجرای تغییرات بر حسب میلی ثانیه
سومی **callback**: تابعی که بعد از اجرای **animate** باید اجرا بشه



مرور جلسه قبل (مثال از animate

۲ - یک پروژه جدا ایجاد می کنیم و jquery را به آن اضافه می کنیم:

```
...
<html>
  <head>
    ...
    <script src="js/jquery1.min.js"></script>
  </head>
  <body>
    <style>
      #example{
        width:200px;
        Height:200px;
        background: blue;
      }
    </style>

    <div id="example"
    </div>
    <button>animate</button>
    <script>
      $( 'button').click (function{
        $('#example').animate ({width:300px},2000);
      }
    </script>
  </body>
</html>
```

میخواهیم با دستورات jquery
عرض باکس را از ۲۰۰ پیکسل به ۳۰۰
پیکسل تغییر بدیم به صورت
انیمیشنی



مرور جلسه قبل (مثال از animate

۳ - یک پروژه جدا ایجاد می کنیم و **jquery** را به آن اضافه می کنیم:

```
...
<html>
  <head>
    ...
    <script src="js/jquery1.min.js"></script>
  </head>
  <body>
    <style>
      #example{
        width:200px;
        Height:200px;
        background: blue;
      }
    </style>

    <div id="example"
    </div>
    <button>animate</button>
    <script>
      $( 'button').click (function{
        // $('#example').animate ({'marginLeft:200px'}, 2000);
        // $('#example').animate ({'marginTop:200px'}, 2000);
        $('#example').animate ({'marginLeft:200px'}, 2000,
          function(){
            alert('end');
          }
        )
      }
    )
  }
</script>
</body>
</html>
```

نکته : بجای کلمه **callback** باید تابع مدنظر خودمون را بنویسیم

میخواهیم با دستورات **jquery** حاشیه خارجی چپ باکس را اضافه کنیم

یا **margin-top** را تغییر دهیم

نکته : در **jquery** خط وسط را برمیداریم و حرف اول **top** یا **left** را بدون خط وسط و به صورت حرف بزرگ مینویسیم:
marginLeft
marginTop

نکته : تابع سه بار نوشته شده هر بار باید یکی اجرا شود



```
...  
<html>  
  <head>  
    ...  
    <script src="js/jquery1.min.js"></script>  
  </head>  
  <body>  
    <style>  
      #example{  
        width:200px;  
        Height:200px;  
        background: blue;  
      }  
    </style>  
  
    <div id="example"  
    </div>  
    <button>animate</button>  
    <script>  
      $( 'button').click (function{  
        $('#example').animate ({'height:200px'}, 2000,  
          function(){  
            $('#example').animate ({'width:200px'}, 2000)  
          } )  
      }  
    </script>  
  </body>  
</html>
```

میتوانیم هر تعداد تابع و دستور در **callback** بنویسیم که به ترتیب اجرا بشه و **animation** درست کنیم یا در قسمت **callback** دوباره از دستور **animate** استفاده کنیم و بترتیب تغییرات به صورت انیمیشن صورت پذیرد

دوباره میتونیم برای **callback** **animate** دوم هم یکم تابع دیگر و همینطور تودرتو ادامه دهیم



مرور جلسه قبل (مقایسه دستور animate و CSS

۵- با CSS هم میتوانیم مقدار یک ویژگی CSS را **jquery** تغییر بدیم اما مانند **animate** با زمان و به تدریج نیست

```
...
<html>
  <head>
    ...
    <script src="js/jquery1.min.js"></script>
  </head>
  <body>
    <style>
      #example{
        width:200px;
        Height:200px;
        background: blue;
      }
    </style>

    <div id="example"
    </div>
    <button>animate</button>
    <script>
      $( 'button').click (function{
        $('#example').css ('marginLeft','200px');
      }
    </script>
  </body>
</html>
```



```
...
<html>
  <head>
    ...
    <script src="js/jquery1.min.js"></script>
  </head>
  <body>
    <style>
      #example{
        width:200px;
        Height:200px;
        background: blue;
        margin-left:200px;
      }
    </style>
    <div id="example"
    </div>
    <button>animate</button>
    <script>
      $( 'button').click (function{
        Var m= $('#example').css
('marginLeft');
        // alert(m);
        alert(parseFloat(m));
      }
    </script>
  </body>
```

من مقدار **margin-left** یک **div** را
دادم میخوام با دستور **CSS** مقدار آن
را بدست بیارم

ابتدا **px200** نمایش داده می شود
بعدهش مقدار **200** نمایش داده می شود
یعنی **parseFloat** فقط قسمت عددی
را استفاده می کند



ادامه پروژه (دستورات jquery مربوط به sliderscroll)

۷- وارد قسمت **script** پروژه شده و دستورات **jquery** زیر را می نویسیم

```
<script>
```

```
Var sliderscroll_tag= $('sliderscroll'); // ذخیره کردن تگ اسلایدراسکرول داخل یک متغیر برای راحتی کار  
Var sliderscroll_UL= sliderscroll_tag.find( '.sliderscroll_main ul'); // ذخیره کردن یوال  
Var sliderscroll_Item= sliderscroll_ul.find( 'li'); // ذخیره کردن ال آی ها در متغیر
```

```
function sliderscroll (direction) // تابع برای حرکت اسلایدر در جهت داده شده با پارامتر  
{  
    var marginrightnew;  
    var marginrightold= sliderscroll_UL.css ('margin-left') // گرفتن مقدار فعلی حاشیه چپ  
    marginrightold=parsfloat ( marginrightold); // برداشتن px  
  
    if (direction=='left'){ // اگر اسکرول به سمت چپ باشد  
        marginrightnew= marginrightold-1100  
    }  
    if (direction=='right'){ // اگر اسکرول به سمت راست باشد  
        marginrightnew= marginrightold+1100  
    }  
    sliderscroll_UL.animate ({'marginRight':marginRightnew},2000)  
}  
$ ( '. Fliptimer').fliptimer ({  
    direction:'down';  
});  
</script>
```

ادامه پروژه (دستورات jquery مربوط به sliderscroll)

۸- کدها کامل شد تابع کلیک روی دکمه چپ و راست را مینویسیم و خروجی را چک می کنیم

```
<script>
```

```
Var sliderscroll_tag= $('sliderscroll');  
Var sliderscroll_UL= sliderscroll_tag.find( '.sliderscroll_main ul');  
Var sliderscroll_Item= sliderscroll_ul.find( 'li');  
  
function sliderscroll (direction) // تابع برای حرکت اسلایدر در جهت داده شده با پارامتر  
{  
    var marginrightnew;  
    var marginrightold= sliderscroll_UL.css ('margin-left') // گرفتن مقدار فعلی حاشیه چپ  
    marginrightold=parsfloat ( marginrightold);  
  
    if (direction=='left'){ // اگر اسکرول به سمت چپ باشد  
        marginrightnew= marginrightold-800  
    }  
    if (direction=='right'){ // اگر اسکرول به سمت راست باشد  
        marginrightnew= marginrightold+800  
    }  
    sliderscroll_UL.animate ({'marginRight':marginRightnew},3000)  
}
```

```
$('#next').click(function(){  
    sliderscroll ('left')  
});  
$('#prev').click(function(){  
    sliderscroll ('right')  
});
```

```
$ ('. Fliptimer').fliptimer ({  
    direction:'down';
```

```
</script>
```

خروجی میگیریم همه چی درست !!
یکبار چپ بزنیم میره به سمت چپ
یکبار راست میزنیم میاد به سمت راست
حالا اگر دو یا چند بار سمت چپ یا راست
را بزنیم چی !!!

نکته : ممکن هست برخی از دستورات
مانند transform در برخی
مرورگرها درست کار نکنند که برای
اطلاع میتونید سرچ کنید



ادامه پروژه (دستورات jquery مربوط به sliderscroll)

۹- باید تعداد li ها و تعداد نمایش آنها در یک اسلاید را داشته باشیم و با تقسیم آنها بر یکدیگر تعداد اسلاید را بدست بیاوریم و اگر اعشاری شد به سمت بالا گرد کنیم با این روش تعداد اسلایدها بدست میاد که در پروژه ما باید تقسیم بر ۴ کنیم

```
<script>
```

```
Var sliderscroll_tag= $('sliderscroll');
Var sliderscroll_UL= sliderscroll_tag.find( '.sliderscroll_main ul');
Var sliderscroll_Item= sliderscroll_ul.find( 'li');
var sliderscroll_item_numbers= sliderscroll_Item.length;// (ها li) شمردن تعداد آیتمها
var sliderscroll_numbers= math.ceil ( sliderscroll_item_numbers/5) // شمردن تعداد اسلایدها
var maxnegativemargin= - (: sliderscroll_numbers-1)*1100) // حداکثر میزان پیکسل منفی مارجین
function sliderscroll (direction) // تابع برای حرکت اسلایدر در جهت داده شده با پارامتر
{
    var marginrightnew;
    var marginrightold= sliderscroll_UL.css ('margin-left') // گرفتن مقدار فعلی حاشیه چپ
    marginrightold=parsfloat ( marginrightold);

    if (direction=='left'){ // اگر اسکرول به سمت چپ باشد
        marginrightnew= marginrightold-800
    }
    if (direction=='right'){ // اگر اسکرول به سمت راست باشد
        marginrightnew= marginrightold+800
    }
    sliderscroll_UL.animate ({'marginRight':marginRightnew},2000)
}
$ ( '. Fliptimer').fliptimer ({
    direction:'down';
</script>
```

نکته : تابع **math.ceil**
تابعی که عدد را به سمت بالا گرد می کند



ادامه پروژه (دستورات jquery مربوط به sliderscroll)

۱۰ - نکته ای که وجود داره اینه که عرض ul با توجه به تعداد li ها ممکن هست تغییر کنه و ثابت نباشه بنابراین ما باید با دستورات jquery مقدار دینامیک برای آن در نظر بگیریم بنابراین در ul مقدار width را بر میداریم و دستور زیر را اضافه می کنیم (تعداد li ضربدر عرض هریک از آنها) البته متغیر تعداد li را هم داریم: sliderscroll_item_numbers

```
<script>
```

```
Var sliderscroll_tag= $('sliderscroll');
Var sliderscroll_UL= sliderscroll_tag.find( '.sliderscroll_main ul');
Var sliderscroll_Item= sliderscroll_ul.find( 'li');
var sliderscroll_item_numbers= sliderscroll_Item.length; // شمردن تعداد آیتمها (liها)
var sliderscroll_numbers= math.ceil ( sliderscroll_item_numbers/4) // شمردن تعداد اسلایدها
var maxnegativemargin= - ( : sliderscroll_numbers-1)*800) // حداکثر میزان پیکسل منفی مارچین
sliderscroll_UL.css ('width', sliderscroll_item_numbers*220) //10
function sliderscroll (direction) // تابع برای حرکت اسلایدر در جهت داده شده با پارامتر
{
    var marginrightnew;
    var marginrightold= sliderscroll_UL.css ('margin-left') // گرفتن مقدار فعلی حاشیه چپ
    marginrightold=parsfloat ( marginrightold);

    if (direction=='left'){ // اگر اسکرول به سمت چپ باشد
        marginrightnew= marginrightold-800
    }
    if (direction=='right'){ // اگر اسکرول به سمت راست باشد
        marginrightnew= marginrightold+800
    }
    sliderscroll_UL.animate ({'marginRight':marginRightnew},2000)
}
$ ('. Fliptimer').fliptimer ({
    direction:'down';
});
</script>
```

```
.sliderScroll-main ul { //10
    padding:0
    Height:100%
    Width: 1100px; // این عرض حذف می شود
}
```

ادامه پروژه (دستورات jquery مربوط به sliderscroll)

- ۱۱- شروط لازم برای چک کردن مقادیر خارج از محدوده را مینویسیم
- ۱۲ - علامت پوئنتر را به چپ و راست اضافه می کنیم

<script>

```
Var sliderscroll_tag= $('sliderscroll');  
Var sliderscroll_UL= sliderscroll_tag.find( '.sliderscroll_main ul');  
Var sliderscroll_Item= sliderscroll_ul.find( 'li');  
var sliderscroll_item_numbers= sliderscroll_Item.length; // (ها) شمردن تعداد آیتمها  
var sliderscroll_numbers= math.ceil ( sliderscroll_item_numbers/4) // شمردن تعداد اسلایدها  
var maxnegativemargin=- ( : sliderscroll_numbers-1)*1100) // حداکثر میزان پیکسل
```

منفی مارجین

```
: function sliderscroll (direction) // تابع برای حرکت اسلایدر در جهت داده شده با پارامتر  
{  
  var marginrightnew;  
  var marginrightold= sliderscroll_UL.css ('margin-left') // گرفتن مقدار فعلی حاشیه چپ  
  marginrightold=parsfloat ( marginrightold);  
  
  if (direction=='left'){ // اگر اسکرول به سمت چپ باشد  
    marginrightnew= marginrightold-800  
  }  
  if (direction=='right'){ // اگر اسکرول به سمت راست باشد  
    marginrightnew= marginrightold+800  
  }  
  
  if(marginrightnew<maxnegativemargin)//11  
  {  
    marginrightnew=0;  
  }  
  
  if(marginrightnew>0)//11  
  {  
    marginrightnew=maxnegativemargin;  
  }  
  
  sliderscroll_UL.animate (/(marginRight:marginRightnew) 2000)
```

خروجی میگیریم کل اسلاید را تست میکنیم و مقدار عرض را میبینیم

```
.sliderScroll-prev .prev{ //12  
  ...  
  Cursor:pointer;  
}  
.sliderScroll-next .next{ //12  
  ...  
  Cursor:pointer;  
}
```

پایان



پایان جلسه
موفق باشید

