

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

ایجاد animation های سفارشی با استفاده از متد animate()

مدرس : مهندس افشین رفوآ

دوره آموزشی JQuery

ایجاد animation های سفارشی با استفاده از متد animate()

در مباحث پیشین، ایفکت های نمایش اسلایدی و پنهان سازی تدریجی را که امکانات درون ساخته ی این کتابخانه ی جاوا اسکریپت هستند ، تشریح کردیم. به کمک متد animate()، می توان animation های سفارشی ایجاد نمود. به طوری که تمامی خواص عددی CSS یک المان توسط این متد قابل مدیریت و دستکاری می باشد. با این متد می توان به طور مثال، یک کادر را به آرامی بر روی صفحه به حرکت درآورد یا به آن قابلیت پرش به بالا و پایین اعمال نمود.

```
<div style="height: 60px;">
  <div id="divTestBox1" style="height: 50px; width: 50px; background-color: #89BC38; position: absolute;"></div>
</div>
<script type="text/javascript">
  $(function () {
    $("#divTestBox1").animate(
      {
        "left": "200px"
      }
    );
  });
</script>
```

اولین و تنها پارامتر الزامی متد animate()، مجموعه خواص CSS است که مایلید تغییر دهید. در این مثال، یک المان div با موقعیت absolute داریم که با کمک متد animate() جی کوئری به آن دستور می دهیم زمانی که مقدار خاصیت left سی اس اس آن به 200 پیکسل رسید، آن را متوقف کند.

دومین پارامتر به شما این امکان را می دهد تا سرعت اتمام انیمیشن و متحرک سازی را بر حسب میلی ثانیه یا با پاس دادن مقادیر "slow" یا "fast" (به ترتیب معادل 600 یا 200 میلی ثانیه) به متد یاد شده، تنظیم نمایید. با این دستور می توان سرعت حرکت کادر سبز رنگ بر روی صفحه را کاهش دهیم.

```
<div style="height: 60px;">
  <div id="divTestBox2" style="height: 50px; width: 50px; background-color: #89BC38; position: absolute;"></div>
</div>
<script type="text/javascript">
  $(function () {
    $("#divTestBox2").animate(
      {
        "left": "200px"
      },
      5000
    );
  });
</script>
```

سومین پارامتر که به متد `animate()` ارسال می شود یک تابع `callback` است که به محض اتمام انیمیشن صدا زده می شود. این امکان در اجرای یک سری انیمیشن مختلف به صورت پشت سرهم بسیار مفید واقع می شود. مثال:

```
<div style="height: 40px;">
  <div id="divTestBox3" style="height: 20px; width: 20px; background-color: #89BC38; position: absolute;"></div>
</div>
<script type="text/javascript">
  $(function () {
    $("#divTestBox3").animate(
      { "left": "100px" },
      1000,
      function () {
        $(this).animate(
          { "left": "20px" },
          500,
          function () {
            $(this).animate({ "left": "50px" }, 500);
          }
        )
      }
    );
  });
</script>
```

در این نمونه، متد **animate** را فراخوانی کرده و خاصیت **left** المان دارای شناسه ی **divTestBox3** را به عنوان پارامتر به آن پاس داده ایم. سپس از آن خواستیم تا زمانی که مقدار این خاصیت به **100** پیکسل نرسیده، انیمیشن را ادامه دهد. همچنین سرعت اتمام انیمیشن را **1** ثانیه (**1000** میلی ثانیه) مشخص کرده ایم که با به پایان رسیدن آن بلافاصله انیمیشن دیگری آغاز می گردد. انیمیشن دوم، همان طور که از کد مشخص است، المان مزبور را به اندازه ی **20** پیکسل و در طی نیم ثانیه به عقب برمی گرداند. بار دیگر با خاتمه یافتن انیمیشن دوم، انیمیشن دیگری شروع می شود که در آن المان به سمت راست رانده شده تا مقدار خاصیت **left** بر روی **50** تنظیم گردد.

با این حال، از آنجایی که **jQuery** دارای امکان صف بندی برای اجرای **animation** ها می باشد، می توانید مثال بالا را بسیار ساده تر پیاده سازی کنید. ناگفته نماند که این امکان بیشتر زمانی بکار می آید که بخواهید چندین انیمیشن به صورت پشت سرهم اجرا نمایید. اگر بخواهید زمانی که **animation** به پایان رسید، اتفاق دیگری رخ دهد، مثال ذکر شده در بالا باز روش مناسب می باشد. نسخه ی صف بندی شده ی انیمیشن بالا را در مثال زیر مشاهده می کنید (انیمیشن ها پشت سرهم اجرا می شوند):

```
<div style="height: 40px;">
  <div id="divTestBox4" style="height: 20px; width: 20px; background-color: #89BC38; position: absolute;"></div>
</div>
<script type="text/javascript">
  $(function () {
    $("#divTestBox4").animate({ "left": "100px" }, 1000);
    $("#divTestBox4").animate({ "left": "20px" }, 500);
    $("#divTestBox4").animate({ "left": "50px" }, 500);
  });
</script>
```