

ژنرال وب دیزاین جلسه چهاردم



یادآوری تابع

برای ادامه کار باید توابع در `jquery` را اول بگم بعد بریم توی پروژه اجراش کنیم

تابع چیست؟

تکه کدی هست که ما مینویسیم و مثل یک متد میتونیم بارها ازش استفاده کنیم

نحوه نوشتن تابع

```
function name (){  
    کدهای لازم  
}
```

مثال

```
function al (){  
    Alert("سلام");  
}
```



یادآوری تابع arrow function

Arrow function چیست؟

همانند تابع هست فقط نحوه تعریف آن بصورت زیر و اخیرا بیشتر ازین روش استفاده میشه

نحوه نوشتن تابع
name = () => {
 کدهای لازم
}

مثال

```
AI = () => {  
    Alert("سلام");  
}
```



تابع eq

تابع eq

این تابع برای اینکه بررسی کند یک مقدار با مقدار دیگر مساوی است یا خیر؟

مثال تابع eq

میخواهیم بطور مثال تگ دوم از کد زیر را انتخاب کنیم و پنهان کنیم:

```
<div id="div-test">  
<p>سلام</p>  
<p>من علی هستم</p>  
<p>چخبر</p>  
</div>
```

```
<style>  
var ptags = $("#div-test p");  
Ptag.eq(1).hide();  
</style>
```



خصوصیت length

خصوصیت length

این خصوصیت برای نمایش طول یک رشته و یا تعداد موجود از یک سلکتور به کار می رود

مثال length

میخواهیم بطور مثال تعداد تگ `p` را شمارش کنیم:

```
<div id="div-test">  
<p>سلام</p>  
<p>من علی هستم</p>  
<p>چخبر</p>  
</div>
```

```
<style>  
  var ptags = $("#div-test p");  
  alert(ptags.length);  
</style>
```



تابع find

پیدا کردن فرزند یک تگ
از تابع find استفاده می کنیم

مثال تابع find

بطور مثال میخواهیم فرزند div- test را پیدا کنیم و سپس مخفی کنیم :

```
<div id="div-test">  
<p>سلام</p>  
<p>من علی هستم</p>  
<p>چخبر</p>  
<div id="child">  
</div>  
</div>
```

```
<style>  
var tag1 = $("#div-test ");  
var ch=tag1.find('#child').html();  
alert(ch);  
var ch1=tag1.find('#child');  
ch1.hide();  
</style>
```



ایجاد اسلایدر

اگر به اسلایدر سایت نگاه کنیم میبینیم که ۵ تا عکس داریم که به ترتیب نشان داده میشن و اگر ما روی عنوان هر کدام نیز کلیک کنیم بقیه مخفی میشن و یکی نمایش داده میشه و با زدن فلش به چپ و راست نیز همین اتفاق میفته

ما ابتدا ۵ تا عکس اسلایدر را پیدا می کنیم و در پوشه عکس های پروژه کپی می کنیم سپس یک عکس که قبلا اضافه کرده بودیم ۴ تا تصویر دیگر را هم داخل slider-img اضافه کنیم:

```
<div id="slider-img"> //3
  <a>

</a>
  <a>

</a>
  <a>

</a>
  <a>

</a>
  <a>

</a>
</div>
```

اگر خوب دقت کنیم می بینیم که این تصاویر الان همه زیر هم در اسلایدر قرار دارند که اگر overflow را برداریم همه را میبینیم



ایجاد اسلایدر

حالا باید با جی کوئری به ترتیب این تصاویر را نشان دهیم
۱- اول همه را مخفی می کنیم

```
<script>//3
function slider(){
}

$('button').click(function(){
    slider();
});
</script>
```

```
1- Slider-img a{
    Display:none;
}
```

۲- به طور موقت یک دکمه برای نمایش اسلایدر میگذاریم

```
<div id="left-content"> //۲
</div>
<button>نمایش اسلایدر</button>
```

۳- حالا در قسمت script دستورات JQuery را
مینویسیم



ایجاد اسلایدر

4. به همه تگ های a اسلایدر یک کلاس می‌دهیم

5. حالا تابع اسلایدر را به مرور تکمیل می‌کنیم

سه متغیر برای پیمایش اسلایدر بصورت زیر ایجاد می‌کنیم

یکی نام خود اسلایدر هست یکی آیتم های اسلایدر هستند و دیگری برای رفتن به اسلاید بعدی است

```
<script>//5
var sliderTag = $('#slider');
var sliderItems =
sliderTag.find('.item');
Var nextSlide = 1;
    function slider(){
    }

    $('#button').click(
        slider());
};
</script>
```

```
<div id="slider-img"> //4
  <a class='item'>

</a>
  <a class='item'>

</a>
  <a class='item'>

</a>
  <a class='item'>

</a>
  <a class='item'>

</a>
</div>
```

حرکت در اسلایدر

۶. در تابع اسلایدر یک دستور برای رفتن به اسلاید بعدی بصورت زیر مینویسیم
۷. همه آیتم ها را مخفی میکنیم
۸. پس از نمایش یک اسلاید متغیر را روی اسلاید بعدی تنظیم میکنیم

```
<script>
```

```
var sliderTag = $('slider');  
var sliderItems = sliderTag.find('.item');  
var nextSlide = 1;
```

```
function slider(){  
    sliderItem.hide(); //7  
    sliderItems.eq(nextSlide-1).fadeIn(100); //6  
    nextSlide++; //8  
}
```

```
$('#button').click(  
    slider());  
);
```

```
</script>
```

nextSlide -1 یعنی شماره اولین اسلاید صفر است و دومین اسلاید یک است و همینطور تا آخر چون مقدار **nextSlide** در بیرون تابع یک قرار دادیم



استایل فلش های اسلایدر

۹- دو علامت فلش قبل و بعد را روی اسلایدر در چپ و راست اضافه میکنیم

```
<div id='slider'  
style='position:relative'>  
  <span id='prev'>  
</span>  
  
<span id='next'>  
</span>  
...  
</div>
```

```
#prev{  
  Width: 20px;  
  Height: 35px;  
  Display: block;  
  Position: absolute;  
  Top: 200px;  
  Right: 5px;  
  Background: url(images/arrow.jpg) no-repeat
```

```
#next{  
  مثل بالاست فقط  
  Left: 5px  
  }  
  Right حذف میشه
```

یک اجرا میگیریم اگر نواقصی در جای فلش ها بود با تغییر
top و right تنظیم میکنیم



معرفی خصوصیت background-position

۱۰ - background-position

از این خاصیت برای قرار دادن دقیق یک عکس استفاده میشود و دارای دو آیتم که اولی افقی و دومی عمودی می باشد که جلوتر در این رابطه مثال خواهیم زد و در پروژه استفاده می شود

Background-position: 10 -20

عدد اول می تواند حرکت در راستای افقی باشد مثبت یعنی پس زمینه به سمت راست و منفی یعنی به سمت چپ حرکت کند
عدد دوم برای حرکت در راستای عمودی ، منفی یعنی پایین و مثبت یعنی بالا می باشد



توابع اسلایدر

۱۱- افزودن تابع next یا بعدی

با کلیک روی فلش های راست و چپ باید یک بار تابع slider اجرا بشه به همین خاطر کد ها را بصورت زیر مینویسیم

```
< script >
var sliderTag = $('slider');
var sliderItems = sliderTag.find('.item');
var nextSlide = 1;
var slideCount = sliderItems.length;

function next(){
    slider();
}
$('#slider #next).click(function(){
    next();
});
</ script >
```

یک اجرا میگیریم و خروجی را میبینیم

۱۲- با اجرای اسلایدر و هر بار کلیک روی فلش به عکس بعدی میرویم. اگر اجرا کنیم میبینیم که بعد از عکس آخر صفحه سفید ظاهر شده یا ارور میدهد. چرا؟ چون شماره اسلاید بعد آخرین اسلاید به شماره عکسی میرسد که اون عکس موجود نیست پس ما باید بعد از اینکه عکس ها تموم شد دستور بدیم که اولین عکس دیده بشه. پس نیاز داریم که تعداد آیتم ها را شمارش کنیم که با length این کار را انجام میدیم



توابع اسلایدر

۱۳- باید شرطی بنویسیم که اگر شماره اسلاید از تعداد اسلایدرها بیشتر شد به یک برگردد یعنی

$$1 \leq \text{nextSlide} \leq \text{sliderCount}$$

پس در تابع اسلایدر شرطی زیر را مینویسیم

```
< script >  
...  
function slider(){  
    if(nextSlide > sliderCount){  
        nextSlide = 1;  
    }  
}  
</ script >
```

یک اجرا میگیریم و خروجی را میبینیم



توابع اسلایدر

۱۴- افزودن تابع `prev` یا قبلی

با کلیک روی فلش های راست و چپ باید یک بار تابع `slider` اجرا بشه به همین خاطر کد ها را بصورت زیر مینویسیم

```
< script >  
....  
function prev(){  
    nextSlide = nextSlide - 2;  
    slider();  
}  
$('#slider #prev).click(function(){  
    prev();  
});  
</ script >
```

بازم یک تابع برای کلیک روی فلش قبلی مینویسیم اما نکاتی که باید در آن رعایت کنیم این است که اگر شماره اسلاید ما خارج از محدوده اسلاید ها شد باید مقدار آن آخرین اسلاید شود

به دلیل اینکه شماره اولیه اسلاید را یک گذاشتیم، باید متغیر `nextSlide` را دوتا کم کنیم. البته میشد مقدار اولیه `nextSlide` را صفر قرار بدیم اما کدها یک مقدار تغییر میکرد

یک اجرا میگیریم و خروجی را میبینیم



توابع اسلایدر

۱۵- در تابع اسلایدر باید شرطی قرار بدیم که اگر شماره اسلاید صفر شد مقدار آن به تعداد آیتم تغییر کند یعنی یک حلقه درست میکنیم و همانطور که در اسلایدهای قبلی گفتیم کنترل میکنیم که با زدن دکمه قبلی و بعدی از محدوده خارج نشیم یعنی همیشه `nextSlide` باید بین یک تا تعداد اسلاید باشد

```
< script >  
....  
function slider(){  
...  
    if(nextSlide < 1){  
        nextSlide = slideCount;  
    }  
}  
</ script >
```

۱۶- شکل پوینتر موس را عوض میکنیم در روی فلش های بعد و قبل خاصیت `z-index` برای این است که اولویت نمایش بین چند المنت را مشخص کنیم مثلاً با دستور زیر ما دکمه های بعد و قبل را روی تصویر می اوریم و دیگر علامت پوینتر از روی فلش ها با جابجایی عکس حذف نمیشود

```
<style>  
#next , #prev{  
    cursor: pointer;  
    Z-index = 2;  
}  
</style>
```

یک اجرا میگیریم و خروجی را میبینیم



دستور index

۱۷- این دستور شماره آیتم یا سلکتور مورد نظر را برمیگرداند

مثال : می خواهیم با کلیک روی هر تگ شماره آن نمایش داده شود این شماره عددی را نشان میدهد که مشخص می کند چندمین تگ از ابتدای تگ والد می باشد (تمرین ۱۴-۱)
در تمرین زیر تگ والد **test** می باشد

```
<div id="test">  
<p>۱ آیتم</p>  
<p>۲ آیتم</p>  
<p>۳ آیتم</p>  
<span>سلام</span>  
<span>خوبی</span>  
</div>
```

نکته: تفاوت **eq** با **index**

با استفاده از **eq** شماره سلکتور را داریم و میخواییم به یکی از تگ های فرزند بر اساس شماره تگ دسترسی داشته باشیم برای مثال مخفیش کنیم.
اما با **index** میتوانیم شماره فرزندهای یک تگ را پیدا کنیم

یک اجرا میگیریم و خروجی را میبینیم

```
< script >  
$( '#test p' ).click( function() {  
  Var numtag=$(this).index();  
  Alert(numtag);  
} )  
</ script >
```



Event attribute

۱۸- خواصی در تگ ها وجود دارد که با استفاده از آن ها میتوان بر اساس اینکه چه رویدادی رخ داده است تابعی را فراخوانی کرد به این خواص **event attribute** میگویند
مثال: میخواهیم با کلیک روی یک تگ **p** یک پیام روی صفحه نمایش داده شود همچنین با کلیک روی تگ **p** بعدی یک تابع اجرا شود

```
<div id="test">  
<p onclick="alert('سلام علی کجایی')">۱ آیتم</p>  
<p onclick="answer()">۲ آیتم</p>  
</div>
```

```
< script >  
Function answer()  
{  
    alert("سلام من تهرانم")  
}  
</ script >
```

یک اجرا میگیریم و خروجی را میبینیم



معرفی رویدادهای windows در html

نام رویداد	html5	توضیح
onafterprint		بعد از عمل پرینت رخ میدهد
onbeforeprint		قبل از عمل پرینت رخ میدهد
onbeforeunload		قبل از لود شدن یک محتوا درون یک صفحه رخ میدهد
onerror		وقتی یک خطا از سمت سرور ایجاد شود این رویداد رخ میدهد
onhashchange		وقتی قسمت لنگر یا anchor از یک URL تغییر کند این رویداد رخ میدهد
onload	-	وقتی لود صفحه به پایان رسید این رویداد رخ میدهد
onmessage		وقتی یک پیغام از سمت سرور ایجاد شود این رویداد رخ میدهد
onoffline		وقتی مرورگر در حالت آفلاین باشد این رویداد رخ میدهد
ononline		وقتی مرورگر در حالت آنلاین باشد این رویداد رخ میدهد
onpagehide		وقتی کاربر از صفحه خارج/دور شود این رویداد رخ میدهد
onpageshow		وقتی یک صفحه به کاربر نمایش داده شود این رویداد رخ میدهد
onpopstate		وقتی در پنجره تاریخچه ویندوز تغییری ایجاد شود این رویداد رخ میدهد
onresize		وقتی اندازه پنجره مرورگر تغییر کند این رویداد رخ میدهد
onstorage		وقتی یک منطقه ذخیره سازی وب به روز شود این رویداد رخ میدهد
onunload	-	وقتی یک صفحه توسط کاربر بسته شود این رویداد رخ میدهد



معرفی رویدادهای فرمها در html

نام رویداد	html5	توضیح
onblur	-	وقتی فوکوس ماوس از روی عنصر ورودی برداشته شود این رویداد رخ میدهد
onchange	-	وقتی مقدار یک عنصر تغییر کند این رویداد رخ میدهد
oncontextmenu	-	وقتی با کلیک راست در یک محدوده یک منو نمایش داده شود رخ میدهد
onfocus	-	وقتی فوکوس ماوس بروی یک عنصر قرار بگیرد این رویداد رخ میدهد
oninput	-	وقتی مقدار یک عنصر توسط کاربر تغییر کند این رویداد رخ میدهد
oninvalid	-	زمانی رخ میدهد که اعتبارسنجی در فرم با خطا همراه باشد
onreset	-	زمانی که دکمه ریست فرم توسط کاربر فشار داده شود رخ میدهد
onsearch	-	با وارد کردن یک مقدار درون عنصر جستجو (>) "input=search" میتواند رخ دهد
onselect	-	وقتی متن توسط کاربر با ماوس انتخاب شود میتواند رخ دهد



معرفی رویدادهای صفحه کلید در html

نام رویداد	html5	توضیح
onkeydown	-	وقتی یک دکمه از صفحه کلید فشار داده شود این رویداد رخ میدهد
onkeypress	-	وقتی یک دکمه از صفحه کلید فشار داده شود این رویداد رخ میدهد
onkeyup	-	وقتی یک دکمه از صفحه کلید رها شود این رویداد رخ میدهد



معرفی رویدادهای ماوس در html

نام رویداد	html5	توضیح
onclick	-	وقتی کلیک چپ ماوس فشار داده شود این رویداد رخ میدهد
ondblclick	-	وقتی بروی یک عنصر دابل کلیک (دوبار کلیک سریع) شود رخ میدهد
onmousedown	-	وقتی کلیک ماوس فشار داده شود این رویداد رخ میدهد
onmousemove	-	وقتی ماوس بروی یک عنصر قرار بگیرد این رویداد رخ میدهد
onmouseout	-	وقتی ماوس از روی یک عنصر خارج شود این رویداد رخ میدهد
onmouseover	-	وقتی ماوس بروی یک عنصر قرار بگیرد این رویداد رخ میدهد
onmouseup	-	وقتی کلیک ماوس رها شود این رویداد رخ میدهد
onmousewheel	-	این رویداد بکل منسوخ شده و باید از onwheel استفاده نمایید
onwheel	-	وقتی دکمه وسط ماوس بسمت بالا یا پایین اسکرول شود رخ میدهد



معرفی رویدادهای drag & drop (کشیدن و انداختن) در html

نام رویداد	html5	توضیح
ondrag		وقتی یک عنصر با ماوس کشیده شود رخ میدهد
ondragend		وقتی عمل کشیدن به پایان برسد این رویداد رخ میدهد
ondragenter		وقتی یک عنصر وارد محلی که قرار است در آن رها شود بگیرد رخ میدهد
ondragleave		وقتی یک عنصر محلی که قرار است در آن رها شود را ترک کند رخ میدهد
ondragover		وقتی عنصر بروی محدوده ای که قرار است در آن رها شود قرار میگیرد رخ میدهد
ondragstart		وقتی عمل کشیدن شروع شود این رویداد رخ میدهد
ondrop		محل/محدوده ای که قرار است یک عنصر در آن رها شود
onscroll		وقتی درون یک عنصر با ماوس اسکرول اتفاق بیوفتد این رویداد رخ میدهد



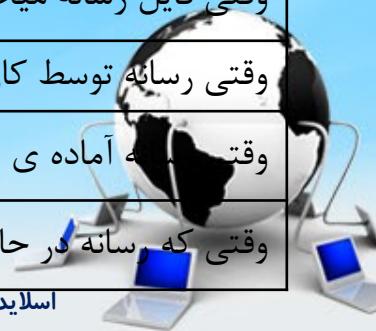
معرفی رویدادهای کلیپ مورد در html

نام رویداد	html5	توضیح
oncopy	-	وقتی یک متن و.. در حافظه کپی (Copy) شود این رویداد رخ میدهد
oncut	-	وقتی یک متن و.. در حافظه کات (Cut) شود این رویداد رخ میدهد
onpaste	-	وقتی یک متن و.. که در حافظه قرار دارد پیست شود این رویداد رخ میدهد



معرفی رویدادهای رسانه ها در html

نام رویداد	html5	توضیح
onabort	-	وقتی در هنگام لود یک رسانه مشکلی (ارتباط قطع شود) پیش بیاید رخ میدهد
oncanplay		وقتی یک رسانه بتواند پلی شود(قابلیت اجرا داشته باشد) رخ میدهد
oncanplaythrough		وقتی یک رسانه بدون توقف بتواند اجرا (پلی) شود رخ میدهد
oncuechange		وقتی تغییراتی درون ترک (Track) اتفاق بیوفتد رخ میدهد
ondurationchange		وقتی تغییراتی درون مدت زمان رسانه اتفاق بیوفتد رخ میدهد
onemptied		وقتی ارتباط بصورت غیرمنتظره قطع شود این رویداد رخ میدهد
onended		وقتی نمایش/پلی یک رسانه به پایان رسد این رویداد رخ میدهد
onerror		وقتی در هنگام بارگذاری فایل خطایی ایجاد شود این رویداد رخ میدهد
onloadeddata		وقتی داده های یک رسانه لود/بارگیری شود این رویداد رخ میدهد
onloadedmetadata		وقتی متاداده های یک رسانه مثله (مدت زمان و...) لود شود رخ میدهد
onloadstart		وقتی فایل رسانه میخواهد شروع به اجرا شدن کند رخ میدهد
onpause		وقتی رسانه توسط کاربر یا.. متوقف شود رخ میدهد
onplay		وقتی رسانه آماده ی پخش شدن باشد رخ میدهد
onplaying		وقتی که رسانه در حال پخش/پلی باشد رخ میدهد



ادامه معرفی رویدادهای رسانه ها در html

نام رویداد	html5	توضیح
onprogress		وقتی که مرورگر در حال دریافت داده های مربوط به رسانه باشد رخ میدهد
onratechange		وقتی در حالت پخش رسانه (مثله آهسته یا تند کردن) تغییری ایجاد شود رخ میدهد
onseeked		وقتی کاربر به نقطه ی پایان از پخش رسانه برسد رخ میدهد
onseeking		وقتی کاربر در حال جابه جایی به نقطه جدید از پخش رسانه باشد رخ میدهد
onstalled		وقتی مرورگر قادر به دریافت اطلاعات رسانه نمی باشد رخ میدهد
onsuspend		وقتی مرورگر نتواند به هر دلیلی اطلاعات رسانه را بارگیری کند رخ میدهد
ontimeupdate		وقتی زمان (مدت زمان رسانه) توسط کاربر تغییر کند رخ میدهد
onvolumechange		وقتی تغییری در صدای رسانه (کوتاه-بلند-قطع) ایجاد شود رخ میدهد
onwaiting		وقتی که داده ها در حال دریافت می باشند و کاربر باید صبر نماید رخ میدهد



جمع بندی معرفی رویدادها در html

ما در پروژه ممکنه به برخی از این رویدادها نیاز داشته باشیم که فکر می کنم بیشتر مربوط به کیبورد و ماوس هست
ما در اینجاسعی کردیم لیست تمام رویدادها را به تفکیک نمایش دهیم لذا در پروژه خودمان از همه اینها استفاده نمی کنیم



پایان جلسه
موفق باشید

