

2014

M. Damrudi

نمره نهایی

- ✓ امتحان میان ترم
- ✓ تمرین
- ✓ امتحان عملی
- ✓ پروژه

✓ مطالب در سایت www.damrudi.ir

2014

M. Damrudi

شبکه

هر شبکه مجموعه‌ای از کامپیوترها است که می‌توانند از طریق یک رسانه انتقالی مشترک با یکدیگر ارتباط برقرار کنند. در هر شبکه درخواستها و داده از یک کامپیوتر به وسیله رسانه‌ی انتقالی (که ممکن است کابل شبکه، خط تلفن و یا ارتباط بی‌سیم باشد) به سوی کامپیوتر دیگر عبور داده می‌شوند.

2014

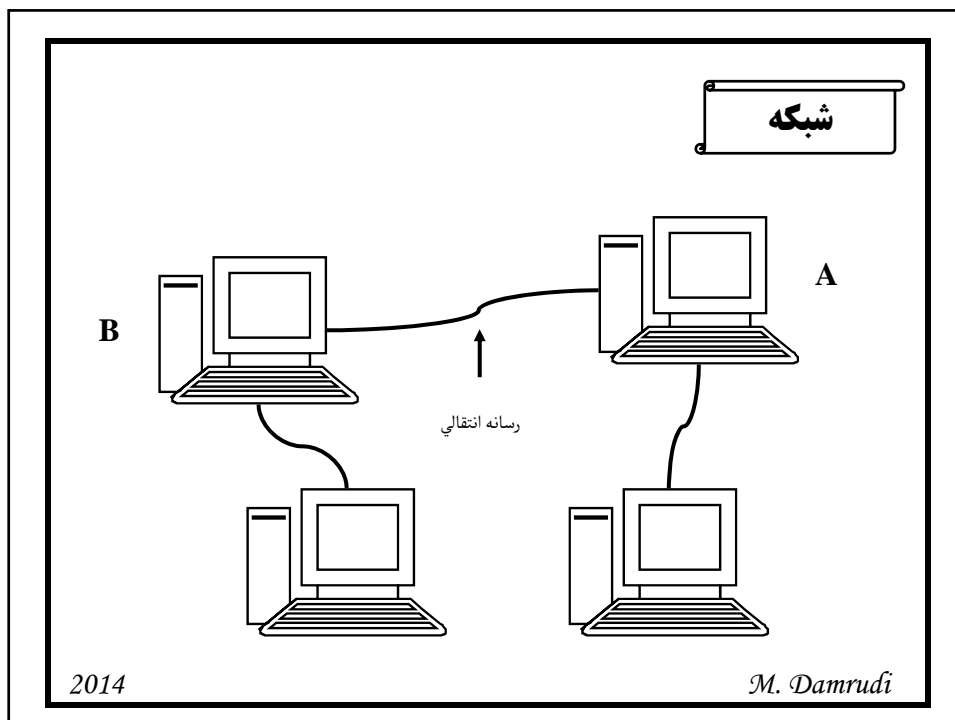
M. Damrudi

شبکه

در شکل بعد کامپیوتر **A** می‌تواند پیام یا درخواستی را به کامپیوتر **B** بفرستد. کامپیوتر **B** قادر است که پیام **A** را درک کند و از طریق ارسال یک پیام به آن پاسخ گوید.

2014

M. Damrudi



وب و شبکه اینترنت

در سال 1984 ایده آن توسط دانشمندان فیزیک مطرح شد که می خواستند روی یک موضوع مشترک تحقیق و بررسی کنند (مثلاً یکی در سوئیس بود و دیگری در آلمان) و لازم بود مستنداتی که تهیه می کنند را خیلی سریع هر دو به اطلاع هم برسانند در واقع روشی پیدا کنند که مستندات خود را به هم **Link** یا متصل کنند.

2014

M. Damrudi

وب و شبکه اینترنت

سپس ایده پروتکلی را مطرح کردند که از طریق آن می توانستند مستندات قابل Link شدن به یکدیگر را در محیط اینترنت داشته باشند در حال حاضر این مستندات Page (صفحه) نامیده می شود و نرم افزاری که قابلیت نشان دادن این Page ها را داشته باشد Browser (مرورگر) می نامند.

Page ، همه اطلاعات اینترنت در قالب page هستند که حاوی Data ، متن ، تصویر ، آدرس و Link هستند.

2014

M. Damrudi

وب و شبکه اینترنت

امکان Browser

- ✓ دسترسی به صفحات web ،
- ✓ برقراری ارتباط با دیگر صفحات (Link) ،
- ✓ نگه داری آدرس صفحات پیموده شده در اینترنت (history) ،
- ✓ forward ،
- ✓ backward ،
- ✓ اتصال سریع به مبدا یا homepage ،
- ✓ امکان print صفحات و ...

2014

M. Damrudi

وب و شبکه اینترنت

دو نمونه از Browser ها ، Internet Explorer و Netscape می باشند.

این سیستم که از طریق صفحات توزیع شده می تواند Link هایی داشته باشد ، WWW (World Wide Web) می گویند.

2014

M. Damrudi

شرکتهای مرتبط با اینترنت

ایده مطرح شده توسط دانشمندان در محیط Text بود. اما بعد دانشجویی ایده ی گرافیکی کردن این محیط را مطرح کرد (تا قبل از سال 1993 اینترنت گرافیکی نبود) این دانشجو سپس بنیانگذار شرکت Netscape شد و یک Browser در سال 1993 بنام Mosaic بوجود آورد.

2014

M. Damrudi

شرکتهای مرتبط با اینترنت

از آن به بعد Browserهایی که ایجاد شد Netscape Navigator یا Communication Netscape نام نهادند. هم اکنون بیش از 30 میلیون کاربر از این برنامه‌ها استفاده می‌کنند. Microsoft برای رقابت با این شرکت Internet Explorer را عرضه کرد.

2014

M. Damrudi

پروتکل

پروتکل اینترنت یعنی زبان مشترکی که همه کامپیوترها متصل به اینترنت برای اتصال به آن رعایت می‌کنند.

پروتکل ، گروه‌بندی برخی از استانداردها است.

2014

M. Damrudi

پروتکل

در مرکزیت تمام پروتکل‌ها TCP/IP قرار دارد که همه پروتکل‌ها روی لایه‌های آن تعریف شده است. پروتکل‌های اینترنت پروتکل‌هایی هستند که برای انتقال فایل، ارسال Mail، جستجو در اینترنت، استفاده از Web و ... به کار می‌آیند.

Transport Control Protocol/Internet Protocol TCP/IP

2014

M. Damrudi

ساختار اینترنت

صنعت شبکه دارای یک مدل 7 لایه‌ای استاندارد برای معماری شبکه می‌باشد که مدل OSI نام دارد. مدل OSI نمایانگر تلاش سازمان استانداردهای بین المللی برای استاندارد کردن طراحی سیستم‌های پروتکلی شبکه برای تولیدکنندگان نرم‌افزار است.

2014

M. Damrudi

ساختار اینترنت

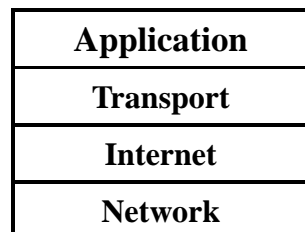
در واقع مدل OSI اغلب به عنوان يك چارچوب كاري عمومي براي درك سیستم‌هاي پروتکلي مورد استفاده قرار مي گيرد. مدل پروتکلي TCP/IP بر طبق مدل‌هاي OSI تعريف مي شوند. در شكل لايه‌هاي تعريف شده در هر يك ملاحظه مي گردد.

Open System Interconnection
OSI

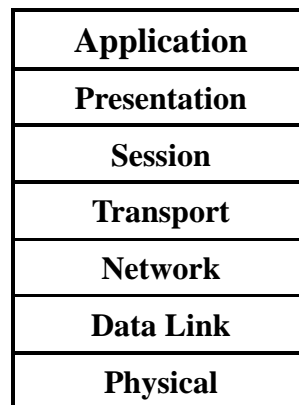
2014

M. Damrudi

ساختار اینترنت



TCP/IP



OSI

2014

M. Damrudi

پروتکل‌های اینترنت

پروتکل‌های اینترنت (FTP, Gopher, HTTP و ...) روی لایه‌ی چهارم مدل TCP/IP فعلی لایه‌ی کاربرد یا Application قرار دارند.

پروتکل (File Transfer Protocol) FTP

FTP پروتکل انتقال فایل است و یکی از قدیمی‌ترین پروتکل‌های انتقال اینترنت می‌باشد خصوصیت FTP اینست که می‌توان توسط آن به سیستم‌هایی که دارای رمز عبور هستند با کمترین سطح دسترسی وارد شد.

2014

M. Damrudi

پروتکل‌های اینترنت

پروتکل (Hyper Text Transfer Protocol) HTTP

پایه و اساس وب روی HTTP می‌باشد به آن Stateless (بدون حالت) می‌گویند. یعنی در این پروتکل هیچ‌گونه آدرس و اطلاعاتی از سندی که انتقال پیدا کرده است نگهداری نمی‌شود. Browser از طریق آن می‌تواند درخواست‌های خود را ارسال کرده و صفحات مورد نظر را ببیند.

2014

M. Damrudi

Host میزبان

میزبان به طور عمومی کار سرویس دهی را انجام می دهد و در این کاربرد خاص به میزبان ، ISP (سرویس دهنده ی اینترنت) گویند. با اتصال به میزبان می توان به اینترنت متصل شد و در اینترنت به مرور موارد دلخواه پرداخت مگر اینکه ISP یا میزبان با استفاده از نرم افزارهایی مانع از دسترسی به برخی از سایت ها شوند.

2014

M. Damrudi

Host میزبان

ISP
Internet Service Provider

(شرکتهای ارائه دهنده سرویس دسترسی به اینترنت)

2014

M. Damrudi

Domain

هر ISP روي شبکه وب داراي يك نام منحصر بفرد است که به آن Domain گویند مانند :

Irna.com
Netradio.net

اسامي فوق اسامي اينترنتي هستند. اما در اينترنت بايد اين اسامي به آدرس هاي فیزیکی تبدیل شود.

2014

M. Damrudi

Domain

شرکت هاي معدودي وجود دارند که در اينترنت اين کار را انجام مي دهند. مي توان با آنها تماس گرفت و IP آدرس و اسم را به آنها گفت تا آن روي Database خود ثبت کنند و باعث معرفي آن به اينترنت شوند.

2014

M. Damrudi

DNS

DNS سرویسی است که وظیفه‌ی تبدیل آدرس‌های اسمی اینترنتی به آدرس‌های فیزیکی را بر عهده دارد. در واقع DNS يك بانك اطلاعاتي مي‌سازد و در آن اسم Domainها و IP که به آنها اختصاص پیدا کرده است را قرار می‌دهد و اسم اینترنتی را هم که برای آن‌ها مشخص کردیم نشان می‌دهد. به طور مثال:

آدرس فیزیکی \rightarrow www.Neda.net : اسم اینترنتی

2014

M. Damrudi

IP address

قرارداد کنترل اطلاعات TCP/IP برای هر کس يك IP address در نظر می‌گیرد. آدرس‌های IP ، نظیر آدرس‌های مکانی تحویل به محل مورد نظر و صحیح را تعیین می‌کنند. با استفاده از نام و شماره آدرس ، محل تحویل شناسایی می‌شود. آدرس IP هم به دو قسمت تقسیم شده است. ID شبکه و ID میزبان.

2014

M. Damrudi

URL

URL (Uniform Resource Locator)

هر سایت در اینترنت يك شناسنامه منحصر به فرد دارد که به آن **Domain** می گوئیم. هر **Page** نیز يك شناسنامه منحصر به فرد دارد که به آن **URL** می گویند. مانند :

[HTTP://www.cnm.com/main/main.htm](http://www.cnm.com/main/main.htm)

2014

M. Damrudi

URL

اجزاء تشکیل دهنده ی URL :

1- قسمت اول نشان دهنده ی نوعی پروتکلی است که جهت بازیابی سند استفاده می شود. **HTTP**

2- بخش دوم نشان دهنده ی کامپیوتر میزبان.

www.cnm.com

3- بخش سوم زیر شاخه روی (فهرست) **main**

4- بخش چهارم نام فایل (نام صفحه **webpage**)

main.htm

2014

M. Damrudi

URL

در پایان نام **Domain** از يك پسوند استفاده مي شود که نشان دهنده ي نوع موسسه يا شرکت و يا کشوري است که آن سایت در آنجا قرار دارد. مانند :

.com	(commercial business)
.edu	(education)
.gov	(government)
.org	(organization)
.net	(network)
.au	Australia
.ir	Iran

2014

M. Damrudi

website

از يك سري **Page** ها تشكيل شده است که اين **Page** ها به هم متصل هستند. (**Page** هاي **Link** شده به هم). اين صفحات روي **web server** هستند. اولين صفحه اي که با نوشتن آدرس يك **Website** روي **Browser** نمايان مي شود **Home page** است.

2014

M. Damrudi

website

اطلاعاتی که بهتر است در **Home Page** باشد عبارتند از:

- 1-جدولي ازعناوين و فهرست‌هاي اطلاعات داخل سايت.
- 2-تعدادي **Link** به صفحات ديگر سايت يا سايت‌هاي ديگر مرتبط با آن سايت.
- 3-تمام صفحات يك **Website** بهتر است يك **Link** به **homepage** داشته باشد.

2014

M. Damrudi

زبانهای markup

عوامل اصلي و اوليه صفحات **Web** هستند. اين زبان‌ها شامل تعدادي نشانه **Tag** هستند که اين نشانه‌ها طريقه نشان دادن سند **HTML** است.

98 درصد صفحات **Web** با زبان **HTML** نوشته مي‌شود و بقيه با زبان‌هاي **Modeling** نوشته مي‌شوند.

2014

M. Damrudi

زبانهای Modeling

مهمترین زبان Modeling، VRML می باشد.
(Virtual Real Modeling Language)

VRML زبانی است که دارای واقعیت مجازی است. برنامه VRML می تواند از Server به کامپیوتر Client انتقال پیدا کند و از طریق Browser دیده شود. VRML یک محیط سه بعدی را می تواند به کاربر نشان دهد و کاربر می تواند در این محیط حرکت کند.

2014

M. Damrudi

زبانهای Modeling

مثلاً برخی از فروشگاهها VRML را روی سایت خود قرار می دهند و به کمک آن مشتریها می توانند کالای خود را خریداری کنند.

از آنجائیکه VRML یک محیط سه بعدی است و دائماً باید Render شود احتیاج به پردازش زیادی دارد.

2014

M. Damrudi

زبانهای Modeling

پس کامپیوتر باید بسیار قوی باشد. از طرفی فایل‌های VRML معمولاً خیلی بزرگ هستند چون برای تعریف یک جسم کوچک تمام قسمت‌ها باید تعریف شود.

Netscape Internet Explorer نرم‌افزار VRML را در داخل خود دارند و VRML را نشان می‌دهد.

2014

M. Damrudi

جستجو در اینترنت

برای جستجوی اطلاعات در اینترنت نیاز به موتورهای جستجو (Search Engines) داریم.

موتورهای جستجو الگوریتم‌های متفاوتی را برای جستجوی اطلاعات استفاده می‌کنند.

انواع موتورهای جستجو عبارتند از:

Google, Yahoo, AltaVista, AOL, Excite

2014

M. Damrudi

نحوه جستجو

Google

- ❖ موضوع
- ❖ نوع فایل
- ❖ تعداد نتایج در يك صفحه
- ❖ زمان
- ❖ زبان
- ❖ ...

2014

M. Damrudi

HTML

HyperText Markup Language

در HTML مي توان Link داشت که این Link ها مي توانند شما را به محلي در همان صفحه یا صفحات ديگري در مکانهاي ديگري منتقل کند.

هر بخشي از متن مي تواند این Link داشته باشد. به این تکه متن که داراي Link است Hypertext گفته مي شود.

2014

M. Damrudi

HTML

HyperText Markup Language

HTML مجموعه‌ای از کدهای منطقی (markup) است که یک مستند وب و محتویاتی که دارد را تشکیل می‌دهد.

کدها در بین علامت < > قرار می‌گیرند. به این کدها tag می‌گویند. tagها case sensitive نیستند به این معنی که نسبت به حروف کوچک و بزرگ حساس نیستند. اما بهتر است tagها را با حروف بزرگ بنویسیم.

اکثر عناصر دارای tag ابتدایی و انتهایی هستند که tag انتهایی دارای "/" است.

2014

M. Damrudi

HTML

اولین کلمه یا کاراکتری که پس از < قرار می‌گیرد عنصر یا element می‌نامند. element عنصری است که به browser می‌گوید چه کاری انجام دهد.

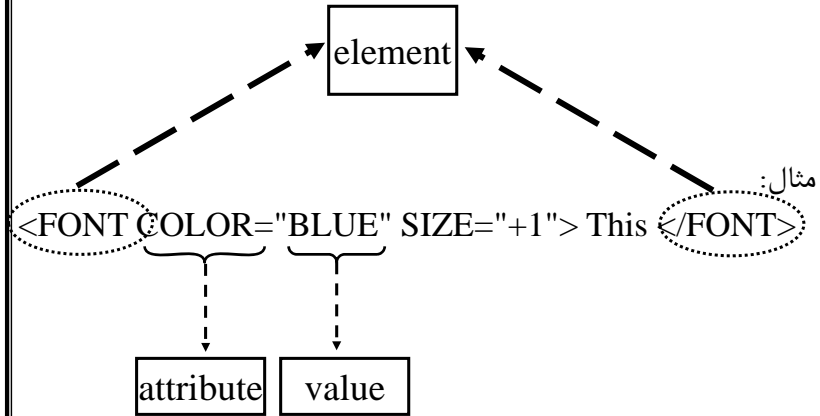
کلمه‌ای که پس از عنصر در داخل < > و پیش از > قرار دارد را صفت یا attribute می‌گویند.

attributeها به وسیله فاصله از یکدیگر جدا می‌شوند. مقدار این صفت‌ها را value می‌گویند که با = این مقداردهی انجام گردد.

2014

M. Damrudi

HTML



2014

M. Damrudi

HTML

هر صفحه HTML داراي دو قسمت اصلي HEAD و BODY است.

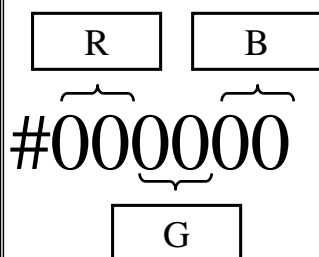
```
<HTML>
<HEAD>
<TITLE> TITLE </TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

2014

M. Damrudi

HTML

کدهای رنگی در HTML براساس RGB است که با مقادیر هگزادسیمال نشان داده می‌شود.



رنگها به صورت زیر می‌باشند:

2014

M. Damrudi

HTML

RGB Color	Hexadecimal Value
White	#FFFFFF
Black	#000000
Red	#FF0000
Green	#00FF00
Blue	#0000FF
Magenta	#FF00FF
Cyan	#00FFFF
Yellow	#FFFF00
Aquamarine	#70DB93
Baker's Chocolate	#5C3317
Violet	#9F5F9F
Brass	#B5A642
Copper	#B87333
Pink	#FF6EC7
Orange	#FF7F00

2014

M. Damrudi

HTML

برای اطلاعات بیشتر درباره رنگها می‌توانید از سایت زیر استفاده نمایید:

<http://www.vgdesign.com/color.html>

در صورتیکه بخواهیم رنگ background را تعیین کنیم از روش زیر استفاده می‌شود. رنگ background به طور پیش‌فرض سفید است.

```
<BODY BGCOLOR=="red" > </BODY>  
<BODY BGCOLOR=="rgb(0,0,0)"> </BODY>  
<BODY BGCOLOR="#FF0000" > </BODY>
```

2014

M. Damrudi

HTML

رنگ متن نوشته شده (text) به طور پیش‌فرض سیاه است که برای تغییر آن باید به روش زیر عمل کرد:

```
<BODY TEXT="#0000FF" > </BODY>
```

باید در نظر داشت که رنگ پیش‌زمینه و پس‌زمینه به‌گونه‌ای باشد که متن به خوبی قابل خواندن باشد.

2014

M. Damrudi

HTML

Link, VLink, ALink

➤ **Link**: ظاهر اولیه لینک که به طور پیش فرض آبی است.
➤ **VLink**: لینکی که بر روی آن قرار گرفته ایم اما کلیک نشده است و به طور پیش فرض بنفش است. (**Visited Link**)
➤ **ALink**: لینکی که کلیک شده است و به طور پیش فرض قرمز است. (**Active Link**)

برای تغییر این موارد به روش زیر عمل می کنیم:

```
<BODY BGCOLOR="#FF0000" TEXT="#0000FF"  
Link="#FFFFFF" VLink="#FF00FF" ALink="#FFFF00">  
</BODY>
```

2014

M. Damrudi

HTML

Background image

می توان به جای رنگ از تصویر برای پیش زمینه استفاده کرد. تصاویر پیش زمینه به صورت **tiled** هستند به این معنی که چند نسخه از عکس در کنار هم و زیر هم قرار می گیرند تا صفحه پر شود.

می توانیم رنگ **background** را نیز تعیین کنیم. مدت زمانی که طول می کشد تا تصویر **load** شود، رنگ پیش زمینه مشخص می شود.

2014

M. Damrudi

HTML

Background image

```
<BODY BACKGROUND="NameOfLogo.gif"  
BGCOLOR="#FF0000"> </BODY>
```

توجه داشته باشید رنگ تصویر روشن باشد تا متن روی آن قابل خواندن باشد.

2014

M. Damrudi

HTML

کدهای مربوطه را در notepad نوشته و به پسوند .htm یا .html ذخیره نمایید.

برای ویرایش آن از کلیک راست روی فایل و انتخاب گزینه Edit استفاده می شود.

2014

M. Damrudi

در داخل عنصر **BODY** می‌توان متن را سطح‌بندی کرد همانند بخش‌های یک کتاب که عنوان‌ها و زیرعنوان‌ها دارای اندازه متناسب خود هستند.

این عنوان‌ها به صورت زیر تعیین می‌گردند که X از 1 تا 6 می‌باشد.

`<Hx > </Hx >`

باید مراقب باشید که ترتیب آن‌ها رعایت گردد و بعد از **H1** مثلا **H3** را قرار ندهید زیرا ابتدا باید **H2** بیاید.

```
<HTML>
<HEAD> <TITLE>Example Page</TITLE> </HEAD>
<BODY>
<H1>Heading 1</H1>
<H2>Heading 2</H2>
<H3>Heading 3</H3>
<H4>Heading 4</H4>
<H5>Heading 5</H5>
<H6>Heading 6</H6>
</BODY>
</HTML>
```

خروجي اين كد به صورت زير مي باشد:

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

زمانیکه از پاراگراف استفاده می شود انتهای خط را اگر مطلب بیش از اندازه پنجره مرورگر باشد متناسب با اندازه پنجره مرورگر قرار می دهد و ادامه مطلب را از خط بعد آغاز می کند. اگر پنجره را کوچکتر یا بزرگتر کنید نیز متن را متناسب با اندازه پنجره قرار می دهد. برای پاراگراف بندی از tag زیر استفاده می شود:

```
<P> </P>
```

متن و تصویر را می توان راست چین ، چپ چین یا وسط چین نمود. برای انجام اینکار از "ALIGN="right" ، ALIGN="center" و ALIGN="left" استفاده می شود.

```
<P ALIGN="center">
```

Welcome

```
</P>
```

HTML

Paragraph

اگر چند خط خالی در صفحه خود قرار دهید ، مشاهده می کنید که تنها یک خط خالی ایجاد می شود.

اگر بین کلمات بیش از یک فاصله قرار دهید ، تنها یک فاصله بین آن ها قرار می دهد.

Browser non-breaking space فاصله های اضافی را در نظر نمی گیرد. برای قراردادن فاصله هایی اضافی از **NBSP** استفاده می شود که به معنی ** ** است و هر ** ** یک فاصله خالی ایجاد می کند.

2014

M. Damrudi

HTML

Break

این tag همانند زدن کلید **enter** رفتار می کند.

با قراردادن این tag می توانیم جمله را در وسط خط و هر مکانی را که مورد نیاز است به اتمام برسانیم.

این tag همانند tag های دیگر دارای ابتدا و انتها نیست و هر جا که نیاز است ادامه مطلب از خط بعد آغاز گردد از این tag استفاده می شود.

2014

M. Damrudi

HTML

Example

Break

```

<HTML>
<HEAD> <TITLE>Example Page2</TITLE>
</HEAD>
<BODY>
<P>This is a test of paragraph in html pages</P>
<P>&nbsp;</P>
<P>This is a test of break in <BR> html pages</P>
</BODY>
</HTML>

```

2014
M. Damrudi

HTML

Break

خروجي اين كد به صورت زير است:

This is a test of paragraph in html pages

This is a test of break in
html pages

2014
M. Damrudi

HTML

Horizontal Rule

با استفاده از این tag می‌توان یک خط افقی ایجاد کرد. این tag نیز مانند `<HR>` ، break tag پایانی ندارد.

Attribute	Description	Default Value
SIZE	height of the rule in pixels	2 pixels
WIDTH	width of the rule in pixels or percentage of screen width	100%
NOSHADE	draw the rule with a flat look instead of a 3D look	not set (3D look)
ALIGN	aligns the line (Left, Center, Right)	Center
COLOR	sets a color for the rule (IE 3.0 or later)	not set

2014

M. Damrudi

HTML

Character formatting

☞ `<PRE>Preformatted </PRE>`

متنی که در داخل این tag قرار می‌گیرد فونت‌ها را با کاراکترهای با فاصله واحد نشان می‌دهد.

☞ ` Emphasis `

مروگرها متن داخل این tag را Emphasize نشان می‌دهند.

☞ ` STRONG `

مروگرها متن داخل این tag را مانند bold نشان می‌دهند.

☞ `<TT> </TT>`

مانند PRE ، tag عمل می‌کند.

2014

M. Damrudi

HTML

Character formatting

☞ ` Two sizes bigger `

اندازه متن با صفت **size** تعیین می‌شود که می‌تواند از 1 تا 7 باشد و یا با یک مقدار نسبی نسبت به آنچه هست تعیین گردد. رنگ متن با **color** تعیین می‌شود.

☞ `Font changed `

☞ ` Bold `

☞ `<I> Italic </I>`

☞ `<U> Underline </U>`

2014

M. Damrudi

HTML



Character formatting

`<P>One Size Larger - Normal - One Size Smaller
Bold - <I>Italics</I> - <U>Underlined</U> - Colored
Emphasized - Strong - <TT>Tele Type</TT>
</P>`

2014

M. Damrudi

خروجي اين كد به صورت زير مي باشد.

One Size Larger - Normal - One Size Smaller
Bold - *Italics* - Underlined - Colored
Emphasized - **Strong** - Tele Type

بعضي از کاراکترها را نمی توان همانند بقیه به کار برد. برای استفاده از آنها باید از علامت & استفاده شود.

جدول اسلاید بعد بعضي از این کاراکترها را نشان می دهد.
 در صورتیکه اطلاعات جامع تري می خواهید از سایت زیر استفاده نمایید.

http://www.w3.org/MarkUp/HTMLPlus/htmlplus_13.html

HTML
Special characters

Character	Entity	IE	Character	Entity	IE
ampersand	&	&	copyright	©	©
greater-than sign	>	>	quotation mark	"	"
asterisk	∗	*	fraction one qtr	¼	¼
less-than sign	<	<	registration mark	®	®
cent sign	¢	¢	fraction one half	½	½
non-breaking space	 		trademark sign	™	™

2014
M. Damrudi

HTML
Special characters

⇒<STRIKE> strike-through text </STRIKE>

⇒<BIG> places text in a big font </BIG>

⇒<SMALL>places text in a small font</SMALL>

⇒_{places text in subscript position}

⇒^{places text in superscript position}

2014
M. Damrudi

HTML	<i>Example</i>	Special characters
<pre> <P><STRIKE> strike-through text </STRIKE>
 <BIG> places text in a big font </BIG>
 <SMALL> places text in a small font </SMALL>
 <SUB> places text in subscript position </SUB> Normal <SUP> places text in superscript position</SUP>
 </P> </pre>		
2014	<i>M. Damrudi</i>	

HTML		Special characters
<pre> strike-through text places text in a big font places text in a small font places text in subscript position Normal places text in superscript position </pre>		
2014	<i>M. Damrudi</i>	

HTML سه نوع لیست دارد:

1- لیست نامرتب (Unordered List):

که آن را با UL نشان می دهند.

2- لیست مرتب (Ordered List):

که آن را با OL نشان می دهند.

3- لیست توضیحی (Definition List):

که آن را با DL نشان می دهند.

عناصر این لیست با یک علامت مانند bullet شروع می شود. که در TYPE نوع آن مشخص می گردد. انواع آن عبارتند از: Disc (•)، Square (■)، Circle (○) است که به طور پیش فرض Disc است.

```
<UL TYPE="SQUARE">
```

```
<LI>List item...</LI>
```

```
<LI>List item...</LI>
```

```
<LI>List item...</LI>
```

```
</UL>
```

- List item ...
- List item ...
- List item ...

HTML**Ordered List**

عناصر این لیست به طور خودکار توسط **browser** شماره گذاری می شوند. نوع شماره گذاری توسط **TYPE** مشخص می گردد. می توان شروع شماره گذاری را با استفاده از **START** مشخص کرد. در مواردی که لیست توسط عکس و یا توضیحاتی جدا می شود مفید است. انواع **TYPE** عبارتند از:

TYPE	Number Style
1	1, 2, 3, ...
a	a, b, c, ...
A	A, B, C, ...
i	i, ii, iii, ...
I	I, II, III, ...

2014

*M. Damrudi***HTML****Ordered List**

```
<OL TYPE="i">
<LI>List item...</LI>
<LI>List item...</LI>
</OL>
<P>Unrelated text ... </P>
<OL TYPE="i" START="3">
<LI>List item...</LI> </OL>
```

i. List item...
 ii. List item...
 Unrelated text ...
 iii. List item...

2014

M. Damrudi

HTML

Definition List

این لیست با لیست‌های دیگر متفاوت است. در این لیست هر **DL** شامل یک یا چند **DT** (Definition Term) است که به دنبال آن یک یا چند **DD** (Definition Description) می‌آید.

```
<DL>
<DT>HTML</DT>
<DD>Hyper Text Markup Language</DD>
<DT>Dog</DT>
<DD>A human's best friend!</DD>
</DL>
```

HTML
Hyper Text Markup Language
Dog
A human's best friend!

2014

M. Damrudi

HTML

Image

فرمت تصاویری که توسط HTML پشتیبانی می‌شود. 

GIF: Graphic Interchange Format

از 256 رنگ استفاده می‌کند. این فرمت بهترین نوع برای **icon**ها، دکمه‌ها و **logo**ها می‌باشد. این فرمت دو ویرایش دارد، **GIF 89a** جدیدتر از **GIF 87** است که کاربردهای بیشتری نسبت به **GIF 87** و **JPEG** دارد. **GIF** دارای ویژگی‌های **Interlacing**، **Transparency** و **Animated GIFs** می‌باشد.

2014

M. Damrudi

HTML

Image

✎ فرمت تصاویری که توسط HTML پشتیبانی می‌شود.

✎ JPG, JPEG: Joint Photographic Expert Group

این فرمت دارای ویژگی‌های Interlacing ، Transparency نیست اما می‌توان درجه فشردگی را مشخص کرد. بنابراین می‌توان تعادلی بین کیفیت و اندازه تصویر ایجاد کرد.

فرمت پیشرفته‌تری به نام Pro-JPEG نیز وجود دارد که دارای ویژگی‌های پیشرفته‌تری از جمله Interlacing است.

2014

M. Damrudi

HTML

Image

✎ فرمت‌هایی که کمتر توسط HTML پشتیبانی می‌شوند و یا اصلاً پشتیبانی نمی‌شوند.

✎ PNG: Portable Network Graphics

توسط Internet Explorer قابل پشتیبانی است. برای ترکیب متن و گرافیک مناسب است. Interlacing ، Transparency دوبرعده‌ای و نمایش مستقل از سخت‌افزار از ویژگی‌های این فرمت است.

✎ TIFF: Tagged Image File Format

2014

M. Damrudi

HTML

Image

➤

برای اضافه نمودن تصویر از این tag استفاده می شود.

➤ SRC: Image File

این مقدار URL (محل تصویر) را مشخص می کند.

➤ ALT: Alternate Text

برای شرح تصویر و یا برجسته استفاده می شود. اگر تصویر در صفحه مرورگر کاربر نمایش داده نشود، متن مشخص شده در آن نمایش داده می شود. اگر موس بر روی تصویر قرار گیرد نیز متن نمایش داده می شود.

2014

M. Damrudi

HTML

Image

➤ ALIGN: Alignment

Bottom, Middle, که می تواند شامل Top, Left, Right, TextTop, , Baseline ABSMiddle, ABSBottom باشد.

➤ WIDTH: Width

عرض تصویر را مشخص می سازد.

➤ HEIGHT: Height

ارتفاع تصویر را مشخص می کند.

➤ BORDER: Border

کادر دور تصویر را بر حسب پیکسل مشخص می کند.

2014

M. Damrudi

HTML

Image

➤ HSPACE:

فضای خالی افقی در دو طرف تصویر که بر حسب پیکسل می باشد.

➤ VSPACE:

فضای خالی عمودی در بالا و پایین تصویر که بر حسب پیکسل می باشد.

بهرتر است اندازه تصویر کوچک نگه داشته شود.

در صورتیکه تصویر به عنوان پیش زمینه در نظر گرفته شود خوانایی متن کاهش می یابد.

```
<P ALIGN="left">
```

```
<IMG BORDER = "3" SRC="c:\picture\test.jpg  
WIDTH = "100" HEIGHT = "200"> </P>
```

2014

M. Damrudi

HTML

Link

برای برقراری ارتباط بین صفحات یک سایت از **Link** استفاده می شود.

URLs (Uniform Resource Locators) مکان و نام فایل را برای


ایجاد **Link** ارائه می دهد.

برای ایجاد **Link** از **tag** `<A>` استفاده می شود.

```
<A HREF="URL">Link </A>
```

2014

M. Damrudi

انواع Link  Internal Link

Link هايي كه در داخل يك document قرار دارند.

 Local Link

Link هايي كه در يك وب سرور محلي و در بين صفحات مختلف وب وجود دارد. URL مي تواند كامل يا نسبي باشد.

 External Link

Link هايي كه در صفحاتي به ديگر وب سرورها وجود دارد. URL همواره كامل است.

 URL: Uniform Resource Locator

URL روش استاندارد براي ارجاع مي باشد.

Service :// Host name : Port / Directory-path

مثال:

<http://www.1stvirtual.com:80/default.htm>

هر نرم افزاري روي سرور در پورت خاصي كار مي كند و پورت استاندارد براي web server ، پورت 80 مي باشد.

HTML

Link

◆ HTTP: HyperText Transport Protocol

پروتکل استاندارد برای اتصال به صفحات وب با محتوای multimedia است.

Service :// Host name .Domain. Zone / Web Page

مثال:

<http://www.1stvirtual.com/default.htm>

2014

M. Damrudi

HTML

Link

◆ FTP: File Transfer Protocol

این سرویس برای Link به فایل‌هایی که قرار است از سرور ftp دریافت شود، استفاده شود.

Service :// Host name / Subdirectory/ file

مثال:

<ftp://ftp.1stvirtual.com/prizes/brkbrk.wav>

2014

M. Damrudi

HTML

Link

◆ E-mail: Electronic mail

این سرویس برای اجرای برنامه ارسال E-mail به کار می‌رود.

mailto :The recipient of the message

مثال:

mailto: webmaster@1stvirtual.com

2014

M. Damrudi

HTML

Internal Link

✿ برای ایجاد این Link ، به قسمتی از متن که قرار است Link به آن برقرار شود ، نامی نسبت داده می‌شود. این نام نباید در این صفحه تکراری باشد.

```
<A NAME="bookmark_name">Book Mark</A>
```

✿ متن مورد نظر که قرار است به مکان فوق Link داشته باشد را با استفاده از روشی که بیان شده ، ایجاد می‌کنیم.

```
<A HREF="#bookmark_name">Goto Book Mark</A>
```

2014

M. Damrudi

HTML
Internal Link

می توان یک **Link** به صفحه دیگری را با **Internal Link** ترکیب کرد.

``

2014
M. Damrudi

HTML
Table

* <code><TABLE> </TABLE></code>	♦ ایجاد جدول
* <code><TR></TR></code>	♦ ایجاد سطر جدید
* <code><TH></TH></code>	♦ ایجاد عنوان ستون ها
* <code><TD> </TD></code>	♦ وارد نمودن داده در سلول ها
* <code><CAPTION> </CAPTION></code>	♦ توضیح جدول

2014
M. Damrudi

HTML

Example

Table

```

<TABLE BORDER="1">
<TR>
<TH>Column 1 Header</TH>
<TH>Column 2 Header</TH>
</TR>
<TR>
<TD>Row 1 - Col 1 </TD>
<TD>Row 1 - Col 2 </TD>
</TR>
<TR>
<TD>Row 2 - Col 1 </TD>
<TD>Row 2 - Col 2 </TD>
</TR>
<TR>
<TD>Row 3 - Col 1 </TD>
<TD>Row 3 - Col 2 </TD>
</TR>
</TABLE>

```

2014
M. Damrudi











HTML











Output

Table

Column 1 Header	Column 2 Header
Row 1 - Col 1	Row 1 - Col 2
Row 2 - Col 1	Row 2 - Col 2
Row 3 - Col 1	Row 3 - Col 2

2014
M. Damrudi

HTML	Table
 BgColor:	 رنگ پیش‌زمینه جدول که به صورت هگزادسیمال RGB و یا از اسامی رنگی استاندارد ویندوز می‌باشد.
 Width:	 عرض جدول که به پیکسل یا درصدی از عرض صفحه می‌باشد.
 Border:	 کادر هر سلول جدول است.
 CellSpacing:	 فضای بین سلول‌ها به پیکسل می‌باشد.
 CellPadding:	 فضای بین کادر و محتوای سلول به پیکسل می‌باشد.
2014	<i>M. Damrudi</i>

HTML	Table
 Align:	 جدول تنظیمات left ، right یا center را دارد.
 Background:	 تصویر پیش‌زمینه که به صورت tiled می‌باشد.
 BorderColor:	 رنگ کادر جدول می‌باشد.
 BorderColorLight:	 رنگ روشنی که در دو طرف سلول یا جدول استفاده می‌شود.
 BorderColorDark:	 رنگ تیره‌ای که در دو طرف سلول یا جدول استفاده می‌شود.
2014	<i>M. Damrudi</i>

☞ CAPTION:

توضیح جدول که می تواند TOP یا BOTTOM می باشد.

مثال:

```
<TABLE BORDER="1" CELLPADDING="2">
<CAPTION ALIGN="BOTTOM">Label for my Table
</CAPTION>
```

☞ Colspan:

تعداد ستونهایی که توسط یک سلول پوشش داده می شود.

☞ Rowspan:

تعداد سطریهایی که توسط یک سلول پوشش داده می شود.

☞ Align:

تنظیمات عرضی سلول که left ، right یا center است.

☞ Valign:

تنظیمات طولی سلول که top ، middle یا bottom است.

HTML
Table Data

Background: تصویر پیش‌زمینه سلول که به صورت tiled است.

BGColor: رنگ پیش‌زمینه سلول را تعیین می‌کند.

Width: عرض سلول که به پیکسل یا درصدی از عرض صفحه است.

Height: ارتفاع سلول که به پیکسل یا درصدی از ارتفاع صفحه است.

2014
M. Damrudi

HTML

Example

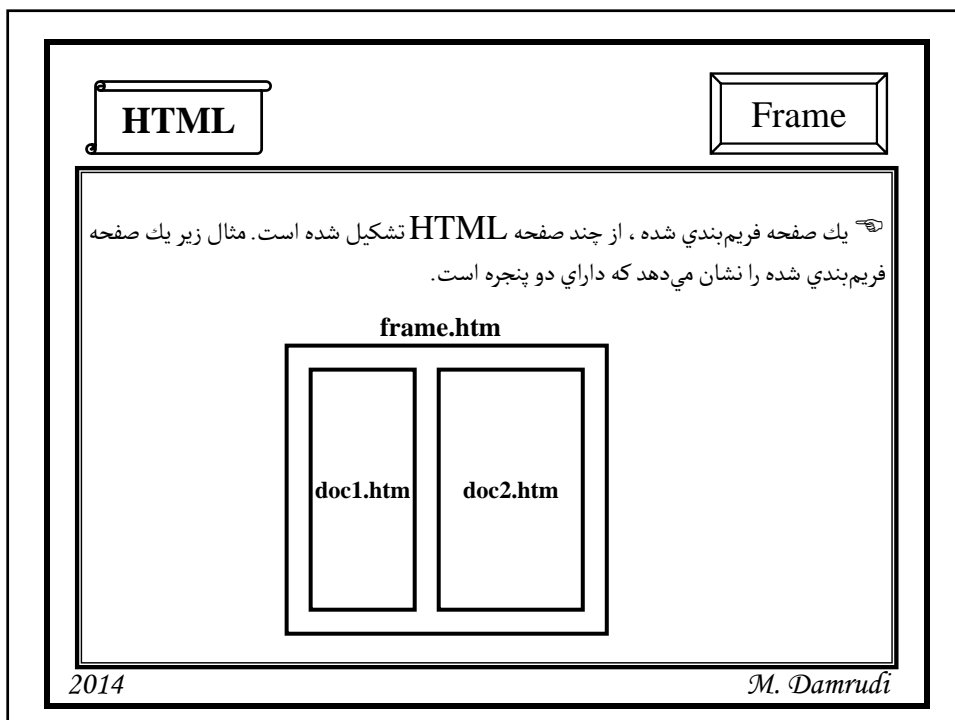
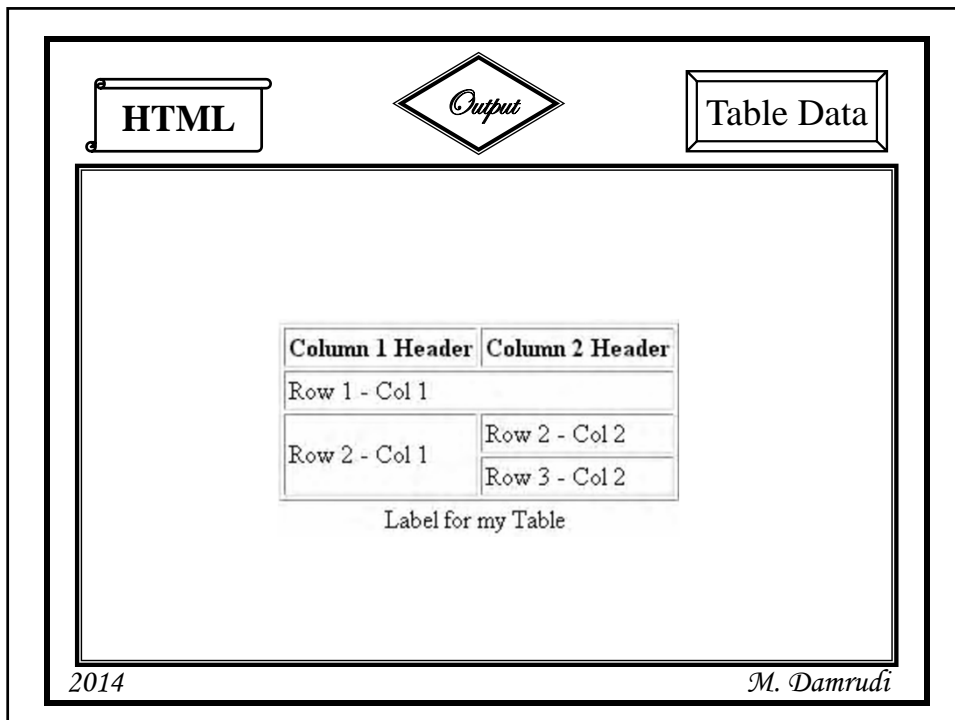
Table Data

```

<TABLE BORDER="1" CELLPADDING="2">
<CAPTION ALIGN="BOTTOM">Label for my Table
</CAPTION>
<TR> <TH>Column 1 Header</TH>
<TH>Column 2 Header</TH> </TR>
<TR> <TD COLSPAN="2">Row 1 - Col 1</TD></TR>
<TR> <TD ROWSPAN="2">Row 2 - Col 1 </TD>
<TD>Row 2 - Col 2 </TD> </TR>
<TR> <TD>Row 3 - Col 2 </TD> </TR>
</TABLE>

```

2014
M. Damrudi



HTML
Frame

👉 تگ **FRAMESET** براي تعيين تعداد و اندازه فریم‌ها و مشخص نمودن عمودي يا افقي بودن فریم‌ها استفاده مي‌شود.

👉 اين تگ **<BODY>** از **<BODY>** قرار مي‌گيرد.

👉 براي ايجاد هر پنجره از يك تگ **FRAME** استفاده مي‌شود.

👉 در هر تگ فریم آدرس يك صفحه **HTML** كه از قبل ايجاد شده است ، قرار مي‌گيرد.

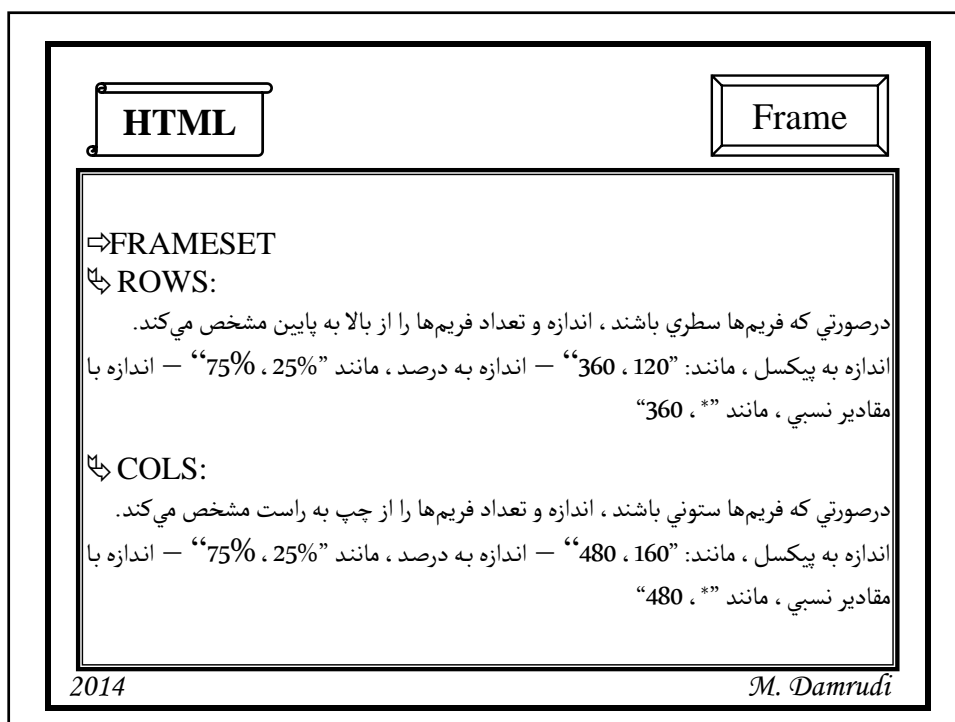
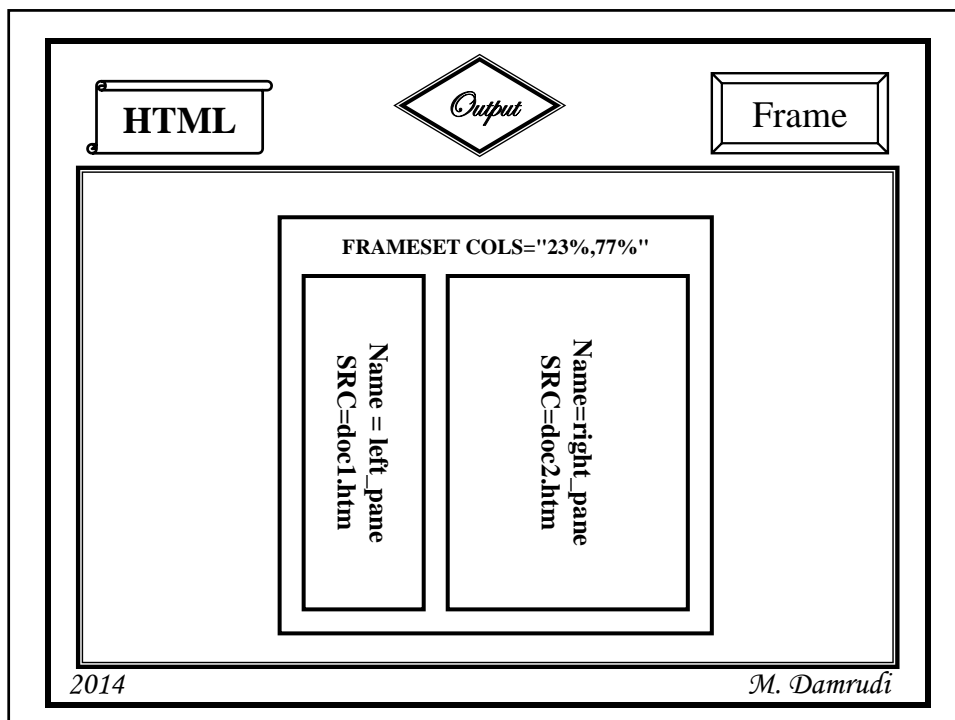
2014
M. Damrudi

HTML
Frame

Example

```
<HTML>
<HEAD> <TITLE>Framed Page </TITLE> </HEAD>
<FRAMESET COLS="23%,77%">
<FRAME SRC="doc1.htm" NAME="left_pane"
SCROLLING="NO">
<FRAME SRC="doc2.htm" NAME="right_pane"
SCROLLING="YES">
<NOFRAMES>
<BODY>
<!-- Your browser that do not support Frames -->
</BODY>
</NOFRAMES>
</FRAMESET>
</HTML>
```

2014
M. Damrudi



HTML	Frame
<p>⇒FRAMESET</p> <p>↳FRAMEBORDER</p> <p>↳FRAMESPACING</p> <p>↳BORDER</p> <p>↳BORDERCOLOR</p>	<p>مقادیر ممکن 0، 1، YES و NO می باشد.</p> <p>فاصله بین فریم ها را مشخص می کند.</p> <p>مشابه FRAMEBORDER است.</p> <p>رنگ کادر فریم را مشخص می کند.</p>
2014	<i>M. Damrudi</i>

HTML	Frame
<p>⇒FRAME</p> <p>◆ SRC:</p> <p>◆ NAME:</p> <p>◆ MARGINWIDTH:</p> <p>◆ MARGINHEIGHT:</p>	<p>URL صفحه ای است که باید در این صفحه نمایش داده شود.</p> <p>نامی برای فریم در نظر گرفته می شود تا در صورتی که این فریم مقصد HTML دیگری باشد، از آن استفاده نماید.</p> <p>فضای خالی افقی به پیکسل است که بین محتوای فریم و کادر فریم در نظر گرفته می شود.</p> <p>فضای خالی عمودی به پیکسل است که بین محتوای فریم و کادر فریم در نظر گرفته می شود.</p>
2014	<i>M. Damrudi</i>

⇒FRAME

◆ SCROLLING:

می‌توان برای فریم‌ها **Scroll bar** قرار داد.
 اگر مقدار آن **yes** باشد، **Scroll** را نمایش خواهد داد و اگر مقدار آن **no** باشد، **Scroll** را نمایش نمی‌دهد. اگر مقدار آن **auto** باشد، **Browser** با توجه به محتوای فریم، در صورت نیاز **Scroll** را قرار می‌دهد.

◆ NORESIZE:

به طور پیش‌فرض کاربر می‌تواند اندازه فریم‌ها را تغییر دهد. با قرار دادن این صفت این امکان از بین می‌رود.

⇒NOFRAME

◆ **Browser**هایی که قابلیت فریم‌بندی را دارند، این **tag** را در نظر نمی‌گیرند.

◆ اما **Browser**هایی که این قابلیت را ندارند، محتوای **NOFRAME** را که با **BODY** آغاز می‌شود، در نظر می‌گیرند.

زمانیکه يك Link ایجاد می‌شود، می‌توان تعیین کرد که Link در کدام FRAME باز شود.

مقصدي که Link باید در آن‌جا باز شود، با Target مشخص می‌شود.

Target و Name، Case Sensitive هستند. اگر نام Target درست نباشد، صفحه در يك Browser جدید بدون فریم باز می‌شود.

اگر Target مشخص نشود، صفحه در فریم جاری باز می‌شود.

چهار نام رزرو شده وجود دارند که نمی‌توان از آن‌ها برای Name در فریم استفاده نمود.

چهار نام رزرو شده برای Target عبارتند از:

① _top

صفحه مورد نظر را همراه URL در پنجره Browser بدون فریم نمایش می‌دهد.

② _blank

یک Browser جدید باز کرده و صفحه را در این پنجره Browser نمایش می‌دهد. این پنجره بدون نام است.

HTML	Frame
<p style="text-align: right;">چهار نام رزرو شده براي Target عبارتند از:</p> <p>③ _self صفحه را در همان محلي که Link وجود دارد Load کرده و نمایش مي دهد.</p> <p>④ _parent FRAMESET پيشين (parent) است که FRAMESET جاري از آن به وجود آمده است.</p>	
2014	<i>M. Damrudi</i>

HTML	Form
<p style="text-align: right;">* براي ايجاد فرم بايد از tag زیر استفاده نمود.</p> <pre style="font-family: monospace;"><FORM> </FORM></pre> <p style="text-align: right;">* هر يك از عناصر فرم بايد در داخل اين tag قرار گیرند.</p> <p style="text-align: right;">* اين tag داراي صفاتي است که در ادامه شرح داده مي شوند:</p>	
2014	<i>M. Damrudi</i>

HTML

Form

**ACTION:

آدرس URL می باشد که اطلاعات فرم برای آن ارسال می شود.

**METHOD:

دو نوع روش برای ارسال اطلاعات وجود دارد:

- 1- روش GET که روش پیش فرض می باشد. در این روش اطلاعات ارسالی به URL در address bar قابل مشاهده است.
- 2- روش POST.

2014

M. Damrudi

HTML

Form

**ENCTYPE:

روش استفاده شده برای به رمز درآوردن اطلاعات ارسالی در این قسمت قرار می گیرد.

**NAME:

نام فرم است که در VBScript یا JavaScript استفاده می شود.

**TARGET:

فریم مقصد است که فرم در آن محل نمایش داده می شود.

2014

M. Damrudi

HTML	Form
<p style="text-align: right;">★ عناصر فرم عبارتند از:</p> <ul style="list-style-type: none"> ➤ Text boxes, ➤ Password boxes, ➤ Checkboxes, ➤ Option (Radio) buttons, ➤ Submit, ➤ Reset, ➤ File, ➤ Hidden and Image. <p style="text-align: right;">★ عناصر فرم در tag <code><INPUT></code> قرار می‌گیرند.</p>	
2014	<i>M. Damrudi</i>

HTML	Form
<p>☞ Text boxes:</p> <p style="text-align: right;">برای وارد کردن اطلاعات استفاده می‌شود.</p> <ul style="list-style-type: none"> ◆ TYPE: text ◆ SIZE: ◆ MAXSIZE: <p style="text-align: right;">اندازه textbox را مشخص می‌کند که پیش فرض 20 است.</p> <p style="text-align: right;">حداکثر تعداد کاراکترهایی که این فیلد می‌پذیرد را تعیین می‌کند.</p>	
2014	<i>M. Damrudi</i>

HTML
Form

☞ Text boxes:

- ◆ NAME: نام متغیر که برای CGI ارسال می شود.
- ◆ VALUE: محتویات textbox را به عنوان مقدار آن قرار می دهد.

اطلاعات فرم به صورت زوج های NAME=VALUE برای CGI فرستاده می شوند.

2014
M. Damrudi

HTML
Form

☞ Password:

- ◆ TYPE: password برای وارد کردن password استفاده می شود.
- ◆ SIZE: اندازه textbox را مشخص می کند که پیش فرض 20 است.
- ◆ MAXSIZE: حداکثر تعداد کاراکترهایی که این فیلد می پذیرد را تعیین می کند.

2014
M. Damrudi

HTML	Form
<p>☞ Hidden: اطلاعاتی که برای CGI ارسال می‌شود و می‌خواهید که مشاهده کنندگان سایت، نتوانند آن را ملاحظه کنند یا تغییر دهند.</p> <p>◆ TYPE: hidden</p> <p>◆ NAME: نام متغیر که برای CGI ارسال می‌شود.</p> <p>◆ VALUE: با مقداری که CGI انتظار دارد، مقداردهی می‌شود.</p>	
2014	<i>M. Damrudi</i>

HTML	Form
<p>☞ Check box: به کاربر اجازه می‌دهد امکان بیش از یک انتخاب را داشته باشد.</p> <p>◆ TYPE: checkbox</p> <p>◆ CHECKED: حالت اولیه checkbox است که در صورت وجود داشتن، حالت انتخاب شده را خواهد داشت.</p> <p>◆ NAME: نام متغیر که برای CGI ارسال می‌شود.</p> <p>◆ VALUE: معمولاً مقداری برای آن در نظر گرفته می‌شود.</p>	
2014	<i>M. Damrudi</i>

☞ Radio Button:

به کاربر اجازه می‌دهد تنها یک انتخاب داشته باشد.

◆ TYPE: radio

◆ CHECKED:

حالت اولیه checkbox است که در صورت وجود داشتن، حالت انتخاب شده را خواهد داشت.

◆ NAME:

نام متغیر که برای CGI ارسال می‌شود.

◆ VALUE:

معمولاً مقداری برای آن در نظر گرفته می‌شود.

☞ File Upload:

برای قرار دادن فایل کاربر در سرور استفاده می‌شود.

◆ TYPE: file

◆ NAME:

نام متغیر که برای CGI ارسال می‌شود.

◆ SIZE:

اندازه textbox را مشخص می‌کند.

HTML	Form
<p>☞ File Upload:</p> <p>◆ MAXSIZE: حداکثر اندازه textbox براساس تعداد کاراکترها مشخص می شود.</p> <p>◆ ENCTYPE="multipart/form-data" این خاصیت مربوط به فرم است که باید دارای مقدار multipart/form-data باشد.</p>	
2014	<i>M. Damrudi</i>

HTML	Form
<p>☞ Submit:</p> <p>برای ارسال اطلاعات فرم به CGI استفاده می شود. آدرس مقصد در ACTION فرم قرار دارد.</p> <p>◆ TYPE: submit</p> <p>◆ NAME: نام متغیر که برای CGI ارسال می شود.</p> <p>◆ VALUE: نام روی دکمه را مشخص می کند.</p>	
2014	<i>M. Damrudi</i>

HTML	Form
<p>☞ Reset:</p> <p style="text-align: right;">به کاربر اجازه می‌دهد اطلاعات فرم را خالی کند.</p> <p>◆ TYPE: reset</p> <p>◆ VALUE:</p> <p style="text-align: right;">نام روی دکمه را مشخص می‌کند.</p>	
2014	<i>M. Damrudi</i>

HTML	Form
<p>☞ <code><TEXTAREA></TEXTAREA></code></p> <p style="text-align: right;">به کاربر اجازه می‌دهد در یک <code>textbox</code> توضیحاتی را ارائه نماید.</p> <p>◆ NAME:</p> <p style="text-align: right;">نام متغیر که برای CGI ارسال می‌شود.</p> <p>◆ ROWS:</p> <p style="text-align: right;">تعداد سطرهاي <code>textbox</code> را مشخص می‌کند.</p> <p>◆ COLS:</p> <p style="text-align: right;">تعداد ستونهای <code>textbox</code> را مشخص می‌کند.</p>	
2014	<i>M. Damrudi</i>

HTML

Form

☞ `<SELECT></SELECT>`

به کاربر اجازه می‌دهد موارد را انتخاب کند.

◆ **NAME:**

نام متغیر که برای CGI ارسال می‌شود.

◆ **SIZE:**

تعداد انتخابهای مشهود را نمایش می‌دهد.

◆ **MULTIPLE:**

اگر این صفت وجود داشته باشد کاربر می‌تواند چندین انتخاب داشته باشد.

2014

M. Damrudi

HTML

Form

☞ `<SELECT></SELECT>`

◆ `<OPTION></OPTION>`

برای قرار دادن هر عنصر در لیست از این tag استفاده می‌شود.

◆ **SELECTED:**

این صفت برای عنصری که به عنوان پیش‌فرض در حالت انتخاب شده است، قرار داده می‌شود.

◆ **VALUE:**

مقداری برای هر عنصر لیست جهت زمان ارسال به CGI در نظر گرفته می‌شود.

2014

M. Damrudi

HTML

Example

Form

```
<HTML>
<HEAD> <TITLE>Partial Order Form</TITLE> </HEAD>
<BODY BGCOLOR="#FFFFFF">
<FORM method="get" action="Test.asp"
ENCTYPE="multipart/form-data">
<P>
Greeting:
<SELECT NAME="Greeting" SIZE="1">
<OPTION VALUE="Mr.">Mr.</OPTION>
<OPTION VALUE="Mrs.">Mrs.</OPTION>
<OPTION VALUE="Ms.">Ms.</OPTION>
</SELECT> <BR>
```

2014

M. Damrudi

HTML

Example

Form

```
First Name:<INPUT TYPE="TEXT" NAME="FirstName"
SIZE="20" MAXLENGTH="25"><BR>
Last Name:<INPUT TYPE="TEXT" NAME="LastName"
SIZE="30" MAXLENGTH="35"><BR>
<BR>
What type of response would you prefer?
<INPUT TYPE="RADIO" NAME="Response"
VALUE="Phone" CHECKED="CHECKED">Phone
<INPUT TYPE="RADIO" NAME="Response"
VALUE="Fax">Fax
<BR>
<BR>
```

2014

M. Damrudi

HTML

Example

Form

I would like the following options:

 DVD Player<INPUT TYPE="CHECKBOX"
 NAME="DVD">

 CD Player <INPUT TYPE="CHECKBOX"
 NAME="CD">

 <TEXTAREA NAME="Description"></TEXTAREA>

 <INPUT TYPE="FILE" NAME="Fileupload">

 <INPUT TYPE="SUBMIT" VALUE="Submit">

 </P>
 </FORM>
 </BODY>
 </HTML>

2014
M. Damrudi

HTML

Output

Form

Greeting:

First Name:

Last Name:

What type of response would you prefer? Phone Fax

I would like the following options:
 DVD Player
 CD Player

2014
M. Damrudi

HTML

Example

Marquee

براي به حرکت درآوردن متن و تصویر:

<MARQUEE>Text is here </MARQUEE>

براي قرار دادن توضیحات در HTML:

<!-- توضیحات -->

2014
M. Damrudi

JavaScript

Introduction

JavaScript يك زبان Interpreter ي مي باشد.

زبانهاي Interpreter ي از نظر برنامه نويسي ساده تر هستند اما در اجرا کندتر مي باشند. هر بار که برنامه اجرا مي شود برنامه خط به خط تفسير مي شود.

JavaScript و Java دو زبان متفاوت مي باشند.

JavaScript توانايي صفحات HTML را افزايش مي دهد.

با JavaScript مي توان بررسي کرد کاربر اطلاعات درستي را وارد فرم کرده است يا خير.

2014
M. Damrudi

◀ توسط JavaScript مي توان صفحات وب داینامیک ایجاد کرد.
 ◀ JavaScript زبانی Case Sensitive است.
 ◀ قراردادن ؛ در انتهای اختیاری است اما اگر در یک خط بیش از یک دستور قرار گیرد باید در انتهای هر دستور ؛ را قرار داد.
 ◀ برای قراردادن JavaScript در HTML از تگ Script استفاده می شود. تگ script دارای خصوصیت language است که مقدار آنرا برابر با javascript است.

```

<script language="javascript">
</script>

```

```

<html>
<body>
<script >
document.write("Hello World!")
</script>
</body>
</html>

```

❖ document.write دستوری است که برای نوشتن خروجی روی صفحه به کار می رود.

JavaScript

Comment

// single line comment

1- توضیحات تک خطی

```
/*  
Multiline comment  
*/
```

2- توضیحات چندخطی

2014

M. Damrudi

JavaScript

محل Script

Script در head:

اسکرپت‌ها زمانی اجرا می‌شوند که یا صدا زده شوند و یا `event` رخ دهد. در این حالت Script در `head` قرار می‌گیرد.

Script در body:

در `body`، اسکرپت‌های محتویات صفحه را درست می‌کند.

0 استفاده از یک فایل اسکرپت `external`:

```
<script src="test.js"></script>
```

فایل `external` با پسوند `.js` و بدون تگ `Script` ذخیره می‌شود.

2014

M. Damrudi

o متغیرها در JavaScript به صورت ضمنی Implicit تعریف می‌شوند به این معنی که نوع آن‌ها مشخص نمی‌شود.

```
var نام متغیر;
var x;
x=12;
or
var t="Hello"
```

❑ در صورتی که متغیر در یک function تعریف شود، تنها در همان function می‌توان به آن دسترسی داشت.

❑ در صورتی که متغیر در خارج از تمام functionها تعریف شود، تمام Functionهای آن صفحه می‌توانند به آن دسترسی داشته باشند.

JavaScript

operators

Logical NOT (!)
Less than (<)
Greater than (>)
Less than or equal to (<=)
Greater than or equal to (>=)
Equality (==)
Inequality (!=)
Logical AND (&&)
Logical OR (||)

2014

M. Damrudi

JavaScript

Arithmetic Operators

Operator	Description	Example	Result
+	Addition	x=2 y=2 x+y	4
-	Subtraction	x=5 y=2 x-y	3
*	Multiplication	x=5 y=4 x*y	20
/	Division	15/5 5/2	3 2.5
%	Modulus (division remainder)	5%2 10%8 10%2	1 2 0
++	Increment	x=5 x++	x=6
--	Decrement	x=5 x--	x=4

2014

M. Damrudi

JavaScript

Assignment Operators

Operator	Example	Is The Same As
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
=	x=y	x=x*y
/=	x/=y	x=x/y
%=	x%=y	x=x%y

2014

M. Damrudi

JavaScript

Comparison Operators

Operator	Description	Example
==	is equal to	5==8 returns false
===	is equal to (checks for both value and type)	x=5 y="5" x==y returns true x===y returns false
!=	is not equal	5!=8 returns true
>	is greater than	5>8 returns false
<	is less than	5<8 returns true
>=	is greater than or equal to	5>=8 returns false
<=	is less than or equal to	5<=8 returns true

2014

M. Damrudi

Operator	Description	Example
&&	and	x=6 y=3 (x < 10 && y > 1) returns true
	or	x=6 y=3 (x==5 y==5) returns false
!	not	x=6 y=3 !(x==y) returns true

```

if (condition)
{
code to be executed if condition is true
}

```

این دستور با حروف کوچک نوشته می شود و با حروف بزرگ خطا است. ↩

JavaScript



if

```
<script>
//Write a "Good morning" greeting if
//the time is less than 10
var d=new Date()
var time=d.getHours()
if (time<10)
{
document.write("<b>Good morning</b>")
}
</script>
```

2014

M. Damrudi

JavaScript

if ... else

```
if (condition)
{
code to be executed if condition is true
}
else
{
code to be executed if condition is not true
}
```

2014

M. Damrudi

JavaScript



if

```
<script>
//Write "Lunch-time!" if the time is 11
var d=new Date()
var time=d.getHours()
if (time==11)
{
  document.write("<b>Lunch-time!</b>")
}
</script>
```

2014

M. Damrudi

JavaScript



if ... else

```
<script>
var d = new Date()
var time = d.getHours()
if (time < 10)
{
  document.write("Good morning!")
}
else
{
  document.write("Good day!")
}
</script>
```

2014

M. Damrudi

JavaScript

if ... else if ...else

```
if (condition1)
{
code to be executed if condition1 is true
}
else if (condition2)
{
code to be executed if condition2 is true
}
else
{
code to be executed if condition 1 & 2 are not true
}
```

2014

M. Damrudi

JavaScript



if ... else if ...else

```
<script>
var d = new Date()
var time = d.getHours()
if (time<10)
{ document.write("<b>Good morning</b>") }
else if (time>10 && time<16)
{ document.write("<b>Good day</b>") }
else
{ document.write("<b>Hello World!</b>") }
</script>
```

2014

M. Damrudi

JavaScript

switch

```
switch(n)
{
  case 1:
    execute code block 1
    break
  case 2:
    execute code block 2
    break
  default:
    code to be executed if n is different from case 1 and 2
}
```

◆ دستور switch در JavaScript مانند C است. برای هر case باید break قرار داد تا پس از اجرای دستور از switch خارج شود.

2014

M. Damrudi

JavaScript



switch

```
<script>
//Note that Sunday=0, //Monday=1, Tuesday=2, etc.
var d=new Date()
theDay=d.getDay()
switch (theDay)
{ case 5: document.write("Finally Friday")
  break
  case 6: document.write("Super Saturday")
  break
  case 0: document.write("Sleepy Sunday")
  break
  default: document.write("which day?") }
</script>
```

2014

M. Damrudi

◆ اغلب براي اطمینان از آنکه اطلاعات از سمت کاربر می آید ، به کار می رود.

◆ در این حالت تنها دکمه OK روی Box وجود دارد.

```
alert("SomeText")
```

◆ زمانیکه می خواهید کاربر اطلاعاتی را تایید کند استفاده می شود.

◆ در این حالت دکمه OK و Cancel روی Box وجود دارد.

◆ اگر OK زده شود مقدار true و اگر Cancel زده شود مقدار false را برمی راند.

```
confirm("SomeText")
```

JavaScript

Prompt Box

◆ زمانیکه می خواهید کاربر اطلاعاتی را وارد کند از این **Box** استفاده می شود.

◆ در این حالت دکمه **OK** و **Cancel** روی **Box** وجود دارد، اگر کاربر مقداری وارد کرده باشد و **OK** کلیک شود، **Box** مقدار وارد شده را برمی گرداند، اگر **Cancel** زده شود، مقدار **null** را برمی گرداند.

```
prompt("SomeText","defaultValue")
```

2014

M. Damrudi

JavaScript

for Loop

```
for (var=startvalue;var<=endvalue;var=var+increment) {  
  //code to be executed  
}
```

★ این حلقه زمانی به کار می رود که دقیقاً مشخص باشد چند بار نیاز است حلقه تکرار گردد.

★ پارامتر افزایشی می تواند کاهششی هم باشد.

2014

M. Damrudi

JavaScript

Example

for Loop

```

<html>
<body>
<script>
var i=0
  for (i=0;i<=10;i++) {
    document.write("The number is " + i)
    document.write("<br>")
  }
</script>
</body>
</html>

```

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9
The number is
10

2014
M. Damrudi

JavaScript

while Loop

```

while (var<=endvalue) {
//code to be executed
}

```

★ تا زمانیکه شرط حلقه برقرار باشد ، دستورات اجرا می شوند.

2014
M. Damrudi

JavaScript

Example

while Loop

```
<html>
<body>
<script>
  var i=0
  while (i<=10) {
    document.write("The number is " + i)
    document.write("<br>")
    i=i+1
  }
</script>
</body>
</html>
```

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9
The number is
10

2014

M. Damrudi

JavaScript

do while Loop

```
do {
  //code to be executed
} while (var<=endvalue)
```

☆ مشابه **while** می باشد با این تفاوت که شرط حلقه در انتهای حلقه قرار می گیرد.

☆ اگر شرط حلقه **true** نباشد، یک بار حلقه اجرا می شود.

2014

M. Damrudi

JavaScript

Example

do while Loop

```

<html>
<body>
<script>
var i=0
do {
document.write("The number is " + i)
document.write("<br>")
i=i+1
} while (i<0)
</script>
</body>
</html>

```

The number is 0

2014
M. Damrudi

JavaScript

break-continue

break ◆ باعث می‌شود که از حلقه خارج شده و اجرای برنامه از بعد از حلقه ادامه یابد.

continue ◆ باعث می‌شود که اجرای دور بعدی حلقه با مقدار جدید انجام گیرد و دستورات بعد **continue** در دور جاری حلقه در نظر گرفته نمی‌شود.

2014
M. Damrudi

JavaScript**break**

```
<html>
<body>
<script>
var i=0
for (i=0;i<=10;i++) {
if (i==3){break}
document.write("The number is " + i)
document.write("<br>") }
</script>
</body>
</html>
```

The number is 0
The number is 1
The number is 2

2014

M. Damrudi

JavaScript**continue**

```
<html>
<body>
<script>
var i=0
for (i=0;i<=10;i++) {
if (i==3){continue}
document.write("The number is " + i)
document.write("<br>")
}
</script>
</body>
</html>
```

The number is 0
The number is 1
The number is 2
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9
The number is 10

2014

M. Damrudi

JavaScript

for In

```
for (variable in object) {  
  //code to be executed  
}
```

این حلقه برای تمام عناصر موجود در آرایه یا **object** تکرار می‌شود. *

2014

M. Damrudi

JavaScript

Example

for In

```
<html>  
<body> <script>  
var x  
var mycars = new Array()  
mycars[0] = "Saab"  
mycars[1] = "Volvo"  
mycars[2] = "BMW"  
for (x in mycars) {  
  document.write(mycars[x] + "<br>")  
}  
</script></body>  
</html>
```

Saab
Volvo
BMW

2014

M. Damrudi

Function بخش مهمی از JavaScript است.

JavaScript دارای توابعی است که به آنها Built-in گفته می‌شود. هر یک از این توابع عملیاتی را انجام می‌دهند.

JavaScript امکان نوشتن توابع را به برنامه‌نویس می‌دهد. این توابع اگر صدا زده شوند و یا eventی اتفاق افتد، اجرا می‌شوند.

یک Function از هرجایی داخل صفحه می‌تواند صدا زده شود. اگر تابع در یک فایل با پسوند .js ذخیره شود، نیز به صورت external به کار می‌رود.

Function در داخل HEAD نوشته می‌شود.

حالت کلی Function به صورت زیر می‌باشد:

```
function function_name(var1,var2,...){
.
.
JavaScript statements go here
.
.
} // end of function
```

Function می‌تواند یک مقدار را با دستور return برگرداند. برای آنکه مقدار بیشتری از یک تابع بازگشت داده شود از متغیر استفاده می‌شود.

JavaScript



Function

```
<head><script>
function add_two_num( a, b) {
var sum = a + b; return sum; }
</script> </head>
<body>
<script>
var x = 5, y = 3, total = 0;
total = add_two_num( x , y );
alert(total);
</script>
</body>
```

مقدار بازگشتی تابع در متغیر total ریخته می‌شود.

2014

M. Damrudi

JavaScript



Function

```
<html>
<head> <script>
function displaymessage()
{ alert("Hello World!") }
</script> </head>
<body>
<form>
<input type="button" value="Click me!"
onclick="displaymessage()" >
</form>
</body>
</html>
```

onclick یکی از event های button است.

2014

M. Damrudi

JavaScript

Event

☞ با استفاده از **event** ها می توان **function** ها را صدا زد. به عنوان مثال **event** می تواند یکی از موارد زیر باشد:

- ☞ **onload**
- ☞ **onUnload**
- ☞ **onFocus**
- ☞ **onBlur**
- ☞ **onChange**
- ☞ **onSubmit**
- ☞ **onMouseOver**
- ☞ **onMouseOut**
- ☞ **onClick**

2014

M. Damrudi

JavaScript

onLoad Event

☞ زمانی که صفحه بارگذاری شد اتفاق می افتد. زمانی که کاربر وارد صفحه می شود این **event** اتفاق می افتد.

☞ این **event** اغلب برای بررسی نوع **browser** کاربر ، **version browser** و بارگذاری **version** درست **browser** با توجه به اطلاعات می باشد.

☞ برای برقراری ارتباط با **cookie** ها و مقاردهی براساس ورود کاربر استفاده می شود.

```
<BODY onLoad="alert('Welcome to our website!');">
```

2014

M. Damrudi

JavaScript

onUnLoad Event

⌚ زمانی که کاربر وارد صفحه دیگری می‌شود و کاربر صفحه را ترک می‌کند این event اتفاق می‌افتد.

⌚ برای برقراری ارتباط با cookieها و مقادیردهی براساس خروج کاربر استفاده می‌شود.

```
<BODY onUnload="alert('Thanks for checking out our site!');">
```

2014

M. Damrudi

JavaScript

onFocus Event

⌚ این event زمانی اتفاق می‌افتد که کاربر با استفاده از tab و یا کلیک وارد یکی از موارد زیر می‌شود:

- password field
- text field
- textarea
- FileUpload field

```
<INPUT TYPE="TEXT" NAME="Month"
onFocus="window.status=('Please enter the Month as two
digits 01 through 12'); return true;">
```

2014

M. Damrudi

JavaScript

onBlur Event

این event زمانی اتفاق می افتد که کاربر یکی از فیلدهای زیر را ترک کند:

- password field
- text field
- textarea
- FileUpload field

```
<INPUT TYPE="TEXT" NAME="Month"
onBlur="window.status=('Do you not care about the value
of the Month!'); return true;">
<INPUT TYPE="TEXT" NAME="Year">
```

2014

M. Damrudi

JavaScript

onChange Event

این event زمانی اتفاق می افتد که کاربر یکی از فیلدهای زیر را ترک کند و مقدار آن فیلد تغییر کرده باشد: (بهتر است که onBlur و onChange همزمان استفاده نشوند)

- password field
- text field
- textarea
- FileUpload field

```
<INPUT TYPE="TEXT" NAME="Month"
onChange="window.status=('The value of the Month
Changed!!!!'); return true;" > <INPUT TYPE="TEXT"
NAME="Year">
```

2014

M. Damrudi

JavaScript

onSubmit Event

☞ برای ارزیابی تمام فیلدهای فرم پیش از ارسال شدن استفاده می‌شود.

```
<FORM action=http://www.xnu.com/formtest.asp  
onSubmit="return checkform();">
```

☞ اگر تابع `checkform` مقدار `false` را برگرداند، `submit` لغو می‌شود و اگر مقدار `true` برگرداند اطلاعات فرم به آدرس `action` ارسال می‌شود.

2014

M. Damrudi

JavaScript

onMouseOver Event

☞ زمانی که کاربر موس را روی تصویر یا `link` قرار دهد، این `event` اتفاق می‌افتد.

☞ اغلب برای ایجاد دکمه‌های انیمیشنی استفاده می‌شود.

```
<A href="http://synergy.simplenet.com/simpsons/"  
onMouseOver="window.status='The best Simpsons  
Webpage on the NET!'; return true;">Click for  
information on the Simpsons.</A>
```

2014

M. Damrudi

JavaScript

onMouseOut Event

زمانی که کاربر موس را از روی تصویر یا link بردارد ، این event اتفاق می افتد.

اغلب برای ایجاد دکمه های انیمیشنی استفاده می شود.

```
<a href="http://www.Test.com"
onmouseover="alert('An onMouseout event'); return
false;">
</a>
```

2014

M. Damrudi

JavaScript

onClick Event

زمانی که کاربر یک button ، checkbox و radio کلیک کند ، این event اتفاق می افتد.

```
<INPUT TYPE="Button" VALUE="Click Me"
onClick="window.alert('You Clicked me');">
```

2014

M. Damrudi

JavaScript

معتبر سازی فرم

```
<Html> <head>
<script type="text/javascript">
function validate_Empty(field,alerttxt)
{ with (field)
  { if (value==null || value=="") { alert(alerttxt); return false ;}
    else { return true }
  } }
function validate_form(thisform)
{ with (thisform) {
if (validate_Empty(UID," Empty UID!") == false) {UID.focus(); return false;}
if (validate_Empty(pwd," Empty pwd!") == false) {pwd.focus(); return false;}
if (validate_Empty(pwdr," Empty pwdr!") == false) {pwdr.focus(); return false;}
if (validate_Empty(ZCode," Empty Zip Code!") == false) {ZCode.focus(); return false;}
if (Equal_Pass(pwd,pwdr,"Password1 <> Password2") == false) { pwdr.focus(); return false; }
} }
function Equal_Pass(field1,field2,alerttxt)
{ if (field1.value != field2.value) { alert(alerttxt); return false ;}
  else { return true } }

```

2014

M. Damrudi

JavaScript

معتبر سازی فرم

```
</script> </head>
<BODY>
<Form method=GET Action="http://Localhost/formpros.asp"
onsubmit="return validate_form(this)">
<P>UserID: <input type =text name=UID size=18> <FONT
COLOR="red"> * </FONT>
<P>Password: <input type =password name=pwd size=20> *
<P>ReType Password: <input type =password name=pwdr size=20> *
<P>Full Name: <input type =text name=FName size=18>
<P>Zip Code: <input type =text name=ZCode size=18> <FONT
COLOR="red"> * </FONT>
<p> <input type =reset value ="Clear" >
<input type =submit value = Send onclick="Validation( )">
</form>
</BODY>
</Html>
```

2014

M. Damrudi