

MYSQAL

PHP

JAVASCRIPT

HTML

NETWORKS

www.ssampad.ir



طراحی وب

محسن نصرتی • محمدرضا جوان



۱. شبکه ۲

۱-۱. چطور می توانم وارد این اجتماع عظیم شوم. ۲

۲-۱. اینترنت چیست؟ ۲

ارتباطات کامپیوتری چگونه بوجود آمدند؟ ۳

پروتکل TCP/IP ۸

بسته های داده ای چگونه مقصد خود را تشخیص می دهند و بازسازی می شوند؟ ۱۰

مکانیزم اولیه ای که برای مشخص کردن کامپیوتر ها ۱۱

روش آدرس دهی IP بر روی کامپیوتر ۱۶

ایجاد اولین شبکه با استفاده از دو کامپیوتر ۱۸

Ping ۱۸

روش شبکه کردن دو تا کامپیوتر بدون سیم ۱۹

۳-۱. مراحل پیاده سازی یک سایت ۲۲

نقشه سایت ۲۲

ثبت نام دامنه ۲۳

چگونه با استفاده از نام دامنه آدرس سایت اینترنتی ما مشخص می شود؟ ۲۴

پیدا کردن ip از طریق نام سایت در ویندوز با استفاده از nslookup ۲۷

مراحل ثبت نام دامنه ۲۸

خرید هاست ۳۱

Error! Bookmark not defined. ساخت اولین صفحه از سایت

۳۲..... 1. برنامه نویسی HTML

۳۳ ۲. برنامه نویسی HTML

۳۳ ۲-۱. هدف از ایجاد وبسایت شخصی

۳۳ ۲-۲. چرا HTML؟

۳۴..... ۲-۳. ساخت اولین صفحه در وب

۳۶ تگ

۳۶ ساختار پایه کد

۳۷ ساختار نوشتاری

۴۳..... ۲-۴. صفحه آلبوم عکس

۴۴ تگ

۴۸ ۲-۵. صفحه درباره من

۵۱..... ۲-۶. صفحه پیوند به دوستان

۵۳ ویژگی target

۵۴ پیوند داخلی

۵۵ ۲-۷. صفحه تماس با من

۵۶..... ۲-۸. صفحه خانه

۶۰	 صفحه ورود اعضا	۹-۲
۶۲	 صفحه ثبت نام خبرنگار	۱۰-۲
۶۳	 ورودی های متنی	
۶۴	 ورودی گزینه ای	
۶۴	 ورودی انتخابی	
۶۵	 ورودی توضیحی	
۶۵	 ورودی تاییدی	
۶۶	 دکمه ها	
۶۷	 آشنایی با CSS	۳
۶۷	 ۱- نیاز به CSS از کجا آغاز شد	۳-۱
۶۷	 ساختار صفحات وب	
۶۸	 اجزای صفحات وب	
۶۸	 ۲-۳ CSS چیست	
۶۸	 چگونه از دستورات CSS در صفحات HTML می توان استفاده کرد	
۷۰	 مزیت های عمده استفاده از CSS	
۷۰	 نحو CSS	
۷۱	 روش اول: با استفاده از ClassSelector	

۷۳ ID Selector با استفاده از

۷۵ ۳-۳. قواعد کار در css

۷۵ قواعد مربوط به رنگ و پس زمینه صفحه

۷۵ Background-image

۷۶ Background-repeat

۷۶ Repeat

۷۷ No-repeat

۷۷ Repeat-x

۷۷ Repeat-y

۷۷ Background-attachment

۷۸ Scroll

۷۸ fixed

۷۸ قواعد مربوط به لینکها

۸۰ قواعد مربوط به متن

۸۳ 4. برنامه نویسی JAVASCRIPT

۸۳ ۴-۱. ساختار زبان های برنامه نویسی تحت وب

۸۵ ۴-۲. برای نوشتن برنامه به زبان جاوا اسکریپت از کجا باید شروع کنیم

۳-۴. شی گرای در جاوا اسکریپت ۸۷

آشنایی با شی document ۸۹

متد document.write() ۸۹

۴-۴. بازی با تصاویر ۹۳

تعریف توابع ۹۳

متد () getElementById ۹۴

شکل دستور ۹۵

رویدادها ۹۶

۴-۴-۱. آشنایی با انواع رویدادها زبان جاوا اسکریپت ۹۸

رویدادهای mouse ۹۸

رویدادهای Keyboard ۹۹

رویدادهای مرورگر ۹۹

4-5. ساخت منوی پویا (rollover image) ۱۰۰

آلبوم تصادفی - نمایش یک تصویر بصورت اتفاقی ۱۰۲

آرایه‌ها ۱۰۳

زمان بندی ۱۰۵

4-6. معتبر سازی فرمها ۱۰۷

لزوم استفاده از معتبر سازی فرمها ۱۰۷

۱۱۲	۷-۴. جعبه‌های پیام
۱۱۳	history و location شی
۱۱۵	۸-۴. JQuery چیست؟
۱۱۵	۹-۴. چگونه کتابخانه JQuery را به برنامه خود اضافه کنیم
۱۱۶	۱۰-۴. چگونه با JQuery برنامه نویسی کنیم
۱۱۷	\$(document).ready
۱۱۹	۱۱-۴. آشنایی با افکتها
۱۲۰	Toggle متد
۱۲۴	۵. برنامه نویسی PHP
۱۲۴	2. برنامه نویسی PHP
۱۲۵	۱-۵. مقدمات برنامه‌نویسی PHP
۱۲۵	۲-۵. از کجا شروع کنیم؟
۱۲۶	۳-۵. دریافت اطلاعات فرم
۱۲۸	آرایه
۱۲۸	آرایه‌های از پیش تعریف شده
۱۳۰	foreach حلقه
۱۳۱	۴-۵. ایجاد قسمت ورود اعضا

شرط if ۱۳۲

۵-۵. تکمیل قسمت ورود اعضا ۱۳۵

ایجاد یک فایل حاوی اطلاعات ۱۳۵

دسترسی به فایل ۱۳۶

خواندن از فایل ۱۳۷

نوشتن بر روی فایل ۱۳۸

بستن فایل ۱۳۹

مشخص کننده انتهای فایل ۱۳۹

توابعی برای کار با رشته‌ها ۱۴۰

strlen() ۱۴۰

substr() ۱۴۰

strpos() ۱۴۲

strstr() ۱۴۳

explode() ۱۴۳

implode() ۱۴۴

حلقه while ۱۴۵

۵-۶. ایجاد فرم ثبت نام ۱۴۷

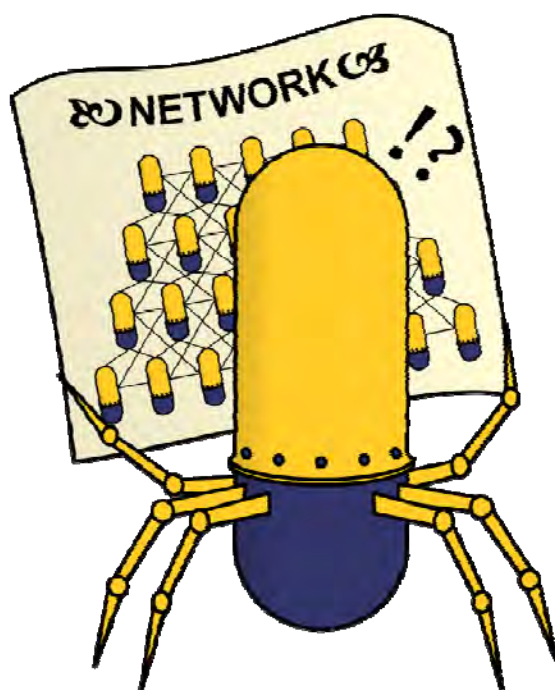
حلقه for ۱۴۷

۱۴۸	date()
۱۵۰	۵-۷. افزودن قسمت‌ها جدید سایت
۱۵۰	دستور include()
۱۵۲	۵-۸. توابع در PHP
۱۵۵	۵-۹. نشست در PHP
۱۶۰	۵-۱۰. AJAX چیست؟
۱۶۰	AJAX چگونه عمل می‌کند؟
۱۷۱	۳. برنامه نویسی MySQL
۱۸۴	۴. پیوست‌ها
۱۸۵	مرحله ۱: نصب XAMPP
۱۹۰	مرحله ۲: نحوه اجرای برنامه های php

به نام خدا

اینترنت یک اشتراک است که می‌توانیم آن را موجودی دیجیتالی فرض کنیم که به درخواست تجربه انسان در حال تکامل است. اما صرف‌نظر از مفهوم اینترنت که در ادامه این کتاب به آن می‌پردازیم، لازم است پیش از اینکه بخواهیم وارد این فضای دیجیتالی بشویم اندکی از آنچه که ما را در این مسیر قرار داده است، تامل کنیم. لازم است این حقیقت را باور کنیم که اینترنت در اصل یک اجتماع شگفت‌آور از انسان‌هاست که روزانه به این اجتماع اضافه می‌گردد و اگر ما بخواهیم در این اجتماع شگفت‌آور که در آن از هر صنفی و قشری جمع هستند به تعامل و گفتگو بپردازیم و بیننده صرف نباشیم، زبان و ابزار این اجتماع شگفت‌آور را باید بیاموزیم. البته این نکته را باید متذکر شد که همانند همه انواع ارتباطاتی که تا به حال در آن شرکت کرده‌اید و یا دیده‌اید، شاید هضم کامل این زبان و این ارتباط برای همه مشکل باشد. اما باید بدانید که حتی با اندکی اطلاعات نیز می‌توان وارد این اجتماع شد و به گفتگو پرداخت. در این کتاب قصد داریم پا را فراتر گذاشته و از این اندک اطلاعات مطالبی عمیق و کاربردی صرفاً جهت ورود به دنیای شگفت‌آور وب قدم بگذاریم.

شبکه



شبکه

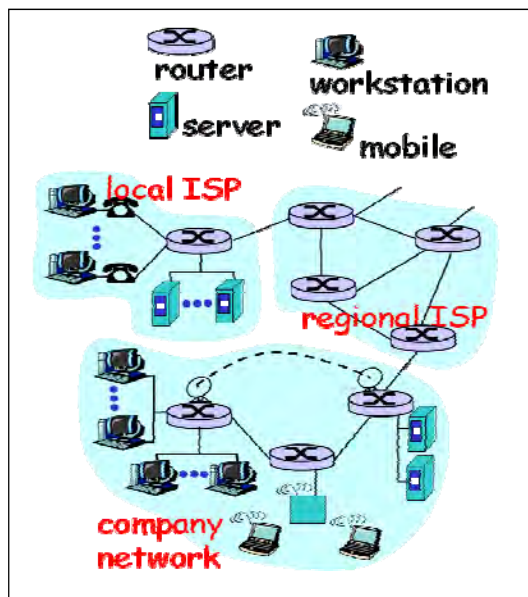
۱-۱. چطور می توانم وارد این اجتماع عظیم شوم.

برای پیوستن به اینترنت لازم است ابتدا تعریفی از آن ارائه گردد. برای درک و مفهوم این دنیای شگفت انگیز و آشنایی با کلمات آن لازم است در ابتدا آن مفاهیم و اصطلاحات را یادآور شویم و سپس پا را فراتر گذاشته و به صورت عملی و اجرایی به ساخت یک وب سایت شخصی با موضوع مورد نظر بپردازیم و خود را وارد این عرصه نماییم.

بنابراین در ابتدا توضیحی از زیرساخت های وب بیان می کنیم زیرا تا این اصطلاحات و کلمات آن برای ما نامفهوم باشد نمی توانیم درک درستی از مطالب آن ارائه دهیم.

۱-۲. اینترنت چیست؟

اینترنت قوی ترین ابزاری است که تاکنون برای مبادله ی اطلاعات اختراع شده است. اگر خواهیم تعریف ساده ای از اینترنت ارائه دهیم، اینترنت یک شبکه بزرگ از مجموعه ای از کامپیوترها است که توسط یک قانون مشترک با یکدیگر ارتباط دارند. باید بدانید که هر کامپیوتری که به این مجموعه متصل می شود به عنوان بخش کوچکی از اینترنت محسوب می شود.



شکل ۱-۱: نمایی کلی از شبکه‌های وب

اینترنت یک بخش مستقل نیست که دیگران کامپیوترهایشان را به آن متصل کنند. بلکه اینترنت خود حاصل بهم پیوستن این کامپیوترهاست. بنابراین اینترنت به هیچ سازمان و موسسه خاصی متعلق نیست. (البته سازمانهایی جهت استاندارد سازی اینترنت تشکیل گردیده‌اند) تا اینجا ما جهت تعریف اینترنت از واژه شبکه‌های کامپیوتری استفاده کرده‌ایم و در مورد این واژه شاید قبلاً هم شنیده باشید، اما توضیحی از این پدیده و تکنولوژی که منجر به پیدایش اینترنت گردید حرفی نزدیم. در این فصل قصد داریم در ابتدا توضیحی در مورد این که این امکانات از کجا آمده و بر چه مبنایی قوانین آن تعریف شده‌اند و اینکه سختی‌های این ارتباطات که تشکیل چنین شبکه عظیمی را می‌دهد چگونه است.

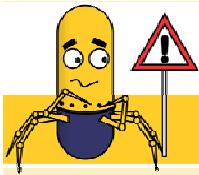
ارتباطات کامپیوتری چگونه بوجود آمدند؟

داستان از اینجا شروع می‌شود که اتحاد جماهیر شوروی سابق در سال ۱۹۵۷ موشکی به نام اسپونیک Sponik را به فضا می‌فرستد و نشان می‌دهد که دارای قدرتی است که می‌تواند شبکه‌های

ارتباطی آمریکا را توسط موشک‌های بالستیک و دوربرد خود از بین ببرد. به دلیل رقابت تنگاتنگ نظامی میان آمریکا و شوروی، اداره دفاع آمریکا به دستور آیزنهاور (رئیس‌جمهور وقت ایالات متحده) اقدام به تشکیل آژانس تحقیقاتی پروژه‌های پیشرفته (آرپا) نمود. هدف از تاسیس چنین موسسه‌ای پژوهش و آزمایش برای پیدا کردن روشی بود که بتوان از طریق خطوط تلفنی کامپیوترها را به هم مرتبط کرد. به طوریکه چندین کاربر بتوانند از یک خط ارتباطی مشترک استفاده کنند. در اصل شبکه‌ای بسازند که در آن داده‌ها به صورت اتوماتیک بین مبدا و مقصد جا به جا شوند. در اصل هدف اصلی ARPA ایجاد شبکه اینترنتی نبود و فقط یک حرکت احتیاطی در مقابل حمله احتمالی موشک‌های اتمی دوربرد بود.

در کل در مورد پیدایش اینترنت باید بدانید که اینترنت نتیجه مجموع کار و تلاش عده کثیری از دانشمندان است که البته بیشترین عامل آن در اوایل دهه ۶۰ میلادی طی تحقیقات و ارائه مقالاتی بود که پایه کار اینترنت امروزی را ایجاد کرد.

در این مقالات اطلاعات و داده‌ها به صورت قطعات و بسته‌های کوچکتری همانند بسته‌های نامه تشبیه شده بود که هر بسته با آدرسی که به آن اختصاص داده می‌شود به مقصد خاص در نظرگرفته شده ارسال می‌شود. با ارسال مداوم این بسته‌ها در مقصد و یکپارچه‌سازی همه محتویات آن داده‌های ما ارسال می‌گردید. البته ارتباط و ارسال این بسته‌ها در این فرضیه همانند کار سلول‌های مغزی الگو گرفته بود. به طوریکه وقتی سلول‌های مغزی از بین بروند شبکه عصبی از آنها دیگر استفاده نمی‌کند و مسیر دیگری را در مغز انتخاب می‌کند.

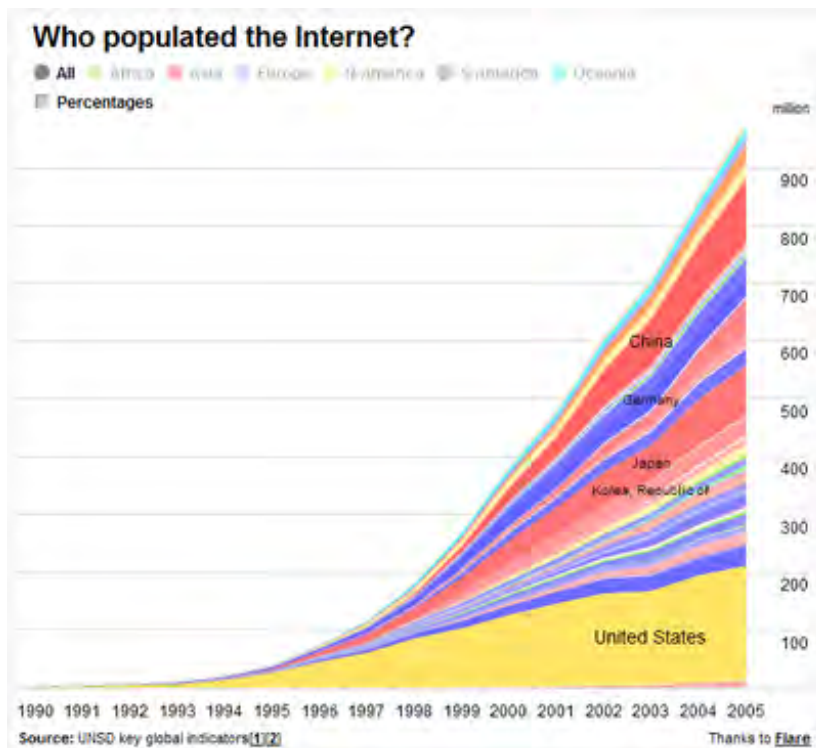


توجه!

از این دیدگاه این امکان وجود دارد که شبکه‌ای با تعداد زیادی اتصالات برای تکرار ایجاد شوند تا در صورت نابودی بخشی از آن، ارتباط شبکه ای همچنان به صورت مجموعه‌ای به هم پیوسته کار کند.

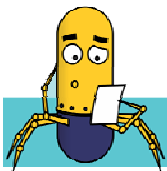
با استفاده از این ایده (بوسیله دانشمندی به نام تیلور، پیاده سازی شد.) در سال ۱۹۶۶ دو کامپیوتر را در شرق و غرب آمریکا به هم متصل گردید که با این اتصال اولین شروع (استارت) وب آغاز شد که با آزمایش‌های مختلف، بخش‌های دولتی و دانشگاهی تمایل خود را جهت اتصال به این شبکه اعلام کردند. البته از اوایل دهه ۱۹۹۰ تقریباً رشد استفاده از این شبکه همانطور که در شکل زیر ملاحظه می‌کنید به صورت تصاعدی افزایش یافت.

نمودار رشد سریع این شبکه بزرگ را همانند شکل زیر می‌توان مشاهده کرد:



شکل ۱-۲: نمودار رشد شبکه اینترنت

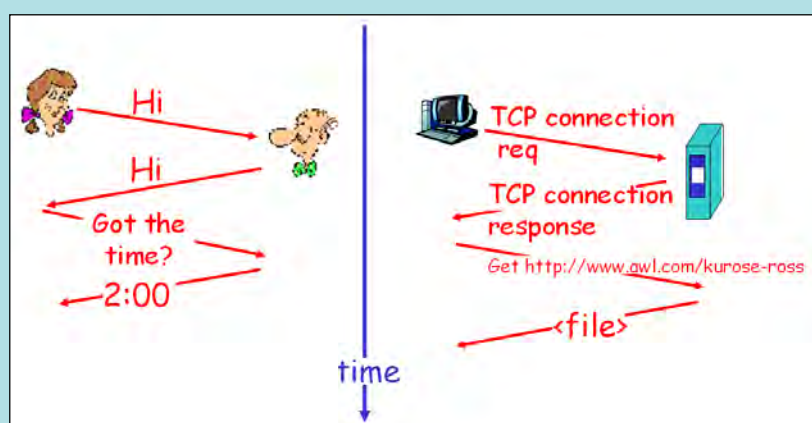
همانطور که در بالا گفتیم اینترنت براساس مجموعه‌ای از شبکه‌ها بنا شده است که این شبکه‌ها شامل انواع بسیار زیادی کامپیوتر می‌باشد. بنابراین برای ارسال اطلاعات و همچنین بهره‌برداری از این امکانات سخت افزاری و برقراری ارتباط بین اجزای شبکه‌ای نیاز به یک مجموعه از قوانین و دستورالعمل‌های مشترک دارد که به آن قوانین اصطلاحاً **پروتکل** می‌گویند.



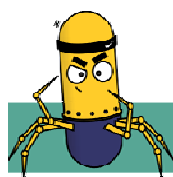
مثال

واژه پروتکل در لغت به معنای قرارداد است و به عنوان مثال ساده‌ترین نوع پروتکل‌ها

هنگامی است که ما پشت تلفن با یکدیگر صحبت می‌کنیم. زمانی که ما حرف می‌زنیم طرف مقابل گوش می‌کند و زمانی که او صحبت می‌کند ما گوش می‌دهیم. به این قرارداد جهت یک ارتباط تلفنی سالم که ما ایجاد می‌کنیم یک پروتکل می‌گویند. ما در طول روز با پروتکل‌های مختلفی برخورد می‌کنیم مثلاً همانند حاضر شدن سرکلاس و برنامه‌ای که جهت نظم شما در مدرسه اجرا می‌گردد یا روش تدریسی که یک معلم در ابتدای سال برای شما مشخص می‌کند.

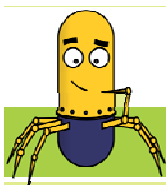


شکل ۱-۳: نمایی ساده از مفهوم پروتکل



تمرین

حال سعی کنید پروتکل‌هایی که در ارتباط روزمره خود مشاهده می‌کنید بیان کنید و



بیشتر بدانیم!

جامعه اینترنت در سال ۱۹۹۲ با هدف مدیریت و راهبری استانداردهای مرتبط به اینترنت، علم و دانش و امنیت و خط مشی تاسیس شد. در سال ۱۹۹۴، Tim Berners Lee سازمان استاندارد جهانی وب (C۳W) را با هدف هدایت اینترنت به پتانسیل موجودش با گسترش و ایجاد پروتکل‌ها و خطوط راهنمایی که رشد وب در طولانی مدت را تضمین کند تاسیس کرد.

حال لازم است به مهمترین پروتکل در شبکه بپردازیم. زیرا در اهمیت پروتکل تنها توجه به این نکته کافی است که ارتباط در اینترنت بدون آن تقریبا غیر ممکن است و اکثر سرویس‌های اینترنت تحت قوانین این پروتکل عرضه می‌شود.

پروتکل TCP/IP

این پروتکل در ابتدا به وسیله وزارت دفاع آمریکا و در سیستم عامل unix ایجاد شد و امروزه تقریبا این پروتکل کلیه رقبا را کنار زده و در اکثر شبکه‌ها اعم از کوچک و بزرگ و بوسیله کلیه سیستم عامل‌ها پشتیبانی می‌شود.

TCP/IP از دو پروتکل مهم (Transmission Control Protocol) و (Internet Protocol) IP گرفته شده است و مهمترین خصوصیات این پروتکل را می‌توان به صورت زیر نام برد.

پشتیبانی انواع شبکه

پشتیبانی انواع سیستم عامل

مسیریابی

حق انتخاب به صورت عادی و انتخابی

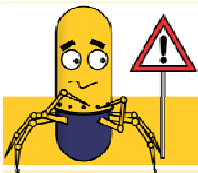
ارسال گروهی

جهت ارتباط با کامپیوتر و یا اینترنت سه نوع مخاطب در شبکه‌ها وجود دارند که با استفاده از TCP/IP می‌توانیم برای این سه نوع اطلاعات را ارسال کنیم.

ارسال برای یک نفر

ارسال برای همه تا محدوده مجاز

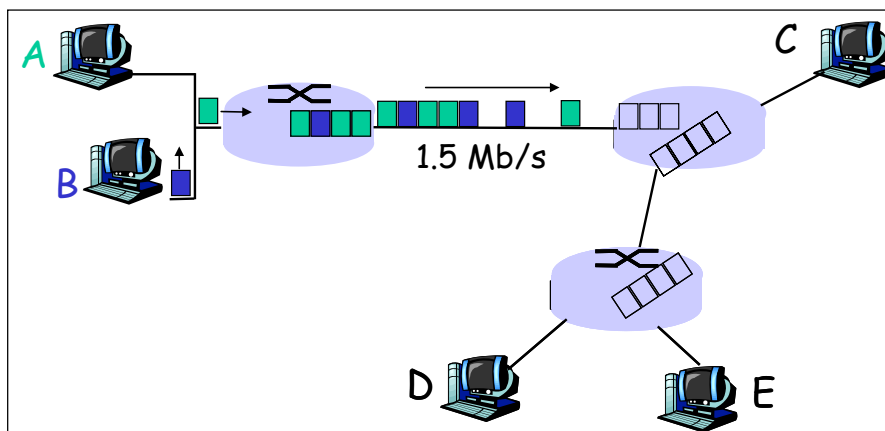
ارسال برای گروهی از استفاده کنندگان



توجه!

در ارسال اطلاعات با استفاده از پروتکل TCP/IP اطلاعات به صورت جریان ثابتی بین کامپیوترها منتقل نمی‌شود. بلکه به بسته‌هایی کوچک به نام Pocket تقسیم‌بندی می‌شوند.

مثلا اگر شما بخواهید پیامی را برای دوستتان ارسال کنید TCP ابتدا آن را به تعدادی بسته تقسیم می کند و سپس بسته ها را از طریق شبکه ارسال می کند.



شکل ۱-۴: نحوه ارسال بسته های اطلاعات

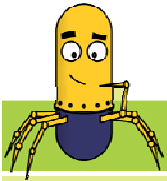
کار IP این هم است که این بسته ها را به میزبان راه دور منتقل کند و TCP که در انتهای دیگر بسته ها را دریافت و وجود خطاها را بررسی می کند و در صورتی که خطایی رخ داده باشد می تواند درخواست ارسال مجدد این بسته بخصوص را از مبدا بنماید. بعد از اینکه ارسال بسته ها به پایان رسید، این وظیفه TCP است که آنها را به ترتیب به هم متصل کرده تا پیام مورد نظر ما به صورت کامل ساخته شود.

بسته های داده ای چگونه مقصد خود را تشخیص می دهند و بازسازی می شوند؟

در قسمت قبل بیان کردیم که پیامی که ما برای دوستان ارسال می کنیم ابتدا به قسمت های کوچکتری به نام Pocket تقسیم می شود. اما مطرح نکردیم که این بسته ها چگونه مقصد خود را می یابند و چگونه دوباره پیام اصلی ما بازسازی می شود.

مکانیزم اولیه‌ای برای مشخص کردن کامپیوترها

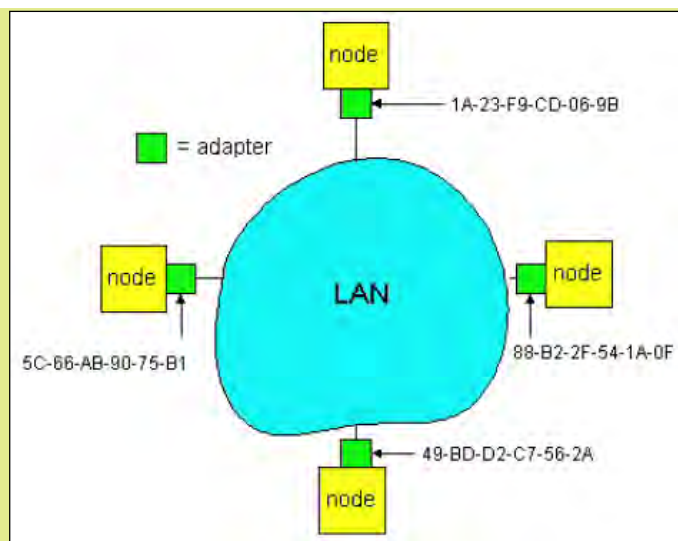
در ابتدا به نظر می‌رسید استفاده از یک شماره یکتا برای هر کامپیوتر همانند کد ملی کافی بود. به این آدرس منحصر به فرد، آدرس فیزیکی یا (Mac address) می‌گویند که این گونه آدرس‌ها نیازمند درایورهای خاصی نخواهند بود.



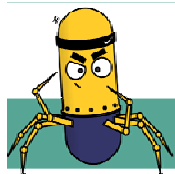
بیشتر بدانیم!

مک آدرس توسط کارخانه سازنده درون تراشه کارت شبکه قرار می‌گیرد. یک Mac address بر روی هر کارت شبکه همواره دارای طولی مشابه و یکسان می‌باشد که بر مبنای شانزده^۱ و به اندازه ۶ بایت می‌باشد که در نتیجه توانایی آدرس‌دهی²⁴⁸ کامپیوتر مختلف را داراست. هر شرکت سازنده کارت شبکه، محدوده خاصی از اعداد این بازه را می‌خرد و آنها را روی کارت شبکه خود ثبت می‌کند. به منظور برخورد مشکلات آدرس‌ها یکسان و مشابه و اختلال در شبکه و ارسال بسته‌های اطلاعاتی، گروهی به نام IEEE بر آن شد تا هر Mac address را به دو بخش مساوی تقسیم کند و اولین بخش آن به منظور شناسایی تولید کننده کارت و دومین بخش به تولیدکنندگان اختصاص داده شده تا آنان یک شماره سریال را در آن درج نمایند.

¹ Hex

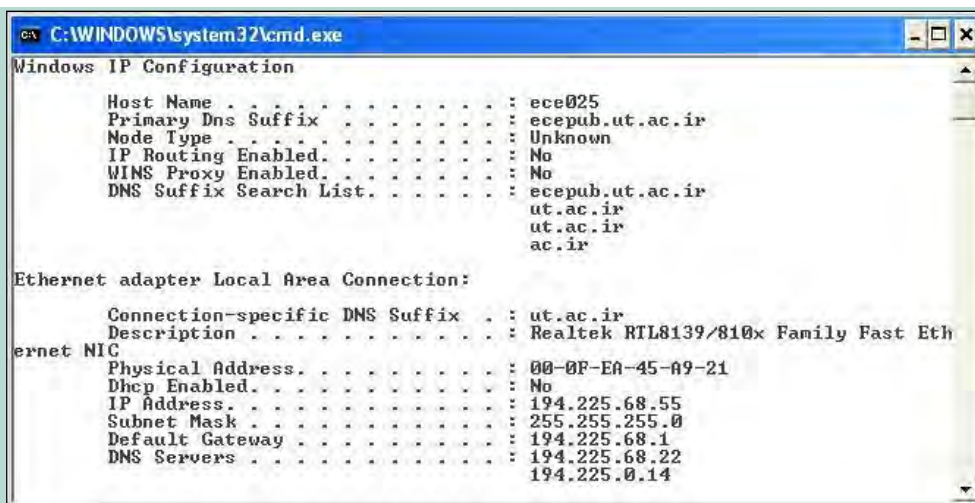


شکل ۱-۵: آدرس‌دهی توسط مک آدرس



تمرین

اگر بخواهید این آدرس را در ویندوز مشاهده کنید می‌توانید از دستور `ipconfig -all` استفاده کنید که با تایپ این دستور در Command Prompt می‌توانید مشاهده کنید.



شکل ۶-۱: نمایشی از کنترل Mac Address

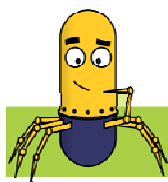
با روش بالا Mac address کامپیوتر خود را کنترل کنید.

چگونه سرویس‌های اینترنتی می‌توانند تنها با داشتن آدرس فیزیکی یا Mac address بفهمند که اطلاعات را به کجا بفرستند و با تغییر و جابه جایی کامپیوترها و تغییر آدرس بتوانند آدرس‌ها را پیدا کنند و بسته‌ها را به مقصد برسانند؟

برای روشن‌تر شدن موضوع فرض کنید می‌خواهید نامه‌ای را پست کنید برای پست نامه لازم است آدرس کشور، استان، شهر، کوچه، پلاک و کدپستی گیرنده را روی پاکت خود بنویسید. ولی برای ارسال غیرپستی توسط شخص و یا دوستان نیاز به نوشتن آدرس دقیق پستی وجود ندارد ولی برای ارسال پستی باید بدانید پستی به آدرس دقیق پستی نیاز دارد. حال فرض کنید ما نامه‌ای را بدون آدرس پستی ارسال کنیم و از پستی بخواهیم نامه را به دست فلان دوست ما برساند.

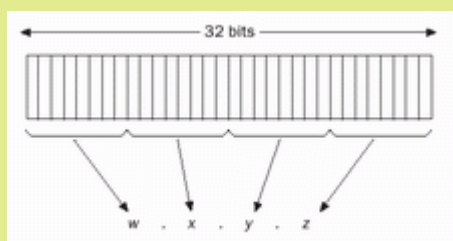
طبیعتاً این کار برای پستیچی غیرممکن است. درحقیقت آدرس منطقی همان آدرس پستی است که هر شخص و نیز پستیچی می‌تواند این نامه را به دوست شما برساند. اینجا بود که IP Address اینترنت را نجات داد.

دراصل پروتکل IP بود که جهت شناسایی هر کامپیوتر مورد استفاده قرار گرفت، این آدرس منطقی (آدرس پستی) از ۴ عدد تشکیل می‌شود که این اعداد در بازه صفر تا ۲۵۵ تغییر کرده است و توسط نقطه از هم جدا می‌شوند. همانند ۱۲۱.۵۱.۲۷.۲۱۷



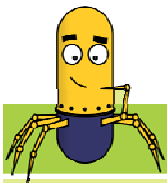
بیشتر بدانیم!

IP آدرس‌ها برای قابل فهم بودن افراد، معمولاً به صورت دهدهی که به صورت چهار بخش که با نقطه از هم جدا شده‌اند، بیان می‌شوند. اما کامپیوترها با شکل باینری (دودویی) این اعداد سروکار دارند.



شکل ۱-۷: IP آدرس‌ها به صورت باینری

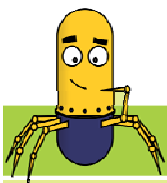
چهار عدد در یک آدرس چون هر عدد به تنهایی دو بایت جا می‌گیرد. در بازه (0,255) بنابراین آدرس IP از اعداد ۳۲ بیتی تشکیل شده است که مجموعاً می‌توان ۲ به توان ۳۲ یعنی ۴۲۹۴۹۶۷۲۹۶ عدد منحصر به فرد داشته باشیم که حدود ۴۰۳ میلیارد آدرس IP منحصر به فرد می‌توانیم معین کنیم.



بیشتر بدانیم!

البته با آمدن IPv6 در مبنای ۱۶ نوشته می‌شوند و به جای آدرس دهی ۳۲ بیتی از آدرس دهی ۱۲۸ بیتی استفاده می‌کنند. با این روش ترکیب آدرس بیشتری خواهیم داشت.

اعداد بین نقطه‌ها را می‌توان به دو قسمت تقسیم کرد. شبکه‌ها و نودها^۲ که اولین عدد (از سمت راست) به معنای شناسایی شبکه‌ای است که آن کامپیوتر به آن متصل می‌باشد.



بیشتر بدانیم!

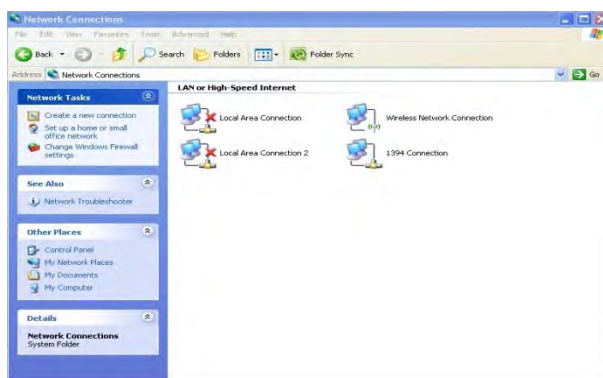
عدد اول IP که از ۱ تا ۱۲۶ می‌باشد، برای کمپانی‌های عظیم بین‌المللی که دارای

² Nodes

شبکه‌های بزرگ می‌باشند و سه عدد بعدی، برای نام سایت‌ها استفاده می‌شود یعنی می‌توانیم بگوییم، هر کمپانی می‌تواند ۱۶۷۷۷۲۱۴ نام سایت زیر مجموعه داشته باشد.

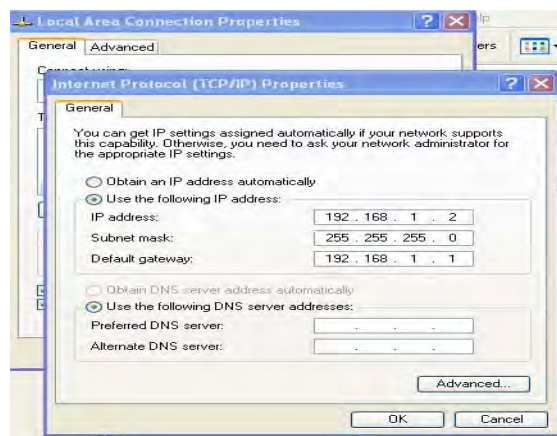
روش آدرس دهی IP بر روی کامپیوتر

برای اینکه بخواهید به یک شبکه اتصال پیدا کنید لازم است که کامپیوتر شما دارای یک کارت شبکه باشد. کارت شبکه، ارتباط بین کامپیوتر و محیط انتقال را فراهم می‌نماید. حال می‌توانید در صورت وجود کارت شبکه بر روی کامپیوتر خود همانند شکل زیر آن را آدرس دهی کنید.



شکل ۱-۸: قسمت شبکه در کنترل پنل

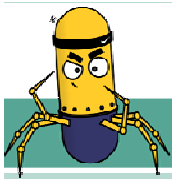
در این مرحله پس از انتخاب کارت شبکه به صورت زیر آن را آدرس دهی نمایید.



شکل ۹-۱: تنظیم IP

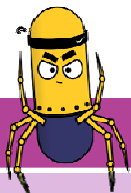
مقدار subnet mask روش فیلترگذاری بر روی آدرس (ip) می‌باشد که به عنوان یک mask برای مشخص کردن زیرشبکه‌های متصل به مسیریاب‌ها استفاده می‌شود.

مقدار Default Gateway را از عددی (IP) مقدار داده‌ایم که نشان می‌دهد ما به کدام کامپیوتر می‌خواهیم متصل شویم و از آن سرویس بگیریم. به عنوان مثال Default gateway من در حال حاضر ۱۹۲،۱۶۸،۱،۱ است. یعنی IP کامپیوتری که من به آن Connect شده‌ام و از آن سرویس می‌خواهم بگیرم ۱۹۲،۱۶۸،۱،۱ است.



تمرین

سعی کنید همانند بالا کارت شبکه خود را IP بدهید و سپس با استفاده از دستور ipconfig آدرسی که به کامپیوتر خود داده‌اید را چک کنید.



تمرین بیشتر

ایجاد اولین شبکه با استفاده از دو کامپیوتر

سعی کنید دو کامپیوتر را با استفاده از کابل به یکدیگر متصل کرده و سپس با توجه مطالبی که در بالا خوانده‌اید به این دو کامپیوتر ip دهید البته باید توجه کنید که یکی از آنها را حتما gateway نمایید. (یعنی شماره یکی باید شماره gateway کامپیوتر دیگری باشد و خود آن لازم نیست دیگر gateway داشته باشد).

از کجا می‌توان فهمید در تمرین بالا این دو کامپیوتر به یکدیگر وصل شده‌اند و تشکیل شبکه کوچکی را داده‌اند.

جهت اینکه بتوانیم ببینیم این دو کامپیوتر با یکدیگر ارتباط دارند، ساده‌ترین راه این است که با استفاده از پروتکل ping اتصال این دو کامپیوتر را با یکدیگر چک کنیم.

Ping^۳

پروتکلی است برای آزمایش اینکه یک کامپیوتر خاص به شبکه متصل است یا خیر. برای این کار بسته‌ای برای دریافت پاسخ به نشانی IP آن ارسال می‌شود و جوابی را در مدت زمان ارسال دریافت می‌کند.

³ Packet Internet Groper

```

C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\mono>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

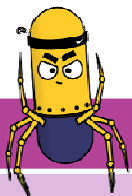
Reply from 192.168.1.1: bytes=32 time=1ms TTL=64
Reply from 192.168.1.1: bytes=32 time=1ms TTL=64
Reply from 192.168.1.1: bytes=32 time=2ms TTL=64
Reply from 192.168.1.1: bytes=32 time=2ms TTL=64

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms

C:\Documents and Settings\mono>

```

شکل ۱-۱۰: نمایی از ping کردن یک کامپیوتر



تمرین بیشتر

روش شبکه کردن دو تا کامپیوتر بدون سیم

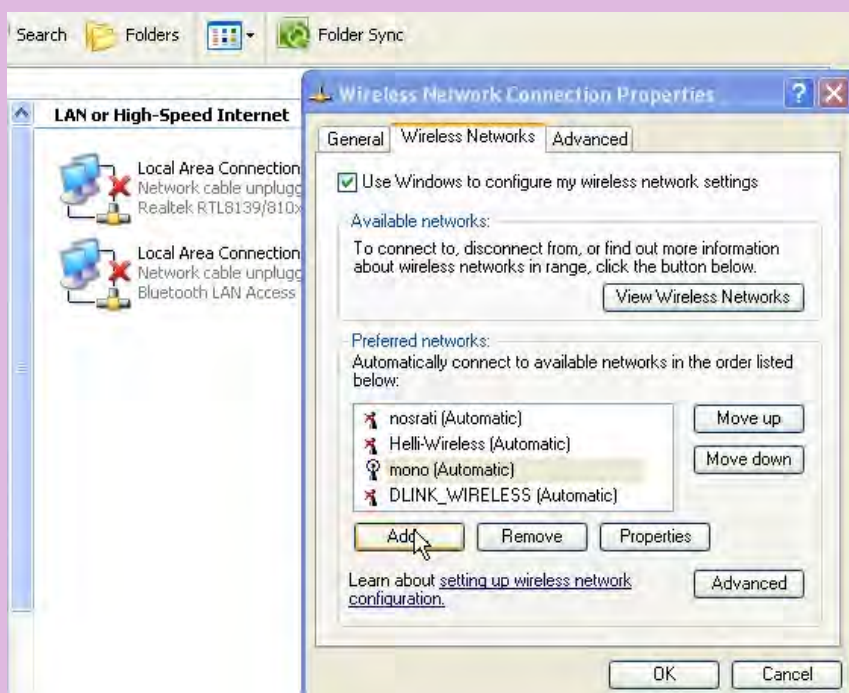
همان طور که می‌دانید برای اتصال دو کامپیوتر با استفاده از کارت شبکه از کابل استفاده کردیم که سرعت بسیار خوبی را هم برای ما فراهم می‌آورد. (در گذشته 100mbps و در حال حاضر 1Gbps)

البته این روش به خاطر محدودیت فاصله دو کامپیوتر از هم می‌تواند مشکلاتی را برای کاربران بوجود آورد. شبکه کردن دو کامپیوتر به صورت بی‌سیم به می‌تواند تا حدی این مشکل را حل کند.

برای ایجاد یک اتصال adhoc در قسمت Network Connection از Control Panel و یا در System Tray (کنار ساعت) بر روی آیکن wireless کلیک کنید.

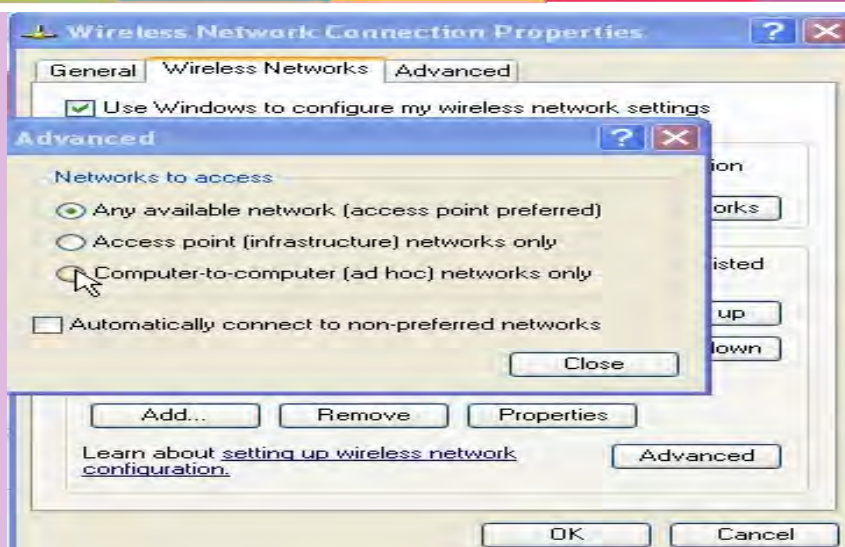


سر برگ Wireless Networks را انتخاب کنید سپس بر روی دکمه Add کلیک کنید. در پنجره جدید برای شبکه خود نامی را انتخاب کنید و تیک هر دو Check box را بزنید.



دو بار بر روی کلید OK کلیک کنید تا پنجره‌ها بسته شوند. اکنون شبکه نصب شده است. Network Connection را باز کنید سپس Wireless Networks را انتخاب کنید. بایستی در پنجره باز شده شبکه ad hoc را به عنوان computer to computer ببینید. ممکن است برای این کار refresh صفحه لازم باشد.

در صورتیکه باز نتوانستید شبکه خودتان را ببینید باید شبکه خودتان را منحصر به adhoc کنید بدین صورت که در سر برگ Wireless networks بر روی advance کلیک کنید و در صفحه باز شده گزینه را انتخاب کنید.



شکل ۱-۱۱: ایجاد شبکه ad hoc

۳-۱. مراحل پیاده سازی یک سایت

در بخش‌های قبلی شما با زیرساخت‌های شبکه‌های کامپیوتری و بطور خاص شبکه اینترنت آشنا شدید. پس از آشنایی مقدماتی با شبکه و اینترنت و تشکیل یک شبکه کوچک لازم است برگردیم به بحث اصلی هرچه زودتر بتوانید به این شبکه اینترنتی بپیوندیم.

نقشه سایت

برای شروع یک سایت اینترنتی لازم است در ابتدا **موضوع طراحی** سایت خود را مشخص کنید.

بنابراین اگر موضوعی در ذهن ندارید پیشنهاد می‌شود با گشت و گذارهای اینترنتی و فکر در مورد موضوع مورد علاقه و دیدن سایت‌های طراحی شده موضوع سایتی خود را مشخص کنید

سپس بخشهای سایت خود را داخل یک برگ بنویسید. در ادامه برای ساخت وب سایت شخصی، برای خود با کتاب حاضر همراه شوید.

ثبت نام دامنه

قبل از اینکه روش انتخاب و ثبت یک دامنه را برای شما شرح دهیم لازم است توضیح دهیم که یک دامنه چیست.

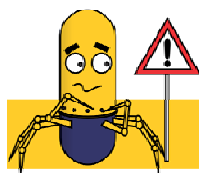
نام دامنه چیست؟

نام دامنه یک نام منحصر به فردی است که برای شناسایی سایت شما در شبکه اینترنتی بکار می‌رود همانند webook.ir که از دو قسمت تشکیل شده است.

۱: نام (webook) که منظور همان نام سایت مورد علاقه شما است.

۲: پسوند (ir) که نشان‌گر نوع فعالیت و محدوده فعالیت شما بر روی اینترنت است.

برای داشتن یک دامنه لازم است ما آن را در ابتدا ثبت نماییم. در ثبت دامنه‌ها محدودیت کمی وجود ندارد ولی نام‌ها با توجه به اینکه چه کسی زودتر برای ثبت آن نام اقدام کرده باشد، ثبت می‌شوند. این نام‌ها برای مدت یک سال در اختیار شما هستند و شما می‌توانید هر ساله آن را تمدید نمایید یا از همان ابتدا نام سایت را به مدت چند سال رزرو نمایید.



چگونه با استفاده از نام دامنه آدرس سایت اینترنتی ما مشخص می شود؟

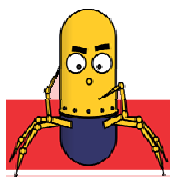
هنگامیکه افراد نام دامنه‌ی شما را در یک مرورگر (internet explorer) تایپ می‌کنند سرورهایی هستند که نامه دامنه‌ی شما را به IP آدرس‌ها تبدیل می‌کنند و با استفاده از ip می‌توانیم به یک کامپیوتر وصل شویم و به این صورت می‌توانیم ارتباط برقرار کنیم. در واقع نام دامنه‌ی شما یک عبارت است که توسط این سرورها به IP تبدیل می‌شود. مزیت این عمل در آنست که خود شما و همه‌ی بازدیدکنندگان وب سایت‌تان نام دامنه را نسبت به IP که شامل تعدادی عدد است بهتر به خاطر می‌سپارید. به زبان ساده DNS مثل یک مرکز ۱۱۸ (مرکز اطلاعات تلفن در ایران) کار می‌کند. یعنی لیست نام‌ها و IP هر کدام از آنها را در یک بانک اطلاعاتی ذخیره کرده و هر وقت سئوالی از آن پرسیده شود با مراجعه به این بانک اطلاعاتی، IP مربوط رو پیدا و به شما اطلاع می‌دهد. (مثل ۱۱۸ که اسم شخص رو می‌پرسد و به آنها شماره را می‌گوید).

امروزه دیگر ما بدون این ابزارها نمی‌توانیم از اینترنت استفاده کنیم و مفاهیم کارهای روی شبکه بر مبنای این ابزارهای و سرویس‌ها و قوانین تعریف می‌شوند. بنابراین لازم می‌بینیم این سرویس مهم اینترنتی را بیشتر بشناسیم.

سرویس DNS چیست و چه وظیفه‌ای دارد؟

در سال ۱۹۸۴ برای ساده‌سازی فرآیند به حافظه سپردن و حفظ نمودن آدرسهای اینترنتی که متشکل از اعداد (آدرس‌های IP) بود،^۴ DNS بوجود آمد. کار DNS تبدیل همان حروف الفبا به اعداد می‌باشد. در این روش شما بجای به خاطر سپردن چند عدد سخت و گنگ، کفایت یک نام را بخاطر بسپارید.

بنابراین برای این کار سرویسی در شبکه‌ها بوجود آمد که امکان تبدیل یک رشته حرفی به آدرس IP را به عهده داشت. این سرویس در اینترنت دارای یک کامپیوتر مرکزی است که هر یک از ما برای تبدیل رشته به آن باید وصل می‌شدیم و از طریق آن IP مربوط به رشته یا نام دامنه خود را می‌گرفتیم.

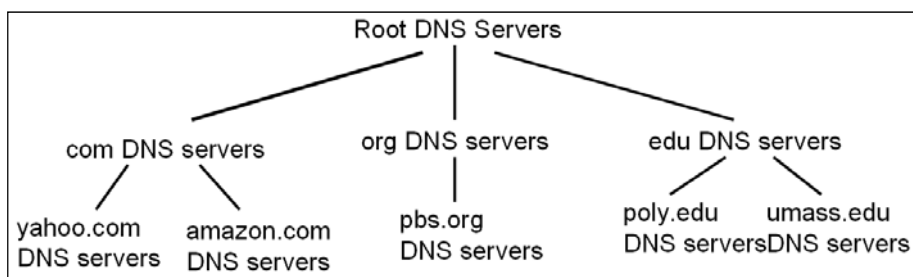


نکته مهم!

اگر یک سرور داشته باشیم شبکه خیلی شلوغ می‌شود، پس کار را باید بین چندین

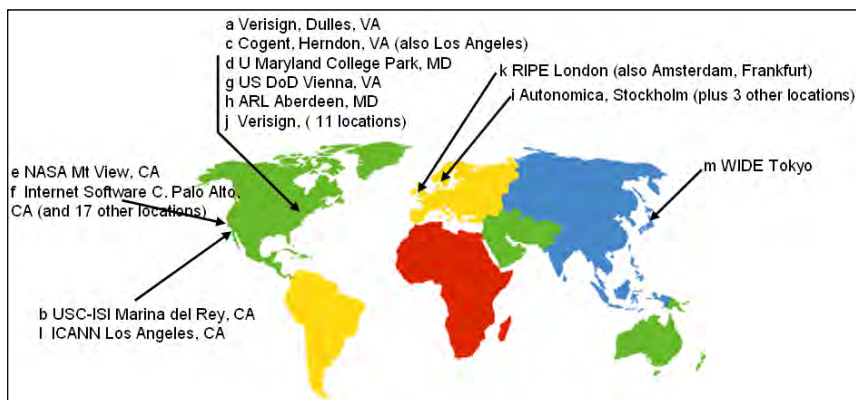
^۴ Domain Name Service

در واقعیت از کامپیوتر مرکزی ۱۷ کامپیوتر دیگر تغذیه می‌شوند. (بر حسب اینکه سایت‌ها با چه پسوندی ثبت شده باشند.) هر یک از ما برای اینکه بتوانیم آدرس یک کامپیوتر را پیدا کنیم باید در ابتدا به سرور مرکزی وصل شویم و از آن به یکی از ۱۷ سرور مرحله دوم و همین‌طور به DNS‌های مراحل پایین‌تر تا اینکه به IP مورد نظرمان برسیم.

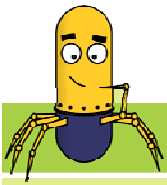


شکل ۱-۱۲: ساختار درختی سرورهای DNS اصلی.

از آنجایی که این ۱۷ سرور محدود و بسیار مهم هستند، محل‌های جغرافیایی قرارگیری آنها در جهان کاملاً مشخص است!

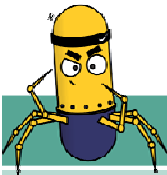


شکل ۱-۱۳: محل جغرافیایی قرارگیری DNS‌های لایه دوم.



بیشتر بدانیم!

DNS سرورها برای سرعت بخشیدن به عمل تبدیل نام با IP عمل Caching را انجام می‌دهد، به این صورت که تمام جواب‌هایی که از دیگر DNS سرورها دریافت می‌شود و در آنها Cache شده در نتیجه در صورت درخواست دوباره نام مشابه نیاز به تکرار مراحل نیست.



تمرین

پیدا کردن ip از طریق نام سایت درویندوز با استفاده از nslookup

NSLookup یک برنامه از نوع خط فرمان (command-line) است که مخفف Name Server Lookup می‌باشد. به وسیله NSLookup می‌توان از Name Server های مختلف IP مورد نظر را در صورت امکان بدست آورد. در واقع NSLookup یک ابزار مدیریتی در شبکه برای آزمایش و رفع اشکال Name Server ها می‌باشد.

اجرای برنامه NSLookup

Command Prompt را باز کنید و دستور nslookup را به همراه نام سایت به صورت زیر تایپ

```

C:\WINDOWS\system32\cmd.exe

C:\>nslookup www.google.com
*** Can't find server name for address 192.168.1.1: Non-existent domain
*** Default servers are not available
Server: Unknown
Address: 192.168.1.1

DNS request timed out.
    timeout was 2 seconds.
Non-authoritative answer:
Name:   www.l.google.com
Addresses: 74.125.230.116, 74.125.230.112, 74.125.230.113, 74.125.230.115
          74.125.230.114
Aliases: www.google.com

C:\>

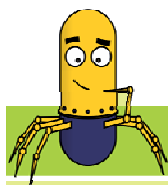
```

مراحل ثبت نام دامنه

برای ثبت نام دامنه مورد نظر باید به شرکت‌هایی که این خدمت را ارائه می‌دهند مراجعه نمایید که دو روش کلی را می‌توان بیان کرد:

۱: ثبت کننده‌های دامنه‌های عمومی که معمولاً شرکت‌هایی هستند که با مجوز مستقیم از استانداردسازی دامنه امکان ثبت دامنه‌های عمومی را دارند.

۲: ثبت دامنه‌های کشوری که کاملاً بر عهده مرکز ثبت دامنه هر کشور است.



بیشتر بدانیم!

در ایران مرکز فیزیک نظری مسئول ثبت دامنه‌های IR میباشد و این مرکز تصمیم

می‌گیرد که چه ثبت‌کننده‌های دامنه‌ای امکان ثبت دامنه را تحت آن دامنه کشوری داشته باشند.

نکات قابل توجه در ثبت دامنه

نام دامنه مهمترین قسمت تشکیل دهنده آدرس یک سایت می‌باشد. این نام از ترکیب ۳ الی ۶۴ حرف، عدد یا خط تیره (بدون فاصله) تشکیل می‌گردد.

✓ نام انتخابی نباید مشابه با نام تجاری معروف و خاصی باشد. مثلاً انتخاب نام microsoftany یا نام‌هائی از این قبیل می‌تواند منجر به مسدود شدن سایت شما و سلب اختیار مالکیت دامنه از شما، توسط شرکت مذکور گردد.

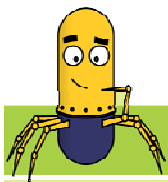
✓ کوتاه و متشکل از حداقل تعداد حروف باشد. (دامنه‌های دارای تعداد حروف کمتری هستند، معمولاً ارزش بیشتری دارند، چون تلفظ و به خاطر سپردن این آدرس‌ها راحت‌تر است.)

✓ با نوع فعالیت سایت ارتباط داشته باشد. هرچه نام انتخابی و فعالیت سایت شما نزدیک‌تر باشند، پیدا کردن سایت شما و تجارت‌تان برای جستجوگران وب و کاربران ساده‌تر خواهد بود.

✓ از اعداد و خط تیره استفاده نشده یا حداقل استفاده صورت گیرد. کوتاه بودن و به اصطلاح رند بودن دامنه بسیار مهم است.

✓ هرگز از آدرس مجهول یا غیر واقعی در مشخصات دامنه خود استفاده ننمایید. در صورت استفاده از آدرس مجهول یا دروغین در هنگام ثبت دامنه، اگر شرکت‌های ثبت‌کننده متوجه این امر شوند به راحتی می‌توانند حق مالکیت دامنه را از شما سلب نموده و شما نیز نمی‌توانید هیچ ادعائی بنمایید.

✓ هرگز از DNS شرکتها و خدمات دهنده‌هایی که مجاز به استفاده از آن نمی‌باشید در دامنه خود استفاده نکنید.

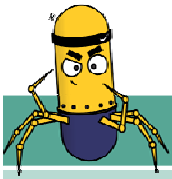


بیشتر بدانیم!

همه‌ی شرکت‌های بزرگ و اصلی ثبت‌کننده‌ی نام دامنه‌ها، یک directory را که شامل اطلاعات کسانی است که صاحب دامنه می‌باشند نگهداری و بروز رسانی می‌کنند. این whois database شامل نام، آدرس، آدرس ایمیل و تلفن صاحبان دامین‌ها می‌باشد و برای هر کسی که به اینترنت متصل شود قابل دسترسی می‌باشد. این کار سبب می‌شود تا صاحبان دامنه بتوانند با مدرک قطعی مالکیت خود را اثبات کنند و همچنین مردم جهت خرید دوباره‌ی دامنه از روی اطلاعات این شخص می‌توانند با او تماس برقرار کنند. همچنین این database صاحب اصلی دامنه را به صورت قطعی و رسمی قابل تشخیص و شناسایی می‌کند. از سوی دیگر در اختیار قرار دادن اطلاعات به همه‌ی مردم سبب ایجاد مزاحمت‌هایی از جمله برداشتن email ها توسط و تلفن‌ها برای صاحبان دامنه‌ها می‌شود.

راه حل چیست؟

البته بعضی از شرکت‌های ثبت‌کننده سرویسی ارائه می‌دهند تا اطلاعات شما محفوظ بماند و در whois نمایش داده نشود. (البته ممکن است بابت این سرویس مقداری هزینه دریافت کنند). چنانچه مایل نیستید تا برای خرید دامنه با شما تماس بگیرند شما می‌توانید نزد شرکت‌هایی دامنه خود را ثبت کنید که اطلاعات شما را آشکار نمی‌کنند.



تمرین

وبوک در سایت webook.ir می‌تواند برای گرفتن whois به شما کمک کند و با تایپ نام مورد نظر ببینید اسمی که شما برای سایت خود انتخاب کرده اید قبلاً ثبت نشده باشد.

خرید هاست

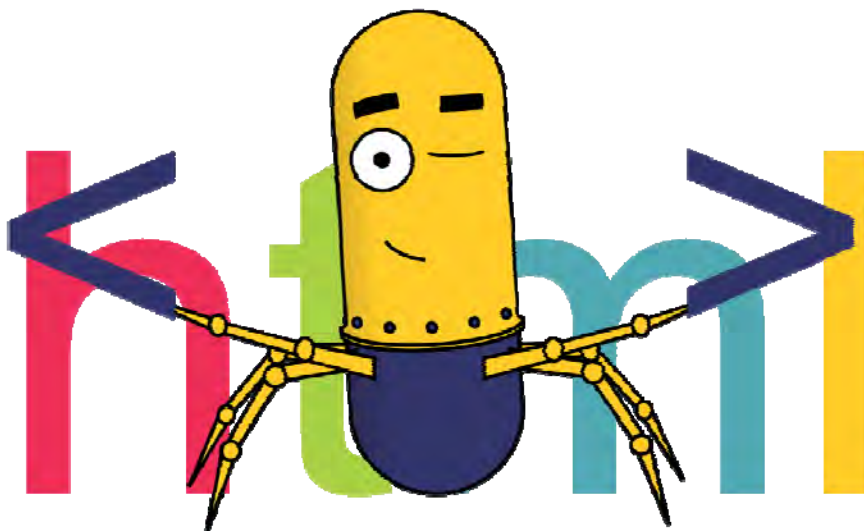
پس از انتخاب نام برای دامنه مورد نظرتان و ثبت آن، شما دارای یک آدرس اینترنتی خواهید شد. ولی با وجود این وقتی آدرس سایتتان را وارد مرورگرتان می‌کنید این جور به نظر می‌رسد که این آدرس ثبت نشده است. اشتباه نکنید. آدرس شما ثبت شده است و تنها چیزی که ندارید داشتن یک منبع برای نمایش اطلاعات آن است.

در اینترنت به چنین مکانی به عنوان منبعی برای اطلاعاتتان، میزبان^۵ گفته می‌شود. میزبان مکانی برای میزبانی سایت شما می‌باشد که می‌توانید اطلاعاتتان را در آن برای عموم به نمایش بگذارید که از یک صفحه ساده تا یک سایت حرفه‌ای را در آن می‌توانید upload (بارگذاری) کنید. این فضا دقیقاً مانند فضایی است که روی هارد دیسک کامپیوتر شما می‌باشد، با این تفاوت که فضا روی هارد دیسک‌های سروری قرار دارد که دائم به اینترنت پرسرعت متصل می‌باشد و فایل‌هایی که توسط صاحب سایت روی هاست قرار می‌گیرد، از سراسر دنیا قابل دسترسی هستند.

⁵ Host

فصل دوم

برنامه نویسی HTML



برنامه نویسی HTML

برای ایجاد وبسایت شخصی لازم است تا صفحاتی طراحی شوند تا بر روی کامپیوتر میزبان^۶ قرار گیرند تا در زمانی که بازدیدکنندگان برای بازدید به سایت رجوع می‌کنند برای آن‌ها نمایش داده شود. برای اینکار لازم است تا از زبان برنامه‌نویسی‌ای استفاده شود تا قابلیت قرار دادن محتوا را در صفحه برای نمایش را داشته باشد.

۲-۱. هدف از ایجاد وبسایت شخصی

وبسایت‌ها برای معرفی یک حرفه، شرکت، محصول، شخص و ... بر روی فضای مجازی قرار می‌گیرند. با توجه به اینکه دامنه مطرح شدن در دنیای وب بسیار وسیع است لازم است تا قبل از اجرای وبسایت روی بخش‌های آن فکر شده و طراحی شوند.

وبسایت شخصی برای معرفی یک شخص در سطح وسیع ایجاد می‌شود. افراد بر اساس موقعیت‌های مختلف اجتماعی از وبسایت شخصی خود جهت انتشار مطالب مختلف استفاده می‌کنند. اما بخش‌هایی به صورت مشخص در همه این سایت‌ها وجود دارد. در این فصل به صورت مرحله به مرحله به معرفی این بخش‌ها می‌پردازیم و نحوه اجرای آن را می‌آموزیم.

۲-۲. چرا HTML؟

برای ساختن صفحات وب، زبان‌های متعددی وجود دارد ولی زبان پایه و ساده برای طراحی صفحات HTML^۷ است. این زبان پایه و اساس طراحی و برنامه‌نویسی وب می‌باشد. HTML یک زبان ساده برای مشخص کردن محل قرارگیری متون و تصاویر و دیگر اجزای یک صفحه است.

^۶ Host

^۷ Hyper Text Markup Language

همانند زبان‌های دیگر برنامه‌نویسی، برای زبان HTML نیاز به یک محیط برای برنامه‌نویسی و محیط دیگر برای اجرا هستیم. برای برنامه‌نویسی به زبان HTML نیاز به هیچ برنامه اضافی نیست. در Notepad می‌توان به این زبان برنامه نوشت و برای اجرا نیز نیاز به یک مرورگر^۸ است که به صورت پیش فرض بر روی هر سیستم عاملی وجود دارد.

در این کتاب برای سهولت در کدنویسی از محیط Eclipse استفاده شده است. در پیوست نحوه نصب و اجرای Eclipse توضیح داده شده است.

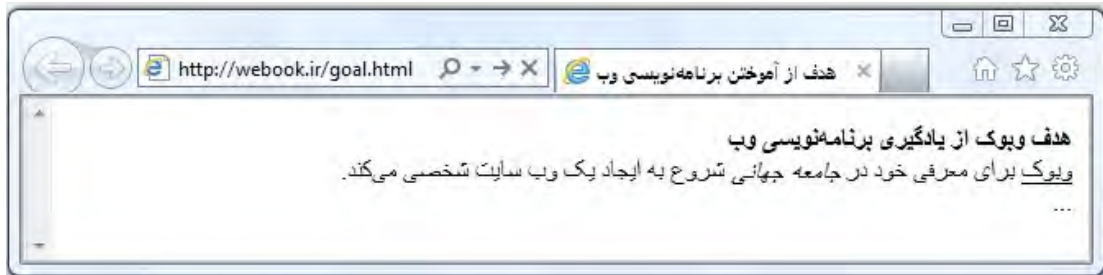


شکل ۲-۱: نمایی از محیط Eclipse

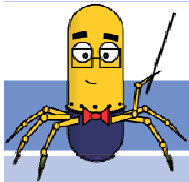
۲-۳. ساخت اولین صفحه در وب

خب وقت آن رسیده که اولین اثر از خود را در وب ایجاد کنیم. به عنوان اولین اثر هدف خود را از آموختن برنامه‌نویسی وب در صفحه‌ای به نمایش می‌گذاریم.

⁸ Brower



شکل ۲-۲: ایجاد اولین صفحه در وب و شرح هدف از آموختن برنامه‌نویسی وب



ویژه دبیران

کد صفحه فوق به شرح زیر است:

```
<html>
<head>
  <title>هدف از آموختن برنامه نویسی وب</title>
</head>
<body dir="rtl">
  <b>هدف وبوک از یادگیری برنامه نویسی وب</b>
  <br>
  <u>وبوک</u> <i>جامعه جهانی</i> برای معرفی خود در <i>جامعه جهانی</i> می‌کند
  <br>
  ...
</body>
</html>
```

لازم است در این قسمت مفهوم تگ و اصول اولیه برنامه‌نویسی HTML، پسوندهای فایل‌های HTML به خوبی توضیح داده شود.

همچنین انتظار می‌رود نحوه ذخیره کردن فایل و مفهوم Encoding برای ذخیره کردن صفحات فارسی انتقال یابد.

تگ^۹

تگ‌ها دستورات HTML هستند که بین دو علامت < و > نوشته می‌شوند.

```
<br>
```

برخی از تگ‌ها مانند تگ br یک قسمتی هستند و برخی دیگر به صورت فوق آغاز و با همان دستور بین > و </ پایان می‌شود.

```
<b>متن برجسته</b>
```

ساختار پایه کد

تگ <html></html>

برای شروع درست کردن یک صفحه ابتدا باید تگ HTML را بنویسیم. تگ HTML از جمله تگ‌هایی است که دو قسمت باز و بسته دارد، پس مطابق توضیح بالا به صورت زیر نوشته می‌شود.

```
<html>
</html>
```

تگ‌های <head></head> و <body></body>

درون تگ HTML به دو قسمت head و body تقسیم می‌گردد.

```
<html>
  <head>
</head>
  <body>
</body>
</html>
```

^۹ Tag

هر متنی که بین تگ باز و بسته body نوشته شود، دقیقا در صفحه مرورگر نشان داده می‌شود. حال بین تگ body هدف خود را از آموزش HTML شرح دهید.

تیتتر صفحه، تگ `<title></title>`

شما می‌توانید برای صفحه خود یک عنوان انتخاب کنید تا در نوار بالایی مرورگر نمایش داده شود. پیشنهاد می‌شود در انتخاب عناوین دقت شود تا بخوبی بیانگر محتوای صفحه باشد زیرا این عنوان‌ها در زمانی که کاربر بر روی صفحات و یا Tab های دیگر قرار دارد، مشخص کننده صفحه برای انتخاب شدن است.

این تگ بین تگ‌های head به صورت زیر نوشته می‌شود.

```
<head>
  <title> تیتتر صفحه اینجا نوشته می‌شود </title>
</head>
```

ساختار نوشتاری

تگ `<p></p>`

با نوشتن متن درون این تگ، آن قسمت به صورت یک پاراگراف مشخص می‌شود و از خطوط بالایی و پایینی خود کمی بیشتر از فاصله دو خط، فاصله ایجاد می‌شود.

تگ `<br>`

شاید متوجه شده باشید که شما فاصله‌هایی که برای رفتن به خط بعد در کد، توسط کلید Enter ایجاد می‌کنید، نقشی در نمایش صفحه ندارد و نمایشگر از نمایش آن صرف‌نظر می‌کند. این

قابلیت برای این ایجاد شده است تا شما بتوانید برای خوانایی کد خود بین قسمت‌های مختلف فاصله ایجاد کنید و این در نمایش شما تاثیر نگذارد.

پس چگونه در نمایشگر بین خطوط فاصله ایجاد کنیم؟

برای ایجاد فاصله بین خطوط و شکستن خطوط از تگ `<br>` استفاده می‌کنیم. همانطور که مشاهده می‌کنید تگ `<br>` یک تگ تک قسمتی است و دیگر نیاز به بسته شدن ندارد.

برای ایجاد چند خط فاصله لازم است تا تگ `<br>` چند با نوشته شود.

تگ‌های `<b></b>` و `<i></i>` و `<u></u>`

گاهی لازم است تا قسمتی از متن را به صورت برجسته^{۱۰} یا مورب^{۱۱} شود و یا خطی زیر آن^{۱۲} کشیده شود. برای این کار آن قسمت از متن را که می‌خواهیم برجسته باشد، بین تگ `<b></b>` می‌نویسیم. برای مورب نوشتن متن آن را بین `<i></i>` و برای کشیدن خط زیر نوشته‌ها آن را بین `<u></u>` می‌نویسیم.

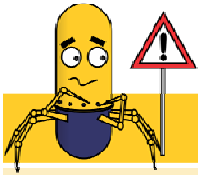
تگ `<font></font>`

تغییرات مانند تغییر اندازه متن، نوع فونت متن و تغییر رنگ متن از طریق تگ `<font></font>` امکان‌پذیر است.

¹⁰ Bold

¹¹ Italic

¹² Underline



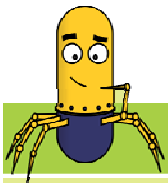
توجه!

ویژگی‌ها^{۱۳}:

تگ‌ها بسته به نوع خود دارای ویژگی‌هایی هستند. ویژگی‌ها قسمت‌هایی از تگ هستند که بوسیله آن‌ها می‌توان تغییراتی در عملکرد تگ ایجاد کرد.

ویژگی‌ها فقط درون تگ با یک فاصله از نام تگ قرار می‌گیرند و در جلوی آن‌ها بعد از یک علامت = درون " " مقدار^{۱۴} آن ویژگی مشخص می‌شود.

```
<p dir="rtl"> </p>
```



بیشتر بدانیم!

اگر تا اینجا اولین صفحه خود را ایجاد کرده باشید، متوجه این شده‌اید که مرورگرها به صورت پیش‌فرض صفحه شما را به صورت چپ به راست نمایش می‌دهند.

در صورتی که صفحه خود را به صورت فارسی می‌نویسید، لازم است تا از ویژگی dir در

¹³ Attributes

¹⁴ Value

تگ <body> استفاده کنید. این ویژگی می‌تواند یکی از دو مقدار rtl^{۱۵} یا ltr^{۱۶} را داشته باشد.

برای نمایش صفحه به صورت راست به چپ <body dir="rtl">

برای نمایش صفحه به صورت چپ به راست <body dir="ltr">

همانطور که قبل‌تر دیدید، این ویژگی در تگ <p></p> نیز قابل استفاده است.

برای تغییر اندازه متن می‌توانیم از ویژگی size به صورت زیر درون تگ font استفاده کنیم.

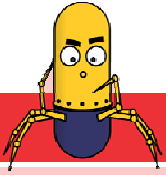
تغییر فونت نیز با ویژگی face امکان‌پذیر است.

این امکان وجود دارد که شما چند فونت جایگزین نیز در نظر بگیرید تا در صورتی که بازدید کننده فونت اول مشخص شده را بر روی مرورگر خود نداشت، آن متن با فونت‌های بعدی نمایش داده شود.

هر چند توصیه اکید می‌شود از فونت‌هایی استفاده کنید که به صورت استاندارد بر روی تمام دستگاه‌ها وجود داشته باشند تا صفحه شما نمایی واحد در تمامی مرورگرها داشته باشد.

¹⁵ Right to left

¹⁶ Left to right

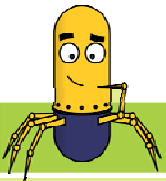


نکته مهم!

ویژگی‌های درون یک تگ اولویتی نسبت به یکدیگر ندارند و همزمان با هم قابل استفاده هستند.

برای تغییر رنگ نوشته متن نیز می‌توان از ویژگی color درون تگ font استفاده کرد. رنگ‌های ساده red, yellow, blue, green, black, white, ... رنگ‌هایی هستند که به عنوان مقدار ویژگی color قابل استفاده هستند.

```
<font size="4" face="Tahoma" color="green"> </font>
```



بیشتر بدانیم!

همانطور که می‌دانید رنگ‌ها از سه رنگ اصلی قرمز (Red)، سبز (Green) و آبی (Blue) تشکیل شده‌اند. در انتخاب رنگ در HTML نیز استفاده از ترکیب این سه رنگ برای ایجاد رنگ دلخواه امکان‌پذیر است.

انتخاب رنگ به این صورت است مقدار هر رنگ به صورت عددی بین ۰ تا ۲۵۵ (که در

مبنای ۱۶ عبارتست از 00 تا FF) مشخص می‌شود.

سپس کد رنگ از قرار گرفتن علامت # و مقدار هر رنگ به صورت زیر مشخص می‌شود.

#RRGGBB

حال رنگی که به این شکل انتخاب می‌شود به صورت زیر قابل استفاده به عنوان رنگ متن می‌باشد. در کد زیر مقدار رنگ سبز بیشترین مقدار و رنگ‌های آبی و قرمز کمترین مقدار خود را دارد و در نتیجه متن به رنگ سبز نمایش داده می‌شود.

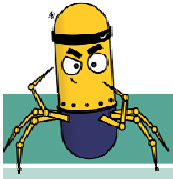
```
<font color="#00FF00"> </font>
```

رنگهای پیش فرض

رنگ‌های زیر به صورت پیش فرض در HTML تعریف شده‌اند و شما می‌توانید از اسامی آن‌ها برای ویژگی color استفاده کنید.

 Black = "#000000"	 Green = "#008000"
 Silver = "#C0C0C0"	 Lime = "#00FF00"
 Gray = "#808080"	 Olive = "#808000"
 White = "#FFFFFF"	 Yellow = "#FFFF00"
 Maroon = "#800000"	 Navy = "#000080"
 Red = "#FF0000"	 Blue = "#0000FF"
 Purple = "#800080"	 Teal = "#008080"
 Fuchsia = "#FF00FF"	 Aqua = "#00FFFF"

شکل ۲-۳: رنگ‌های پیش فرض



تمرین

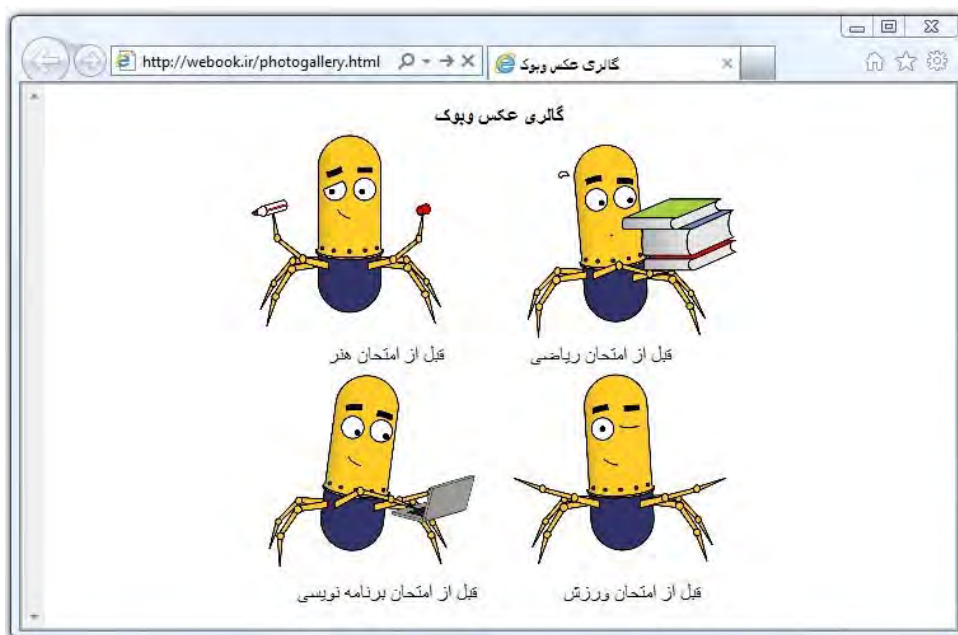
حال با اطلاعاتی که تا کنون در این فصل آموختید، اولین صفحه وب خود را ایجاد کرده و در آن اهداف خود را از آموختن برنامه‌نویسی وب شرح دهید.

۲-۴. صفحه آلبوم عکس

عکس‌ها جز جدایی ناپذیر از صفحات وب هستند. امروزه تقریباً نمی‌توان صفحه‌ای را در اینترنت پیدا کرد که در آن عکسی وجود نداشته باشد. همانطور که تاکنون محیط برنامه‌نویسی HTML را مشاهده کردید، محیطی کامل متنی است. پس لازم است تا ابتدا فایل عکس‌های مورد نیاز به صورت جداگانه ذخیره شوند.

برای شروع یک عکس را در پوشه^{۱۷} مشابه که فایل HTML خود را آن ذخیره کرده‌اید، قرار دهید.

¹⁷ Folder



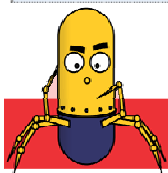
شکل ۲-۴: ایجاد صفحه گالری عکس

تگ

از تگ `img` برای قرار دادن عکس در صفحات HTML استفاده می‌شود. تگ `img` دارای ویژگی به نام `src` می‌باشد، که آدرس قرار گرفتن عکس را مشخص می‌کند.

```

```



نکته مهم!

توجه کنید که پسوند عکس را هم باید به طور دقیق در آدرس ذکر کنید. همچنین به بزرگی و کوچکی حروف توجه کامل داشته باشید تا صفحات نوشته شده توسط شما در همه

سرویس دهنده‌ها به درستی نمایش داده شوند.

آدرس دهی در وب:

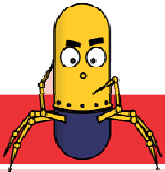
به دو صورت **مطلق**^{۱۸} و **نسبی**^{۱۹} می‌توان آدرس یک عکس را مشخص کرد. در صورتی که عکس بر روی سایت دیگری قرار دارد و قصد استفاده از آن را داریم، می‌توان از روش آدرس دهی مطلق استفاده کرد.

روش آدرس دهی مطلق به این صورت است که باید آدرس به صورت کامل از ابتدا به همراه شیوه اتصال ذکر شود.

به طور مثال آدرس مطلق لوگو سایت Google به صورت زیر است:

`http://www.google.com/images/logos/ps\_logo2.png`

همانطور که مشاهده می‌کنید، در آدرس مطلق نحوه اتصال (http) و نوع فایل عکس (png) حتما ذکر گردد.



نکته مهم!

توجه کنید که در آدرس تنها از ^{۲۰}/ استفاده کنید. استفاده از ^{۲۱}\ سبب می‌شود تا در نمایش صفحات شما بر روی برخی از سرویس دهنده^{۲۲}ها مشکل ایجاد شود و تصویر نمایش داده نشود.

¹⁸ Global

¹⁹ Local

²⁰ Slash

²¹ Back-slash

²² Server

حال اگر که فایل تصویر را در اختیار داریم، در صورتی فایل را در کنار صفحه HTML در یک پوشه باشد، به عنوان آدرس تنها نام فایل را به همراه پسوند آن ذکر می‌کنیم.

در صورتی که در یک پوشه کنار فایل HTML باشد، ابتدا نام پوشه و سپس یک / و بعد نام فایل را همراه با پسوند ذکر می‌کنیم.

در صورتی نیز که بخواهیم به پوشه‌های بالاتر برویم، از ../ به ازای هر پوشه‌ای که بخواهیم بالا برویم در آدرس استفاده می‌کنیم.

در شکل زیر حالت‌های مختلف آدرس نسبی ارائه شده است.

شکل!!!

حال با استفاده از تگ `img` یک صفحه آلبوم عکس از عکس‌های خود و یا یک موضوع مورد علاقه خود ایجاد کنید. این آلبوم عکس در دو ستون و حداقل دارای ۱۰ عکس باشد.

ویژگی width

توسط ویژگی width در تگ `<img>` این امکان وجود دارد تا شما عکس خود را با عرض مشخص و از قبل تعیین شده نشان دهید. مقدار این ویژگی می‌تواند بر حسب pixel و یا درصدی از آخرین محدودیت اعمال شده بر آن عکس باشد.

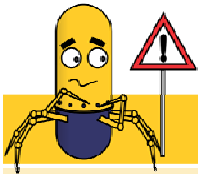
<code></code>	نمایش عکس با عرض ۱۰۰ پیکسل
<code></code>	نمایش عکس با عرض ۵۰ درصد محدودیت نمایش

ویژگی height

این ویژگی برای محدود کردن عکس استفاده می‌شود و کاربرد آن مانند width است. ویژگی‌های width و height در چند تگ دیگر کاربرد دارند که با برخی از آن‌ها در ادامه آشنا خواهید شد.

```


```

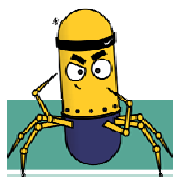


توجه!

لازم است قبل از استفاده از عکس‌ها، آن‌ها را توسط نرم‌افزارهای ویرایش عکس به ابعاد دلخواه خود درآوردید. زیرا درست است که شما می‌توانید در زمان نمایش عکس را با ویژگی‌های width و height به صورت کوچک‌تر نمایش دهید ولی در واقع عکس به صورت کامل به کامپیوتر بازدید کننده منتقل شده و کوچک نمایش داده می‌شود.

به این صورت چون عکس حجم بیشتری دارد، زمان فراخوان صفحه در کامپیوتر مخاطب بیشتر می‌شود. برای حل این مسئله ابتدا عکس را توسط ویرایشگر به ابعاد دلخواه درآوردید.

معمولاً از این دو ویژگی برای کنترل نمایش صحیح عکس استفاده می‌شود تا در صورتی که ابعاد عکس‌ها صحیح نبودند، ساختار صفحه به هم ریخته نشود.



نحوه استفاده از عکس در صفحات HTML را در این بخش آموختید. حال با آموخته‌های خود، آلبوم عکسی مانند آلبوم عکس **وبوک** حداقل دارای ۸ عکس برای خود ایجاد کنید. همچنین می‌توانید آلبوم‌های دیگری در زمینه علایق ورزشی، طبیعت و یا دیگر موارد نیز ایجاد کنید.

۲-۵. صفحه درباره من

به صورت معمول در وبسایت‌ها قسمتی وجود دارد که توضیحاتی را در مورد آن وبسایت ارائه می‌کند. در سایت شرکت‌ها معمولاً در این صفحه مشخصات شرکت ذکر می‌شود و در سوابقی از فعالیت‌های پیشین آن‌ها ذکر می‌شود.

این قسمت معمولاً در سایت‌های شخصی نیز دیده می‌شود. در این صفحه اطلاعاتی از صاحب سایت و سوابق تحصیلی و کاری او ذکر می‌شود.

برای ایجاد این صفحه ابتدا لیستی از سوابق تحصیلی و مدارس که در آنها تحصیل کرده‌اید را بنویسید. سپس لیستی از علایق ورزشی و لیستی از علایق هنری خود را تشکیل دهید.

تگ‌های لیست `<ul>` و `<ol>`

حال باید این لیست‌ها را به صورت کد HTML بنویسیم. دو حالت برای تشکیل لیست وجود دارد.

لیست‌های دارای ترتیب و شمارش `<ol>` `</ol>`

لیست‌های بدون شمارش `<ul>` `</ul>`

برای تشکیل لیست‌ها کافی است هر یک از لیست‌ها را که انتخاب کردیم، هر مورد را بین یک تگ `<li>` `</li>` درون تگ لیست آن بنویسیم.



شکل ۲-۵: نمایی از صفحه درباره من

در شروع صفحه یک لیست با شمارش از مواردی که در بالا ذکر شد بنویسید.

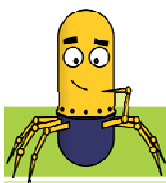
سپس برای هر کدام یک تیتراژ جدید ایجاد کرده و زیر آن در یک لیست بدون شمارش فعالیت‌های خود را شرح دهید.

تگ‌های `<h1>`/`</h1>` تا `<h6>`/`</h6>`

برای ایجاد تیترها ۶ تگ پیش فرض در HTML وجود دارد که از آن‌ها می‌توان به عنوان تیتتر اصلی، تیتتر داخلی و ... استفاده کرد.

اندازه متن نمایش داده شده درون این تگ‌ها از بزرگترین اندازه `<h1>` تا کوچکترین اندازه `<h6>` به ترتیب قرار گرفته‌اند.

برای تیتتر هر قسمت از بخش‌های بالا می‌توان از تگ `<h1>`/`</h1>` استفاده کرد.



بیشتر بدانیم!

تگ `<sup>`/`</sup>`

با استفاده از این تگ می‌توان برای نوشته‌ها بالا نویس ایجاد کرد. کافی است عبارتی را که می‌خواهید بالا نویس شود درون این تگ قرار دهید.

حال برای هر یک از مدارسی که در سوابق تحصیلی خود ذکر کرده‌اید یک شماره بالا نویس ایجاد کرده و در پایین صفحه منطقه و شهر آن مدرسه را ذکر کنید.

تگ `<sub>`/`</sub>`

این تگ برای ایجاد پایین نویس در نوشته‌ها استفاده می‌شود. در فرمول نویسی و موارد مشابه آن استفاده می‌شود.

۲-۶. صفحه پیوند به دوستان

در سایت‌های شخصی بخشی وجود دارد که مجموعه پیوند^{۲۳} به سایت دوستان و سایت‌های مورد علاقه شخص ایجاد می‌شود.



شکل ۲-۶: صفحه پیوند به دوستان

تگ `<a></a>`

این تگ برای ایجاد پیوند در صفحات وب ایجاد می‌شود. متنی را که قرار است بتوان بر روی آن کلیک کرد بین این تگ نوشته می‌شود. فرض کنید می‌خواهید در سایت خود پیوندی به سایت وبوک ایجاد کنید. لازم است تا در ابتدا کدی مانند زیر بنویسید.

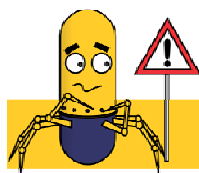
```
<a>Webook</a>
```

حال لازم است تا مشخص کنید تا اگر کاربر روی این نوشته کلیک کرد به سایت وبوک یعنی <http://www.webook.ir> هدایت شود.

تگ `<a></a>` دارای ویژگی‌ای به نام href است. href مشخص میکند تا اگر بر روی نوشته کلیک شد کاربر به چه صفحه‌ای منتقل شود.

```
<a href="http://www.webook.ir">Webook</a>
```

²³ Link



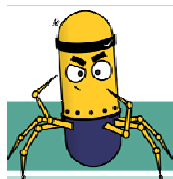
توجه!

لازم است بدانید که تمام قوانین آدرس‌دهی مطلق و نسبی برای آدرس‌های پیوند برقرار است. برای مثال همانطور که در بالا مشاهده کردید، آدرس‌دهی مطلق به <http://www.webook.ir> به صورت مطلق و با نوع ارتباط ذکر شده است.

حال اگر به عنوان مثال صفحه آلبوم عکس‌های خود را با نام `photoalbum.html` ذخیره کرده باشید و آن را در کنار این صفحه ذخیره کرده باشید، آدرس نسبی آن صرفاً `photoalbum.html` خواهد بود و کد آن به این صورت می‌باشد.

```
<a href="photoalbum.html">گالری عکس</a>
```

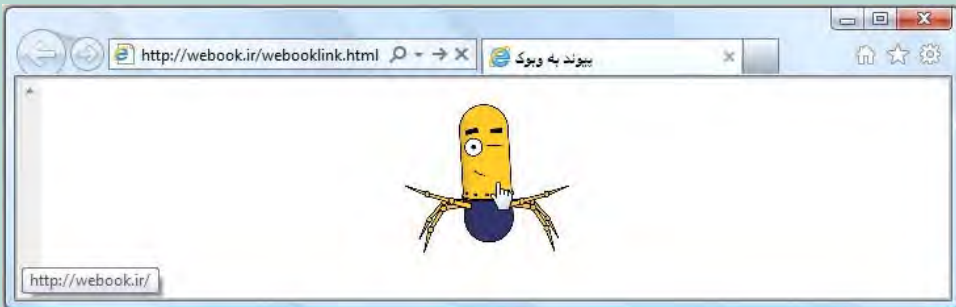
شما همچنین می‌توانید از عکس‌ها برای پیوند دادن به صفحات دیگر استفاده کنید. به این صورت که یک عکس به صورت پیوند ایجاد کنید تا زمانی که بر روی آن کلیک شد، کاربر به صفحه دیگری هدایت شود.



تمرین

تصویر وبوک را به صورتی در سایت خود قرار دهید تا زمانی که بر روی آن کلیک شد،

کاربر به سایت <http://webook.ir> وارد شود.



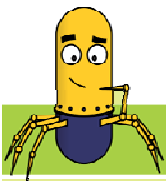
شکل ۲-۷: صفحه پیوند به وبوک

ویژگی target

گاهی ممکن است شما پیوند بخواهید پیوند سایتی دیگر را در صفحه خود بگذارید که کاربر با کلیک بر روی آن به سایتی دیگر برود ولی نمی‌خواهید در صورتی که این کلیک انجام شد، بازدید کننده از سایت شما خارج شود. به عبارت دیگر می‌خواهید وقتی بر روی آن پیوند کلیک شد، سایت مقصد در صفحه جداگانه برای بازدید کننده باز شود. شما می‌توانید برای اینکار از ویژگی target با مقدار blank استفاده کنید.

```
<a href="http://webook.ir" target="_blank">Webook</a>
```

مقدار این ویژگی به صورت پیش‌فرض مقدار self است.



بیشتر بدانیم!

در صورتی که با frame آشنا نباشید، لازم است بدانید این ویژگی می‌تواند مقادیر

_top و یا _parent را هم به خود بگیرد.

پیوند داخلی

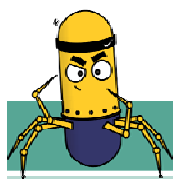
ممکن است صفحات وبای را دیده باشید که محتوای زیادی را داشته باشند و پیدا کردن مطلب مورد نظر شما در این بین سخت باشد. گاهی نیز وقتی شما به آخر صفحه می‌رسید می‌خواهید به راحتی به بالای صفحه برگردید. برای اینکار می‌توانید در صفحه خود پیوند داخل صفحه ایجاد کنید.

برای ایجاد پیوند داخلی شما قسمت مقصد را باید بین تگ `<a>` قرار داده و برای مقدار ویژگی name آن یک اسم انتخاب کنید.

```
<a name="section">مقصد</a>
```

حال محلی را که می‌خواهید زمانی که بر روی آن کلیک شد، به این مقصد برود را به صورت زیر مشخص کنید.

```
<a href="#section">مبدأ</a>
```



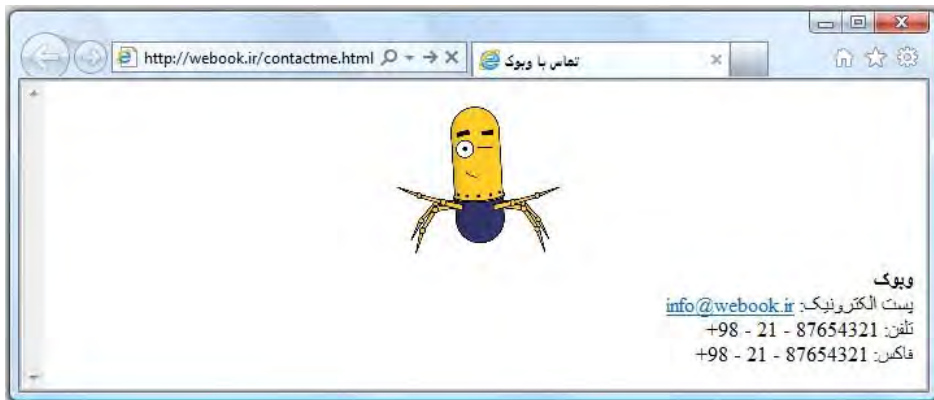
تمرین

با استفاده از آموزه‌های خود در این بخش، صفحه **درباره من** خود را به گونه‌ای تغییر دهید که بین تیترهای ابتدای صفحه و بخش‌های زیرین پیوند داخلی باشد.

۲-۷. صفحه تماس با من

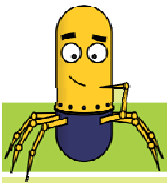
صفحه تماس با من، قسمت دیگری از وبسایت‌های شخصی است. در این قسمت اطلاعات تماس با صاحب سایت ذکر می‌گردد.

این اطلاعات شامل آدرس سایت و در صورت لزوم آدرس پستی، شماره تماس، ایمیل آدرس و قسمت‌های دیگر است.



شکل ۲-۸: صفحه تماس با من

تگ‌های لازم برای ایجاد این صفحه را در قسمت‌های قبل آموخته‌ایم.



بیشتر بدانیم!

این امکان وجود دارد تا وقتی بر روی آدرس ایمیل کلیک می‌کنیم، نرم‌افزار ارسال ایمیل باز شده تا بتوان برای آن آدرس ایمیل ارسال کرد.

برای اینکار کافی است تا آدرس ایمیل را درون تگ `<a>` نوشته و در قسمت href

عبارت `mailto:` را قبل از آدرس ایمیل ذکر می‌کنیم.

```
<a href="mailto:info@webook.ir">info@webook.ir</a>
```

تگ `<hr>`

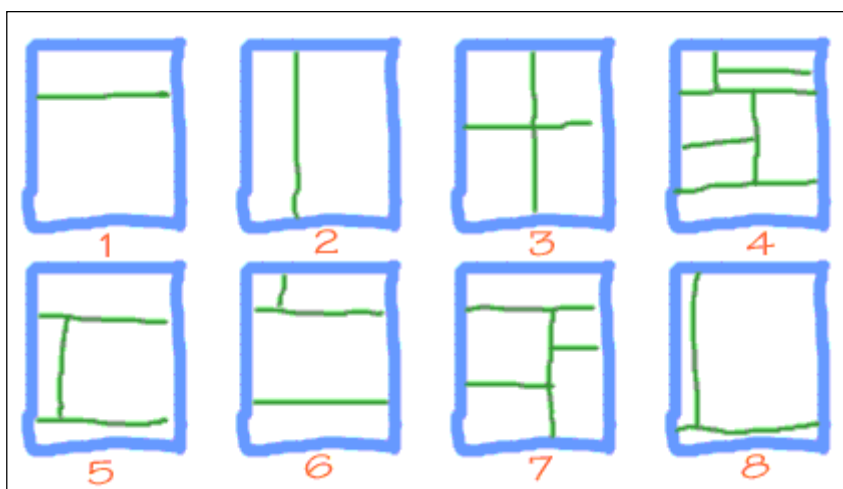
این تگ برای ایجاد یک خط افقی در صفحه استفاده می‌شود. وقتی از این تگ استفاده می‌شود یک خط افقی سرتاسری در محدودیت صفحه کشیده می‌شود.

۲-۸. صفحه خانه

بخش اصلی هر وبسایتی، صفحه خانه است. به اصطلاح صفحه اولی که کاربر از آن طریق وارد سایت شده و به بخش‌های دیگر دسترسی پیدا می‌کند، صفحه خانه نامیده می‌شود.

قبل از مطالعه این بخش از کتاب یک کاغذ و قلم بردارید و تمام قسمت‌هایی را که فکر می‌کنید وبسایت شخصی شما لازم است داشته باشد را بر روی کاغذ یادداشت کنید.

سپس به گشتی در سایت‌های مختلف زده و یک شکل حدودی از صفحه خانه خود بر روی کاغذ ترسیم کنید.



شکل ۲-۹: مثالی هایی از انواع تقسیم بندی صفحات

تگ `<table></table>`

تا کنون تگ های مختلف را برای قرار دادن محتوا در وبسایت خود فرا گرفتید. حال لازم است تا نظمی در نمایش آن ها ایجاد کنید. برای این کار از تگی به نام `<table></table>` استفاده می شود.

در ابتدا ساده ترین جدول، یعنی یک جدول دو در دو را ترسیم می کنیم.

به هر یک از خانه های جدول سلول^{۲۴} گفته می شود. به عنوان مثال جدول فوق دارای ۴ سلول است. همچنین جدول فوق دارای ۲ سطر و ۲ ستون می باشد.

تگ `<tr></tr>`

به ازای هر سطری از جدولی که می خواهیم ترسیم کنیم باید یک تگ `<tr></tr>` درون تگ `<table></table>` نوشته شود. پس کد جدول فوق تا اینجا به شکل زیر است.

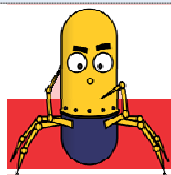
```
<table>
  <tr>
  </tr>
  <tr>
  </tr>
</table>
```

²⁴ Cell

تگ `<td></td>`

این تگ برای ایجاد سلول در هر سطر استفاده می‌شود. پس لازم است تا در هر یک از تگ‌های `<tr></tr>` دو بار این تگ نوشته شود.

```
<table>
<tr>
  <td></td>
  <td></td>
</tr>
<tr>
  <td></td>
  <td></td>
</tr>
</table>
```



نکته مهم!

توجه کنید که تنها عباراتی که بین `<td></td>` نوشته می‌شوند به صورت صحیح در سلول نشان داده می‌شود و هر عبارتی که خارج از این تگ درون جدول نوشته شود، برحسب نوع مرورگر جایی خارج از جدول نمایان خواهد شد.

تگ `<table></table>` دارای ویژگی به نام `border` است که مقدار آن عددی بر حسب `pixel` می‌باشد. به این ترتیب `border="0"` سبب می‌شود تا جدول به صورت فرضی ترسیم شود و خطوط آن قابل مشاهده نباشد.

همچنین برای رسم جدول با ضخامت مرز `1pixel` باید از ویژگی `border="1"` استفاده کرد.

ویژگی colspan

گاهی لازم است تا دو سلول کنار هم در یک سطر را به هم متصل کنیم. برای این کار کافی است تا فقط تگ `<td></td>` اولی را نوشته و در به عنوان ویژگی colspan آن مقدار ۲ را قرار دهیم. این عمل بدین معناست که باید این سلول معادل دو سلول سطرهای دیگر قرار بگیرد.

```
<table>
<tr>
  <td></td>
  <td></td>
</tr>
<tr>
  <td colspan="2"></td>
</tr>
</table>
```

همینطور برای اتصال سلولهای بیشتر درون یک سطر می‌توانیم از مقادیر بالاتر برای ویژگی colspan استفاده کنیم.

ویژگی rowspan

برای اتصال دو سلول زیر هم از ویژگی rowspan استفاده می‌شود. برای اینکار لازم است که تگ `<td></td>` سلول پایینی را حذف کنیم و در سلول بالایی از ویژگی rowspan با مقدار ۲ استفاده کنیم.

```
<table>
<tr>
  <td></td>
  <td rowspan="2"></td>
</tr>
<tr>
```

```
<td></td>
</tr>
</table>
```

با استفاده از این ترکیب این دو ویژگی، شما می‌توانید هر جدولی را که می‌خواهید ایجاد کنید.

Width , height

۲-۹. صفحه ورود اعضا

در اکثر وب‌سایت‌ها بخشی به نام ورود اعضا وجود دارد. در این قسمت نام کاربری و کلمه عبور از کاربر گرفته شده و در صورت تطابق با داده‌های موجود در سایت کاربر مجاز است که وارد قسمت شخصی خود در سایت شود.

این قسمت شامل دو بخش است. بخش اول دریافت اطلاعات در فرم از کاربر و بخش دوم پردازش آن در سرور و ارجاع به صفحات بعدی.

ایجاد بخش دوم نیاز به دانش PHP دارد که در فصل بعد توضیح داده خواهد شد. در این قسمت سعی بر آن است تا فرم ورود نام کاربری و کلمه عبور را ایجاد کنیم.

شکل ۲-۱۰: صفحه ورود اعضا

تگ `<form></form>`

تگ فرم دربرگیرنده بخش‌هایی است که کاربر می‌تواند در صفحه آن موارد را وارد کند.

این تگ دو ویژگی مهم `action` و `method` را داراست. `action` مشخص کننده آدرس صفحه‌ای است که عبارات وارد شده کاربر باید برای آن ارسال شود. `method` نیز نحوه ارسال را مشخص می‌کند.

ویژگی `method`

ویژگی `method` می‌تواند یکی از مقادیر `get` و `post` را دارا باشد. در نحوه ارسال فرم به روش `get` اطلاعات فرم به صورت زیر به آدرس اضافه می‌شوند.

در روش ارسال `post` اطلاعات وارد شده فرم به صورت کد شده ارسال می‌گردند.

به نظر شما کدام مورد برای ارسال فرم‌ها مناسب‌تر است؟

تگ `<input>`

حال برای ایجاد هر قسمت ورود اطلاعات متنی از تگ `<input>` استفاده می‌کنیم. این تگ دارای ویژگی به نام `type` می‌باشد. برای ورود نام کاربری مقدار `text` و برای ورود کلمه عبور مقدار `password` را این ویژگی قرار می‌دهیم.

همچنین هر فرم باید یک دکمه ارسال نیز داشته باشد. دکمه ارسال نیز با تگ `<input>` ایجاد می‌شود و مقدار ویژگی `type` آن برابر `submit` است.

ویژگی name

هر `<input>` باید دارای یک نام باشد تا به وسیله آن نام مقدار آن در صفحه مقصد قابل خواندن باشد. این نام توسط ویژگی name در تگ `<input>` مشخص می‌شود.

ویژگی value

فرم‌ها ممکن است در ابتدا دارای اطلاعاتی پیش‌فرض در خود باشند. برای تنظیم این اطلاعات پیش‌فرض از ویژگی value در تگ `<input>` استفاده می‌کنیم.

با توجه به اطلاعات بالا کد قسمت ورود به سایت به صورت زیر است.

```
<form action="form.php" method="post">
  Username:<input type="text" name="username"><br>
  Password:<input type="password" name="password"><br>
  <input type="submit" value="Login">
</form>
```

۲-۱۰. صفحه ثبت نام خبرنامه

با توجه به اینکه سر زدن روزانه به تمامی سایت‌ها ممکن نمی‌باشد، در سایت‌ها بخشی به نام خبرنامه ایجاد می‌شود تا کاربران با ثبت نام در آن از اخبار و اطلاعات جدید سایت با خبر شوند. کاربران با توجه به علایق خود می‌توانند در سایت‌های مختلف عضو شوند و این بر عهده طراح سایت است تا قسمتی برای جمع‌آوری اطلاعات بازدید کنندگان در سایت خود ایجاد کنند.

در بخش خبرنامه‌ها آدرس پست الکترونیک کاربر و اطلاعات شخصی دیگر از او گرفته می‌شود و در زمان به روز شدن سایت به کاربر از طریق ایمیل اطلاع داده می‌شود.

ایجاد این قسمت نیاز به طراحی یک فرم برای دریافت اطلاعات از کاربر و یک صفحه برای ذخیره این اطلاعات دارد. صفحه اول با معلومات HTML قابل طراحی است ولی صفحه دریافت را در بخش PHP فرا می‌گیریم.

شکل ۲-۱۱: صفحه اشتراک خبرنامه

هر فرم از اجزایی مانند صفحه بالا ایجاد شده است. در این قسمت با اجزای مختلف فرم‌ها آشنا می‌شویم.

ورودی‌های متنی^{۲۵}

قسمت‌ها نام، نام خانوادگی و پست الکترونیک در فرم بالا مثالی برای اینگونه ورودی‌ها هستند. در این قسمت می‌توان اطلاعات کوتاه متنی از کاربر گرفت.

شکل ۲-۱۲: نمونه ورودی متنی

کد این نوع ورودی به این صورت می‌باشد.

²⁵ Textbox

```
<input type="text" name="username" value="Enter username here.">
```

ویژگی type در این تگ نوع ورودی را مشخص می‌کند. ویژگی name نام این فیلد را تعیین می‌کند و در قسمت value می‌توان به عنوان پیش‌فرض متنی در این فیلد بنویسید.

ورودی گزینه‌ای^{۲۶}

می‌توان در فرم‌ها یک قسمت انتخاب بین چند گزینه ایجاد کرد که کاربر باید یکی از آن‌ها را انتخاب کند. این نوع فیلد radio box نام دارد و به صورت زیر نوشته می‌شود.



شکل ۲-۱۳: نمونه ورودی گزینه‌ای

```
<input type="radio" name="gender" value="Male">
```

برای استفاده از radio box ها در یک گروه باید آنها دارای name یکسان باشند و در مقصد از روی مقدار آن می‌توان متوجه شد که کدام گزینه انتخاب شده است.

ورودی انتخابی^{۲۷}

ورودی انتخابی به صورتی است که یک لیست از انتخاب‌ها پیش روی بازدید کننده قرار می‌دهد و کاربر مجاز است که یکی از آن‌ها را انتخاب کند.



شکل ۲-۱۴: نمونه ورودی انتخابی

²⁶ Radio box

²⁷ Select

برای ایجاد این ورودی در فرم باید از تگ `<select></select>` استفاده کنیم. به ازای هر یک از گزینه‌ها، گزینه مربوطه را در تگ `<option></option>` می‌نویسیم.

```
<select name="set">
  <option value="daily">روزانه</option>
  <option value="weekly">هفتگی</option>
  <option value="monthly">ماهانه</option>
</select>
```

ورودی توضیحی^{۲۸}

گاهی در فرم‌ها لازم است تا توضیحاتی بیش از چند کلمه از بازدید کننده گرفته شود. برای این فیلدها می‌توان از تگ `<textarea></textarea>` به صورت زیر استفاده کرد.

شکل ۲-۱۵: نمونه ورودی توضیحی

```
<textarea name="comments">
</textarea>
```

ورودی تاییدی^{۲۹}

این نوع ورودی برای گرفتن گزینه‌های تاییدی از کاربر است. به صورتی که می‌توان درون آن را علامت زد و یا علامت را برداشت.

شکل ۲-۱۶: نمونه ورودی تاییدی

برای ایجاد یک ورودی تاییدی می‌توان از کد زیر استفاده کرد.

²⁸ Textarea

²⁹ Checkbox

```
<input type="checkbox" name="confirm">
```

دکمه‌ها^{۳۰}

معمولا در پایین هر فرم HTML یک دکمه ثبت و یا ارسال اطلاعات فرم وجود دارد. زمانی که کاربر بر روی این دکمه کلیک می‌کند، اطلاعات آن فرم به صفحه مشخص شده، ارسال می‌گردد. فرم ارسال یک فیلد با تگ <input> و از نوع submit است.



شکل ۲-۱۷: نمونه دکمه‌های ارسال و از نو

برای ایجاد یک دکمه ثبت و یا ارسال در فرم، از تگ زیر در <form></form> استفاده می‌کنیم.

```
<input type="submit" value="ثبت">
```

مقدار ویژگی value در این تگ، مشخص کننده متن نوشته شده بر روی دکمه می‌باشد.

در فرم‌ها یک دکمه نیز برای پاک کردن فرم می‌توان قرار داد. به اینصورت که کاربر اگر قسمتی را اشتباه پر کرد، با کلیک کردن بر روی این دکمه، تمامی اطلاعات موجود در فرم پاک شده و فرم برای پر شدن مجدد، آماده گردد.

با استفاده از کد زیر می‌توانید یک دکمه از نو در فرم خود ایجاد کنید.

```
<input type="reset" value="از نو">
```



آشنایی با CSS

۳-۱. نیاز به CSS از کجا آغاز شد

فرض کنید وب سایت شخصی خود را طراحی کردید و بخواهید فونت تیتريهای صفحات خود را تغییر دهید با مطالبی که شما در فصل دوم مطالعه کرده اید برای شما کارچندان دشواری نمی باشد زیرا به راحتی به کدهای صفحات HTML خود مراجعه کرده و هر جا که تیتري را دیدید تگ فونت را پیدا کرده و فونت خود را تغییر می دهید. فرض کنید صفحات شخصی شما صفحات سایتی باشد برای یک شرکت یا یک جای سازمان بزرگ که نویسنده و طراح آن سایت شما باشید و تعداد صفحات آنجا بیش از ۱۰۰ صفحه باشد و کارفرما از شما بخواهد تیتريهای متن را با فونت دیگری تغییر دهد چه کاری باید کرد؟ آیا منطقی این است مانند قبل دنبال تگ فونت در هر صفحه گشت؟

برای حل کردن این مشکل مجموعه ی W3C استاندارد CSS را پیشنهاد داده است. با استفاده از این استاندارد، ساختارهای سبک نمایش صفحه در فایل با پسوند CSS. ذخیره می شود و همراه فایل اصلی HTML ارسال می شود. بدین ترتیب برای تغییرات تنها کافی است این فایل CSS مورد بازبینی قرار گیرد.

ساختار صفحات وب

در روزهای اولیه صفحات وب طرح HTML بعنوان ساختاری منطقی برای متون داخل

صفحه ها بوجود آمده بود البته هیچ مرورگری در آن زمان برای نمایش این ساختار وجود نداشت اما با پیشرفت شبکه و استفاده مردم از این صفحات طراحان برای جذاب کردن بازدیدکنندگان از صفحات خود، شروع به استفاده بی رویه از این تگها را کردن و از آنجایی که هدف اصلی HTML برای نمایش دادن و شکل دادن به صفحه در نظر گرفته نشده بود. ساختار طراحی صفحات به صورت پیچیده از تگهای بی رویه تشکیل شده بود. بنابراین همین امر موجب شد که بحث CSS و ساختار صفحات وب شکل بگیرد.

اجزای صفحات وب

اجزای یک صفحه وب را می توان به سه لایه تقسیم بندی کرد

۱- محتوا: که بوسیله تگهای HTML ایجاد و فرمت دهی می شوند.

۲- نمایش: که به وسیله CSS ایجاد می شوند.

۳- رفتار: که بوسیله JAVASCRIPT ایجاد می گردد.

۲-۳ CSS چیست

CSS مخفف Cascading Style Sheets می باشد و زبانی است برای مشخص کردن ویژگی های ظاهری یک وب سایت بنابراین توسط CSS شما این قدرت را خواهید داشت که ظاهر وب سایت خود را هر گونه که بخواهید تغییر دهید و لایه نمایش را از لایه محتوا جدا کنید و بتوانید راحت تر ظاهر صفحات را کنترل نمایید.

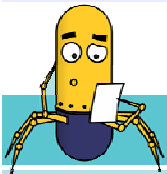
چگونه از دستورات css در صفحات html می توان استفاده کرد

برای نوشتن CSS ما می توانیم برحسب کارایی به سه صورت در وب سایت خود تعریف نماییم.

۱- به صورت داخلی در این نوع، style نوشته شده در تمامی صفحه تاثیر خواهد گذاشت، از این رو بین دو تگ `<head></head>` به صورت زیر نوشته خواهد شد.

```
<style type="text/css">
.....
</style>
```

۲- به صورت خطی نوشته شده است این نوع style در داخل تگ‌های HTML به کار گرفته می‌شوند و فقط بر روی همان تگ به خصوص تاثیر خواهد گذاشت.



مثال

ظاهر صفحه را با استفاده از تگ پاراگراف طوری تغییر دهید که کل متن پاراگراف صفحه به رنگ آبی تبدیل شود.

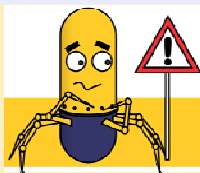
```
<p style="color:red">....</p>
```

۳- به صورت خارجی بیرون صفحه داخل یک فایل با پسوند CSS بنویسیم و با یک دستور آن را به صفحه به صورت زیر لینک کنیم.

```
<link rel="stylesheet" type="text/css" href="myfile.css" >
```

یکی از مزیت‌های این نوع style این می‌باشد که می‌توانید با داشتن یک فایل style برای

چندین صفحه استفاده کنید و از دیگر مزیت‌های آن با این کار کدهای قالب را به چند فایل تقسیم کرده و قالب را منظم‌تر می‌شود.



توجه!

مزیت‌های عمده استفاده از CSS

- ۱- سرعت طراحی صفحات به مراتب بالا می‌رود.
- ۲- اعمال تغییرات به راحتی و در زمان کوتاه انجام می‌شوند.
- ۳- حجم کدنویسی کم می‌شود.
- ۴- زمان بارگذاری صفحه کم می‌شود.

نحو CSS

ساختار سبک‌های تعریف شده در فایل CSS از عباراتی سه قسمتی شامل انتخاب‌گر، خاصیت و مقدار تشکیل شده است. این محتوا بصورت زیر در فایل قرار می‌گیرد:

h1

{ color:blue; font-size:12px; }

انتخابگر

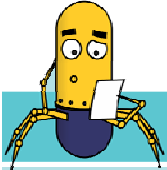
خاصیت

مقدار

خاصیت

مقدار

انتخاب‌گر معمولاً جزء تگ‌های HTML است و خاصیت نیز مجموعه‌ای از خاصیت‌های مربوط به تگ انتخاب‌گر می‌باشد. مقادیر مجاز خاصیت، محتوایی هستند که باید در قسمت مقدار قرار گیرند.



مثال

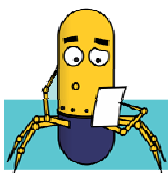
برای اینکه بخواهیم رنگ پس زمینه کل صفحه را تغییر دهیم می‌توانیم از تکه کد زیر استفاده کنیم.

```
body {color: black}
```

در صورتی که بخواهیم سبک های متفاوت دیگری را به یک تگ اختصاص دهیم باید یکی از روشهای زیر را پیاده‌سازی نماییم.

روش اول: با استفاده از ClassSelector

ما با استفاده از این روش یک کلاس (مجموعه) برای یک تگ در CSS تعریف می‌کنیم و سپس یک خاصیت را به یک کلاس اختصاص می‌دهیم. با استفاده از این روش می‌توانیم خاصیت‌های متفاوتی را به یک تگ در صفحه اختصاص دهیم. قبل از نام کلاس کاراکتر نقطه قرار می‌گیرد و در صورتی که تگی برای آن در نظر گرفته نشود برای تمامی تگهایی که از این کلاس استفاده می‌کنند قابل استفاده می‌باشد.



مثال

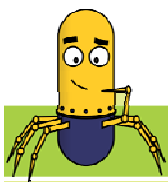
می‌خواهیم در صفحه دو نوع پاراگراف داشته باشیم، یکی با متن مشکی و دیگری با متن قرمز. برای مشخص کردن این دو نوع پاراگراف از ساختار کلاس استفاده می‌کنیم.

```
p.black {color: black}
p.red {color:red}
```

بعد از نوشتن خطوط بالا به یکی از سه شکلی که توضیح داده شد، باید برای هر پاراگرافی که می‌خواهیم به رنگ مشکی باشد از شناسه `class="black"` در تگ `<p>` و برای پاراگراف قرمز از شناسه `class="red"` استفاده کنیم.

البته نام کلاس‌ها اختیاری است ولی بهتر است آنها را طوری انتخاب کنید که مفهوم داشته باشند تا وقتی که حجم `style` شما بیشتر شد گنگ نباشد.

```
<p class="black">این متنی است که به رنگ مشکی نمایش داده می‌شود</p>
<p class="red">این متن به رنگ قرمز نمایش داده می‌شود</p>
```



بیشتر بدانیم!

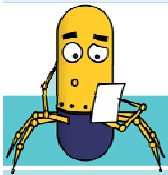
ما می‌توانیم بیش از یک کلاس برای یک تگ HTML تعریف کنیم. اما برای این کار نباید دو بار از شناسه `class` استفاده کرد. بلکه باید در یک شناسه `class` نام هر دو مورد را با

یک فاصله بین آنها وارد کرد.

اگر بخواهیم کلاسی برای همه تگ‌ها عمل کند کافی است انتخابگر کلاس را بدون تگ بنویسیم و در برنامه برای هر تگی که بخواهیم از نام این انتخابگر در عنوان کلاس آن استفاده کنیم.

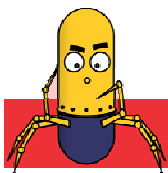
روش دوم : با استفاده از ID Selector

استفاده از ID همانند خاصیت کلاس می‌باشد که ترجیحا برای انتخاب گروهی تگ‌ها که خاصیت مشترک دارند مورد استفاده قرار می‌گیرد.



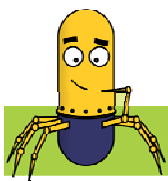
مثال

```
#border { border: green thin solid }
<h3 id="border"> این یک متن است قابدار با رنگ سبز است </h3>
#parag
{
text-align:center;
color:red;
}
<p id="parag">سلام دوستان</p>
```



نکته مهم!

در هر صفحه نباید بیش از یک بار از هر id استفاده شود. اگر بیش از یک بار از یک id استفاده شود، کارایی صفحه تغییری نمی‌کند ولی صفحه اعتبار خود را از دست می‌دهد. انتخابگر id با علامت # مشخص می‌شود.



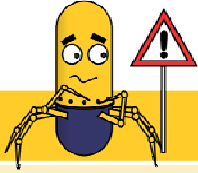
بیشتر بدانیم!

کلاسهای کاذب

در صورتی که بخواهیم در یک تگی که با یک کلاس از قبل تعریف شده رفتار آن را با کلاس جدید عوض کنیم، از کلاسهای کاذب به صورت زیر استفاده می‌کنیم. در اصل کلاسهای کاذب رفتار تگهای دیگر را تغییر می‌دهند.

کلاسهای کاذب با : آغاز می‌شوند که نمونه بارز آن در تگ لینک قابل مشاهده می‌باشد. زیرا بر چسب a در HTML نسبت به بقیه از تنوع بیشتری برخوردار است. با تعیین خصوصیات Link , Vlink , Alink در body می‌توانید رنگ لینکهای خود را تغییر دهید.

```
a:link { color #FF0; }
a:visited { color : #FF00FF; }
a:hover { color : 00ccff; }
a:active { color : #FF0000; }
```



توجه!

کلاس و ID را می‌توانیم برای یک عنصر مشخص تعریف کنیم.

```
p.border{border:green}
p#border{border:green}
```

۳-۳. قواعد کار در css

با یادگیری پیاده‌سازی CSS در صفحه خود و با دانستن مجموعه قواعد کاری در این زبان و با استفاده از عناصری مانند مدل، اندازه، رنگ و نوع آرایش متن، رنگ زمینه و حاشیه صفحات و جدول‌ها، تنظیمات لیست‌ها و همچنین عکس‌ها را می‌توانید کنترل کنید. در این قسمت می‌خواهیم با استفاده از این قواعد عناصر صفحات خود را به راحتی تغییر دهیم.

قواعد مربوط به رنگ و پس زمینه صفحه

مهمترین قسمتی که می‌توان با استفاده از آن تغییرات جدی در صفحه ایجاد کرد، قواعد مربوط به رنگ و پشت زمینه عناصر در صفحه می‌باشد. با استفاده از قواعد این بخش شما به راحتی می‌توانید برای عناصر صفحه خود رنگ و تصویر، تکرار و موقعیت تعیین نمایید.

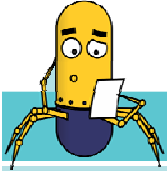
Background-image

این ویژگی برای مشخص کردن تصویر زمینه به صورت زیر به کار می‌رود.

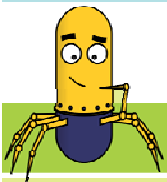
```
Background-image:url
```

Background-repeat

این ویژگی تعیین کننده نوع تکرار شدن تصویر زمینه است. به مثال زیر جهت استفاده این دو دستور توجه کنید.

**مثال**

```
<html>
<head>
  <style type="text/css">
    body
    {
      background-image:url('webook.png');
      background-repeat:repeat-x;
    }
  </style>
</head>
<body>
  <h1>Hello webook</h1>
</body>
</html>
```

**بیشتر بدانیم!****Repeat**

این مقدار تعیین کننده این است که در صورتی که تصویر زمینه از اندازه پنجره کوچکتر بود، هم به صورت افقی و هم به عمودی تکرار شود تا همه زمینه پنجره را بپوشاند.

No-repeat

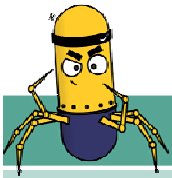
این گزینه مشخص می‌کند که تصویر زمینه نباید تکرار شود. حتی اگر از اندازه پنجره کوچکتر باشد.

Repeat-x

این گزینه مشخص می‌کند که تصویر باید به صورت افقی تکرار شود و در راستای عمودی تکرار نشود.

Repeat-y

این گزینه مشخص می‌کند که تصویر باید به صورت عمودی تکرار شده و در جهت عمودی تکرار نشود.

**تمرین**

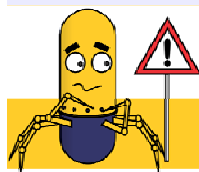
در این قسمت سعی کنید عکسی برای پشت زمینه صفحه شخصی خود ایجاد نمایید و آن را برای کل صفحه قرار دهید.

Background-attachment

از این ویژگی به صورت زیر برای تعیین موقعیت تصویر زمینه بالا و پایین بردن صفحه به وسیله نوار جابجاگر^{۳۱} در پنجره مرورگر استفاده کرد.

`background-attachment: fixed;`

³¹ Scrollbar



توجه!

Scroll

اگر این گزینه را انتخاب کنید تصویر زمینه به همراه محتویات صفحه به بالا و پایین می‌رود.

fixed

با انتخاب این گزینه مشخص می‌کنیم که تصویر در هنگام حرکت کردن محتویات صفحه باید ثابت بماند.

قواعد مربوط به لینکها

برای کار با لینکها در CSS می‌توانیم مشخصات یک لینک را به چهار بخش تقسیم کنیم که در زیر می‌توانیم آنها را مشاهده کنید.

۱- a:link

این مشخصه‌ها نحوه لینک‌هایی را که در صفحه وجود دارند و هنوز روی آنها کلیک نشده است تنظیم می‌کنند.

۲- a: hover

این قسمت مشخص کننده وضعیت لینکی است که بر روی آن کلیک می‌شود.

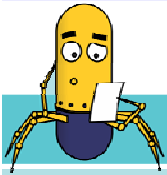
۳- a:visited

این مشخصه برای لینک‌هایی که در بازدیدهای قبلی مورد استفاده قرار گرفته‌اند، استفاده می‌شود.

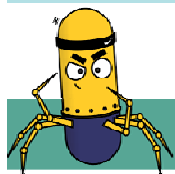
۴- a:active

این مشخصه برای لینک‌هایی که در بازدیدهای قبلی مورد استفاده قرار گرفته‌اند، استفاده می‌شود.

به مثال زیر توجه کنید:

**مثال**

```
<html>
<head>
  <style type="text/css">
    a:link {color:#FF0000;} /* unvisited link */
    a:visited {color:#00FF00;} /* visited link */
    a:hover {color:#FF00FF;} /* mouse over link */
    a:active {color:#0000FF;} /* selected link */
  </style>
</head>
<body>
  <p>
    <b>
      <a href="#" target="_blank">This is a link</a>
    </b>
  </p>
</body>
</html>
```



تمرین

سعی کنید با توجه به قواعدی که یاد گرفتید، منوی اصلی صفحه خود را طوری تغییر دهید که با کلیک کردن آن تصویر زمینه آن عوض شود.

قواعد مربوط به متن

در این قسمت برخی از ویژگی‌های متن را که در CSS استفاده می‌شوند، به شما معرفی می‌کنیم. با کمک این قواعد می‌توانیم نحوه قرار گرفتن متن در صفحه، فاصله بین حروف و بسیاری از خواص یک متن را تغییر دهیم.

Letter-spacing

این ویژگی به ما اجازه می‌دهد تا فاصله میان حروف یک کلمه را در یک متن تعیین کنیم. این ویژگی می‌تواند مقداری را در مقیاس پیکسل به خود بگیرد یا از مقدار پیش‌فرض normal استفاده کند.

```
h1
{
  letter-spacing: -1px;
}
text-decoration
```

این ویژگی به ما اجازه می‌دهد تا برای قسمت خاصی زیر خط تعریف کنیم. به عنوان

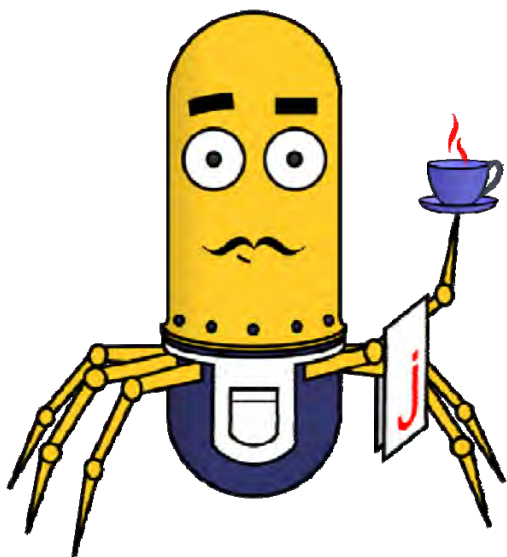
مثال می‌توانیم برای تگ لینک به این صورت بنویسیم.

```
a:link {  
    font-weight: bold;  
    text-decoration: none;  
}  
a:visited {  
    font-weight: bold;  
    text-decoration: line-through;  
}  
a:hover, a:focus {  
    text-decoration: underline overline;  
}
```

تا اینجا مطالبی مختصر از قواعد مورد استفاده شما برای کار با CSS مطرح شد لذا چنانچه می‌خواهید سایتی را با استفاده از این تکنولوژی طراحی کنید لازم است قواعد بیشتری برای سبک نمایش سایت خود بدانید و استفاده کنید.

فصل سوم

برنامه نویسی JAVASCRIPT



برنامه نویسی JAVASCRIPT

پویا سازی صفحات

اطلاعات موجود در صفحات وب می‌تواند بصورت ایستا و پویا ایجاد گردند. در مواردی که اطلاعات بصورت ایستا در صفحات وب قرار می‌گیرند، مولفین صفحات از قبل آنها را آماده کرده و بر روی یک هاست ذخیره می‌نمایند. همانطور که در فصل قبل مشاهده کردید در این روش نظرات بازدیدکنندگان سایت هیچگونه دخالتی در محتویات سایت ندارد و همچنین صفحات سایت به صورت ثابت می‌باشد و هیچ پویایی ندارد و بازدید کننده بعد از مدتی از صفحه سایت کسل کننده شما خارج می‌شود. (البته به محتوای صفحه شما نیز مهم است).

در صورتی که سایت شما یک صفحه پویا داشته باشد شما می‌توانید با شرایط متعدد و سفارشی، وب سایت را برای بازدیدکننده طوری طراحی نمایید که ثابت و خسته کننده نباشد.

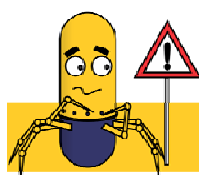
۴-۱. ساختار زبان‌های برنامه نویسی تحت وب

اگر بخواهیم کامپیوترهای شبکه اینترنت را تقسیم‌بندی کنیم، می‌توانیم آنها را به دو دسته کلی تقسیم کنیم.

کامپیوتر درخواست کننده (کاربر)

کامپیوتر سرویس دهنده (سرور)

بنابراین برای ایجاد صفحات پویا و متحرک نیز دو نوع زبان برنامه‌نویسی تولید گردید. زبان‌هایی که توسط کامپیوتر سرویس‌دهنده تفسیر^{۳۲} می‌شوند و همچنین زبان‌هایی که توسط کامپیوتر سرویس گیرنده پردازش می‌شود که اصطلاحاً به این زبان‌ها، زبان‌های سمت سرویس دهنده و زبان‌های سمت سرویس گیرنده نامیده می‌شوند که از توضیح بالا می‌توانیم تفاوت این دو زبان را درک کنیم.



توجه!

برنامه‌های سمت سرویس گیرنده هرگز توسط سرویس دهنده وب پردازش نمی‌شوند بلکه توسط مرورگر خود شما پردازش می‌شوند یعنی سرویس دهنده برای شما هیچ عملی انجام نمی‌دهد و تنها مرورگر کامپیوتر شماست که کدها را خط به خط می‌خواند و به شما نشان می‌دهد.

برنامه‌های سمت سرویس دهنده تماماً بر روی سرویس دهنده (هاست) پردازش می‌شود و تنها خروجی آن برای درخواست کننده ارسال می‌شود.

جاوا اسکریپت چیست؟

JavaScript دقیقاً یک زبان برنامه‌نویسی سمت درخواست کننده می‌باشد که از پرکاربردترین و عمومی‌ترین زبان‌های اسکریپتی در دنیا محسوب می‌شود به وسیله آن می‌توان بین کاربر و سایت ارتباط برقرار نمود و صفحاتی متحرک و پویا ایجاد نمود.

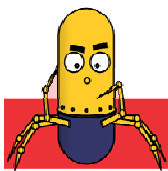
³² Compile

جاوا اسکریپت در میلیونها صفحه وب جهت بهبود طراحی، اعتبار سنجی فرمها، نمایش اطلاعات مرورگر، و چیزهای زیاد دیگری مورد استفاده قرار می گیرد.

۴-۲. برای نوشتن برنامه به زبان جاوا اسکریپت از کجا باید شروع کنیم

می توان هم به صورت مستقیم در فایل HTML قرارداد (بصورت بلوکی از کد بین دو تگ مشخص script)

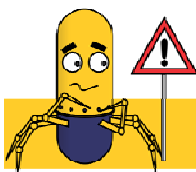
یا اینکه دستورات و توابع را در فایل دیگری نوشت و سپس به آن ارجاع داد.



نکته مهم!

به دلیل خط به خط خوانده شدن فایل HTML (برای به نمایش در آمدن محتوای آن)، باید کد مورد نظر جاوا اسکریپت را در جای مناسبی قرار دهید.

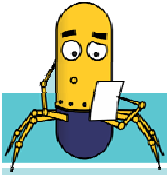
برای شروع این مثال را نوشته صفحه را با مرورگر خود باز کنید.



توجه!

برای نوشتن کدهای جاوا اسکریپت از هر ویرایشگر متنی می توان استفاده نمود.

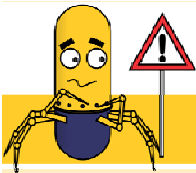
مثال



حال در این مثال یک پیام خوش آمد گویی برای بازدید کننده نمایش داده می شود.

```
<html>
<head>
<script>
  alert("به فصل سوم کتاب خوش آمدید");
</script>
</head>
<body>
  <h1> page heading </h1>
</body>
</html>
```

در این دستور از تگ script جهت نوشتن اولین دستور در زبان جاوا اسکریپت استفاده کردیم. به جهت معروفیت این زبان می توانید خصوصیت های آن را ننویسید.



توجه!

تگ script مانند بقیه تگها دارای خصوصیتی می باشد، یکی از آنها language است که مقدار آن را باید برابر با javascript قرار دهید، همانند <"script language="javascript>

خصوصیت type هم دارد که برای این زبان باید آنرا برابر با text/javascript قرار دهید.

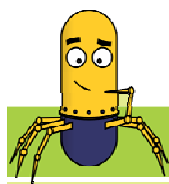
در صورتی که مثال فوق را اجرا کنید می بینید پنجره ای با پیغام نوشته شده بر روی آن برای شما نمایان می شود

دستورات ساختاری زبان جاوا اسکریپت بسیار شبیه دستورات زبان سی پلاس پلاس که در سال اول خوانده اید می باشد

در صورتی که به مفاهیم وب و تگهای html تسلط داشته باشید به راحتی می توانید این زبان را بیاموزید . بنابراین در موقع نوشتن برنامه های این کتاب ویا نوشتن برنامه ای به عنوان تمرین برای خودتان فکر کنید که دارید با زبان سی برنامه نویسی می کنید البته تفاوت هایی دستوری هم وجود دارد که ما در این کتاب سعی می کنیم آن تفاوتها را بازگو کنیم.

۴-۳. شی گرایی در جاوا اسکریپت

زبان جاوا اسکریپت یک زبان شی گرا است این زبان به برنامه نویس ، قابلیت استفاده از اشیای پیش ساخته را می دهد . که شما می توانید با استفاده از آنها وب سایت هایی با جذابیت فوق العاده ایجاد کنید.



بیشتر بدانیم!

یک شی (object) چیست؟

اشیا کلید فهم تکنولوژی شی گرایی است . شما می توانید با نگاه به محیط اطراف خود

مثال های زیادی از اشیای دنیای واقعی را ببینید

دستگاه تلویزیون - دوچرخه - گربه - درخت

این اشیای واقعی دارای دو مشخصه هستند :

ویژگی و رفتار

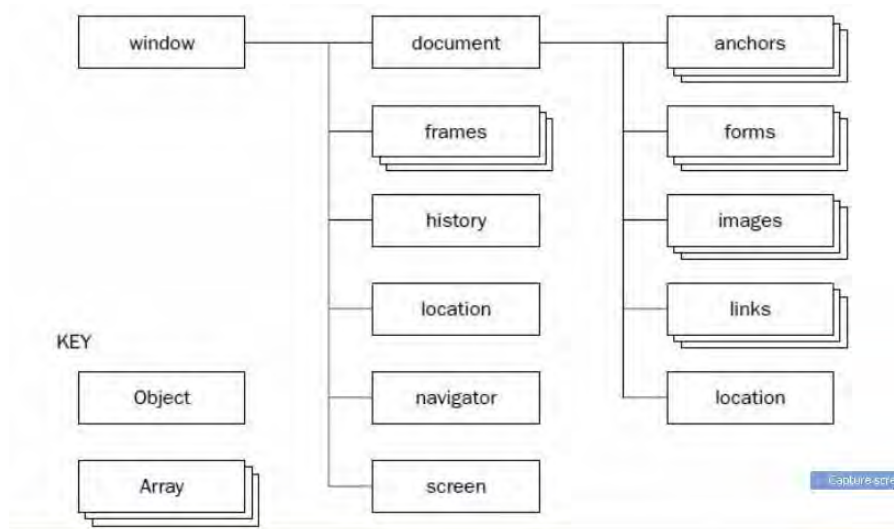
اشیای یک برنامه نرم افزاری هم همانند اشیای دنیای واقعی مدل می شوند و دارای ویژگی ها و رفتار ها هستند.

ویژگی یک شی برنامه نرم افزاری با یک یا چند متغیر از نوع صحیح - اعشاری - کاراکتری یا انواع دیگر بیان میشود.

رفتار های یک شی برنامه نرم افزاری با متد (method) بیان میشود . متد یک تابع (function) یا مرتبط با شی می باشد.

پس یک شی یک مجموعه از متغیر ها و متد ها است.

شی اصلی در جاوا اسکریپت، شی window بوده که خود دارای ۶ شی مهم و اصلی می باشد. با کمک این اشیا می توان تمام فرآیندها را در صفحه کنترل کرد. ساختار اشياء پیش ساخته این زبان را می توانید به صورت زیر مشاهده نمایید.

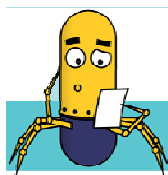


شکل ۴-۱: ساختار اشیای پیش‌ساخته زبان JAVASCRIPT

آشنایی با شی document

در واقع این شی خصلتی از شی اصلی در window می‌باشد. واز مهمترین شی های پیش ساخته زبان جاوا اسکریپت است این شی به ارائه اشیا و خصوصیات و متدهایی برای کار کردن با تمام ویژگی‌های تگها، مانند فرم‌ها و پیوندها و کلیدها و ... می‌پردازد

متد document.write()

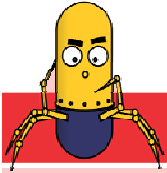


مثال

در این قسمت برنامه‌ای می‌نویسیم که متنی را داخل متغیر قرار داده و سپس آن را با استفاده از متد `document.write` چاپ می‌نماید.

```
<script type="text/javascript">
```

```
var matn = "Hello Student" ;
document.write ( matn);
</script>
```



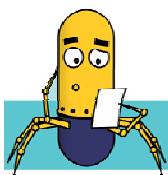
نکته مهم!

۱- تنها زمانی که چند دستور در یک خط نوشته شود نیاز است دستورها با یک سمیکالن^{۳۳} (یا همان ";") از هم جدا شوند.

۲- متغیرهای جاوا اسکریپت با کلمه‌ی var تعریف می‌شوند و نوع ندارند. یعنی می‌توان هر مقداری در درون آنها ریخت و از آنها خواند. البته استفاده از این کلمه واجب نیست و اگر از آن استفاده نکنید و با قراردادی یک مقدار در نام متغیر جدید خود به خود متغیر تعریف می‌شود. اما این کار کد جاوا اسکریپت را بسیار ناخوانا می‌کند. همچنین می‌توانید یک متغیر را دوبار تعریف کنید که عملاً از دفعه‌ی دوم آن چشم‌پوشی شده و مقدار قبلی را حفظ می‌کند.

۳- دستورات در زبان جاوا اسکریپت به کوچک و بزرگ بودن حساس می‌باشند.

دیدید که ما با استفاده از این دستور توانستیم یک متن را بر روی صفحه وب خود چاپ کنیم . در مثال قبل متنی که ما چاپ کردیم یک متن ساده بود بنابراین مرورگر آن را به همان صورت بر روی صفحه ما چاپ کرد حال در نظر بگیرید که متن ما یکی از تگهای html باشد و مرورگر بخواهد آن را چاپ کند . همانطور که در فصل قبل مطالعه کردید دیدید که مرورگرها وقتی به تگهای html برخورد می کنند آنها را نمایش نمی دهند . به مثال زیر توجه کنید

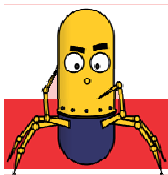


مثال

در این مثال با استفاده از متد فوق زمان سیستم را نمایش می دهیم.

```
<html>
<body>

<h1>Date is</h1>
<script type="text/javascript">
document.write("<p>" + Date() + "</p>");
</script>
</body>
</html>
```



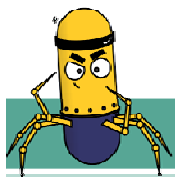
نکته مهم!

با استفاده از Date() شما می توانید تاریخ و زمان سیستم را هنگام اجرای کد دریافت کنیم

برای جمع دو تا رشته (کنارهم قرار گرفتن) دورشته می توانید از + استفاده کنید

همیشه نمی توان از document.write استفاده کرد زیرا دستور write تنها در هنگام بارگزاری صفحات می تواند به کار گرفته شود، در غیر این صورت منجر به پاک شدن بارگذاری کل صفحه و نوشته شدن اطلاعات از نو خواهد شد.

با فهمیدن مثال فوق متوجه می شوید که می توانید تنها با دانستن همین یک دستور (document.write) و دستورات حلقه و شرط صفحات جدیدی خلق کنید. حال برای تسلط بیشتر و اهمیت این دستور سعی کنید تمرین زیر را انجام دهید.



تمرین

برنامه ای بنویسید که یک جدول ضرب ۱۰*۱۰ را مانند شکل زیر در صفحه سایت خود نمایش دهید.

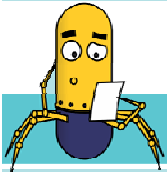
1	3	5	7	9	11	13	15	17	19
4	8	12	16	20	24	28	32	36	40
6	10	14	18	22	26	30	34	38	42
8	12	16	20	24	28	32	36	40	44
10	14	18	22	26	30	34	38	42	46
12	16	20	24	28	32	36	40	44	48
14	18	22	26	30	34	38	42	46	50
16	20	24	28	32	36	40	44	48	52
18	22	26	30	34	38	42	46	50	54
20	24	28	32	36	40	44	48	52	56

شکل ۴-۲: ایجاد جدول ضرب توسط جاوا اسکریپت

۴-۴. بازی با تصاویر

تعریف توابع

توابع در جاوا اسکریپت نیز بسیار شبیه توابع در C است. تعریف هر تابع با استفاده از کلیدواژه‌ی `function` صورت می‌گیرد. بعد از این کلیدواژه نام تابع و لیست ورودی‌ها و بعد از آن بدنه‌ی تابع معرفی می‌شود. برگشت دادن خروجی توسط تابع دلخواه است (یعنی تابع می‌تواند خروجی داشته باشد یا نداشته باشد) و توسط کلیدواژه‌ی `return` صورت می‌پذیرد.



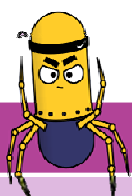
مثال

در این مثال تابعی می‌نویسیم که ورودی تابع را با استفاده از دستور `alert()` بر روی پنجره نمایش دهد.

```
function sayHello(theName){
    alert ("Hello"+ the Name + "!!")
}
```

در زبان‌های برنامه نویسی وقتی ما بدنه یک تابع را می‌نویسیم ، تابع را تا زمانی فراخوانی نکنیم تابع اجرا نمی‌شود در زبان جاوا اسکریپت هم وقتی یک تابع را تعریف می‌کنیم باید آن را فراخوانی کنیم .

شما می‌توانید در هر جایی از صفحه خود تابع را فراخوانی کنید به شرط آنکه فراخوانی تابع خود را نیز داخل تگ script قرار دهید(البته در رویدادها می‌توانید بدون این تگ فراخوانی کنید که در ادامه می‌خوانید.

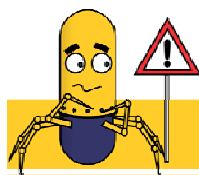


تمرین بیشتر

حال برنامه جدول ضرب را با استفاده از تابع پیاده سازی نمایید.

متد () getElementById

ما می‌توانیم با استفاده از این متد به هر عنصری در یک صفحه HTML که دسترسی داشته باشیم و هر اطلاعاتی را درباره آن کسب نماییم.



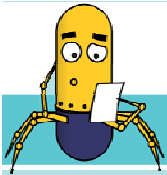
توجه!

همانطور که در بخش آموزش HTML آموختید، تگ در یک صفحه، می‌تواند دارای یک شناسه یا id منحصر به فرد و غیر تکراری باشد. به عبارت دیگر هیچ دو عنصری در یک صفحه HTML، نمی‌توانند دارای id یکسان باشند. از این id برای شناسایی و دسترسی به

عنصر مورد نظر در صفحه استفاده می شود.

شکل دستور

```
document.getElementById ( " id عنصر مورد نظر " );
```



مثال

حال تابعی می نویسیم که بوسیله آن دو عکس را در یک صفحه با یکدیگر جابجا نماید.

```
<html>
<head>
<title>Change Picture</title>
<script language="javascript" type="text/javascript">
function changePic()
{
    k=document.getElementById('pic1').src;
    document.getElementById('pic1').src=document.getElementById('pic2').src;
    document.getElementById('pic2').src=k;
}
</script>
</head>
</body>

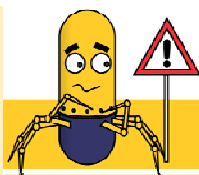

<script>
changePic();
</script>
</body>
</html>
```

همانطور که در مثال بالا مشاهده می کنید بدنه تابع را در قسمت head می نویسیم تا به محض اجرا شدن صفحه تابع ایجاد گردد. همچنین در قسمت <body> آن را فراخوانی کرده ایم.

آیا با دستوراتی که تا اینجا مطالعه کردید توانسته اید پویایی به صفحه خود ایجاد کنید و با کاربر تعامل برقرار کنید؟

رویدادها

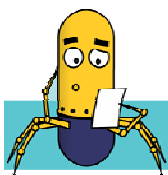
در ابتدا فصل گفتیم که با استفاده از زبان جاوا اسکریپت می توانیم پویایی به صفحه ی وب اضافه کنیم ، این پویایی زمانی احساس می شود که به کمک تشخیص رخدادها و اعمالی است که کاربر در روی صفحه انجام می دهد احساس می شود.



توجه!

تعاملات جاوا اسکریپت با html از طریق رویدادهایی که به واسطه دستکاری هایی که کاربر یا مرورگر بر روی صفحه انجام می دهد شکل می گیرد.

زمانی که صفحه ما بارگذاری می شود یک رویداد اتفاق می افتد (onLoad) یا زمانی که کاربر بر روی دکمه ای کلیک می کند باز هم رویدادی اتفاق می افتد . به عنوان مثال رخداد onClick برای دریافت کلیک شدن بر روی یک کلید است و یا نسبت دادن یک تابع به آن، می توان عملیات مرتبط با آن رخداد را انجام داد. این رخدادها همه در درون تگ های HTML تعریف می شوند حالا سعی می کنیم مثال فوق را بازنویسی کنیم.

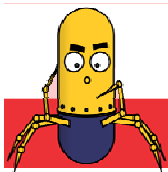


مثال

با استفاده از آموخته های این بخش مثال فوق را به گونه ای بازنویسی می کنیم که در صورت کلیک بر روی یک دکمه، جای دو عکس با هم عوض شوند.

```
<html>
<head>
<title>Change Picture</title>
<script language="javascript" type="text/javascript">
function changePic()
{
    k=document.getElementById('pic1').src;
    document.getElementById('pic1').src=document.getElementById('pic2').src;
    document.getElementById('pic2').src=k;
}
</script>
</head>
<body>


<input type="button" height="50" width="50" value="change" onClick="changePic()">
</body>
</html>
```

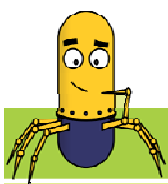


نکته مهم!

به محل فراخوانی تابع دقت کنید. همانطور که می بینید لازم است جهت فراخوانی یک تابع به صورت زیر عمل کنیم:

" () نام تابع = رویداد"

اگر دقت کنید در این قسمت ما برای فراخوانی تابع از تگ script استفاده نکردیم.



بیشتر بدانیم!

می توانیم دستورات جاوااسکریپت را داخل کوئیشن یک رویداد بنویسیم.

۴-۴-۱. آشنایی با انواع رویدادها زبان جاوااسکریپت

رویدادهای mouse

زمانی که از طریق ماوس ما کارهایی انجام می دهیم.

جدول ۴-۱: رویدادهای Mouse

توضیحات	رویداد
کلیک کردن دکمه چپ موس توسط کاربر.	onClick
برای دوبار کلیک دکمه چپ موس.	onDbClick
فشار آوردن بر روی دکمه چپ موس.	onMouseDown
رها کردن دکمه موس پس از فشردن آن.	onMouseUp
قرار گرفتن نشانگر موس بر روی یک چیزی.	onMouseOver
خارج شدن موس از روی یک چیزی.	onMouseOut

رویدادهای Keyboard

زمانی اتفاق می افتد که ما دکمه‌ای در کیبورد تغییر وضعیت می‌دهد.

جدول ۴-۲: رویدادهای Keyboard

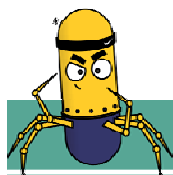
توضیحات	رویداد
فشردن یک کلید کیبورد.	onKeyPress
پایین نگه داشتن یک کلید کیبورد.	onKeyDown
رها کردن یک کلید فشرده شده توسط کاربر.	onKeyUp

رویدادهای مرورگر

زمانی می باشد که تغییراتی در پنجره مرورگر انجام می‌شود.

جدول ۴-۳: رویدادهای مرورگر

توضیحات	رویداد
تکمیل شدن یک صفحه وب یا یک تصویر در مرورگر.	onLoad
بستن مرورگر یا خروج از یک صفحه وب.	onUnload
تغییر اندازه پنجره مرورگر.	onResize



تمرین

با استفاده از جاوا اسکریپت صفحه (پازلی) طراحی کنید که سه عکس را به شکل زیر در صفحه به صورت زیر نمایش داده و با کلیک بر روی هر عکس، در صورتی که خانه جانی آن خالی بود، عکس به آن خانه منتقل شود.

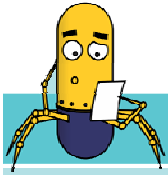


توجه داشته باشید که شکل سمت چپ بعد از کلیک بر روی خانه‌ی بالا سمت راست می‌باشد.

۴-۵. ساخت منوی پویا (rollover image)

مفهوم rollover بسیار ساده می‌باشد، بدین معنا که ابتدا دو تصویر وجود داشته‌اند. تصویر اول در هنگام دریافت صفحه وب توسط کاربر در browser نمایش داده می‌شود، سپس بعد از آنکه کاربر ماوس را بر روی اولین تصویر ببرد، browser تصویر دوم را با تصویر اول در عوض چند لحظه عوض می‌کند. در صفحات وب این صحنه را بارها زمانی که بر روی یک منو با ماوس حرکت می‌کردید دیده‌اید در این قسمت می‌خواهیم آن را با استفاده از جاوا اسکریپت پیاده سازی نماییم.

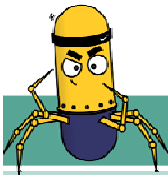
در مثال قبل ما با استفاده از تابع changPic توانستیم جای دو عکس در صفحه را تغییر دهیم و همچنین باشناختی که از رویدادهای ماوس پیدا کردید کافیت از دو رویداد ماوس بر روی تگ لینک استفاده نماییم و مثال زیر را به راحتی انجام دهیم .



مثال

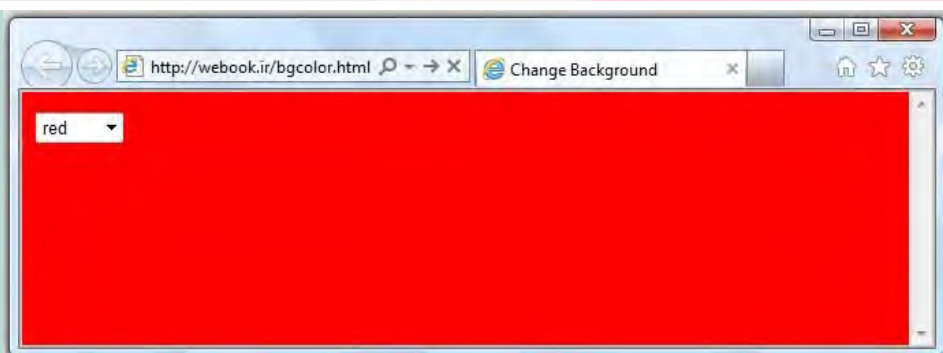
برنامه‌ی زیر با قرار گرفتن ماوس بر روی لینک مورد نظر و همچنین با خارج شدن از روی آن جای دو تصویر در صفحه تغییر می‌دهد.

```
<html>
<body>
  <p align="center">
    <a href="#" onmouseover="document.getElementById('arrow').src='1.jpg'"
      onmouseout="document.getElementById('arrow').src='3.jpg'">
      
    </a>
  </p>
</body>
</html>
```



تمرین

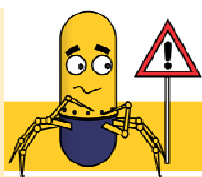
با استفاده از مثال قبل یک لیست برای انتخاب رنگ پیش زمینه در سایت ایجاد نمایید.



شکل ۴-۳: تغییر رنگ پس زمینه

آلبوم تصادفی - نمایش یک تصویر بصورت اتفاقی

برای نمایش تصاویر تصادفی لازم است ما ابتدا تعدادی تصاویر تهیه کنیم و همچنین با استفاده از دستور random در زبان جاوا اسکریپت آن را پیاده‌سازی نماییم دستور random در این زبان در مجموعه توابع کتابخانه‌ای ریاضی قرار دارد.



توجه!

توابع ریاضی

در جاوا اسکریپت برای کار با اعداد و عملیات ریاضی توابع و مقادیر متنوعی در نظر گرفته شده است. دسته‌ی اول این امکانات توابع آزاد هستند که مانند دیگر توابع قابل فراخوانی هستند. مانند `parseInt(x)` (سعی می‌کند رشته‌ی حرفی ورودی را به عدد صحیح تبدیل کند). اما دو دسته‌ی دیگر وابسته به کلاس `Math` هستند مانند `random()` (مقداری

تصادفی بین ۰ و ۱ باز می‌گرداند) و به عنوان مقدار یا تابع آن و با استفاده از "." باید فراخوانی شوند. شما می‌توانید در انتهای کتاب لیستی از توابع پرکاربرد ریاضی را در زبان جاوا اسکریپت پیاده سازی نمایید.

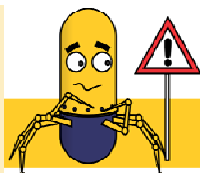
برای اینکه بتوانیم راحت‌تر با این مجموعه تصاویر کار کنیم لازم است از آرایه‌ها در زبان جاوا اسکریپت استفاده نماییم. تعریف آرایه‌ها در این زبان به صورت زیر می‌باشد.

آرایه‌ها

مفهوم آرایه‌ها در جاوا اسکریپت مانند زبان سی می‌باشد. به تکه کدهای زیر توجه کنید:

```
var arr1 = new Array ();
var arr2 = new Array (10);
```

کد بالا arr1 را به عنوان یک آرایه‌ی جدید ایجاد می‌کند که طول آن صفر است، یعنی فعلاً عضوی ندارد. همچنین arr2 را با طول 10 می‌سازد. برای دسترسی به عناصر آرایه و خواندن و نوشتن آنها، می‌توان از اندیس استفاده کرد (همانند C، شروع اندیس آرایه از صفر است).

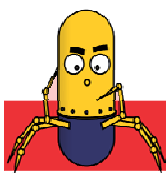


توجه!

در زبان جاوا اسکریپت این آزادی عمل وجود دارد تا طول آرایه در حین برنامه‌نویسی زیاد شود و نیازی به تعریف آن از ابتدا نیست.

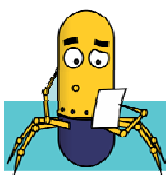
اگر هم به ترتیب خانه‌های آرایه مقداردهی نشوند، مثلاً ابتدا خانه‌ی صفر و سپس

خانه‌ی ۱۰ مقدار دهی شود، تمام خانه‌های این فاصله هم ایجاد شود (یعنی خانه‌های یکم تا نهم آرایه)، اما مقدار آنها برابر undefined خواهد بود.



نکته مهم!

ما با استفاده از شکل `length()` اسم آرایه می‌توانیم طول یک آرایه را محاسبه نماییم.



مثال

برنامه‌ای بنویسید که اسمی عکسها را داخل یک آرایه ریخته با هر بار لود شدن صفحه عکسهای تصادفی ایجاد نماید.

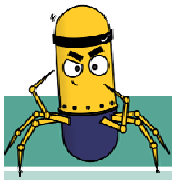
```
<html>
<head>
<script language="javascript" type="text/javascript">
myPix=newArray("webook1.jpg","webook2.jpg","webook3.jpg")
imgCt = myPix.length;
function choosePic() {
    if(document.images){
        randomNum = Math.floor((Math.random()*imgCt))
        document.getElementById(myPicture).src = myPix[randomNum];
    }
}
</script>
```

```

</head>
<body onload="choosePic()">

</body>
</html>

```



تمرین

سعی کنید با استفاده از مثال فوق برای صفحه شخصی خود یک آلبوم تصادفی ایجاد کنید که با هر بار اجرا شدن صفحه تعدادی از عکس‌های شما در صفحه نمایش داده شود.

زمان‌بندی

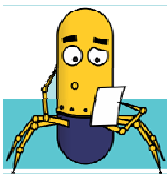
یکی از امکانات جالب و پر کاربرد در جاوا اسکریپت، امکانات زمان‌بندی آن است. این امکانات به برنامه‌سازی اجازه می‌دهد در بازه‌های زمانی مختلف کدهایی را اجرا کند. این اجرای زمان‌بند کمک شایانی در راستای پویانمایی در صفحات جاوا اسکریپت انجام می‌دهد. مجموع این امکانات با کمک دو تابع بدست می‌آید که عبارتند از `setTimeout` و `clearTimeout`.

روش استفاده از `setTimeout` بدین صورت است:

```
var t=setTimeout("نام تابع یا کد مورد نظر", milliseconds);
```

مقدار ورودی اول در این تابع تکه کدی است که باید بعد از گذشت زمانی که توسط ورودی دوم مشخص شده، اجرا گردد. همچنین خروجی این تابع برای وقتی مناسب است که بخواهید رخداد زمانی تنظیم شده را قبل از اجرا لغو کنید. برای لغو کافی است این مقدار را به تابع clearTimeout با الگوی زیر بدهید:

```
clearTimeout(setTimeout_variable);
```



مثال

برنامه جا به جایی عکس را با استفاده از این دو تابع بازنویسی نمایید.

```
<html>
<head>
  <title>Change Picture</title>
  <script language="javascript" type="text/javascript">
    function changePic()
    { k=document.getElementById('pic1').src;
      document.getElementById('pic1').src=document.getElementById('pic2').src;
      document.getElementById('pic2').src=k;
      r=setTimeout("changePic()", 5000);
    }
  </script>
</head>
<body>
  
  
  <input type="button" height="50" width="50" value="change" onclick="changePic()">
</body>
</html>
```

لزوم استفاده از معتبر سازی فرمها

در صفحات وب بازدیدکنندگان می‌توانند بوسیله فرمها اطلاعات خود را برای سرویس دهند(سرور) ارسال نمایند. البته این را باید بدانید که هکرها و مهاجمان نیز از طریق همین فرمهای دریافت اطلاعات، کدهای خرابکاری خود را در برنامه‌های ما تزریق می‌کنند. بنابراین اولین اقدام برای داشتن یک وب سایت امن نوشتن برنامه‌ای می‌باشد که اطلاعات دریافتی کاربران را قبل از ارسال از طریق فرمها برای ما کنترل نمایند و در صورت هرگونه مقدار نامعتبری از ارسال آن جلوگیری نماییم.

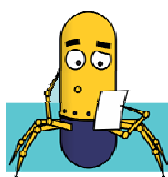
بسته به عناصر فرم و ارسال اطلاعات، عوامل مختلفی را باید در یک فرم کنترل کرد. در زیر نمونه‌هایی از عواملی که رایج است و در بیشتر فرمها استفاده می‌شود. برای مثال ذکر شده است.

چک کردن فیلد خالی یک textbox

چک کردن عدد بودن یک textbox

چک کردن الفبا بودن یک textbox

چک کردن ساختار یک آدرس پست الکترونیک تایپ شده در یک textbox



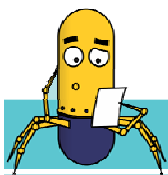
مثال

تابعی بنویسید که قبل از ارسال فرم خانه‌های خالی یک textbox را چک کند و در صورت

³⁴ Form validation

خالی بودن به کاربر پیغام دهد.

```
<html>
<head>
  <script type="text/javascript">
    function notEmpty(elem, helperMsg){
      if(elem.value.length == 0){
        alert(helperMsg);
        elem.focus(); // به موقعیت تایپ کردن مجدد کنترل برمی گردد
        return false;
      }
      return true;
    }
  </script>
</head>
<body>
  <form>
    Required Field: <input type='text' id='req1'>
    <input type="button"
      onclick="notEmpty(document.getElementById('req1'), 'Please Enter a Value')"
      value="Check Field">
  </form>
</body>
</html>
```



مثال

تابعی بنویسید که قبل از ارسال فرم خانه یک textbox را چک کند و در صورت عدد نبودن به کاربر پیغام دهد.

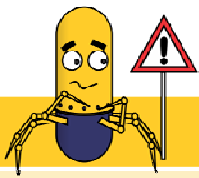
```
function isNumeric(elem, helperMsg){
  var numericExpression = /^[0-9]+$/;
```



```

if(elem.value.match(numericExpression)){
    return true;
}else{
    alert(helperMsg);
    elem.focus();
    return false;
}
}

```



توجه!

علامات [] مجموعه ای از کاراکترها را با هم گروه بندی می کند.

نقطه (.) به معنای همه ی کاراکترها می باشد.

علامت ^ در داخل براکت به معنای نقیض (NOT) می باشد. پس [a-z^] به معنای هر کاراکتر به غیر از a تا z می باشد.

s\ به معنای فضای خالی می باشد.

؟ به معنای صفر یا یک کاراکتر می باشد.

* به معنای صفر یا تعداد بیشتری از المان قبلی می باشد.

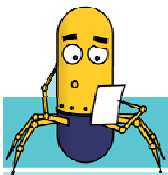
+ به معنای یک یا تعداد بیشتری از المان قبلی می باشد. (همانطور که ما در مثالمان استفاده کردیم.)

اگر نیاز به هر ترکیبی از حروف و اعداد داشته باشید می توانید از (۹-۰a-zA-Z) را استفاده کنید.

اگر نیاز به ترکیبی از حروف و سپس یک شماره داشته باشید از کد ([۹-۰A-Za-z]+[۰-۹]) استفاده کنید.

اگر نیاز به هر حرفی داشته باشید از (*) استفاده کنید.

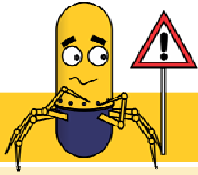
برای ذخیره‌ی جداگانه‌ی حروف و اعداد در متغیرها از کد (+a-zA-Z) ([۹-۰]+) استفاده کنید.



مثال

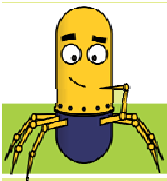
تابعی بنویسید که قبل از ارسال فرم مقدار داخل یک textbox را چک کند و در صورت عدد نبودن و همینطور کاراکتر نبودن به کاربر پیغام دهد.

```
function isAlphanumeric(elem, helperMsg){
    var alphaExp = /^[0-9a-zA-Z]+$/;
    if(elem.value.match(alphaExp)){
        return true;
    }else{
        alert(helperMsg);
        elem.focus();
        return false;
    }
}
```



توجه!

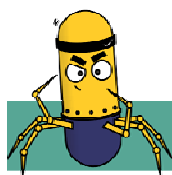
متد `match()` جهت مقایسه مقدار یک المان با ورودی آن به کار می‌رود.



بیشتر بدانیم!

```
<html>
<head>
  <script type="text/javascript">
    function emailValidator(elem, helperMsg){
      var emailExp = /^[w\-\.\+]+\@[a-zA-Z0-9\.\-]+\.[a-zA-z0-9]{2,4}$/;
      if(elem.value.match(emailExp)){
        return true;
      }else{
        alert(helperMsg);
        elem.focus();
        return false;
      }
    }
  </script>
</head>
<body>
  <form>
    Email: <input type="text" id="emailer">
    <input type="button"
      onclick="emailValidator1(document.getElementById('emailer'), 'Not a Valid Email')"
      value="Check Field">
```

```
</form>
</body>
</html>
```



تمرین

با استفاده از مطالبی که تاکنون یاد گرفتید به صفحه شخصی خود بروید و در قسمت فرم ورود اعضاء برنامه معتبرسازی فرم را بنویسید. (ابتدا ببینید به چه توابعی نیاز دارید).

۴-۷. جعبه‌های پیام

جاوا اسکریپت توابع و ابزارهای متنوع و زیادی دارد. اما سه تابع که در این قسمت معرفی می‌کنیم برای برنامه‌نویسی در ابتدای کار و خصوصا برای اشکال زدایی، بسیار مفید هستند. این سه تابع عبارتند از alert، confirm و prompt.

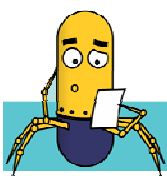
تابع اول ورودی خود را در یک پنجره به صورت پیام نمایش می‌دهد. اجرای خطوط بعدی کد جاوا اسکریپت تا زدن دکمه ok این پنجره توسط کاربر به تاخیر می‌افتد. این تابع برای نشان دادن محتوای متغیرهای مختلف و یا اعلام یک وضعیت به کاربر مفید است.

تابع بعدی confirm است که مانند alert عمل می‌کند ولی خروجی هم دارد. این خروجی برای تشخیص وارد شدن ok یا cancel است. معمولا از این تابع برای گرفتن تایید از کاربر استفاده می‌شود.

تابع prompt برای گرفتن ورودی از کاربر استفاده می‌شود. prompt دو ورودی دارد که ورودی اول متن پیام و ورودی دوم مقداری است که بطور پیش‌فرض در قسمت ورودی قرار می‌گیرد.

شی location و history

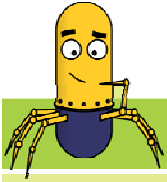
از دیگر اشیا موجود در window می‌توان به location و history اشاره کرد. شی Location مشخص‌کننده‌ی مکان فعلی صفحه است و می‌توان با تابع location.replace مکان آن را عوض کرد. مثلاً با استفاده از این تابع، صفحه‌ای زیر را می‌توان ساخت که دکمه‌ای با عنوان go to google! در آن است باشد و هرگاه آن کلید شود، صفحه به google برود.



مثال

انتقال آدرس

```
<html>
<head>
<title>Location Changer!</title>
<script language= "java script" type="text/javascript">
  function changeURL() {
    window.location.replace("http://www.google.com");
  }
</script>
<head>
<body>
  <input type = "button" value = "go to google!" onClick="changeURL()">
</body>
</html>
```



بیشتر بدانیم!

برای توضیح کار `go()` باید گفت که `go()` صفحه را بارگذاری مجدد^{۳۵} می‌کند. البته اگر این تابع با ورودی خوانده شده در مثال `go (1)` کار `go (1)` و حالت `go (-1)` کار `Back ()` را انجام می‌دهد. البته `go()` مقادیر دیگری نیز به عنوان ورودی می‌گیرد.

رشته‌ها

از جمله اشیا معروف در زبان جاوا اسکریپت رشته‌ها هستند. از رشته‌ها می‌توان مانند یک متغیر عادی استفاده کرد و یا اینکه با استفاده از تابع‌هایی که بر روی آن تعریف شده، امکانات بیشتری از آن را به کار گرفت.

آنچه تا اینجا گفته شد، مختصری از توانایی‌های بسیار زیاد جاوا اسکریپت است و امکانات آن به همین جا ختم نمی‌شود! سایت‌ها و کتاب‌های بسیاری درباره‌ی این زبان وجود دارند شما می‌توانید در بین آنها به دنبال موضوعات مورد علاقه خود بگردید!

³⁵ Reload



JQuery

۴-۸. JQuery چیست؟

امروزه صفحات وب محیطی جذاب هستند. این صفحات معمولاً از فن‌آوری‌هایی که مخاطب را به خود جلب می‌کند استفاده می‌کنند. طراحان سایتها از استانداردهای بالایی برای طراحی سایت‌هایشان استفاده می‌کنند و همچنین هم از ابزار مختلف برای جذابیت و ساده‌سازی عملکردهای پیچیده بهره می‌برند. با این روند رو به رشد، جاوا اسکریپت زبانی نبود که به وسیله آن خلق آن جذابیت‌ها به سادگی امکان‌پذیر باشد. بنابراین طراحان با استفاده از توابع این زبان، کتابخانه‌ای ایجاد کردند که به راحتی بتوان این جذابیت‌ها را بر روی سایت استفاده کرد. البته کتابخانه‌های دیگری نیز طراحی شده بود که سادگی و قدرت JQuery را نداشتند. بنابر این می‌توان اینطور گفت برنامه‌نویسان وب با استفاده از زبان برنامه‌نویسی جاوا اسکریپت، یک سری قاعده و قانون تعریف کرده‌اند. همچنین یک دستور زبان جدید ساخته که شما با آن قواعد یک کد به مراتب بسیار ساده‌تر می‌نویسید، سپس آن فایل دستورات شما را به دستورات استاندارد جاوا اسکریپت ترجمه می‌کند و آن‌ها را اجرا می‌کند.

۴-۹. چگونه کتابخانه JQuery را به برنامه خود اضافه کنیم

برای اضافه کردن این کتابخانه به پروژه وب خود و کار با توابع آن لازم است آخرین

نسخه^{۳۶} آن را از سایت jquery.com دانلود کرده یا متن آن را داخل یک فایل با پسوند js ذخیره نموده و سپس به صفحه وب خود به همانند جاوا اسکریپت به صورت زیرلینک کنید.

```
<script type="text/javascript" src="jquery.js"></script>
```

بعد از لینک کردن صفحه می توانید کدهای JQuery را داخل تگ script به صورت جدا بنویسید یا می توانید داخل یک فایل جاوا اسکریپت بنویسید و آن را به برنامه خود لینک نمایید.

برای برنامه نویسی به زبان JQuery لازم است شما به زبان html و css و جاوا اسکریپت تسلط کافی داشته باشید.

۴-۱۰. چگونه با JQuery برنامه نویسی کنیم

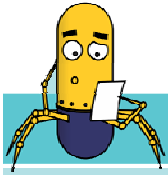
ساختار JQuery

```
$(selector).action();
```

\$ مختصر دستور JQuery می باشد

Selector معادل نام یک تگ یا نام کلاس یا id برای مشخص شدن تعدادی تگ می باشد.

action() معادل عملیاتی تعریف شده در دستورات JQuery می باشد که قرار است بر تگ مشخص شده عمل نماید.



مثال

```

$("p").hide();
$("p.myclass").hide();
$("h1").hide();

```

\$(document).ready

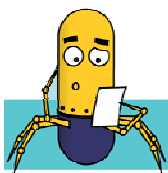
در JQuery تمام دستورات ما زمانی اتفاق می افتد که صفحه مورد نظر ما اجرا شده باشد. بنابراین برای آن یک رویداد کلی تعریف شده که تمام دستورات و رویدادهای صفحه در آن اتفاق می افتد.

حال با آگاهی دستورات جاوا اسکریپت بعد از اتفاق افتادن این رویداد می توانیم یک تابع تعریف کنیم و دستورات را که قرار است بر روی تگ اعمال شود را بنویسیم و هم می توانیم آن رویداد را صدا بزنیم.

```

$(selector).event(function(){...})
$(selector).event()

```



```
<html>
<head>
  <title>Test jquery</title>
  <script type="text/javascript" src="jquery.js" > </script>
  <style type="text/css">
    #changeBG
    {background: red;
    width: 300px;
    height: 300px;
    }
  </style>

  <script type="text/javascript">
    $(function(){
      $('a').click(function(){
        $('#changeBG').hide();
      });
    });
  </script>
</head>
<body>
  <div id="changeBG"></div>
  <a href="#">Hello Webook</a>
</body>
</html>
```

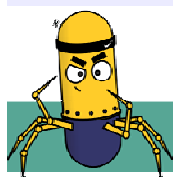
در مثال فوق ما به ترتیب کارهای زیر را برای صفحه خود نوشتیم.

۱- JQuery را به صفحه خود لینک دادیم. (در ابتدای صفحه)

۲- با استفاده از روش idSelector که دربخش css مطالعه کردیم برای تگ div یک سری خصوصیت تعریف کردیم.

۳- با استفاده از رویداد کلیک تگ لینک یک تابع تعریف کردیم.

۴- تابع را طوری تعریف شده که با زدن لینک ایجاد شده در صفحه آن id (برای تگ div) را با استفاده از متد hide در JQuery ناپدید می کند.



تمرین

۱- برنامه مثال را اجرا کنید و ورودی متد hide() را مقدار ۱۰۰۰ بدهید (به این صورت hide(1000) چه اتفاقی می افتد؟

۲- سعی کنید به جای تگ div از یک عکس برای مثال استفاده کنید و برای آن همانند مثال id تعریف کنید و پس از اجرا به صفحه شخصی خود این افکت را اضافه کنید.

۳- سعی کنید این بار به جای متد hide() از متد show() استفاده کنید. می بینید که تگ مورد نظر نمایان می شود. حال سعی کنید برنامه را طوری بنویسید که با استفاده از این دو متد و همچنین دو لینک در صفحه با کلیک بر روی لینک ها به ترتیب متن خود را بالا و پایین ببرید.

۴-۱۱. آشنایی با افکتها

متد Toggle

با این افکت می‌توانید وضعیت متد show() را به hide() و بالعکس تبدیل نمایید.

این افکت هم همانند متدهای قبل دارای ورودی (جهت تنظیم سرعت) می‌باشد.

متد slide

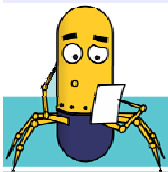
این دارای دو متد به نام slideDown و slideUP می‌باشد که جهت جمع کردن تگ با استفاده از خاصیت تعریف شده به سمت بالا و پایین دارد.

متد fade

این متد هم برای بالا و پایین بردن یک عنصر استفاده می‌شود. با این تفاوت که شفافیت آن به تدریج کم و زیاد می‌تواند بشود که دارای دو متد fadeTo و fadeOut می‌باشد. ورودی این تابع مقداری بین ۰ و ۱ می‌باشد که میزان شفافیت را تعیین می‌کند.

متد animate

با استفاده از این متد می‌توانید هر پارامتری را به عنوان ورودی به آن بدهید.

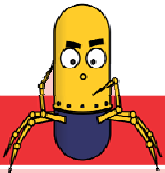
**مثال**

```
<html>
<head>
<title>Test jquery</title>
```

```
<script type="text/javascript" src="jquery.js" > </script>
<style type="text/css">
#changeBG
{background: red;
width: 300px;
height: 300px;
position: relative;
}
</style>

<script type="text/javascript">
$(function(){
    $('a').click(function(){
        $('#changeBG').animate({left:"100px",opacity:0.4,fontSize:"3em"},"slow");
    });
});
</script>

</head>
<body>
    <div id="changeBG">Animate Webook</div>
    <a href="#">Hello Webook</a>
</body>
</html>
```

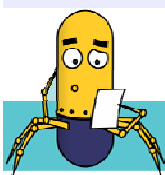


نکته مهم!

توجه داشته باشید ما می‌توانیم متدهای گفته شده فوق را به صورت زنجیره‌ای با هم بر روی یک عنصر به صورت زیر بکار ببریم.

```
$('#changeBG').fadeIn(100).animate({left:"100px",'slow'}).fadeOut(100)
```

این بار می‌خواهیم با استفاده از خاصیت mouseover و mouseout به جای رویداد click در مثال قبل با رفتن بر روی لینک مورد نظر عنصر ما به بالا و پایین برود.



مثال

```
<html>
<head>
  <title>Test jquery</title>
  <script type="text/javascript" src="jquery.js" > </script>
  <style type="text/css">
    #changeBG
    {background: red;
    width: 300px;
    height: 300px;
    position: relative;
    }
  </style>

  <script type="text/javascript">
    $(function(){
      $('a').mouseover(function(){
        $('#changeBG').animate({top:"100px",opacity:0.4,fontSize:"3em"},"slow");
      });
      $('a').mouseout(function(){
        $('#changeBG').animate({top:"-100px",opacity:1,fontSize:"1em"},"slow");
      });
    });
  </script>

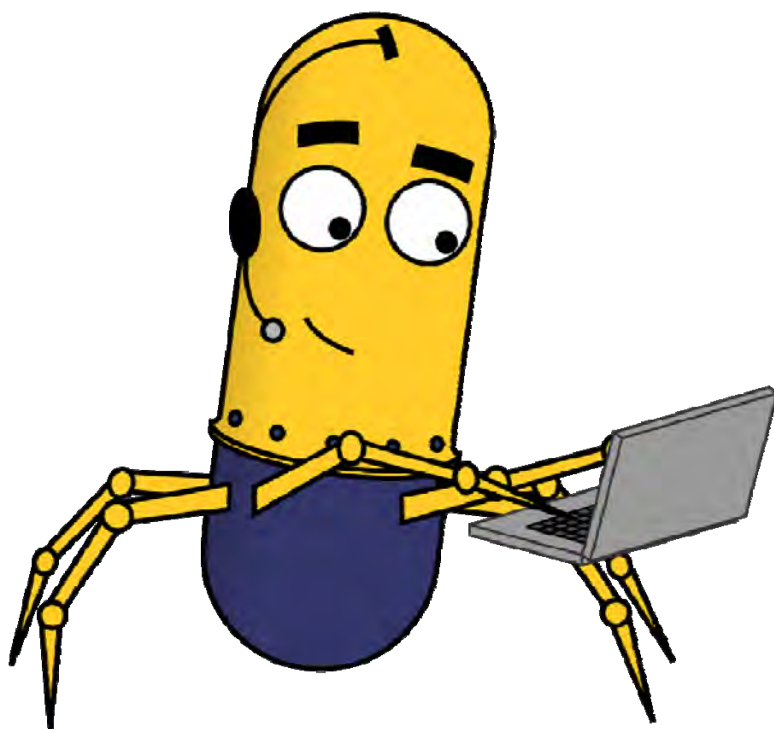
</head>
<body>
  <div id="changeBG">Animate Webook</div>
  <a href="#">Hello Webook</a>
</body>
```

</html>

در مثالهای فوق دیدید که ما با استفاده از افکتهای JQuery و همچنین با استفاده از رویدادهای عناصر صفحه می‌توانیم به جذابیت سایت خود بیفزاییم و همچنین با استفاده از این دستورات محتوا صفحه و نیز مقادیر را در عناصر HTML تغییر داده لذا برای ادامه و استفاده از این دستورات توصیه می‌شود برحسب نیاز طراحی خود به خواندن متدهای این کتابخانه بپردازید. این متن مختصر توانایی‌های کتابخانه JQuery است تا شما را با روش استفاده آن در صفحات خود آشنا نماید. برای علاقه‌مندانی که تمایل استفاده بیشتر از این توابع را دارند لازم است که با مراجعه به سایت JQuery برحسب نیاز خود از افکتهای موجود آن استفاده نمایند.

فصل چهارم

برنامه نویسی PHP



برنامه نویسی PHP

در دنیای امروز پویایی و تغییرات فعال در صفحات وب نقشه ویژه‌ای دارد. دیگر صفحات وب کمتری در اینترنت دیده می‌شوند که ظاهر و محتوای آنها در مراجعات مختلف به آن سایت یکسان است.

همانطور که توضیح داده شده، ایجاد تغییرات در صفحات وب به دو صورت امکان‌پذیر است. اول تغییراتی که در هنگام نمایش و توسط کاربر ایجاد می‌شوند و دوم تغییراتی که سرور در زمان آماده کردن صفحه برای نمایش بر روی آن اعمال می‌کند.

همانطور که در مبحث قبل دیدید، جاوا اسکریپت یکی از زبان‌هایی است که ایجاد تغییر در سمت کاربر را امکان‌پذیر می‌سازد.

زبان‌های مختلفی برای ایجاد تغییرات در سمت سرور وجود دارد. در این کتاب برای ایجاد تغییرات در سمت سرور از زبان PHP استفاده می‌کنیم.

۴-۱۲. مقدمات برنامه‌نویسی PHP

همانطور که در بالا دیدید زبان PHP نیاز به یک سرور برای اعمال تغییرات قبل از نمایش دارد. پس در شروع لازم است تا دسترسی به یک سرور را ایجاد کنید. اگر در حال حاضر به سروری دسترسی دارید برای اجرای فایل‌های PHP باید ابتدا فایل‌ها را به سرور منتقل کرده و بعد از طریق آدرس اینترنتی فایل را فراخوان کنید.

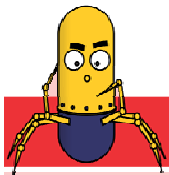
در صورتی که دسترسی به سرور اینترنتی نداشته باشید، می‌توانید برنامه WAMP را مطابق پیوست ... بر روی کامپیوتر خود نصب کرده تا از کامپیوتر خود به عنوان سرور اینترنتی استفاده کنید.

۴-۱۳. از کجا شروع کنیم؟

برای شروع لازم است تا ابتدا پسوند فایل‌هایی را که می‌خواهیم در آن‌ها کد PHP بنویسیم از .html به .php تغییر دهیم. این تغییر برای این انجام می‌شود تا سرور این فایل را قبل از نمایش توسط تحلیلگر PHP اجرا کرده و خروجی HTML آن را تولید نماید.

برنامه‌نویسی PHP درست در دل کدهای HTML انجام می‌شود. برای این کار در هر قسمتی از کد، که بخواهیم از زبان PHP استفاده کنیم با دستورها را بین `<?php` و `>` قرار می‌دهیم.

```
<html>
<head>
<title>PHP Test</title>
</head>
<body>
<?php echo '<p>Hello World</p>'; ?>
</body>
</html>
```



نکته مهم!

به یاد داشته باشید که تمام خطوط دستوری زبان PHP با علامت `;` تمام می‌شوند.

حال این فایل را با پسوند .php ذخیره کرده و به سرور منتقل کنید. سپس نتیجه اجرا شده را در مرورگر ببینید.

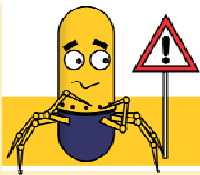
۴-۱۴. دریافت اطلاعات فرم

متغیرها و آرایه‌ها

متغیرها در زبان PHP با علامت \$ شروع می‌شوند و در نام متغیرها مجاز هستید از حروف بزرگ و کوچک انگلیسی از اعداد و از علامت _^{۳۷} استفاده کنید. فقط دقت داشته باشید که استفاده از اعداد به عنوان کاراکتر شروع نام متغیر مجاز نمی‌باشد.

\$a
\$a12
\$_a
\$12a

در PHP نیازی به تعیین نوع متغیر و تعریف اولیه آن نیست. در هر زمان که از نام متغیری استفاده می‌کنید به عنوان تعریف آن محسوب می‌شود. همچنین نوع متغیر در این زبان ثابت نیست و قابل تبدیل از نوعی به نوع دیگر است.



توجه!

انواع متغیرها

جدول ۴-۴: انواع متغیرها در زبان PHP

نوع متغیر	نام متغیر	مثال
عدد صحیح	Integer	\$i = 5;
عدد با ممیز شناور (عدد اعشاری)	Floting Point	\$f = 5.5;
رشته	String	\$s = "Bye"
منطقی (بولی)	Boolean	\$b = true;
آرایه	Array	\$a = array(0=>10, 1=>20);
شیء	Object	\$o = new MyClass();

³⁷ Underscore

\$r = mysql_connect(...);	Resource	منبع
\$n = null;	NULL	هیچ مقدار

آرایه

آرایه‌ها نیز در زبان PHP محدودیت کمی دارند. در برنامه‌نویسی PHP الزامی نیست که همه خانه‌های آرایه از یک نوع باشد و آرایه نیازی نیست در ابتدا تعریف شود.

```
$a[3]="salam";
$a["hi"]=8;
```

همانطور که در بالا میبینید، با توجه به اینکه از \$a قبلاً استفاده نشده است، با رسیدن به این خط خانه‌ای با کلید ۳ در آرایه a از نوع string با مقدار salam ایجاد می‌شود. در خط بعد نیز خانه‌ای با کلید hi از نوع integer با مقدار 8 ایجاد می‌شود. همانطور که متوجه شدید، کلید آرایه‌ها می‌توانند عدد و یا رشته باشند و مهم نیست که این اعداد متوالی و از پیش تعریف شده باشند.

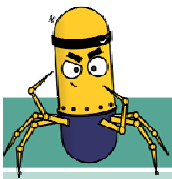
آرایه‌های از پیش تعریف شده

php دارای متغیرهایی از پیش تعریف شده است. دو دسته از این متغیرها در آرایه‌هایی به نام‌های \$_GET و \$_POST قرار دارند. همانطور که احتمالاً متوجه شده‌اید این متغیرها هم نام با نحوه ارسال فرم هستند. در واقع هر کدام در برگیرنده اطلاعات فرم‌ای هستند که با این نحوه به این صفحه ارسال شده‌اند. به ازای هر فیلدی از فرم که برای این صفحه ارسال شده است، خانه‌ای در این آرایه در نظر گرفته می‌شود.

اگر در نظر بگیریم فرمی با یک فیلد به نام username و از نوع متن توسط نحوه GET به این صفحه ارسال شده باشد، عبارتی که کاربر قبل از زدن دکمه ارسال در آن نوشته در خانه‌ای به نام username در آرایه GET ذخیره می‌شود.

در صورتی که فرم به صفحه‌ای که کد زیر در آن نوشته شده ارسال شود، محتوای فیلد username بر روی صفحه نمایش داده خواهد شد.

```
<?php
echo ($_GET['username']);
?>
```



تمرین

با استفاده از اطلاعات فصل دو، فرمی به صورت زیر آماده کنید.

عکس

حال این فرم را به صفحه form.php با نحوه post ارسال کنید.

اکنون با استفاده از اطلاعات این بخش، صفحه form.php را به صورتی کد نویسی کنید که اطلاعات دریافتی را به صورت زیر نشان دهد.

عکس

حلقه foreach

با توجه به اینکه خانه‌های آرایه‌ها لزوماً ترتیب عددی ندارند، پس دسترسی به تمامی خانه‌های آرایه با استفاده از شماره خانه آنها مقدور نیست. برای حل این مساله می‌توان از دستور foreach استفاده کرد. foreach به اینصورت عمل می‌کند که در هر بار تکرار خود یک خانه روی آرایه جلو می‌رود و اطلاعات آن را در متغیری که مشخص شده می‌ریزد.

```
foreach( مقدار as آرایه )
{
    دستورات تکرار شونده
}

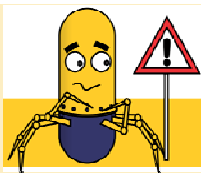
<?php
$arr = array(1, 2, 3, 4);
foreach ($arr as $value) {
    echo $value . '<br>' ;
?>
```

در کد بالا در هر بار تکرار یک مقدار از آرایه درون متغیر value ریخته می‌شود. حال اگر کلید هر خانه آرایه را هم نیاز داشته باشیم، می‌توان از foreach به صورت زیر استفاده کرد.

```
foreach( مقدار => کلید as آرایه )
{
    دستورات تکرار شونده
}

<?php
$a = array(
    "one" => 1,
    "two" => 2,
    "three" => 3,
    "seventeen" => 17
);
```

```
foreach ($a as $k => $v) {
    echo $a[$k] . '=>' . $v . '<br>';
}
?>
```



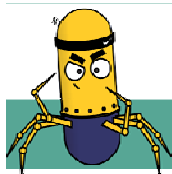
توجه!

به نحوه‌های مختلف تعریف کردن آرایه در مثال‌های بالا دقت کنید.

۴-۱۵. ایجاد قسمت ورود اعضا

امروز اکثر سایت‌ها بخشی را محدود به کاربران خاصی می‌کنند. این بخش ممکن است بخش مدیریت سایت که از طریق آن اطلاعات و یا اخبار سایت و یا قسمت‌های اینچنین قابل تغییر است، باشد و یا بخش اطلاعات خاص برای یک کاربر. این قسمت‌ها بعد از مشخص شدن هویت کاربر از طریق ورود به قسمت کاربری امکان نمایش داده شدن را دارند.

عموماً تعیین هویت کاربران در اینترنت از طریق یک شناسه و کلمه عبور امکان‌پذیر است.



تمرین

برای ایجاد این قسمت از سایت لازم است ابتدا یک فرم HTML با دو فیلد شناسه کاربر

و کلمه عبور ایجاد کنیم.



شکل ۴-۴: صفحه ورود

شرط if

پس از دریافت لازم است تا آن‌ها را با مقادیر از قبل مشخص شده تطابق دهیم. برای اینکار ابتدا یک شناسه کاربر و یک کلمه عبور در نظر بگیریم. برای مقایسه دو مقدار از دستور if مطابق زیر استفاده می‌کنیم.

```
(شرط) if
{
    دستوراتی که در صورت برقرار بودن شرط اجرا می‌شوند.
}
else
{
    دستوراتی که در صورت برقرار نبودن شرط اجرا می‌شوند.
}
```

قسمت شرط if دو مقدار را با یکی از علامت‌های جدول زیر با هم مقایسه می‌کند. اگر نخواهیم عملی در صورت انجام نشدن شرط انجام شود نیازی به نوشتن قسمت else نیست.

جدول ۴-۵: عملگرهای مقایسه‌ای PHP

عملگر	توضیحات	کاربرد
==	برابری مقدار دو متغیر	<code>\$a==\$b</code>
===	برابری مقدار و نوع دو متغیر	<code>\$a=== \$b</code>
!=	نابرابری متغیر	<code>\$a!= \$b</code>
!==	نابرابری مقدار و نوع دو متغیر	<code>\$a!== \$b</code>
>	بزرگ‌تر است	<code>\$a>\$b</code>
<	کوچک‌تر است	<code>\$a<\$b</code>
>=	بزرگ‌تر مساوی است	<code>\$a>= \$b</code>
<=	کوچک‌تر مساوی است	<code>\$a<= \$b</code>

هر کدام از این دو مقدار می‌توانند نام یک متغیر باشند. در زیر چند نمونه از شرط‌ها آمده است.

`$a==3`

`$a==$b`

`1==1`

`$_GET['username']=='ali'`

این امکان وجود دارد تا با ترکیب ifها در صورتی که یک شرط غلط بود، شرط دیگری بررسی شود. برای این کار از ساختار نوشتاری زیر استفاده می‌کنیم.

```

if (شرط ۱)
{
    دستوراتی که در صورت برقرار بودن شرط ۱ اجرا می‌شوند.
}
else if (شرط ۲)
{
    دستوراتی که در صورت برقرار نبودن شرط ۱ و برقرار بودن شرط ۲ اجرا می‌شوند.
}
else
{
    دستوراتی که در صورت برقرار نبودن شرط ۱ و برقرار نبودن شرط ۲ اجرا می‌شوند.
}

```

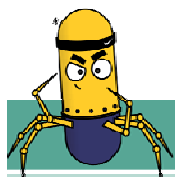
همچنین این امکان وجود دارد که بر روی این شرط‌های عملیات منطقی زیر را نیز انجام داد.

جدول ۴-۶: عملگرهای منطقی PHP

عملگر	نام	توضیحات	کاربرد
!	NOT	"مخالف" منطقی	(\$a != \$b)
	OR	"یا" منطقی	(\$a == 3 \$b == 2)
&&	AND	"و" منطقی	(\$a < 10 && \$a > 1)

با توجه به توضیح بالا برای کنترل چند شرط به صورت همزمان می‌توان از عملگر $\&\&$ ^{۳۸} بین آن شرط‌ها استفاده کرد.

³⁸ AND



تمرین

حال برای تمرین قبل، صفحه دریافتی ایجاد کنید. در این صفحه کنترل کنید در صورتی که نام کاربری و کلمه عبور با عباراتی از پیش تعیین شده (webook و 123) برابر بودند، عبارت "کاربر وبوک خوش آمدید" به رنگ سبز و در صورتی که هر کدام از آن‌ها تطابق نداشت، عبارت "بروز خطا در نام کاربر و یا کلمه عبور" به رنگ قرمز نمایش داده شود.

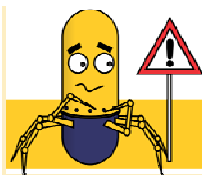
۴-۱۶. تکمیل قسمت ورود اعضا

می‌دانیم که در سایت‌ها این امکان وجود دارد تا کاربر کلمه عبور خود را تغییر دهد و یا کاربرهای جدید به وجود آورد. پس کنترل نام کاربری و کلمه عبور در کد برنامه به صورت ثابت اصولی به نظر نمی‌رسد. در صورتی که کنترل نام کاربر و کلمه عبور به صورت بالا باشد قابلیت تغییر و اضافه کردن کاربران جدید صرفاً با تغییر کد امکان‌پذیر می‌شود و این عملاً انجام این کارها را از طریق صفحات وب غیر ممکن می‌سازد.

برای کامل کردن قسمت ورود اعضا لازم است تا اطلاعات مربوط به نام کاربری و کلمه عبور را از درون کد خارج کرد و از برنامه را طوری بازنویسی کنیم تا نام کاربری و کلمه عبور با اطلاعاتی خارج از صفحه بررسی شده و مطابق داده شوند.

ایجاد یک فایل حاوی اطلاعات

برنامه Notepad را باز کنید و یک نام کاربری و کلمه عبور که با هم جدا شده‌اند را در آن بنویسید و از طریق منوی File، گزینه Save آن را ذخیره کنید.



توجه!

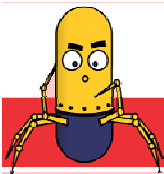
توجه داشته باشید که در صورتی که از حروف فارسی استفاده می‌کنید، نوع Encoding را قبل از ذخیره سازی بر روی UTF-8 قرار دهید.

دسترسی به فایل

برای دسترسی به فایل ابتدا باید به فایل متصل شویم. برای این کار در PHP تابعی به نام `fopen` وجود دارد.

```
متغیر = fopen ('آدرس فایل', 'نحوه باز کردن');
```

با استفاده از این تابع فایلی که آدرس آن مشخص شده است باز شده و اطلاعات مربوط به باز شدن آن درون متغیر قرار می‌گیرد. در این کتاب نام این متغیر را `$file` قرار می‌دهیم. در صورتی که خطایی در باز شدن فایل صورت گیرد، مقدار `false` به متغیر `$file` بازگردانده می‌شود.



نکته مهم!

دقت کنید که این فایل بر روی سرور باز می‌شود و نه بر روی کامپیوتر بازدید کننده و

آدرس فایل می‌تواند مطابق فصل HTML به صورت مطلق و یا نسبی باشد.

همچنین توجه داشته باشید که \$file شامل محتویات فایل نیست و صرفاً اطلاعات مربوط به باز شدن فایل را در خود جای داده است.

فایل را می‌توان به منظورهای مختلفی باز کرد. در جدول خلاصه‌ای از نحوه باز شدن فایل ارائه شده است.

جدول ۴-۷: لیست نحوه‌های باز شدن فایل در تابع fopen()

نحوه	توضیحات
'r' ³⁹	باز کردن فایل برای خواندن. اشاره‌گر در ابتدای فایل قرار می‌گیرد.
'r+'	باز کردن فایل برای خواندن و نوشتن. اشاره‌گر در ابتدای فایل قرار می‌گیرد.
'w' ⁴⁰	باز کردن فایل برای نوشتن. اشاره‌گر در ابتدای فایل قرار می‌گیرد و اطلاعات قبلی فایل پاک می‌شود. در صورت عدم وجود فایل در صورت داشتن دسترسی‌های لازم فایل ایجاد می‌شود.
'w+'	باز کردن فایل برای خواندن و نوشتن. اشاره‌گر در ابتدای فایل قرار می‌گیرد و اطلاعات قبلی فایل پاک می‌شود. در صورت عدم وجود فایل در صورت داشتن دسترسی‌های لازم فایل ایجاد می‌شود.
'a' ⁴¹	باز کردن فایل برای نوشتن. اشاره‌گر در انتهای فایل قرار می‌گیرد. در صورت عدم وجود فایل در صورت داشتن دسترسی‌های لازم فایل ایجاد می‌شود.
'a+'	باز کردن فایل برای خواندن و نوشتن. اشاره‌گر در انتهای فایل قرار می‌گیرد. در صورت عدم وجود فایل در صورت داشتن دسترسی‌های لازم فایل ایجاد می‌شود.

خواندن از فایل

³⁹ Read
⁴⁰ Write
⁴¹ Append

برای خواندن از فایل، از تابع `fgets()` استفاده می‌کنیم. این تابع یک خط از فایل را خوانده و در متغیر مشخص شده قرار می‌دهد. متغیر باز شدن فایل `$file` به عنوان ورودی این تابع قرار دارد. همچنین یک ورودی اختیاری نیز برای این تابع وجود دارد که به وسیله آن می‌توان یک حداکثر طول بر حسب بایت برای آن خط در نظر گرفت.

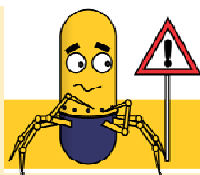
```
$متغیر = fgets ( $file [,طول] );
```

با تکرار این خط به ترتیب خط‌های بعدی فایل خوانده شده و درون متغیر ریخته می‌شود.

نوشتن بر روی فایل

توسط تابع `fputs()` می‌توان بر روی فایل نوشت. این تابع یک رشته را بر روی فایل می‌نویسد. مانند `fgets()` قبل از استفاده از این تابع فایل باید توسط `fopen()` باز شده باشد. `fputs()` دارای یک ورودی که همان `$file` و ورودی دوم رشته‌ای که می‌خواهیم بر روی فایل بنویسیم و یک ورودی اختیاری می‌باشد. توسط ورودی سوم می‌توان طول رشته‌ای را که قرار است در فایل نوشته شود را محدود کرد.

```
fputs ( $file, رشته [,طول] );
```



توجه!

برای رفتن به ابتدای خط بعد در زمان نوشتن در فایل، می‌توان از دستور زیر استفاده کرد.

```
fputs ( $file, '\r\n' );
```

بستن فایل

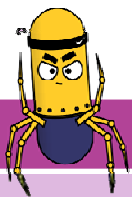
بعد از اتمام کار با فایل توسط تابع `fclose()` فایل را می‌بندیم. متغیر که فایل را با آن باز کردیم به عنوان ورودی این تابع تعیین می‌شود.

```
fclose ( $file );
```

مشخص کننده انتهای فایل

وقتی فایلی را خط به خط می‌خوانیم، لازم است تا بدانیم چه زمانی به انتهای فایل می‌رسیم. در PHP تابعی به نام `feof()` وجود دارد، زمانی که این تابع مقدار `true` رو برمی‌گرداند به این معنی است که آخرین خط فایل خوانده شده است. در بخش بعدی با نحوه استفاده از این تابع بیشتر آشنا خواهید شد.

```
feof ( $file )
```



تمرین بیشتر

با استفاده از توضیحات این بخش برای وب سایت خود یک شمارنده تعداد بازدید ایجاد کنید. به این صورت که بعد از هر بار دیده شدن سایت شما، یک واحد به آن افزوده شود.



شکل ۴-۵: صفحه شمارنده

توابعی برای کار با رشته‌ها

با توجه به استفاده زیاد از رشته‌ها در زبان PHP توابع گسترده‌ای در این زمینه وجود دارند. توسط این توابع می‌توان عملیات دلخواه را بر روی رشته‌ها اعمال کرد. در این بخش تعدادی از این توابع معرفی می‌شوند.

strlen()

این تابع طول یک رشته را به صورت یک متغیر integer باز می‌گرداند. این تابع به صورت زیر استفاده می‌شود.

(رشته strlen)

```
<?php
$str = 'abcdef';
echo strlen($str); // 6

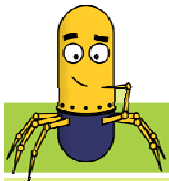
$str = ' ab cd ';
echo strlen($str); // 7
?>
```

substr()

این تابع امکان جدا کردن جزئی از یک رشته ایجاد می‌کند. استفاده از این تابع به این صورت است که در ورودی یک رشته را مشخص می‌کنیم و بعد محل شروع را با یک عدد معین می‌کنیم. به عنوان یک ورودی اختیاری می‌توانیم طول رشته جدا شده را نیز مشخص کنیم. در صورت مشخص نکردن طول رشته، نتیجه تا انتهای رشته جدا می‌شود.

`substr( [طول], محل شروع, رشته )`

```
<?php
echo substr('abcdef', 1);      // bcdef
echo substr('abcdef', 1, 3);   // bcd
echo substr('abcdef', 0, 4);   // abcd
echo substr('abcdef', 0, 8);   // abcdef
echo substr('abcdef', -1, 1);  // f
echo substr('abcdef', -1);     // f
echo substr('abcdef', -2);     // ef
echo substr('abcdef', -3, 1);  // d
echo substr('abcdef', 0, -1);   // abcde
echo substr('abcdef', 2, -1);  // cde
echo substr('abcdef', 4, -4);  //
echo substr('abcdef', -3, -1); // de
?>
```



بیشتر بدانیم!

اگر مقدار منفی به عنوان محل شروع مشخص شود به این معنی است که طول رشته با این عدد جمع شده و محل شروع این عدد مشخص می‌شود. به عبارت دیگر محل شروع از انتها مشخص شده است.

در صورتی که طول با عدد منفی مشخص شود، به این معنی است محل انتهای رشته عبارتست از طول رشته بعلاوه طول منفی مشخص شده. به عبارت دیگر طول رشته از انتها

strpos()

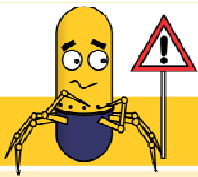
ممکن است بخواهیم محل شروع یک رشته را در رشته‌ای دیگر پیدا کنیم، برای این کار از تابع strpos() استفاده می‌کنیم. ورودی اول این تابع رشته اصلی است و ورودی دوم رشته‌ای است که قصد پیدا کردن آن را در رشته اصلی داریم.

به عنوان یک پارامتر اختیاری می‌توان محلی را که جستجو از آنجا شروع می‌شود را نیز مشخص کرد.

`strpos( [محل شروع], رشته , رشته اصلی )`

```
<?php
$string = 'abcdef abcdef';
$findme = 'a';
$pos = strpos($string, $findme); // 0

$pos = strpos($string, $findme, 1); // 7
?>
```

**توجه!**

توجه داشته باشید که شماره اولین کاراکتر رشته صفر است و همچنین دقت کنید که در صورتی که رشته دوم در رشته اصلی پیدا نشد، مقدار `false` بازگردانده می‌شود.

strstr()

این تابع مانند strpos() یک رشته را در رشته دیگر جستجو می‌کند. اما خروجی آن یک عدد نیست بلکه رشته اصلی از محل وقوع رشته دوم تا انتها جدا می‌شود.

(رشته , رشته اصلی) strstr()

```
<?php
$email = 'user@example.com';
$domain = strstr($email, '@'); // @example.com
?>
```

توجه داشته باشید که strstr() دارای پارامتر اختیاری برای شروع از محل دیگری نمی‌باشد.

explode()

توسط تابع explode() می‌توان یک رشته را با یک جدا کننده به یک آرایه که اجزای آن قسمت‌هایی از رشته اصلی هستند تبدیل نمود. این تابع دارای یک ورودی اختیاری با عنوان حد است که حداکثر قسمت‌هایی explode() ایجاد می‌کند را مشخص می‌کند.

([حد, 'رشته', 'جداکننده'] explode()

```
<?php
$str = 'one|two|three|four';
$a = explode('|', $str);
$b = explode('|', $str, 2);
?>
/*
```

آرایه *a* دارای چهار خانه با مقادیر زیر است:

```
$a[0] => one
$a[1] => two
$a[2] => three
$a[3] => four
```

آرایه *b* دارای دو خانه با مقادیر زیر است:

```
$b[0] => one
$b[1] => two|three|four
*/
```

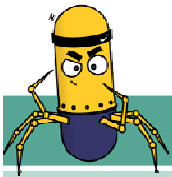
implode()

این تابع دقیقاً برعکس explode() عمل می‌کند و خانه‌های یک آرایه را به وسیله یک چسباننده به هم پیوند داده و یک رشته به وجود می‌آورد.

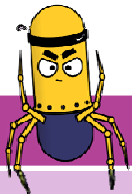
implode('چسباننده', آرایه)

```
<?php
$array = array('lastname', 'email', 'phone');
$comma_separated = implode(" ", $array);

echo $comma_separated; // lastname, email, phone
?>
```

**تمرین**

حال با استفاده از اطلاعات به دست آمده در این بخش، فایل کنترل شناسه کاربری و کلمه عبور را به گونه‌ای بازنویسی کنید که این دو را با محتویات فایل که به صورت webook و 123 است مقایسه کند.

**تمرین بیشتر**

به عنوان تمرین بیشتر می‌توانید قسمت تغییر کلمه عبور را برای سایت خود ایجاد کنید.

توجه داشته باشید که اطلاعات ورود را درون فایل ذخیره کنید.

حلقه while

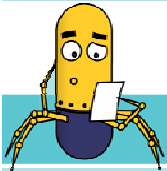
دستور while برای ایجاد یک حلقه تا زمان برقراری یک شرط استفاده می‌شود. در حلقه while دستوراتی که درون این حلقه قرار می‌گیرند، تا زمانی که شرط حلقه برقرار است تکرار می‌شوند. تمامی قوانینی که برای شرط if برقرار است، در شرط while هم قابل استفاده است.

```
( شرط )
while
{
    دستورات تکرار شونده
}

<?php
$i = 1;
while ($i <= 10) {
    echo $i++;
}
?>
```

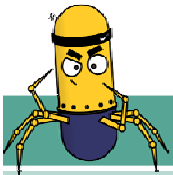
حال اگر بعد از باز کردن فایل دستور fgetc() را درون while قرار دهیم، می‌توانیم به سطرهای مختلف یک فایل دسترسی پیدا کنیم. برای این کار لازم است تا مشخص کنیم تا چند سطر امکان خواندن از فایل را داریم (یا به عبارت دیگر مشخص کنیم که فایل چند سطر اطلاعات دارد). برای این کار در شرط while از مقدار برعکس تابع feof() استفاده می‌کنیم. شرط به این معنی است که تا موقعی که انتهای فایل نشده بود!

مثال



کد زیر تمامی خطوط یک فایل را خوانده و در صفحه نمایش می‌دهد.

```
<?php
$filename="file.txt;"
$file=fopen($filename,"r");
if ($file)
{
    while (!feof($file))
    {
        $line = fgets($file);
        echo $line.'<br>';
    }
}
?>
```



تمرین

با دانسته‌های خود از بخش بالا، صفحه ورود اعضا را به گونه‌ای تغییر دهید که اطلاعات شناسه کاربر و کلمه عبور را با لیستی درون یک فایل که در هر خط آن یک شناسه کاربر و یک کلمه عبور وجود دارد تطابق دهد.

در صورتی که شناسه کاربر و کلمه عبور منطبق در لیست پیدا شد پیام ورود و در غیر اینصورت پیام خطا دهد.

۴-۱۷. ایجاد فرم ثبت نام

با توجه به اینکه سایت شما قسمت ورود اعضا دارد، لازم است قسمتی ایجاد کنید تا کاربران جدید نیز بتوانند ثبت نام کنند. در ابتدا باید مشخص کنیم که چه اطلاعاتی لازم است تا برای ثبت نام از کاربر گرفته شود.

معمولا فرم‌ها دارای بخش گرفتن اطلاعاتی چون نام، نام خانوادگی، آدرس ایمیل، تاریخ تولد و اطلاعاتی اینچنین می‌باشند. برای مثال فرمی مطابق **زیر** برای ثبت نام در نظر می‌گیریم.

عکس

قسمت‌های اولیه را طبق اطلاعات HTML خود ایجاد کنید.

برای قسمت تاریخ تولد این امکان وجود دارد که برای قسمت روز، ۳۱ انتخاب نوشته شود. ولی ایجاد این قسمت توسط حلقه PHP ساده‌تر است.

حلقه for

دستور for برای ایجاد یک حلقه استفاده می‌شود. حلقه با یک مقدار اولیه شروع شده و تا زمانی که شرط حلقه برقرار باشد ادامه می‌یابد و در هر مرحله دستور انتهایی حلقه (که معمولا مقدار تغییر شمارنده حلقه است) و دستورات درون حلقه اجرا می‌شوند.

(دستور حلقه ; شرط پایان حلقه ; مقدار اولیه) for

```
{
دستورات تکرار شونده
}
```

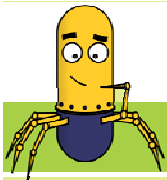
```
<?php
for ($i = 1; $i <= 10; $i++) {
echo $i. '<br>';
}
?>
```

حال با استفاده از توضیح فوق، قسمت روز و ماه فرم ثبت نام را کامل کنید.

date()

در PHP تابعی به نام date() به صورت پیش فرض وجود دارد که بوسیله آن امکان خواندن اطلاعات مربوط به تاریخ و ساعت سرور وجود دارد. تابع date() دارای ورودی‌ای است که به وسیله آن فرمت خروجی ساعت و تاریخ مشخص می‌شود.

کاراکتر	توضیحات	مقادیر مثال
<i>d</i>	مشخص کننده شماره روز در ماه به صورت کاراکتر دوتایی.	از 01 تا 31
<i>j</i>	مشخص کننده شماره روز در ماه.	از 1 تا 31
<i>m</i>	مشخص کننده شماره ماه به صورت کاراکتر دوتایی.	از 01 تا 12
<i>n</i>	مشخص کننده شماره ماه.	از 1 تا 12
<i>L</i>	مشخص کننده سال کبیسه.	مقدار 1 کبیسه مقدار 0 عادی
<i>Y</i>	شماره چهار رقمی سال.	2011
<i>y</i>	شماره دو رقمی سال.	08
<i>h</i>	ساعت ۱۲ ساعته به صورت کاراکتر دوتایی.	از 00 تا 12
<i>g</i>	ساعت ۱۲ ساعته.	از 0 تا 12
<i>i</i>	دقیقه به صورت کاراکتر دوتایی.	از 00 تا 59
<i>s</i>	ثانیه به صورت کاراکتر دوتایی.	از 00 تا 59



بیشتر بدانیم!

با توجه به وجود اختلاف ساعت در کشورهای مختلف، PHP این امکان را به شما می‌دهد تا قبل از استفاده از تابع date() اختلاف زمان پیش فرضی را که می‌خواهید زمان‌ها را بر

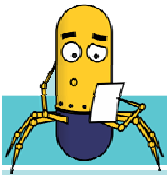
حسب آن به دست آورید، تعیین کنید.

برای مثال ایران اختلاف زمان +3:30 با مرجع زمان دارد. تنظیم زمان پیش فرض از طریق کد زیر امکان پذیر است.

```
date_default_timezone_set ('Iran');
```

همچنین شما میتوانید کارکترهای جدا کننده تاریخ را در ورودی آن مشخص کنید.

برای بهتر متوجه شدن این موضوع به مثال ها توجه کنید.



مثال

```
<?php
```

```
date_default_timezone_set ('Iran');
```

```
// فرض کنید تاریخ امروز بدین قرار است:
```

```
// March 10th, 2001, 5:16:18 pm
```

```
$today = date("m.d.y");           // 03.10.01
```

```
$today = date("j, n, Y");         // 10, 3, 2001
```

```
$today = date("Ymd");             // 20010310
```

```
$today = date('\i\t \i\s \t\h\l\jS \d\ay.');
```

```
// It is the 10th day.
```

```
$today = date('H:m:s \m \i\s\ \m\o\n\t\h'); // 17:03:17 m is month
```

```
$today = date("H:i:s");           // 17:16:17
```

```
?>
```

۴-۱۸. افزودن قسمت‌ها جدید سایت

با توجه به اینکه قسمت‌های جدید سایت را در این فصل طراحی کردید، حال لازم است تا این قسمت‌ها را به صفحاتی که قبل از این فصل طراحی کرده بودید، اضافه کنید.

دستور include()

گاهی ممکن است ما تکه کدی را در یک صفحه PHP نوشته باشیم و لازم است آن را در یک صفحه دیگر عیناً استفاده کنیم. مثلاً ما در این فصل یک صفحه برای فرم ورود اعضا نوشتیم و بخواهیم آن را در صفحه خانه اضافه کنیم. برای اینکار می‌توان محتوای آن را کپی کرده و در کد صفحه خانه اضافه کرد. ولی زمانی که لازم باشد تغییری در آن فرم ایجاد کنیم، باید تمام آن کدها را در صفحات مختلفی که استفاده کردیم، پیدا کرده و آن‌ها را اصلاح کنیم.

PHP توسط دستور include این امکان را به ما می‌دهد که محتویات یک فایل را درون فایل دیگر فراخوان^{۴۲} کنیم.

به عنوان مثال اگر قسمت ورود اعضا را در فایلی به نام login.php ذخیره کرده باشیم برای اضافه کردن آن به صفحه index.php کافیست کد زیر را در قسمت مربوطه خانه اضافه کنید.

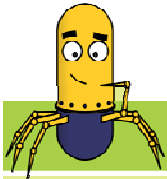
```
<?php
include 'login.php';
?>
```

گاهی ممکن است ما در کد خود دو فایل file1.php و file2.php را نیاز داشته باشیم و این دو فایل را در کد خود include کنیم. حال ممکن است file2.php خود در file1.php فراخوان شده باشد و این برای برنامه مشکل ایجاد کند. برای جلوگیری در فراخوان‌های مجدد میتوان از دستور include_once استفاده کنیم.

⁴² Load

```
<?php
include 'file1.php';
include_once 'file2.php';
?>
```

در این صورت در صورتی که file2.php در file1.php فراخوان شده باشد، دیگر مجددا صدا زده نمی‌شود.



بیشتر بدانیم!

توضیحات^{۴۳}

توضیحات بخشی از کد هستند که برای کامپیوتر هیچ مفهومی ندارند، یعنی پردازشگر کد، کاری به این که شما در بخش توضیحات خود چه نوشته‌اید ندارد و از روی آن رد می‌شود. تنها فایده‌ی توضیحات، برای استفاده‌ی انسان‌ها است. توضیحات را در PHP می‌توان به ۳ طریق مشخص کرد:

```
// Hello! I'm a comment!
```

```
# Hello! I'm a comment too!
```

```
/* Hello! I'm another comment! */
```

دو نوع اول (// و #) تا آخر خط را به توضیح تبدیل می‌کنند. نوع سوم (/* ... */) از ابتدای /* تا رسیدن به /* (که لزومی ندارد در همان خط باشد) را به توضیح تبدیل می‌کند.

⁴³ Comments

۴-۱۹. توابع در PHP

توابع در PHP با دستور function مشخص می‌شوند.

```
<?php
function f() {
    echo "Hello!";
}
echo 123;
f();
echo 456;
?>
```

اجرای برنامه بالا منجر به تولید این خروجی می‌شود: "123Hello!456"

ابتدا تابع f را به عنوان تابعی که ورودی نمی‌گیرد (با استفاده از `()`) تعریف کرده‌ایم که کار این تابع این است که روی صفحه بنویسد "Hello!". سپس بین دو `echo`، این تابع را فراخوانی کرده‌ایم.

گرفتن ورودی در توابع آسان است. تنها این نکته را در اینجا مد نظر داشته باشید که متغیرهای ورودی نوع ندارند و فقط باید تعریف شوند.

```
<?php
function f($x, $y) {
    if ($x == $y)
        echo "Yes!";
    else
        echo "No!";
    echo "\n";
}

f(10, 20);
f("abc", 'abc');
f(100, true);
f(0, false);
?>
```

خروجی برنامه بالا به این شکل خواهد بود:

No!

Yes!

No!

Yes!

کار تابع f این است که دو ورودی می‌گیرد، اگر با هم برابر بودند می‌نویسد **Yes!** و اگر نبودند می‌نویسد **No!** و در انتها به ابتدای خط بعد می‌رود. همان‌طور که می‌بینید، در تعریف تابع f گفته- ایم که این تابع دو ورودی می‌گیرد (با (\$x, \$y)). سپس این تابع را سه بار با مقادیر مختلف (و انواع مختلف: int و string و bool) فراخوانی کرده‌ایم.

سؤال: چرا خروجی دوم برنامه با وجودی که ورودی‌های آن یکی میان علامت ' و دیگری میان " آمده بودند، "Yes!" بود؟

سؤال: چرا آخرین خروجی این برنامه "Yes!" بود؟ مگر دو مقدار 0 و false با هم برابر هستند؟

توابعی که تا به حال تعریف کرده‌ایم، مقدار برگشتی نداشتند. برای مشخص کردن این که یک تابع مقدار برگشتی دارد، هیچ تغییری در تعریف تابع نمی‌دهیم. در واقع توابع در PHP، می‌تواند مقدار برگشتی (از هر نوعی، حتی از انواع مختلف) داشته باشد، یا هیچ مقدار برگشتی‌ای نداشته باشد. مقدار برگشتی یک تابع با دستور `return` مشخص می‌شود.

در زیر می‌توانید تابعی را که دو ورودی گرفته و حاصل ضرب آن‌ها را برمی‌گرداند، مشاهده کنید.

```
<?php
function f($x, $y) {
    return $x * $y;
}
echo f(10, 5); // نتیجه: 50
?>
```

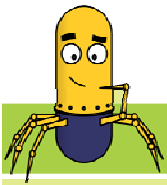
برای اینکه بیشتر با مفهوم توابع و امکانات آن آشنا شوید، در ادامه دو تعریف تابع برای محاسبه‌ی فاکتوریل یک عدد و `var_dump` آمده است.

```

<?php
function fact($n) {
    $r = 1;
    for($i = $n; $i > 1; $i--)
        $r = $r * $i;
    return $r;
}
?>

<?php
function my_var_dump($v) {
    if (is_array($v)) {
        echo "array {\u";
        foreach ($v as $k => $n) {
            echo "[\u]=$>".gettype($n). "(\u)\u";
        } // foreach
        echo "\u";
    } // if
    else // در صورتی که متغیر ورودی یک آرایه نباشد
        echo gettype($v). "(\u)\u";
} // function
?>

```



بیشتر بدانیم!

۴-۲۰. نشست ۴۵ در PHP

حال در ادامه سؤالی پیش می‌آید مبنی بر اینکه به طور منطقی بررسی نام کاربری و کلمه‌ی عبور فقط یک بار به هنگام ورود کاربر به سایت باید انجام بگیرد. پس از این بررسی

و در صورت صحت ورودی‌ها، چگونه در صفحات دیگر سایت بفهمیم که این کاربر قبلاً وارد سایت شده است؟ (یعنی از کجا بفهمیم این کاربر، همان کاربر قبلی است؟) در واقع مشکل از این‌جا ناشی می‌شود که پروتکل مورد استفاده‌ی ما (یعنی HTTP)، یک پروتکل بدون حالت^{۴۶} می‌باشد (در مقایسه با پروتکلی مانند FTP که حالت خود را حفظ می‌کند). یعنی همان‌طور که تا به حال دیدید، با هر بار کلیک کردن کاربر بر روی یک لینک در مرورگر، یک درخواست جدید و بدون هیچ ارتباطی با درخواست‌های قبلی به سرور وب فرستاده می‌شود.^{۴۷} حال سرور چگونه بفهمد که این شخص، همان شخص قبلی است؟ (که در یکی از درخواست‌های قبلی وارد سایت شده بود)

پاسخی که ممکن است ابتدا به نظر شما برسد، تشخیص کاربر از روی آدرس IP دستگاه فرستنده‌ی درخواست است. مشکلی که این روش دارد، این است که هیچ تضمینی برای این که فقط یک نفر از هر آدرس IP استفاده کند وجود ندارد. یعنی ممکن است در یک شبکه چندین دستگاه قرار داشته باشند، اما کل این شبکه فقط یک آدرس IP داشته باشد و از طریق NAT^{۴۸} این یک آدرس IP را بین این دستگاه‌ها به اشتراک بگذارد. پس از طریق آدرس IP نمی‌توان کاربر را شناخت.

راه حلی که برای حل این مشکل ارائه شده است، استفاده از نشست‌ها است. به این صورت که یک نشست مربوط به هر کاربر بر روی سرور ذخیره می‌کنیم و به آن یک شناسه اختصاص می‌دهیم. سپس این شناسه را در اختیار کاربر قرار می‌دهیم و کاربر در هر درخواست، این شناسه را برای سرور وب ارسال می‌کند و در نتیجه سرور کاربر را خواهد

⁴⁶ State-less

^{۴۷} به نظر شما چرا در FTP این مشکل پیش نمی‌آید؟ زیرا در FTP یک اتصال دائمی بین client و سرور برقرار می‌شود و تمامی عملیات از طریق این اتصال انجام می‌شود. اما در این‌جا برای هر درخواست، یک اتصال جدید باید برقرار شود.

⁴⁸ Network Address Translation

برای شروع یک نشست در PHP از تابع `session_start()` استفاده می‌کنیم:

`bool session_start ( void )`

پس از صدا زدن این تابع در برنامه، می‌توانید از متغیر `$_SESSION` به عنوان محل ذخیره‌سازی داده‌های نشست به صورت یک آرایه استفاده کنید. این آرایه مانند متغیر `$GLOBALS` در مورد متغیرهای سراسری برنامه عمل می‌کند، یعنی کلید آرایه برابر نام متغیر، و مقدار آن خانه برابر مقدار آن متغیر می‌باشد. برای روشن‌تر شدن موضوع، به یک مثال توجه کنید. فرض کنید می‌خواهیم صفحه‌ای بسازیم که تعداد دفعات دیده شدن آن توسط یک کاربر خاص را نمایش دهد (یعنی بار اول بنویسد ۱ و هر بار این عدد را برای همین کاربر یکی زیاد کند). برای این منظور تعداد دفعات دیده شدن صفحه را در یک متغیر به نام `count` در نشست قرار می‌دهیم و آن را چاپ می‌کنیم (دقت کنید که بار اول متغیر `count` در نشست وجود ندارد. دلیل استفاده از `if` اول همین است)

```
<?php
session_start();
if (empty($_SESSION['count']))
    $_SESSION['count'] = 1;
else
    $_SESSION['count']++;
?>
```

```
Hello! You have seen this page <?php echo $_SESSION['count']; ?> times!
```

حال با استفاده از نشست، می‌خواهیم صفحه‌ی ورود به سیستم و باقی صفحات محافظت شده را بازنویسی کنیم.

^{۴۹} زبان PHP برای پیاده‌سازی نشست‌ها، از موجوداتی به نام `Cookie` استفاده می‌کند. برای توضیح بیش‌تر درباره‌ی کوکی‌ها، به پیوست کوکی مراجعه کنید.

```

<?php
$u = $_GET["username"];
$p = $_GET["password"];
...
// 3) Execute SELECT query
$query = "SELECT * FROM `tblUsers` WHERE `UserName`='$_u' AND
`Password`='$_p'";
$result = mysql_query($query);
if ($result === false) {
    echo mysql_error();
    mysql_close($link);
    exit;
}

// 4) Fetch result(s)
$row = mysql_fetch_assoc($result);
if ($row === false) {
    echo "Access denied";
}
else {
    session_start();
    $_SESSION['loggedIn'] = true;
    echo "Hello ". $row["FirstName"]. "!";
}

// 5) Disconnect
mysql_close($link);
?>

```

اگر کاربر اطلاعات صحیح وارد کند، یک نشست برای او ایجاد شده و مقدار متغیر loggedIn در این نشست برابر true می‌شود. حال یک صفحه‌ی محافظت شده به کمک متغیر درون نشست می‌سازیم.

```

<?php
session_start();
if (!isset($_SESSION['loggedIn']) || ($_SESSION['loggedIn'] !== true))
{
    die('Access denied.');
```

PHP

```
}  
...  
?>
```



AJAX

۴-۲۱. AJAX چیست؟

AJAX یک زبان برنامه نویسی جدید نیست بلکه یک تکنیک برای جذاب تر کردن صفحات وب است به صورتی که زبان های برنامه نویسی شناخته شده وب را به گونه ای غیر معمول ترکیب می کند تا به برنامه نویسی این امکان را بدهد برنامه های وب را جذاب تر از آن چیزی که به صورت معمول رواج دارد به نمایش بگذارد.

AJAX چگونه عمل می کند؟

همانطور که در بالا گفته شد، AJAX یک تکنیک برای جذاب کردن صفحات وب و از طرف دیگر بالا بردن سرعت بروز رسانی صفحات است.

دربازدید از یک صفحه وب با کلیک بر روی هر لینک یا دکمه ای برای تغییر محتوای صفحه و یا ارسال یک فیلد به سرور باید تمام صفحه بروز شود و این به معنی درخواست تمام عکسها، لوگوها و کدها از سرور می باشد. اما شما با استفاده از این تکنیک فقط قسمتهای مورد نیاز را بروز می کنید و برای انجام هر عملیاتی نیاز به بروز کردن کل صفحه نیست این کار توسط فناوری AJAX انجام پذیر می شود.

در این قسمت یاد می گیریم تا فرم ثبت نام سایت را به صورتی تغییر دهیم که قبل از ثبت نام بدون بارگذاری مجدد^{۵۰} کل صفحه بتوان کنترل کرد که آدرس ایمیل بازدید کننده

⁵⁰ Reload

قبلا به عنوان کاربر سایت ثبت نام کرده است، یا نه.

کار را از تغییر صفحه فرم HTML شروع می‌کنیم. تا این مرحله یک صفحه ایجاد کرده‌ایم که فقط ظاهر صفحه مورد نظر را در بر گرفته و یک دکمه هم با عنوان check دارد که فعلا هیچ عملی را انجام نمیدهد ولی قرار است با کلیک بر روی این دکمه آدرس ایمیل کاربر رو چک شود.

```
<form action="index.php" method="post">
<table width="300" border="0">
  <tr>
    <td><label for="name">name </label></td>
    <td><input type="text" id="name"></td>
    <td>&nbsp;</td>
  </tr>
  <tr>
    <td><label for="user">user </label></td>
    <td><input type="text" id="user" ></td>
    <td style="text-align:left"><input type="button" value="check" ></td>
  </tr>
  <tr>
    <td><label for="pass">password</label></td>
    <td><input type="password" id="pass"></td>
    <td>&nbsp;</td>
  </tr>
  <tr>
    <td></td>
    <td colspan="2"><input type="submit" id="send"></td>
  </tr>
</table>
</form>
```

حالا باید تغییرات را به گونه‌ای اعمال کنیم که با کلیک روی دکمه check وجود یا عدم وجود آدرس ایمیل در فایل ثبت نام بررسی شود .

تنها کاری که لازم است انجام دهیم، ارسال آدرس ایمیل به سرور و چک کردن آن است که باید مرحله ارسال را از طریق جاوا اسکریپت و چک آن را از طریق PHP انجام دهیم.

برای این کار باید مقدماتی رو فراهم کنیم. در این قسمت به نوشتن کدهای سمت بازدیدکننده یا همان کدهای جاوا اسکریپت برای ارسال داده‌ها می‌پردازیم.

برای ایجاد ارتباط باید از پروتکل http استفاده کنیم باید یک شی از نوع xml http request بسازیم که از طریق آن درخواست‌های خود را به وسیله‌ی این پروتکل ارسال و پاسخ‌های ارسالی را کنترل کنیم.

به خاطر تفاوت‌هایی که در مرورگرهای مختلف وجود دارد نمیتوان از یک روش مشخص برای ساخت این شی استفاده کرد برای همین این کار را به صورت زیر انجام می‌دهیم.

```
function GetXmlHttpRequest()
{
var xmlhttp=null;
try
{
// Firefox, Opera 8.0+, Safari
xmlhttp=new XMLHttpRequest();
}
catch (e)
{
//Internet Explorer
try
{
xmlhttp=new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e)
{
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
```

```

}
return xmlHttp;
}

```

روش اول ساخت نمونه روشی است که در تمام مرورگرهای استاندارد پشتیبانی می‌شود ولی مشکل در IE⁵¹ پیش میاد که مجبوریم در IE برای هر کدام از نسخه‌ها از یک روش استفاده کنیم.

این مشکل به دلیل این هست که IE این شی رو به صورت activeX پیاده‌سازی کرده در حالی که در مرورگرهای معروف دیگر مانند Safari , Opera , Firefox این شی به صورت اشیاء داخلی کتابخانه‌های جاوا اسکریپت پیاده‌سازی شده است. به خاطر این تعارض، کدی که برای ساخت xmlhttprequest استفاده می‌کنیم باید شرطی برای ساخت از طریق activeX و یا اشیاء داخلی جاوا اسکریپت داشته باشد.

بعد از تعریف تابعی برای ساخت شی XMLHttpRequest در داخل جاوا اسکریپت حالا یک تابع به نام checkEmail برای ارسال آدرس ایمیل به سرور در نظر می‌گیریم.

```

function checkEmail()
{
xmlHttp=GetXmlHttpRequestObject();
if (xmlHttp==null)
{
alert ("Browser does not support HTTP Request");
return;
}
}
}

```

تا اینجا توسط تابع یک نمونه از شی XMLHttpRequest را ساختیم و در خط بعد متغیر را

⁵¹ Internet Explorer

با null چک کردیم. این کار به خطر این است که اگر مرورگری این شی را پشتیبانی نکرد پیام مناسبی برای کاربر نمایش داده شود.

حالا فرض را بر این می‌گیریم که نمونه‌های از آن کلاس ساخته شده، حالا این شی دارای یک متدهایی است که برای ادامه کار باید از این متدها استفاده کنیم.

برای ارسال آدرس ایمیل به صورت GET باید ایمیل آدرس رو در کنار آدرس صفحه قرار بدهیم و آن را به سرور ارسال کنیم.

ادامه کد را در زیر می‌بینید و سپس متدهای این شی را توضیح خواهیم داد.

```
function checkEmail()
{
    xmlHttp=GetXmlHttpRequestObject();
    if (xmlHttp==null)
    {
        alert ("Browser does not support HTTP Request");
        return;
    }

    url='index.php?user='+document.getElementById('user').value;
    xmlHttp.onreadystatechange=stateChanged;
    xmlHttp.open("GET",url,true);
    xmlHttp.send(null);
    return false;
}
```

تابع onreadystatechange با تغییر وضعیت تابع متناظر با خودش رو فراخوانی می‌کند.

۵ وضعیت مختلف برای ارسال اطلاعات ممکن است پیش بیاد که عبارتند از:

۱- هیچ عملیاتی صورت نگرفته

۲- در حال بارگذاری

۳- عملیات بارگذاری به اتمام رسیده

۴- خاتمه دادن عملیات

۵- کامل شدن اطلاعات ارسالی

اسم تابع نوشته شده در مقابل تابع `onreadystatechange` در صورت تغییر هر کدام از این وضعیت ها فراخوانی می شود.

تابع `open` سه پارامتر می گیرد که پارامتر اول نوع ارسال داده ها را مشخص می کند. پارامتر دوم آدرس صفحه ای را که می خواهیم اطلاعات را برای آن ارسال کنیم. چون در اینجا ما اطلاعات رو به صورت `GET` ارسال میکنیم فیلد `email` رو هم در `url` قرار داده و ارسال می کنیم و پارامتر سوم هم نشان دهنده همگام یا ناهمگام بودن فراخوانی است .

مقدار متغیر `url` در صورتی که هنگام پر کردن فرم مقدار عبارت `email` رو برابر `info@webook.ir` قرار بدهیم مقدار متغیر `url` برابر با مقدار زیر خواهد بود .

`index.php?email=info@webook.ir`

تابع `send` اگر اطلاعات ارسالی به صورت `data` وجود داشته باشد از این تابع برای ارسال استفاده می کنیم اما چون ما در اینجا اطلاعات رو به صورت `GET` ارسال کردیم مقدار این تابع رو برابر `null` قرار می دهیم .

تا اینجا کار هر یک از توابع را توضیح دادیم و حالا یک بار برنامه را تا اینجا چک می کنیم .

کاربر وارد سایت می‌شود و اطلاعات خود را پر می‌کند و برای چک email بر روی دکمه check کلیک می‌کند. در این هنگام ما باید تابع checkEmail را فراخوانی کنیم. بنابراین دکمه check رو به صورت زیر تغییر می‌دهیم .

```
<input type="button" value="check" onclick="checkEmail()">
```

تابع checkEmail فراخوانی می‌شود و در ابتدا یک نمونه از شی XmlHttp ساخته می‌شود و بعد در صورت درست نشدن این نمونه از شی XmlHttp یک پیام به کاربر نمایش می‌دهد و از ادامه اجرای تابع صرف نظر می‌کند اما در صورت رد کردن این مرحله و ادامه کار متغیری که تعریف کردیم، ادرس صفحه ای را که نام کاربری باید به آن ارسال بشود همراه با پارامتر آدرس ایمیل در بر می‌گیرد.

توسط تابع open فراخوانی مورد نظر را تنظیم می‌کنیم. در خواست هنگامی ارسال می‌شود که متد send اجرا بشود و در انتها متد send را با مقدار null فراخوانی کردیم یعنی هیچ مقداری رو به صورت post نمی‌خواهیم ارسال کنیم. در ادامه کار در صورت تغییر هر کدام از وضعیت‌ها، تابع stateChanged فراخوانی می‌شود که البته هنوز این تابع را تعریف نکردیم.

حالا تنها کاری که مانده در سمت بازدیدکننده انجام بدهیم تابع stateChanged هست که در صورت تغییر وضعیت باید فراخوانی بشود.

این تابع را به صورت زیر تعریف می‌کنیم.

```
function stateChanged()
{
```

```
if (xmlHttp.readyState==4 || xmlHttp.readyState=="complete")
{
    if (xmlHttp.responseText==1)
    {
        document.getElementById("user").style.border="1px solid #FF0000";
    }
    else{
        document.getElementById("user").style.border="1px solid #00CC00";
    }
}
}
```

متد readyState وضعیت درخواستمان را نگه داری میکند در بالا دو حالت مختلف را ما چک کردیم. اول وضعیت ۴ و بعد وضعیت complete در واقع هر دوی این وضعیت ها یکی هستند و به کامل بودن صفحه اشاره می کنند.

متد responseText هم نتیجه ارسالی از سرور را بر میگرداند

در سرور اگر آدرس ایمیل وجود داشته باشد قرار است مقدار یک و در غیر اینصورت قرار مقدار صفر رو برگردانده شود. حالا در تابع stateChanged چک می کنیم اگر مقدار برگشتی از سرور برابر یک باشد حاشیه جعبه متن مربوط به email را برابر با قرمز و در غیر اینصورت برابر با سبز قرار می دهیم .

حال فقط لازم است که در سمت سرور به فایل وصل شده و نتیجه را برای این صفحه برگردانیم.

کدهای صفحه اصلی یا همان صفحه ajax به صورت زیر شد.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" >
<title>Untitled Document</title>
```

```

<script type="text/javascript">
    function GetXmlHttpRequestObject()
    {
        var xmlHttp=null;
        try
        {
            // Firefox, Opera 8.0+, Safari
            xmlHttp=new XMLHttpRequest();
        }
        catch (e)
        {
            //Internet Explorer
            try
            {
                xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");
            }
            catch (e)
            {
                xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
            }
        }
        return xmlHttp;
    }

    function checkEmail()
    {
        xmlHttp=GetXmlHttpRequestObject();
        if (xmlHttp==null)
        {
            alert ("Browser does not support HTTP Request");
            return;
        }

        url='index.php?user='+document.getElementById('user').value;
        //alert(url);
        xmlHttp.onreadystatechange=stateChanged;
        xmlHttp.open("GET",url,true);
        xmlHttp.send(null);

    }

```

```
function stateChanged()
{
if (xmlHttp.readyState==4 || xmlHttp.readyState=="complete")
{
    if (xmlHttp.responseText=='1')
    {
        document.getElementById("user").style.border="1px solid #FF0000";
    }
    else{
        document.getElementById("user").style.border="1px solid #00CC00";
    }

}
}
```

```
</script>
</head>
```

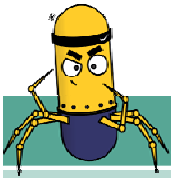
```
<body>
<form action="index.php" method="post">
<table width="300" border="0">
<tr>
<td><label for="name">name </label></td>
<td><input type="text" id="name" /></td>
<td>&nbsp;</td>
</tr>
<tr>
<td><label for="email">email </label></td>
<td><input type="text" id="user" /></td>
<td style="text-align:left"><input type="button" value="check"
onclick="checkUser()"/></td>
</tr>
<tr>
<td><label for="pass">password</label></td>
<td><input type="password" id="pass" /></td>
<td>&nbsp;</td>
</tr>
<tr>
<td></td>
<td colspan="2"><input type="submit" id="send" /></td>
</tr>
</table>
```

```
</form>
</body>
</html>
```

این صفحه را با نام index.html ذخیره می‌کنیم .

در فایل برای ذخیره اطلاعات با نام Ajax.txt ایجاد کرده و فیلدهای لازم برای هر کاربر را تعریف می‌کنیم. از جمله email و password . چند تا کاربر را هم داخل آن وارد می‌کنیم.

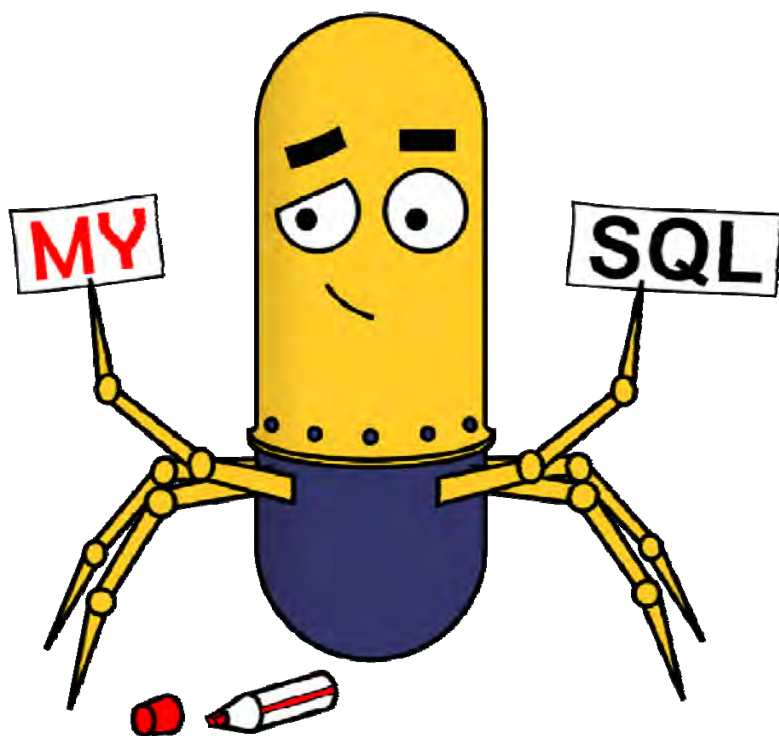
حال فایل را باز می‌کنیم و تمام رکوردهایی که با آدرس ایمیل فرستاده شده از طرف بازدیدکننده که آن رو به صورت get ارسال کردیم وجود دارد را جستجو می‌کنیم. با بررسی نتیجه می‌توانیم متوجه شویم که آدرس ایمیل قبلا ثبت شده است یا نه.



تمرین

برنامه‌ی PHP ای را بنویسید که در ورودی آدرس ایمیل را گرفته و در یک فایل چک کند. اگر آن آدرس وجود داشت مقدار یک و در غیر اینصورت مقدار صفر را برگرداند.

برنامه نویسی MySQL



برنامه نویسی MySQL

۵-۱. بررسی اجزای تشکیل دهنده ی یک بانک اطلاعاتی

قبل از شروع کار با MySQL لازم است نکات کلی در مورد بانک اطلاعاتی را بدانید و با آن ها آشنا شوید. بانک اطلاعاتی یک سیستم نگهداری داده هاست که داده ها را به صورت منظم و جزء به جزء دسته بندی می کند. یعنی در واقع یک بانک اطلاعاتی مجموعه ای از داده های زیاد و یا کم را برای شما سازماندهی می کند. به طور کلی در یک بانک اطلاعاتی ما با سه بخش زیر سرو کار داریم : جدول ها ، رکوردها و فیلدها.

جدول : هر جدول در بانک اطلاعاتی مجموعه ای از داده های دسته بندی شده و مرتبط با آن بانک اطلاعاتی هستند که در واحدی مجزا به نام جدول نگهداری می شوند. اما نام گذاری این مجموعه به نام جدول به آن خاطر است که داده های شما به صورت سطر و ستون ها دسته بندی می شوند و همان طور که می دانید مجموعه ای از سطر و ستون ها در دنیای خارج از کامپیوتر نیز جدول نامیده می شوند.

نام	نام خانوادگی	سن
علی	علوی	۱۶
فاطمه	احمدی	۱۵

رکوردها یا سطرها : رکوردها در واقع همان سطرهای جدول را تشکیل می دهند که جزیی از

جدول می باشند. (که جدول نیز خود جزیی از بانک اطلاعاتی می باشد) رکوردها شامل اطلاعات مرتبط به هم در مورد یک فرد شی و یا ... می باشند که ارتباط نزدیکی به هم دارند و همدیگر را کامل می کنند.

فیلدها: هر سطر از جدول ها نیز از بخش های کوچکی به نام فیلد تشکیل شده است. دقت کنید که فیلد را با ستون اشتباه نگیرید.

پرس و جو^{۵۲}: پرس و جو یک سوال و یا درخواست است. پرس و جو یک سوال و یا درخواست است توسط آن می توانیم اطلاعات لازم را درون پایگاه داده جستجو کنیم. نمونه ای از دستور پرس و جو به صورت زیر است.

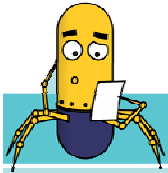
```
SELECT LastName FROM Persons
```

۵-۲. نحوه اتصال به پایگاه داده

قبل از اینکه بخواهید به اطلاعات پایگاه داده دسترسی پیدا کنید، باید یک اتصال به پایگاه داده ایجاد کنید. با استفاده از PHP، این اتصال با تابع `MYSQL_CONNECT()` و به صورت زیر انجام می شود.

```
mysql_connect(servername,username,password);
```

برای استفاده از دستور فوق باید اطلاعات لازم برای دسترسی را به جای سه قسمت مشخص شده از میزبان وارد کرد.



مثال

نحوه اتصال و قطع سطح به پایگاه داده

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code
mysql_close($con);
?>
```

نحوه ایجاد جدول در پایگاه داده

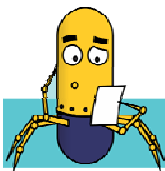
برای استفاده از پایگاه داده لازم است تا در ابتدا جدول‌های لازم را در آن ایجاد کنیم. برای

ایجاد جدول از دستور زیر استفاده می‌کنیم.

```
CREATE TABLE table_name
(
    column_name1 data_type,
    column_name2 data_type,
    column_name3 data_type,
    ....
)
```

در صورتی که بخواهیم ایجاد آن را به صورت کد کامل در PHP بنویسیم به صورت زیر ایجاد

می‌شود.



مثال

نحوه ایجاد جدول در پایگاه داده

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

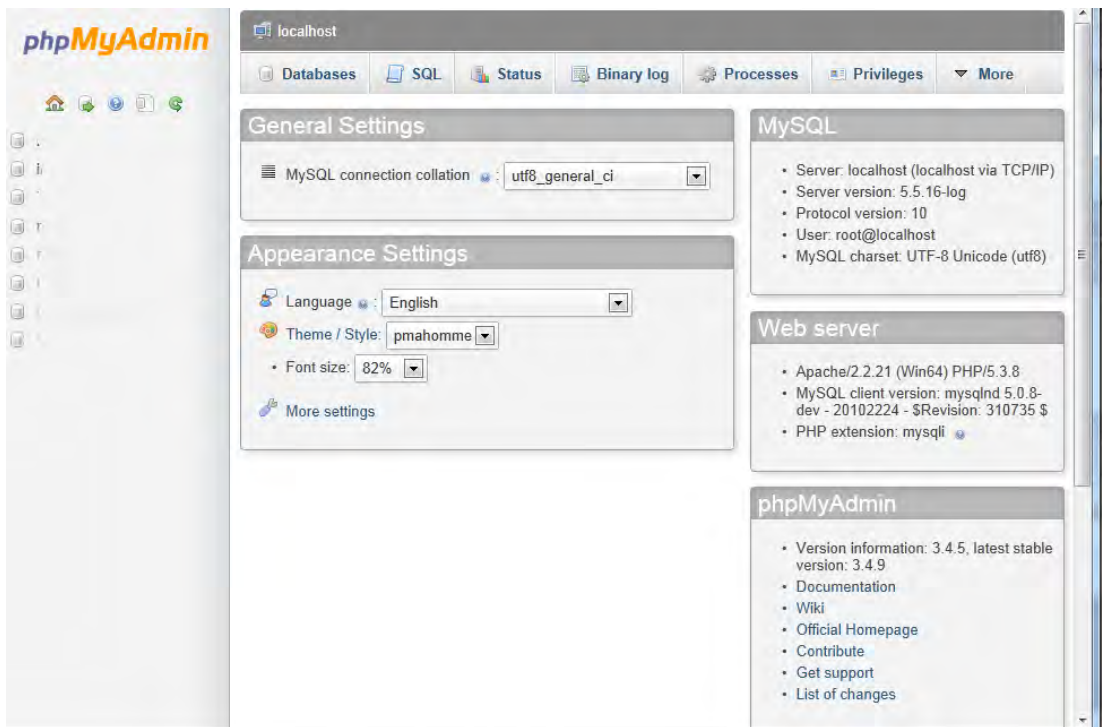
// Create database
if (mysql_query("CREATE DATABASE my_db",$con))
{
    echo "Database created";
}
else
{
    echo "Error creating database: " . mysql_error();
}

// Create table
mysql_select_db("my_db", $con);
$sql = "CREATE TABLE Persons
(
    FirstName varchar(15),
    LastName varchar(15),
    Age int
)";

// Execute query
mysql_query($sql,$con);
```

```
mysql_close($con);
?>
```

البته امروزه بیشتر از نرم‌افزارهای رایجی مثل PhpMyAdmin برای ایجاد و مدیریت پایگاه داده استفاده می‌شود.



شکل ۵-۱: نمایی از phpMyAdmin

کار با استفاده از phpMyAdmin بسیار ساده است و به این علت است که کاربران برای اصلاح و ساخت جدول از آن استفاده می‌کنند.

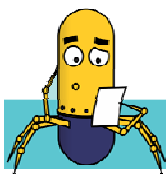
نحوه ورود اطلاعات در جدول در پایگاه داده

برای وارد کردن اطلاعات در پایگاه داده به صورت زیر است.

```
INSERT INTO table_name
VALUES (value1, value2, value3,...)
```

در صورتی که بخواهیم ایجاد آن را به صورت کد کامل در PHP بنویسیم به صورت زیر ایجاد

می شود.



مثال

صفحه PHP ایجاد کنیم که اطلاعات یک فرم را درون پایگاه داده ذخیره نماید.

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$sql="INSERT INTO Persons (FirstName, LastName, Age)
VALUES
('$_POST[firstname]',$_POST[lastname'],$_POST[age])";

if (!mysql_query($sql,$con))
{
    die('Error: ' . mysql_error());
}
echo "1 record added";
```

```
mysql_close($con)
?>
```

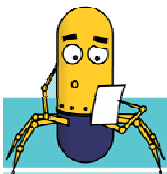
نحوه خواندن اطلاعات از جدول در پایگاه داده

برای خواندن اطلاعات از پایگاه داده به صورت زیر است.

```
SELECT column_name(s)
FROM table_name
```

در صورتی که بخواهیم ایجاد آن را به صورت کد کامل در PHP بنویسیم به صورت زیر ایجاد

می شود.



مثال

صفحه PHP ایجاد کنیم که اطلاعات یک فرم را درون پایگاه داده ذخیره نماید.

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons");
```

```
while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br />";
}

mysql_close($con);
?>
```

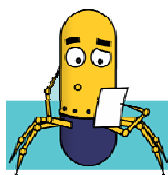
نحوه پرس و جو اطلاعات از جدول در پایگاه داده

برخی مواقع مانند مواقعی که می‌خواهید اطلاعات نام کاربری و کلمه عبور را در پایگاه داده پیدا کنید لازم است تا بتوانید درون اطلاعات پایگاه داده جستجو کنید. برای پرس و جو اطلاعات از پایگاه داده به صورت زیر عمل می‌کنیم.

```
SELECT column_name(s)
FROM table_name
WHERE column_name operator value
```

در صورتی که بخواهیم ایجاد آن را به صورت کد کامل در PHP بنویسیم به صورت زیر ایجاد

می‌شود.



مثال

صفحه پرس و جو اطلاعات از درون پایگاه داده.

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons
WHERE FirstName='فاطمه'");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br />";
}
?>
```

همچنین برای مرتب کردن داده‌های خروجی نیز می‌توانید به صورت زیر عمل کنید.

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name(s) ASC|DESC
```

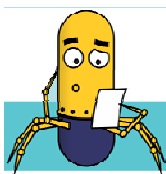

اصلاح و به روز کردن اطلاعات در جدول پایگاه داده

گاهی لازم است تا تغییراتی در اطلاعات موجود در جدول ایجاد کنید و اطلاعات آن‌ها را به روز کنید.

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value
```

در صورتی که بخواهیم ایجاد آن را به صورت کد کامل در PHP بنویسیم به صورت زیر ایجاد

می‌شود.



مثال

صفحه PHP ایجاد کنیم که اطلاعات جدول را به اطلاعات جدید تغییر دهد.

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

mysql_query("UPDATE Persons SET Age = '36'
WHERE FirstName = 'علی' AND LastName = 'علوی');
```

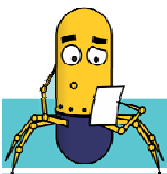
```
mysql_close($con);
?>
```

حذف یک سطر از اطلاعات در جدول پایگاه داده

ممکن است بخواهید یک سطر از اطلاعاتی را که قبلاً وارد جدول کرده‌اید را از آن حذف کنید. برای این کار می‌توانید مطابق دستور زیر عمل کنید.

```
DELETE FROM table_name
WHERE some_column = some_value
```

در صورتی که بخواهیم ایجاد آن را به صورت کد کامل در PHP بنویسیم به صورت زیر ایجاد می‌شود.



مثال

صفحه PHP ایجاد کنیم که یک سطر از اطلاعات را حذف نماید.

```
<?php
$con = mysql_connect("localhost","ali","pass");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
```

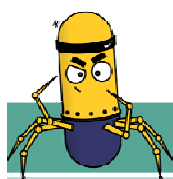
```
}
```

```
mysql_select_db("my_db", $con);
```

```
mysql_query("DELETE FROM Persons WHERE LastName='علوی'");
```

```
mysql_close($con);
```

```
?>
```



تمرین

با دانسته‌های خود از بخش بالا، در این مرحله برای سایت خود یک بخش اخبار شامل قسمت‌های زیر ایجاد کنید.

یک قسمت برای ورود اخبار در نظر بگیرید و در آن امکان افزودن خبر جدید با داشتن عنوان، خلاصه خبر، متن خبر و تاریخ را قرار دهید.

یک قسمت برای نمایش این خبرها زیر هم ایجاد کنید.

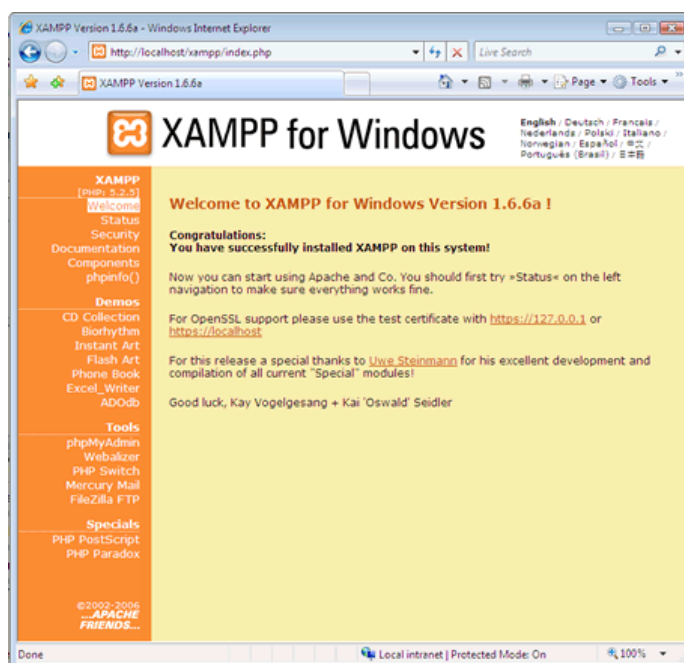
در یک قسمت دیگر امکان ویرایش این خبرها را برنامه‌نویسی کنید.

ضمیمه

پیوست‌ها

پیوست یک:

در این پیوست نحوه نصب نرم افزار XAMPP توضیح داده شده است.
 با نصب این نرم افزار دو موتور MySQL و Apache نیز نصب خواهد شد که می‌توانید با استفاده از آن پروژه‌های خود را بنویسید و اجرا کنید.

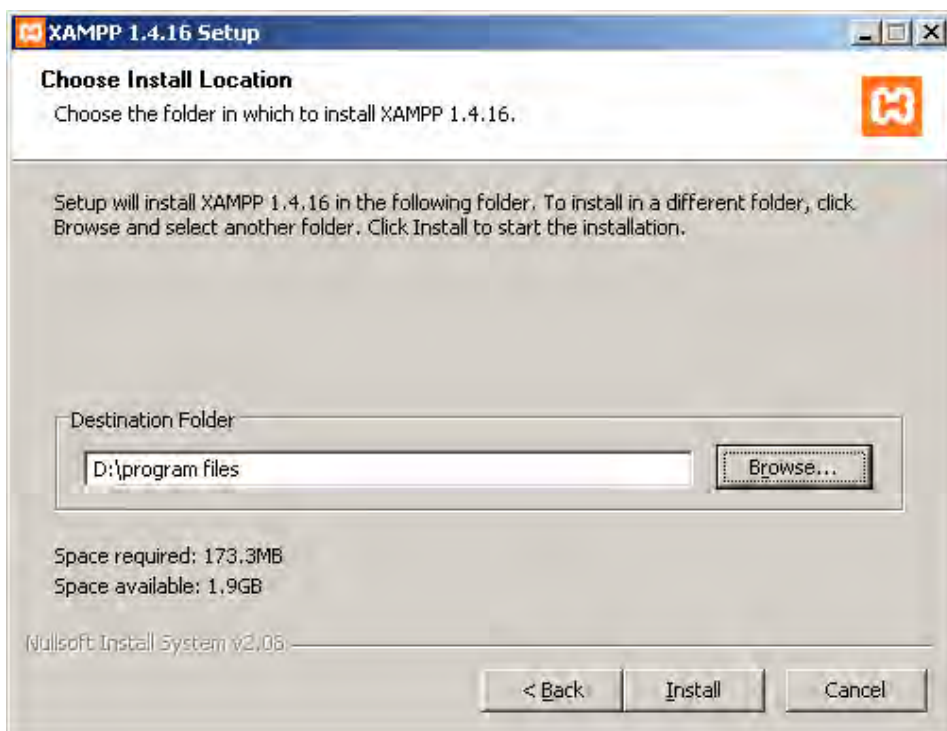


مرحله ۱: نصب XAMPP

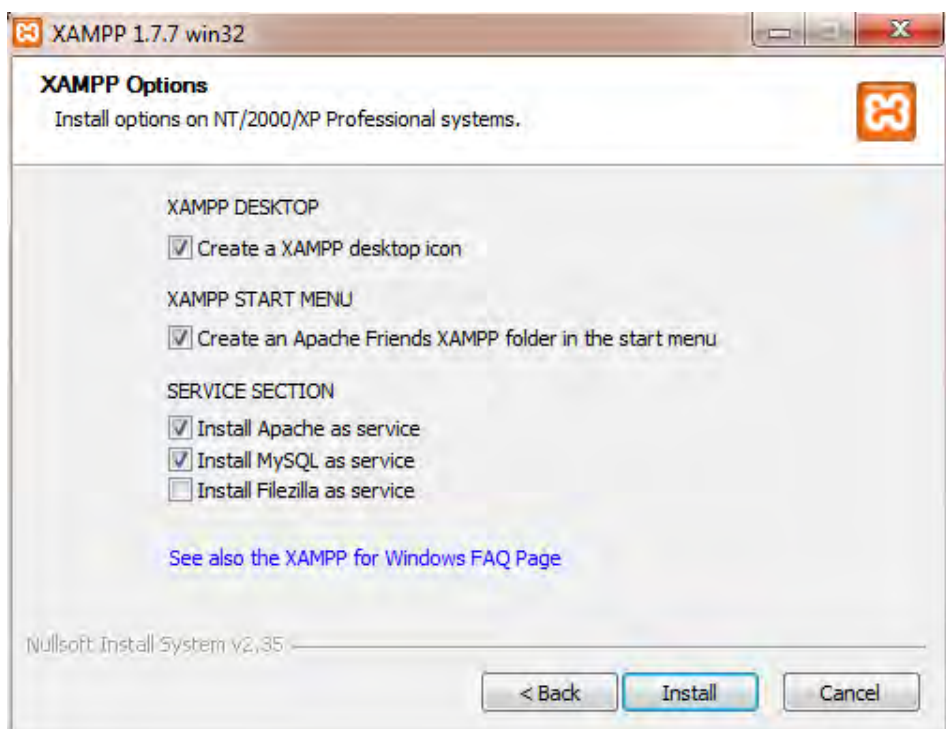
برای شروع کار با PHP نیاز به نصب مفسر PHP داریم که ما پکیج XAMPP را برای انتخاب کردیم. مراحل نصب رو باید به صورت زیر دنبال کنیم:
 در اولین مرحله خوش آمدگویی داریم پس Next رو کلیک می‌کنیم.



مسیر نصب را مشخص کنید.

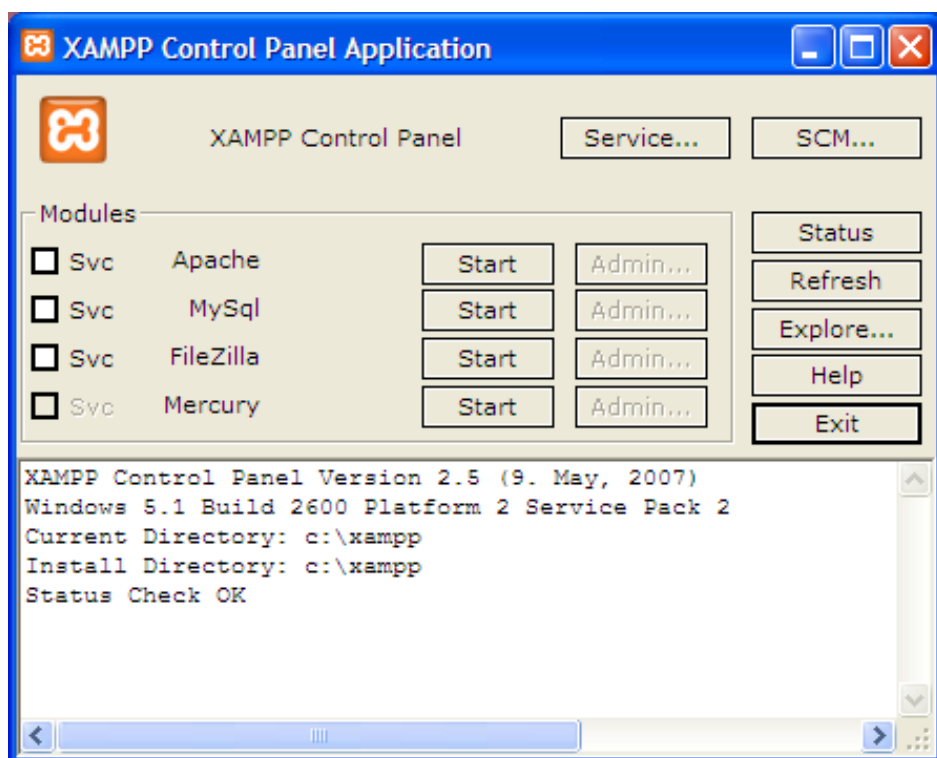


در صورتی که می‌خواهید Apache و MySQL به عنوان سرویس‌های ویندوز نصب شوند، گزینه‌های Install MySQL as Service و Install Apache as Service رو تیک بزنید! انتخاب این گزینه‌ها اختیاری است. تنها نکته‌ای این که در صورتیکه IIS رو سیستم شما نصب باشد و Apache را نصب کنید، باید حتما پورت IIS یا Apache را تغییر بدهید. در غیر اینصورت Apache پیغام خطای Port Busy می‌دهد! حال دکمه Install را بزنید.



اگر به مشکل خاصی بر نخورده باشید کار نصب تمام شده است.

کنترل پنل XAMPP مطابق شکل پایین پیش روی شماست و آن میتوانید Apache و MySQL را Start یا Stop کنید. فراموش نکنید اگر گزینه Install as Service را انتخاب نکرده باشید باید هر بار که می‌خواهید روی وب سایت خود کار کنید Apache و MySQL فعال باشند یعنی باید روی دکمه Start آن کلیک کنید. (به صورتی که کلمه Running کنار Apache و MySQL نمایش داده شود).

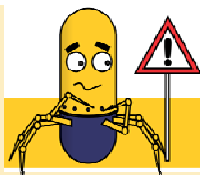
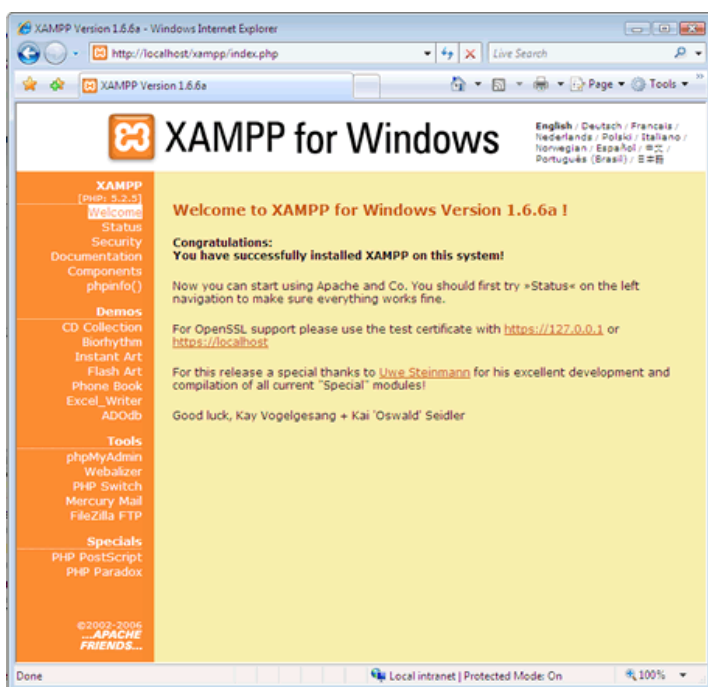


حالا مرورگر خود را باز کنید و در قسمت ورود آدرس مرورگر، عبارت localhost یا 127.0.0.1 را تایپ کنید. اگر همه مراحل درست انجام شده باشد، باید صفحه زیر را در مرورگر مشاهده کنید. در این صفحه می‌توانید زبان صفحات XAMPP را انتخاب کنید.



[English](#) / [Deutsch](#) / [Français](#) / [Nederlands](#) / [Polski](#) / [Italiano](#) / [Norsk](#) / [Español](#) / [中文](#) / [Português \(Brasil\)](#) / [日本語](#)

حالا باید صفحه زیر برای شما نمایش داده شود.



توجه!

شما می‌توانید آخرین نسخه نرم‌افزار XAMPP را از سایت <http://webook.ir> دریافت کنید.

مرحله ۲: نحوه اجرای برنامه های php

بعد از نصب نرم‌افزار XAMPP برای اجرای یک پروژه می‌توانید به این روش عمل کنید.

فایل‌های پروژه را در مسیر کپی کرده `xampp/htdocs/sample/..` و بعد از استارت کردن `apache` ، `mysql` ، این آدرس `http://127.0.0.1/sample` را تایپ کنید تا پروژه اجرا شود.

پیوست دو:

تابع	توضیحات
isFinite(x)	کنترل می‌کند که آیا عدد وارد شده متناهی است یا خیر.
isNaN(x)	کنترل می‌کند که مقدار وارد شده عدد است یا نه.
Number(x)	سعی می‌کند ورودی را به عدد تبدیل کند.
parseInt(x)	سعی می‌کند رشته‌ی حرفی ورودی را به عدد صحیح تبدیل کند.
parseFloat(x)	سعی می‌کند رشته‌ی حرفی ورودی را به عدد اعشاری تبدیل کند.

. توابع آزاد کار با اعداد در جاوا اسکریپت

متغیر	توضیحات
abs(x)	قدرمطلق متغیر ورودی را باز می‌گرداند.
acos(x)	آرک کسینوس متغیر ورودی را باز می‌گرداند.
asin(x)	آرک سینوس متغیر ورودی را باز می‌گرداند.
atan(x)	آرک تانژانت متغیر ورودی را باز می‌گرداند.
ceil(x)	کوچکترین عدد صحیح بزرگتر مساوی متغیر ورودی را باز می‌گرداند.
cos(x)	کسینوس متغیر ورودی را باز می‌گرداند.
exp(x)	مقدار e^x را باز می‌گرداند.
floor(x)	بزرگترین عدد کوچکتر از متغیر ورودی را باز می‌گرداند.
log(x)	لگاریتم طبیعی متغیر ورودی را باز می‌گرداند.

max(a, b)	ماکزیمم دو مقدار ورودی را باز می‌گرداند
min(a, b)	مینیمم دو مقدار ورودی را باز می‌گرداند
pow(x, y)	مقدار x^y را باز می‌گرداند
random()	مقداری تصادفی بین ۰ و ۱ باز می‌گرداند
round(x)	مقدار گرد شده‌ی متغیر ورودی را باز می‌گرداند.
sin(x)	سینوس متغیر ورودی را باز می‌گرداند.
sqrt(x)	ریشه‌ی طبیعی متغیر ورودی را باز می‌گرداند.
tan(x)	تانژانت متغیر ورودی را باز می‌گرداند.

توابع کلاس Math برای کار با اعداد

متغیر	توضیحات
E	عدد ثابت e برای لگاریتم پایه‌ی طبیعی.
PI	عدد ثابت پی برای محاسبه‌ی محیط و مساحت دایره.
SQRT2	عدد گنگ ریشه‌ی صحیح ۲.

تابع	توضیحات
length	طول رشته را باز می‌گرداند.
charAt(x)	مقدار حرف x ام در رشته را بر می‌گرداند.
concat(v1, v2, ...)	مقدار ورودی‌های v1, v2 و ... را به هم چسبانده و به عنوان خروجی باز می‌گرداند. در صورت استفاده از این تابع

رشته‌ی اصلی تغییر نمی‌کند.	
این تابع اندیس شروع بخشی از رشته را که مشابه <code>substr</code> است را باز می‌گرداند. در صورت لزوم می‌توان ورودی دوم این تابع را هم مقدار داد تا جستجو را از کاراکتر اندیس دوم شروع کند.	<code>indexOf(substr[,start])</code>
دقیقاً مشابه <code>indexOf</code> عمل می‌کند با این تفاوت که جستجو را از آخر رشته شروع می‌کند. واضح است که ورودی <code>start</code> محل شروع جستجو به عقب را می‌تواند مشخص کند.	<code>lastIndexOf(substr[,start])</code>
با استفاده از این تابع تمام مقادیری از رشته که با <code>regex</code> مطابقت داشته باشند را می‌توان با <code>newtext</code> تعویض کرد.	<code>replace(regex, newtext)</code>
زیر رشته‌ای را که از محل <code>start</code> شروع شده و تا انتهای رشته ادامه دارد باز می‌گرداند. در صورت استفاده از ورودی دوم، زیر رشته تا مقدار ورودی دوم ادامه خواهد داشت. دقت شود که کاراکتر شماره‌ی <code>end</code> در خروجی نخواهد بود.	<code>slice(start[, end])</code>
رشته را به توجه به جداکننده‌ای که در <code>delimiter</code> مشخص شده است می‌شکند و یک آرایه از زیر رشته‌ها را باز می‌گرداند.	<code>split(delimiter[, limit])</code>
زیر رشته‌ای را که از محل <code>start</code> شروع شده و تا انتهای رشته ادامه دارد باز می‌گرداند. در صورت استفاده از ورودی دوم، طول زیر رشته به مقدار آن محدود می‌شود.	<code>substr(start[, length])</code>

substring(from[, to]) زیر رشته‌ای را که از محل from شروع شده و تا انتهای رشته ادامه دارد باز می‌گرداند. در صورت استفاده از ورودی دوم، زیر رشته تا مقدار ورودی دوم ادامه خواهد داشت. دقت شود که کاراکتر شماره‌ی to در خروجی نخواهد بود.	
toLowerCase() رشته را بطور کلی به حروف کوچک تبدیل کرده و باز می‌گرداند.	
toUpperCase() رشته را بطور کلی به حروف بزرگ تبدیل کرده و باز می‌گرداند.	