考试说明

闭卷考试。

考试时长：120 分钟。

考试题型及分数分布：

1) 单选题（每小题1 分，共30 分）

2) 填空题（每空1 分，共10 分）

3) 简答题（每题4 分，共20 分）

4) 代码填充题（每段语句2 分，共30 分）

5) 代码设计题（10 分）

**1 WWW 简介**

**WWW 的基本工作原理（输入URL 到浏览器显示页面，后台处理过程）**

过程：

1. 输入URL

**2．**浏览器根据特定规则解析URL,获取域名

**3．**浏览器获取IP（DNS操作）

按优先度排列

* **浏览器缓存 -**浏览器会缓存DNS记录一段时间。
* **操作系统缓存** - 如果浏览器缓存不包含所需的记录，则浏览器进行系统调用

如windows系统中的hosts文件

* **路由器缓存** - 请求继续到您的路由器，路由器通常具有自己的DNS缓存。
* **ISP DNS缓存** - 检查的下一个位置是缓存ISP的DNS服务器。
* **递归搜索** - ISP的DNS服务器开始搜索（递归解析与迭代解析）
* **返回IP**

**4.** 浏览器向web服务器发送http请求

* 浏览器构建http请求（可用chrome或火狐的开发者工具查看）
* 通过TCP/IP协议簇传输http请求（比如TCP有名的三次握手与四次挥手）

**5．**服务器获得请求，进行处理（比较复杂，比如PHP档要经过服务器运行后，才将结果返回）返回一个HTML回应

**6.**浏览器处理HTML响应

* 根据响应头解析出HTML页面
* 浏览器开始渲染HTML
* 浏览器发送嵌入HTML的对象的请求（此时就回到步骤2。）我感觉就是一个递归的过程。。。。

**WWW 的资源定位方式**

http://www.cxyzf.cn:8080/news/index.php?boardID=5&ID=24618&page=1#name

从上面的URL可以看出，一个完整的URL包括以下几部分：

1. 协议部分：该URL的协议部分为“http：”，这代表网页使用的是HTTP协议。在Internet中可以使用多种协议，如HTTP，FTP等等本例中使用的是HTTP协议。在"HTTP"后面的“//”为分隔符

**2.**域名部分：该URL的域名部分为“www.cxyzf.cn”。一个URL中，也可以使用IP地址作为域名使用

**3.** 端口部分：跟在域名后面的是端口，域名和端口之间使用“:”作为分隔符。埠不是一个URL必须的部分，如果省略埠部分，将采用默认端口，比如http协议默认端口为80端口

**4.**虚拟目录部分：从域名后的第一个“/”开始到最后一个“/”为止，是虚拟目录部分。虚拟目录也不是一个URL必须的部分。本例中的虚拟目录是“/news/”

5.文件名部分：从域名后的最后一个“/”开始到“？”为止，是文件名部分，如果没有“?”,则是从域名后的最后一个“/”开始到“#”为止，是档部分，如果没有“？”和“#”，那么从域名后的最后一个“/”开始到结束，都是文件名部分。本例中的文件名是“index.php”。文件名部分也不是一个URL必须的部分，如果省略该部分，则使用预设的文件名

**6.**锚部分：从“#”开始到最后，都是锚部分。本例中的锚部分是“name”。锚部分也不是一个URL必须的部分

#代表网页中的一个位置。其右面的字符，就是该位置的标识符。比如，http://www.example.com/index.html#print就代表网页index.html的print位置。浏览器读取这个URL后，会自动将print位置滚动至可视区域。

#是用来指导浏览器动作的，对服务器端完全无用。所以，HTTP请求中不包括#。

比如，访问下面的网址

http://www.example.com/index.html#print

浏览器实际发出的请求是这样的：

http://www.example.com/index.html

**7.**参数部分：从“？”开始到“#”为止之间的部分为参数部分，又称搜索部分、查询部分。本例中的参数部分为“boardID=5&ID=24618&page=1”。参数可以允许有多个参数，参数与参数之间用“&”作为分隔符。

**静态网页与动态网页**

直观感觉上：

静态网页：指web页面中供人们浏览的数据始终是不变。

动态网页：指web页面中供人们浏览的数据是由程序产生的，不同时间点访问web页面看到的内容各不相同。

开发技术上：  
静态网页：HTML

动态网页：JSP/Servlet、ASP、PHP等等

**2 HTML**

**特殊字符，如非换行空格**

& nbsp

HTML中中各种转义字符表：<http://tool.oschina.net/commons?type=2>（感觉其实也就用过空格符，而且有自动补全，为什么还要求死记？？？？）

**水平线元素，宽度控制**

<hr/>我是一条分割线

Width属性

**顺序列表元素**

<ol>

<li>Coffee</li>

<li>Milk</li>

</ol>

**表格：**

表格由 <table> 卷标来定义。每个表格均有若干行（由 <tr> 标签定义），每行被分割为若干单元格（由 <td> 标签定义）。字母 td 指表格数据（table data），即数据单元格的内容。数据单元格可以包含文本、图片、列表、段落、窗体、水平线、表格等等。

<table border="1">

<tr>

<td>row 1, cell 1</td>

<td>row 1, cell 2</td>

</tr>

<tr>

<td>row 2, cell 1</td>

<td>row 2, cell 2</td>

</tr>

</table>

**表格水平对齐、表格边框、表格宽度、背景颜色；**

Align属性为水平对齐

Border属性为边框

Width属性为宽度

Bgcolor属性为背景颜色

**单元格内容与单元格框之间的距离、单元格与单元格之间的距离**

Cellpadding属性

Cellspacing属性

以像素为单位

**行元素、单元格元素、行背景颜色**

<tr><tr/>——行元素

<td><td/>——单元格元素

Bgcolor属性为背景颜色

1、块级元素 行级元素

块级（block）元素的特点：

①总是在新行上开始；

②高度，行高以及外边距和内边距都可控制；

③宽度缺省是它的容器的100%，除非设定一个宽度；

④它可以容纳内联元素和其他块元素。

内联（inline）元素的特点：

①和其他元素都在一行上；

②高，行高及外边距和内边距不可改变；

③宽度就是它的文字或图片的宽度，不可改变；

④内联元素只能容纳文本或者其他内联元素。

**行合并、列合并**

colspan :行合并

rowspan：列合并

**单元格内容水平对齐、单元格内容垂直对齐；**

Align属性为水平对齐

vertical-align属性为垂直对齐

**超链接元素：在新窗口打开链接、电子邮件链接、定义书签（锚点）、跳转到锚点处**

<a> 标签定义超链接，用于从一张页面连结到另一张页面。

<a> 元素最重要的属性是 href 属性，它指示链接的目标。

如 <a href="http://www.baidu.com">百度</a>

**相对路径：上一级目录、子活页夹（？？？？？？）**

../表示源文件所在目录的上一级目录，../../表示源文件所在目录的上上级目录，以此类推。

**HTML5 的doctype**

<!DOCTYPE html>

**HTML5 不再支持的HTML4 元素**

感觉我们都是直接上的HTML5，还有必要考这个，根本没用过HTML4规则啊啊啊！！！！

(1)能使用CSS替代的元素

对于basefont、big、center、font、s、strike、tt、u这些元素，由于他们的功能都是纯粹为画面展示服务的，而HTML5中提倡把画面展示性功能放在CSS样式表单中统一编辑，所以将这些元素废除了，并使用编辑CSS、添加CSS样式表单的方式进行替代。其中font元素允许由“所见即所得”的编辑器来插入，s元素、strike元素可以由del元素替代，tt元素可以由CSS的font-family属性替代。

(2)不再使用frame框架

对于frameset元素、frame元素与noframes元素，由于frame框架对网页可用性存在负面影响，在HTML5中已不支持frame框架，只支持iframe框架，或者用服务器方创建的由多个页面组成的复合页面的形式，同时将以上这三个元素废除。

(3)只有部分浏览器支持的元素

对于applet。bgsound、blink、marquee等元素，由于只有部分浏览器支持这些元素，特别是bgsound元素以及marquee元素，只被Internet Explorer所支持，所以在HTML5中被废除；其中applet元素可由embed元素或者object元素替代，bgsound元素可由audio元素替代，marquee可以由javascript程序设计的方式所替代。

(4)其他被废除的元素

1）废除rb元素，使用ruby元素替代；

2）废除acronym元素，使用abbr元素替代；

3）废除dir元素，使用ul元素替代；

4）废除isindex元素，使用form元素与input元素相结合的方式替代；

5）废除listing元素，使用pre元素替代；

6）废除xmp元素，使用code元素替代；

7）废除nextid元素，使用GUIDS元素替代；

8）废除plaintext元素，使用"text/plian"MIME类型替代；

**播放视频、音频的HTML5 元素**

视频：

## <video> 卷标的属性

|  |  |  |
| --- | --- | --- |
| **属性** | **值** | **描述** |
| [autoplay](http://www.w3school.com.cn/tags/att_video_autoplay.asp) | autoplay | 如果出现该属性，则视频在就绪后马上播放。 |
| [controls](http://www.w3school.com.cn/tags/att_video_controls.asp) | controls | 如果出现该属性，则向用户显示控件，比如播放按钮。 |
| [height](http://www.w3school.com.cn/tags/att_video_height.asp) | *pixels* | 设置视频播放器的高度。 |
| [loop](http://www.w3school.com.cn/tags/att_video_loop.asp) | loop | 如果出现该属性，则当媒介文件完成播放后再次开始播放。 |
| [preload](http://www.w3school.com.cn/tags/att_video_preload.asp) | preload | 如果出现该属性，则视频在页面加载时进行加载，并预备播放。  如果使用 "autoplay"，则忽略该属性。 |
| [src](http://www.w3school.com.cn/tags/att_video_src.asp) | *url* | 要播放的视频的 URL。 |
| [width](http://www.w3school.com.cn/tags/att_video_width.asp) | *pixels* | 设置视频播放器的宽度。 |

音讯：

## <audio> 卷标的属性

|  |  |  |
| --- | --- | --- |
| **属性** | **值** | **描述** |
| [autoplay](http://www.w3school.com.cn/tags/att_audio_autoplay.asp) | autoplay | 如果出现该属性，则音频在就绪后马上播放。 |
| [controls](http://www.w3school.com.cn/tags/att_audio_controls.asp) | controls | 如果出现该属性，则向用户显示控件，比如播放按钮。 |
| [loop](http://www.w3school.com.cn/tags/att_audio_loop.asp) | loop | 如果出现该属性，则每当音频结束时重新开始播放。 |
| [preload](http://www.w3school.com.cn/tags/att_audio_preload.asp) | preload | 如果出现该属性，则音频在页面加载时进行加载，并预备播放。  如果使用 "autoplay"，则忽略该属性。 |
| [src](http://www.w3school.com.cn/tags/att_audio_src.asp) | *url* | 要播放的音讯的 URL。 |

**3 层迭样式表单CSS**

**将CSS 插入HTML 文档的几种方法：**

1.内部样式

最直接最简单的一种，直接对HTML标签使用style=""，例如：

<p style="color:#F00; "></p>

缺点：HTML页面不纯净，档体积大，不利于爬虫爬取，后期维护不方便。

2.内嵌（联）样式

内嵌样式就是将CSS代码写在<head></head>之间，并且用<style></style>进行声明，例如：

<style type="text/css">

body,div,a,img,p{margin:0; padding:0;}

</style>

优缺点：页面使用公共CSS代码，也是每个页面都要定义的，如果一个网站有很多页面，每个档都会变大，后期维护难度也大，如果档很少，CSS代码也不多，这种样式还是很不错的。

3.链接（外部）样式

链接样式是使用频率最高，最实用的样式，只需要在<head></head>之间加上<link…/>就可以了，如下：

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

优缺点：实现了页面框架代码与表现CSS代码的完全分离，使得前期制作和后期维护都十分方便

4.导入样式（不建议使用）

优先度：

一般而言，所有的样式会根据下面的规则层迭于一个新的虚拟样式表单中，其中数字 4 拥有最高的优先权。

1. 浏览器缺省设置
2. 外部样式表单
3. 内部样式表单（位于 <head> 标签内部）
4. 内联样式（在 HTML 元素内部）

因此，内联样式（在 HTML 元素内部）拥有最高的优先权，这意味着它将优先于以下的样式声明：<head> 卷标中的样式声明，外部样式表单中的样式声明，或者浏览器中的样式声明（缺省值）。

**定义内部样式表单**

如上

**定义内联样式**

如上

**引用外部样式表单等**

如上

**常见CSS 选择符：**

**HTML 元素选择符**

html {color:black;}

h1 {color:blue;}

h2 {color:silver;}

**ID 选择符**

#intro {font-weight:bold;}

**伪类a:hover**

a:link {color: #FF0000} /\* 未访问的连结 \*/

a:visited {color: #00FF00} /\* 已访问的连结 \*/

a:hover {color: #FF00FF} /\* 鼠标移动到链接上 \*/

a:active {color: #0000FF} /\* 选定的连结 \*/

**CSS 选择符定义、样式声明的基本格式**

selector {declaration1; declaration2; ... declarationN }

**CSS 样式表单文件语句注释**

/\*\*/

**字体大小属性**

font-size

**背景颜色样式属性**

background-color

**列表横向排列：**

将li 元素定义为行内元素，即将显示属性设为inline

或设定float 属性

Float方法

如下：

<style type="text/css">

#a{float:left}

#b{float:left}

</style>

<ol>

<li id="a">

dfsfd

</li>

<li id="b">

dfs

</li>

</ol>

**列表项项目符号显示样式属性**

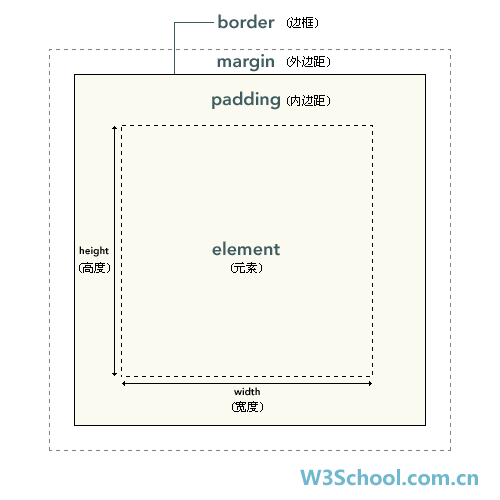
**使用CSS 的好处**

1.结构与内容分离，增强复用性（应该算是本质吧）

2.功能强大，使页面的字体变得更漂亮，更容易编排，使页面真正赏心悦目，轻松地控制页面的布局

3.便于搜索引擎搜索，seo。。。。

**盒子模型**



**使用CSS 层（div）布局网页：如何让层水平居中**

text-align:center

**4 窗体**

**单行文本输入框**

<input type="text"/>

**显示星号的口令输入框**

<input type="password"/>如果非得显示星号，可以用js实现

**下拉列表框**

<select>

<option value="a">膜蛤</option>

<option value="b">苟利国家生死以</option>

<option value="c">岂因祸福避趋之</option>

</select>

**一组单选按钮**

<label><input type="radio" value="test1" />test1</label>

<label><input type="radio" value="test2" />test2</label>

<label> <input type="radio" value="test3" />test3</label>

**复选框**

<input type="checkbox">

**窗体元素的action 属性**

action 属性规定当提交窗体时，向何处发送窗体数据。

<form action="test.asp" method="get">

<input type="submit" value="Submit" />

</form>

**提交按钮**

<input type="submit" value="Submit" />

**POST 与GET 两种提交数据方式的区别**

1.get是从服务器上获取数据，post是向服务器传送数据。

2.get是把参数数据队列加到提交窗体的ACTION属性所指的URL中，值和窗体内各个字段一一对应，在URL中可以看到。post是通过HTTPpost机制，将窗体内各个字段与其内容放置在HTML HEADER内一起传送到ACTION属性所指的URL地址。用户看不到这个过程。

3.服务器端用不同方式处理get与post（如php中get在超全局变量$\_GET中，post在$\_POST中

4.get传送的数据量较小，不能大于2KB。post传送的数据量较大，一般被默认为不受限制。但理论上，IIS4中最大量为80KB，IIS5中为100KB。

5.get安全性非常低，post安全性较高。

**5 JavaScript & DOM**

**script 元素**

HTML中插入一段js

<script type="text/javascript">

document.write("Hello World!")

</script>

**脚本与事件相绑定**

1. 直接在DOM里绑定事件

<div id="btn" onclick="clickone()"></div>

<script>

function clickone(){ alert("hello"); }

</script>

1. 脚本里面绑定

<div id="btn"></div>

<script>

document.getElementById("btn").onclick = function（）{ alert("hello"); }

</script>

3.通过侦听事件处理相应的函数（比较少用）

<div id="btn"></div>

<script>

document.getElementById("btn").addeventlistener("click",clickone,false); function clickone(){ alert("hello"); }

</script>

**单击事件**

Onclick

**浏览器装载网页时所触发的事件**

Onload

**document.write() 方法**

语法：document.write(exp1,exp2,exp3,....)

说明：document.write是JavaScript中对document.open所开启的文档流(document stream操作的API方法，它能够直接在文文件流中写入字符串，一旦文文件流已经关闭，那document.write就会重新利用document.open打开新的文档流并写入，此时原来的文档流会被清空，已渲染好的页面就会被清除，浏览器将重新构建DOM并渲染新的页面。

**document.getElementById() 方法**

语法：oElement = document .getElementById ( sID )

参数：sID――必选项。字符串 (String) 。

返回值：oElemen――对象 (Element) 。

说明：根据指定的 id 属性值得到对象。返回 id 属性值等于 sID 的第一个对象的引用。假如对应的为一组对象，则返回该组对象中的第一个。 如果无符合条件的对象，则返回 null 。

**innerHTML 属性**

innerHTML 属性设置或返回表格行的开始和结束标签之间的 HTML。

**模运算%**

**。。。。。**

**字符串连接运算**

JavaScript是弱类型，直接+吧。。。。其实应该具体情况具体分析的

**6 PHP**

**HTML 里如何嵌入PHP**

<?php

?>

**echo(), print(), print\_r()**

echo是PHP语句, print和print\_r是函数,语句没有返回值,函数可以有返回值(即便没有用)

print只能打印出简单类型变量的值(如int,string)

print\_r可以打印出复杂类型变量的值(如数组,对象)

echo -- 输出一个或者多个字符串

Description

void echo ( string arg1 [, string ...] ) //返回值为空

echo "你好"," 朋友";

print --输出一个字符串

Description

int print ( string arg )//返回值为整形

print "你好朋友";

可以进行下面操作

$name=print "nihao \n";

$str = 'test print value is $name .';

eval("$print=\"$str\";");

echo $print;

print\_r -- 打印关于变量的易于理解的信息。

bool print\_r ( mixed expression [, bool return] ) //返回值是布尔型的,参数是mix类型的,可以是字符串,整形,数组,对象类print\_r() 显示关于一个变量的易于理解的信息。如果给出的是 string、integer 或 float，将打印变量值本身。如果给出的是 array，将会按照一定格式显示键和元素。object 与数组类似。

print\_r() 将把数组的指针移到最后边。

你可以

print\_r(str);

print\_r(int);

print\_r(array);

print\_r(obj);

**获取窗体POST 或GET 方式所提交数据的方法**

$\_POST[]

$\_GET[]

**cookie 与session 的区别**

1、cookie数据存放在客户的浏览器上，session数据放在服务器上。

2、cookie不是很安全，别人可以分析存放在本地的COOKIE并进行COOKIE欺骗

考虑到安全应当使用session。

3、session会在一定时间内保存在服务器上。当访问增多，会比较占用你服务器的性能

考虑到减轻服务器性能方面，应当使用COOKIE。

4、单个cookie保存的资料不能超过4K，很多浏览器都限制一个网站最多保存20个cookie。

**连接mysql 数据库**

mysql\_connect(servername,username,password);

|  |  |
| --- | --- |
| **参数** | **描述** |
| servername | 可选。规定要连接的服务器。默认是 "localhost:3306"。 |
| username | 可选。规定登录所使用的用户名。默认值是拥有服务器进程的用户的名称。 |
| password | 可选。规定登录所用的密码。默认是 ""。 |

**查询表格数据，在网页上显示表格数据**

mysql\_query(“sql语句”)。。。。。。数据库。。。。。。。。。。。

**关闭数据库连接**

mysql\_close()

附：

以访问**bilibili主站为例http报头如下**

**Accept:\*/\***

**Accept-Encoding:gzip, deflate, sdch**

**Accept-Language:zh-CN,zh;q=0.8**

**Connection:keep-alive**

**Cookie:fts=1483267128; sid=ikbg4hch; pgv\_pvi=6076668928; buvid3=D278D79A-C25C-4EED-A5BB-D6EBC036831C2743infoc; LIVE\_BUVID=bbe0b3c75135fe7fe57024008ed0c7b6; LIVE\_BUVID\_\_ckMd5=9a0f15ca57a87087; UM\_distinctid=15ab64c8d23f-0ee117b065be3-57e1b3c-144000-15ab64c8d24d9; rpdid=kxmwkqiqlwdopqllopxxw; finger=7360d3c2; purl\_token=bilibili\_1496988883; pgv\_si=s7473775616**

**Host:api.bilibili.com**

**Referer:http://www.bilibili.com/index.html**

**User-Agent:Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.81 Safari/537.36**

百度首页回应头：

BDPAGETYPE:2

BDQID:0xf5274c6300032051

BDUSERID:2908674435

Cache-Control:private

Connection:keep-alive

Content-Encoding:gzip

Content-Type:text/html;charset=utf-8

Date:Fri, 09 Jun 2017 09:58:47 GMT

Expires:Fri, 09 Jun 2017 09:58:47 GMT

Server:bfe/1.0.8.18

Set-Cookie:BDSVRTM=82; path=/

Set-Cookie:H\_PS\_PSSID=1437\_21092\_17001\_20718; path=/; domain=.baidu.com

Set-Cookie:\_\_bsi=17939640596780274260\_00\_0\_I\_R\_83\_0303\_C02F\_N\_I\_I\_0; expires=Fri, 09-Jun-17 09:58:52 GMT; domain=www.baidu.com; path=/

Set-Cookie:BD\_HOME=1; path=/

Strict-Transport-Security:max-age=172800

Transfer-Encoding:chunked

X-UA-Compatible:IE=Edge,chrome=1

2、块级元素：

div:文档节

section:文档节

nav:导航

header:页眉

article:文章

aside:文章侧栏

footer:页脚

details:元素的细节

summary:details元素可见的标题

dialog:对话框窗口

h1,h2,h3,h4,h5,h6:标题

p:段落

ul:无序列表

ol:有序列表

dir:目录列表

li:项目

dl:列表

dt:列表项目

dd:项目描述

menu:命令的菜单,列表

menuitem:菜单项目

command:命令按钮

form:窗体

fieldset:围绕元素的边框(可用于窗体内元素分组)

legend:在边框上的标题

select:选择列表(内联元素)

optgroup:组合选择列表选项

option:选择列表选项(也可做datalist选项)

datalist:下拉列表(id要与input中list属性值绑定)

table:表格

caption:表格标题

thead:组合表头内容(th)

tbody:组合主体内容(td)

tfoot:组合表注内容(td)

tr:表格行

th:表头单元格

td:表格单元

col:表格列属性;(空标签)

colgroup:表格格式化列组;

iframe:内联框架

figure:媒介内容分组

figcaption:figure标题

map:图像映射

area:图像区域

canvas:图形容器(脚本来绘制图形)

video:视频

source:媒介源

track:文本轨道

audio:声音内容

br:换行(空标签)

hr:水平分割线(空标签)

pre:格式文本

blockquote:块引用

address:文文件联系信息

center:居中文本(不赞成使用)

spacer:在水平和垂直方向插入空间(空元素)

3、行级（内联）元素：

span:内联容器

abbr:缩写

em:强调

strong:粗体强调

mark:突出显示的文本

b:粗体

i:斜体

bdi:文本方向

bdo:文字方向

big:大字体

small:小字体

sup:上标

sub:下标

del:被删除的文本

strike:删除线

s:删除线

ins:被插入的文本

u:底线

nobr:禁止换行

wbr:单词换行时机(空标签)

tt:打字机文本

kbd:键盘文本

time:日期/时间

cite:引用

q:短引用("")

font:字体设定(不常用)

acronym:缩写(html5不支持)

dfn:字段(不常用)

a:锚点

img:图片

embed:内嵌(空标签)

label:input标记(点击文本触发控件)

input:输入框

button:按钮

keygen:生成秘钥(空标签)

textarea:多行文本输入框

output:输出结果

ruby:中文注音

rt:注音

rp:浏览器不支持ruby元素显示的内容

progress:进度条

meter:度量

var:定义变量

code:计算机代码文本

samp:计算机代码样本

select:下拉列表

参考：

<http://www.w3school.com.cn/>（w3school网址）

<http://igoro.com/archive/what-really-happens-when-you-navigate-to-a-url>（What really happens when you navigate to a URL）

<http://www.cnblogs.com/kongxy/p/4615226.html>（从输入URL到浏览器显示页面发生了什么）

<http://blog.csdn.net/longxibendi/article/details/41630389>（浏览器缓存机制详解）

<http://blog.csdn.net/whuslei/article/details/6667471/>（三次握手与四次握手）

<http://blog.csdn.net/wytheonly/article/details/37925067>（DNS解析的工作原理及递归与迭代的区别）

<http://blog.csdn.net/u013063153/article/details/52524132>（HTML块级元素和行级元素）

<http://www.cnblogs.com/fullhouse/archive/2012/01/05/2313811.html>（窗体提交中get和post方式的区别）

<http://blog.csdn.net/lushuaiyin/article/details/7527239>（js 事件绑定的方法）