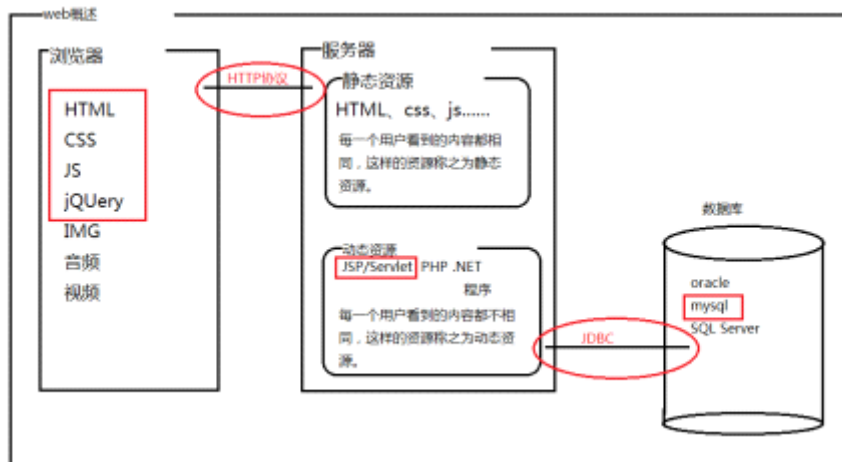


1. web概述图



C/S: client 和 Server 客户端和服务端 LOL QQ 百度云客户端

优点: 画面精美, 不依赖网速 (网络不畅通可能会丢包)。用户体验好。

缺点: 需要下载一个庞大的客户端, 如果客户端需要更新则必须被停机等待, 待服务器更新完毕之后仍然需要下载一个更新包。

B/S: Browser 和 Server 浏览器和服务端 网游 WEBQQ 百度云

优点: 不需要下载庞大的客户端, 不需要停机更新。有些时候停机更新会造成致命的操作。

缺点: 画面不精美, 非常依赖网速, 如果没有网络则无法正常执行工作。

WEB

购物网站 -- 广告 商位

门户网站 -- 天涯 虎扑 百度贴吧 -- 广告 会员

金融网站 -- 广告

游戏网站 -- 5173 u9 -- 广告 会员

大数据

大数据 -- 数据收集 -- 爬虫 -- 埋点

-- 数据处理 -- ETL工程师

-- 数据分析 -- 数据分析

-- 数据展示 -- 数据可视化 DATAV tableau

AI -- 人工智能

文字识别

语音识别

图像识别

视频换脸

仿生

1. HTML是什么？

HTML语言 -- HyperText Markup Language 超文本标记语言

HTML是一门新的语言，与java无关。

HTML是由标签/元素组成。

HTML是文档的一种。

2. HTML基本结构

```
<!DOCTYPE HTML>
```

```
<HTML>
```

```
<HEAD>
```

```
<meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
```

```
<title>我的第一个网页aaa</title>
```

头信息标签：存储优先加载的内容，比如HTML页面的配置信息。

```
</HEAD>
```

```
<BODY>
```

体标签：存储组成页面的标签。

```
</BODY>
```

```
</HTML>
```

页面中的标签自上而下的加载。head中的内容会在body之前加载。

3. HTML语法

a. HTML标签可以写成起始标签和结束标签。

b. HTML标签可以写成自闭标签。

c. HTML注释：

```
<!-- -->
```

d. 在HTML页面中多个回车、制表符、空格都会合并成一个空格来显示。

e. 回车换行标签： --- 自闭标签

```
<br/>
```

f. <hr/>：页面中画一条线

g. 转义字符：

空格

< <

> >

" "

' '

& &

h. 在HTML标签身上可以添加属性：

i. 方式：

属性名 = "属性值"

ii. 在起始标签身上添加一个以 属性名 = "属性值"形式的属性，在一个标签身上

可以添加多个属性。

4. HTML标签

a. 字体标签

font标签中间可以添加一段文本，通过font标签身上的属性可以修改文本的样式-- 颜色和大小

书写方式：

`我是font`

属性：

color	颜色 颜色名称 Hex #AAAAAA rgb(255,255,255)
size	字体大小 1~7 默认大小3

b. 标题标签

i. 在标题标签中的文本都会具有一定字体样式和大小

`<h1 align="center">一级标签</h1>`

`<h2 align="right">二级标签</h2>`

`<h3>三级标签</h3>`

`<h4>四级标签</h4>`

`<h5>五级标签</h5>`

`<h6>六级标签</h6>`

属性：

align	对其方式
left	左对齐
center	居中
right	右对齐
justify	自适应

c. 列表标签

i. 无序列表

``

``

``

``

``

无序列表中有几项就书写几个li标签。

属性：

type	无序图标样式
circle	空心圆
square	实心方块

disc	实心圆
------	-----

ii. 有序列表

```
<ol>
  <li></li>
  <li></li>
  <li></li>
</ol>
```

d. img图片标签

用来显示一张图片。

```

```

属性：

src	图片的路径
alt	在图片不显示的时候提示的文字
width	宽度 px像素 %原图百分比显示
height	高度 px像素 %原图百分比显示

e. 超链接-a-锚标签

i. 超链接 -- 在页面中的链接直接点击就会实现跳转

在文档与文档之间进行跳转 -- 在页面之间跳转

```
<a href="http://www.baidu.com">点击跳转到百度</a>
```

ii. 文档内部跳转

1) 在页面中添加一个书签 -- 就是一个a标签

a) 在页面中就是一个像素点，不会显示任何内容。

b) 点击返回上层

iii. 属性：

href	存储一个url地址
name	a标签的名称
target	打开页面的方式 _self _blank

f. 表格标签-table

table	表格
tr	行
td	单元格
th	标题单元格

o table的重要属性：

border 边框宽度

cellspacing 单元格之间的空白距离

cellpadding 边框与单元格内容之间的距离

bgcolor 背景颜色

bordercolor 边框颜色

width 宽度

align 对齐方式

- tr重要属性:

- align 对齐方式

- bgcolor 背景颜色

- th/td重要属性:

- align 对齐方式

- bgcolor 背景颜色

- width 宽度

- height 高度

- colspan 可横跨的列数

- rowspan 可竖跨的行数

- <caption> 定义表格的标题

g. 表单标签- form

1. 发送数据的方式

a. 手动拼接

之前在利用浏览器访问网站的时候，需要携带参数时，必须要在请求地址后手动拼接上要传递的参数名和参数值信息，这种方式如果传递少量参数还比较可取，如果传递的参数数量众多，参数值复杂，则极易造成拼接错误，对请求参数的拼接要求较高，不好实现。

b. form 表单标签可以用来发送请求参数

用户可以在一个表单中填写自己的数据，在用户填写数据的时候，会输入用户自己所需的数据，输入完成，将用户的数据以请求参数拼接的形式发送到服务器，这个拼接的过程是由浏览器完成的，不需要用户再手动拼接了。正确性得以保证，而且对于用户来说增加便利性。

2. form 标签使用方式：

<form>

</form>

属性：

action:	指定表单提交的url地址
method:	表单提交方式--常用的有两个 get 和post

a. get和post提交方式的不同：

get提交：

地址栏中拼接传递的参数

安全性较低

地址栏中参数的长度有限，限制为1kb -- 和不同的浏览器有关
(如果参数过长，则无法再地址栏中拼接)

post提交：

在底层以流的形式传输

安全性较好

在数据传输的大小上，post在理论上是没有长度的限制的。

提交方式一共有7种：

常用的有两种：get和post

b. <input> 输入框

○ 重要属性：

▪ type属性

文本框 text	输入的文本信息直接显示在框中
密码框 password	输入的文本以圆点或者星号的形式显示
单选框	进行单项的选择 如性别选择 多个radio的name属性相同会被当作一

radio	组来使用 必须用value为选项指定提交的值
复选框 checkbox	进行多项选择，爱好的选择。 多个checkbox具有相同的name属性时会被当作一组来使用 必须用value为选项指定提交的值
隐藏字段 hidden	如果有一些信息，不希望用户看见，又希望表单能够提交，就可以用隐藏字段隐含在表单中
提交按钮 submit	实现表单提交操作的按钮 可以通过value属性指定按钮显示的文字
重置按钮 reset	重置表单到初始状态
按钮 button	普通按钮, 没有任何功能 需要配合JavaScript为按钮指定具体的行为。可以用value属性指定按钮显示的文字。
文件上传项 file	提供选择文件进行上传的功能。
图像 image	利用一张图片替代提交按钮的功能, 不常用

c. name属性：

在form表单中的参数如果需要被传递，则在当前input框中必须要添加一个name属性，这样才能将参数传递，如果不添加，虽然书写但是仍然不会作为参数发送。

d. value属性：

当前input框可以具有一个值，通过value属性可以设置这个值，如果选中当前input框，则设置后的值就会以参数值的形式发送。

▪ readonly属性

使当前输入项变为只读,不能修改,但是提交时仍会被提交

▪ disabled

使当前输入项不可用,不能修改值,也不会被提交

▪ size属性

指定当前输入框的宽度

▪ checked属性

指定单选框/复选框被选中

e. 下拉框select option

<select>

<option></option>

</select>

select属性：

size	下拉框展示选项的数量
multiple	下拉框多选

option属性：

selected="selec	下拉框默认选中
-----------------	---------

ted"	
value="wuhan"	option中传递参数的值如果不设置value，则默认传递option标签中间的内容。

f. 文本域 `textarea`

`<textarea name="test" rows="20" cols="50">我是一个textarea</textarea>`

属性：

rows	行数
cols	列数

1. 页面的发展

- a. 在之前构建页面的时候都是使用table来进行绘制，可以多个table绘制的页面过于简洁，而且维护起来十分繁琐，所以在使用一段之后就被抛弃了，现在已经没有人在开发中使用了。

- b. div+css

2. DIV --- 盒子模型

div标签

div标签会自己独占一行，块级元素。

span标签

span不会自己独占一行，多个元素可以同在一行，行内元素

p标签 -- 段落标签

p标签会自己独占一行，块级元素。

3. CSS层叠样式表

- a. 层叠样式表就是页面中的样式。
- b. 因为在页面中可能会有大量的标签需要样式的修改，而且这些修改的标签可能在样式的修改上是相同的，多次书写相同代码会造成代码的冗余。

4. css引入方式：

- a. 第一种：

直接在标签中引入：

```
<div style="color:red;background:green"></div>
```

- b. 第二种：

在head标签中书写一个style标签：

```
<style>
```

```
</style>
```

- c. 第三种：由于在多个页面都需要引入大量相同的样式，可以将这些样式引入到一个样式文件中，然后再将文件引入。

- i. 在页面的head标签中添加一个link标签，设置内容如下：

```
<link rel="stylesheet" href="css样式文件的路径"/>
```

- d. 第四种：

- i. style标签内部添加内容如下：

```
@import url('6.css');
```

5. css引入顺序：

- a. 样式引入的位置距离标签越近，就越优先生效。
- b. 样式修改选择器书写越具体，越有优先生效。

6. css选择器：

- a. 在页面中可以利用选择器选择中一个或几个元素，进行样式的修改，这种修改方式需要通过css选择器实现。

b. 选择器：

i. 基本选择器

元素名选择器：

```
div{
```

```
}
```

id选择器：

```
#id{
```

```
}
```

类选择器：

```
.class1{
```

```
}
```

ii. 扩展选择器

后代选择器：在div中的全部后代span都会进行样式的修改

```
div span{
```

```
}
```

子元素选择器：在div中的span只有儿子级别样式会修改

```
div>span{
```

```
}
```

相邻兄弟选择器：在div的下一个相邻的兄弟元素样式会修改

```
div+span{
```

```
}
```

属性选择器：

```
div[name]{
```

```
}
```

```
div[name='li']{
```

```
}
```

分组选择器：

```
div,#span1,.class1{
```

```
}
```

伪元素选择器：

其实就在html中预先定义好的一些选择器。称为伪元素。是因为CSS的术语。

:link 未点击的状态

:visited 被点击过的状态

注：在不设置visited状态时active生效，否则会出现visited覆盖active效果

:hover 鼠标移动到元素之上但是仍然未点击的状态

:active 被鼠标点击着的状态