

# 项目咨询日

云服务器的使用 -> 上传自己的代码 -> Git -> 使用 Github.com

## 1.使用云服务器

自己搭建私有服务器：费用昂贵，技术要求高  
租用云服务器：费用划算，维护成本低  
主流云服务器提供商：AWS、阿里云、新浪云

## 2.使用新浪云服务器

### (1)注册新浪微博账号

官网：weibo.com

### (2)使用新浪微博绑定新浪云账号

官网：sae.sina.com.cn



注意：新浪云中的“安全邮箱”和“安全密码”用于管理代码，与微博账号名和密码可以不同。

### (3)登录新浪云开通云应用，获得免费三级域名 官网：sae.sina.com.cn



### (4)管理云应用，上传自己的代码

如果创建的是 PHP 服务器，直接在版本库中

上传代码文件即可；

如果创建的是 Node.js 服务器，需要使用 Git 客户端上传代码；

## (5)客户端访问测试

扩展知识：

一级域名：.com、.cn 无法购买

二级域名：jd.com、tmoooc.cn 可以购买，万网(www.net.cn)

三级域名：mail.jd.com、ftp.jd.com 无需购买

## 3.在新浪云应用中使用数据库

控制台 -> 数据库与缓存服务 -> 共享型 MySQL

MySQL



提示：(1)共享型 MySQL 中每个用户有且只有一个默认数据库，不允许删除或者创建新的库，即禁用 CREATE DATABASE / DROP DATABASE！(2)共享型 MySQL 服务器的主机地址、端口、用户名、密码与上课都不一样，新浪云提供了一套常量！——注意查看手册

Version：版本 —— 每一次大的功能更新

Revision：修订版、版本 —— 每一次的修改提交

## 4.使用 Git 来维护代码——重点&难点

版本管理软件：用于记录软件的开发过程，实现团队成员间的协作。实现代码的每一次提交人、提交时间、提交内容 —— 每一次提交称为一个版本。最终实现代码的存储、共享、合并、追踪、回退等功能你。

常见的版本管理软件：CVS、

SVN(Subversion)、Git

简单的 Git 教程：

<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000>

版本管理软件的基本概念：

**(1)Repository/Repo:** 仓库，一个项目所涉及所有文件以及每个文件的创建修改过程，称为

仓库，即 仓库 = 文件 + 历史

(2)**Commit**: 提交，每一次修改保存入仓库，称为提交

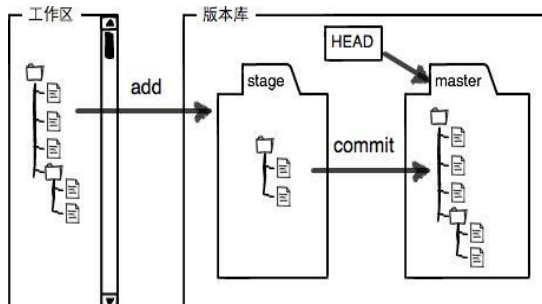
(3)**Workspace**: 工作区，程序员修改代码的地方，就是一个目录

(4)**Stage/Cache**: 缓存区/暂存区/舞台，用于临时存储项目修改的内  
存区域

(5)**Clone**: 克隆，将远程 Git 仓库(文件+历史)复制一份到本地——第一次

(6)**Push**: 推送，将本地仓库(文件+历史)复制到远程仓库

(7)**Pull**: 拉取，将远程 Git 仓库(文件+历史)的最新修改更新到本地



Git 常用命令:

(1)**git config**: 配置 Git 的使用特性

```
git config --global user.name 'LiWenhua'
git config --global user.email
'liwenhua@tedu.cn'
```

(2)**git init**: 把当前目录初始化为一个 Git 仓库——整个项目的第一个仓库，其它同事都克隆此仓库即可

(3)**git status**: 查看当前工作区/暂存区状态，有没有修改/增加/删除

(4)**git add**: 把工作区中的修改添加到暂存区

```
git add 文件名    添加指定文件的修改
git add .         添加所有修改
```

(5)**git commit**: 把暂存区中的修改提交到仓库中，生成一个新的版本

```
git commit -m '提交的原因说明'
```

提示: 若没有提供-m 以及原因说明，会自动进入 vi 编辑器，让用户输入。

(6)**git log**: 查看仓库中的提交日志 (谁+时间+提交了什么)

练习: 修改 index.html 中的内容，查看状态; 添加到暂存区，最后提交到仓库。再次查看状态。

删除 css/index.css 文件，查看状态; 添加到暂存区，最后提交到仓库。再次查看状态。

修改 index.html 中的已有内容，后面再追加两行内容，查看状态; 添加到暂存区，最后提交到仓库。再次查看状态。查看仓库的提交历史。

5.使用 Git 来维护新浪云中的代码——PHP

说明: 每个新浪云中的云应用都对应一个 Git 仓库，可以克隆到本地，修改提交，最后推送本地仓库中的最新内容到远程仓库

(1)克隆云仓库到本地

```
git clone 云仓库的 URL 地址
```

(2)在本地工作区中修改项目文件

(3)从本地工作区添加修改到暂存区

```
git add .
```

(4)从本地暂存区提交到本地仓库中

```
git commit -m "原因说明"
```

(5)将本地仓库中最新历史推送到云仓库

```
git push 云仓库的 URL 地址
```

练习: 把自己的新浪云仓库克隆到另一个目录下，修改 index.html，添加两个 img 元素，同时

在本地仓库添加这两个图片文件，添加/提交到本地仓库，推送到云仓库。

6.使用 Git 来维护新浪云中的代码——Node.js

(1)克隆云仓库克隆到本地

```
git clone 云仓库地址
```

(2)进入仓库目录，编写 Node.js 程序

注意: 项目必需存在 **package.json** 文件，

其中必需有:

```
"scripts":{
  "start": "node xx.js" #没有的话无法启动
}
"dependencies":{
  ....所依赖的第三方模块
  #node_modules 没有都可以
}
```

新浪云启动 Node.js 项目默认使用命令: **npm i && npm start**

此外还需要注意: Node.js 服务器必需监听

5050 端口

(3)把修改添加到暂存区

(4)把暂存区提交到本地仓库

(5)把本地仓库的更新推送到云仓库

## 7.使用 Github

Git：是一款版本管理软件。

Github.com：是一个网站，提供了免费的源代码托管服务，如果想管理被托管的代码，底层使用 git 软件。

使用 Github 的步骤：

(1)注册任一个邮箱，用于接收验证邮件

(2)在 Github.com 注册一个账户

官网：github.com——完成三步输入即可进入邮箱，点击 Github 的注册确认邮件按钮

(3)登录 Github 账户，创建一个应用/仓库

点击 “Start a project” 或者 “new repository” 按钮

(4)在本地克隆 Github 仓库/添加/提交/推送本地的修改

可以点击 “Create new file” 或者

“Upload files” 在线编辑或者上传代码，也可以使用 Git 工具访问该仓库的 URL 地址推送文件。

# HTMLday01

自我介绍

李然 QQ: 1535212067

一.课程安排

0.数据库+服务器(准备数据，准备服务器)

1.HTML5 Basic 2天+(搭建网页结构，可以显示数据，很丑)

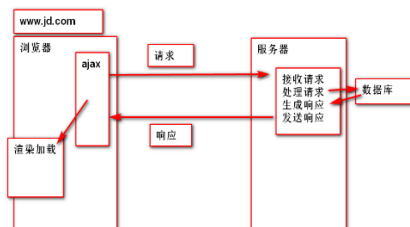
2.Ajax 异步数据交互 3天(页面和服务器的交互，为了得到数据在页面中显示)

3.Ajax 项目 2天

4.CSS3 4+2天 (负责给页面化妆，让页面变得好看)

5.Bootstrap 框架 4天

6.Boot 项目 3天



面临的问题

- 1.知识点细碎，没有条理
- 2.单词量巨大，关键词巨长
- 3.键速
- 4.笔记不要跟抄，

## 二.Web 基础知识

HTML5: HTML4.01 升级版，大前端技术，泛指前端的所有技术

HTML4.01 在 99 年 12 发布

XHTML1.0 在 2000 年 1 月发布，语法更严谨

<img> </img>

## 1.web 与 Internet

Internet: 全球性计算机互联网络

俗称: 互联网，因特网，交换网，际网

www 服务: world wide web (万维网)，访问网站的服务

BBS: 电子公告版，俗称论坛

Email: 电子邮件

telnet: 远程登录

FTP: 文件的上传下载

## 2.internet 上的应用程序

1.C/S client 客户端/ server 服务器

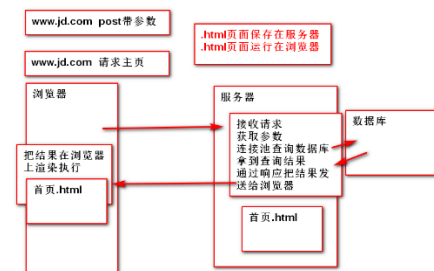
2.B/S Browser 浏览器 / server 服务器

C/S 和 B/S 的区别

C/S 需要升级 B/S 不需要升级

我们工作主要是 B/S

## 3.web 运行原理



web: 运行在 internet 上一种 B/S 结构的应用程序，俗称网站

Internet: 为 web 提供网络环境

web 的工作原理:

基于浏览器和服务器继续通信协议来实现数据的传输和展示

通信协议: 规范了数据和打包和传送的

服务器:

#### 1.功能

存储数据, 接收用户的请求并给出响应、  
提供了程序的运行环境  
具备一定的安全功能

#### 2.服务器产品

tomcat(java)

Apache(php)

IIS(.net)

#### 3.服务器端的技术

PHP/JAVA/.NET/python/nodejs

浏览器

#### 1.功能

代表用户发送请求  
作为 HTML、css 和 js 解析器  
以图形化的解码展示给用户

#### 2.浏览器产品

Chrome

FireFox

IE--edge

Oprea

Safari

#### 3.浏览器支持的技术

H5,CSS,JS

### 三.HTML 快速入门

#### 1.什么是 HTML

HTML:HyperText Markup Language 超文本  
标记语言

##### a 普通文本

<a> </a> 超文本标记

标记: 超文本的组成形式 <关键字>

具有自己的功能

语言: 有语法结构

#### 2.HTML 的特点

1.以.html 或.htm 为后缀

2.由浏览解析执行

3.用带有尖括号的"标记"来标识

4.可以在网页中执行 js 脚本

#### 3.HTML 的基础语法

标记, 标签, 元素, 节点, 对象

主要是表示特殊的功能

标记在使用的时候, 必须使用<>包裹起来

标签的分类

##### 1.双标记(封闭类型)

<关键字> </关键字>

双标记必须成对出现, 有开始就有结束

<p> </p>

##### 2.单标记(非封闭类型)

<关键字> 或者 <关键字/>

#### 练习

创建一个 text 文档, 名称为 first.html

在文档中写出标记

写一对 html 标记

一对 head 标记

一对 title 标记

#### 2.标签的嵌套

##### 1.什么是嵌套

一对标记中出现另一个(对)标记, 形成了层级。

<div>

<p> </p>

</div>

##### 2.语法

<关键字 1>

<关键字 2>

<关键字 3> </关键字 3>

</关键字 2>

</关键字 1>

#### 3.属性和值

允许通过属性和值对标记进行修饰

语法:

1.属性和值必须声明在开始标记中

<关键字 属性="值"> </关键字>

2.多个属性之间用空格分开

<关键字 属性 1="值 1" 属性 2="值 2" ...> </关键字>

##### 3.标准属性(通用属性)

所有的元素都支持的属性

id 定义此元素在页面中的唯一标识

title 鼠标悬停在元素上时, 显示的文本

style css 中, 定义行内样式

class css 中, 引用类选择器

学习 HTML, 就是学习固定关键字, 学习此关键字作用, 学习固定的标签属性, 学习嵌套关系

#### 4.注释

语法: <!--注释内容-->

注意: 注释不能嵌套注释, 注释也不能出现在标记中

## 四.HTML 的文档结构

### 1.HTML 文档的构成

#### 1.文档类型的声明

<!DOCTYPE html>告诉浏览器，我这篇 html 使用 h5 的规则解析

#### 2.网页的结构

<!doctype html>

<html>

<head> </head>

<body> </body>

</html>

<html> </html>表示网页的开头和结束  
在 html 中包含两个标签

<head> </head> 网页的头部，定义全局的信息

<body> </body> 网页的主题，展示的内容

#### 3.head 元素是其他头元素的容器

<head>

<title> </title> 网页的标题

<meta/> 元数据，定义网页的全局信息

<meta charset="utf-8">

chrome 62 以上的版本，自动识别中文

<meta name="Keywords" content="关键词的内容">

<meta name="description" content="描述的内容">

<style> </style>定义网页的内部样式

<script> </script>定义或者引用 js 文件

<link/> 引用外部样式文件

</head>

#### 4.body 元素

定义网页的主体

属性：

bgcolor 背景颜色

text 字体颜色

取值：合法的颜色值->颜色的应为单词

### 练习

在 01\_ex 中，把所有代码删除

1.重新编写一遍，html 的基本结构

2.在 head 中添加网页标题，添加 utf-8

3.在 body 中添加任意文字，改变 body 背景颜色，和字体颜色

red green yellow black white pink blue

## 五.文本标记

### 1.标题元素

作用：在页面中以醒目的方式显示文本

语法：<h1> </h1>一级标题

...

<h6> </h6>六级标题

特点：

1.文本加粗

2.字号发生变化

3.独占一行

4.上下有垂直间距

属性 align 设置标签内容水平方向对齐方式

取值：left/center/right

### 2.段落标记 paragraph

语法：<p> </p>

作用：突出的表示一段文字

特点：独立成行，上下有垂直的间距

属性：align

注意：p 标签里不允许套块级元素

练习：模拟个人简历

### 3.换行标记

<br>或者<br/>

### 4.水平线(分割线)

<hr>或者<hr/>

width="50%" 表示水平线的宽度，取值为%或者 px 为单位的数字

size="5px" 表示水平线的尺寸(粗细)，取值为%或者 px 为单位的数字

align="left" 对齐方式

color="red" 设置水平线的颜色

### 5.预格式化标签

<pre> </pre>

标签内部的文本，保留其格式，在页面上显示  
空格折叠现象，一个或多个空格，会被浏览器渲染成一个空格显示。一个或多个回车，会被浏览器渲染成一个空格显示

### 6.特殊字符(实体)

空格 &nbsp;

< &lt;

> &gt;

© &copy;

® &reg;

¥ &yen;

x &times;

练习：“Copyright © 2004 - 2018 <京东 JD.com> 版权所有”，p 标签中输出上面这句话

## 7.文本样式

粗体	<code>&lt;b&gt;&lt;/b&gt;</code>	<code>&lt;strong&gt;&lt;/strong&gt;</code>
斜体	<code>&lt;i&gt;&lt;/i&gt;</code>	<code>&lt;em&gt;&lt;/em&gt;</code>
下划线	<code>&lt;u&gt;&lt;/u&gt;</code>	
删除线	<code>&lt;s&gt;&lt;/s&gt;</code>	<code>&lt;del&gt;&lt;/del&gt;</code> 语意
上标	<code>&lt;sup&gt;&lt;/sup&gt;</code>	
下标	<code>&lt;sub&gt;&lt;/sub&gt;</code>	

## 8.分区元素

1.块分区，用于页面中布局 <code>&lt;div&gt;&lt;/div&gt;</code> 特点：独立成行 作用：网页布局	2.行分区，处理同一行文本的不同样式 <code>&lt;span&gt;&lt;/span&gt;</code> 特点：可以与其它 span 和文字共用一行 作用：同一行文字设置不同样式时使用
---	--

## 9.行级元素和块级元素（元素的显示形式：块级元素、行内元素、行内块、table）

1.块级元素 在网页中，单独成行，就是块级元素 从上往下排列 常见块级元素 h1~h6 p div	2.行内元素(行级元素) 在页面中，与其他行内元素/行内块共用一行 从左往右排列 页面中常见行内元素 i em b strong u sup sub span	3.行内块元素 显示的方式与行内元素相同，但是具备块级元素的特征	4.table 表格的宽高是由表格的内容决定的 内容多，就宽一些，高一些， 表格中所的数据都预读到内存，一次性画到页面上
---	---	-------------------------------------	---

## 六.图像和链接

### 1.图像

1.使用： <code>&lt;img&gt;</code> 或者 <code>&lt;img/&gt;</code> 属性： <code>src="url"</code> url--图片资源路径
---

### 2.URL

Uniform Resource Locator 统一资源定位符  
简称：路径  
目录结构：web 站点中文件的名称  
web 站点主目录成为根目录

### 3.URL 路径的表现形式

1.绝对路径，一个写完整的路径  
通信协议+主机地址+文件目录结构+文件的具体名称

合适使用：

- (1) 使用网络资源的时候，使用绝对路径  
优点：不占据自己服务器的存储空间  
缺点：不稳定

- (2) 使用本地资源，使用绝对路径(项目中 100%用不到)

从最高盘符开始查找 C/D/E/F/G，项目中不用

2.相对路径（参照物是 html 本身）

- (1) 图片与 html 文件在同一个目录下，src 直接引用图片名称 `src="3.jpg"`

打开 tmooc，按 f12-elements 左边有小箭头  
鼠标去页面中选中元素，右侧会显示对应代码  
鼠标去 src 的图片路径上右键点击，open in new tab，把新页面中的地址栏所有内容复制到自己的 img 的 src 中

### 开发小技巧

- 1.标签只写完整的第一个，按 tab 补全
- 2.属性如果拼写正确，会变色
- 3.属性值可以不添加" 和 "，但是不推荐裸写值
- 4.每次修改要记得保存 ctrl+s
- 5.页面刷新 F5
- 6.添加 eclipse 快捷方式  
tools--首选项--keyboard--最右边 load 加载 1.key  
退回 ctrl+z  
前进 ctrl+y  
删除当前行 ctrl+d  
整行复制 ctrl+alt+↓  
整行位移 alt+↓  
其他快捷方式  
windows+d  
alt+f4



# HTMLday02

## 一.图像和链接

### 1.URL

1.绝对路径

1.1 完整的路径

使用网络图片资源

通信协议+主机名称+文件结构目录+文件名称

http://www.codeboy.com/img/index/banner1.png

1.2 本机资源

从最高盘符开始查找

c:\xxx\xxx\1.jpg

项目中不用

2.相对路径---参照物, 本 html 文件

2.1 在同一个目录下(兄弟关系), 直接引用兄弟文件 

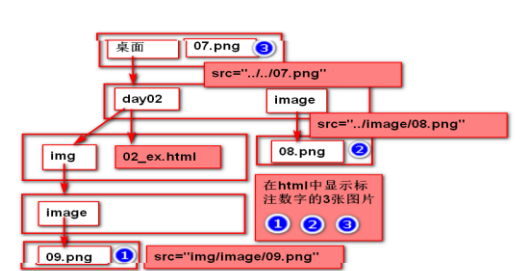
2.2 子集目录, 文件在兄弟文件夹的内部, 先找兄弟, 在找兄弟的孩子



2.3 父级目录, 文件是父级的兄弟, 先找父亲(..),再找父亲的兄弟



练习:



3.属性

src: source 源, 设置资源路径

alt: 图片资源出错, 显示的文字

width: 设置图片的宽度, 以 px 为单位的数字

height: 设置图片的高度, 以 px 为单位的数字

由于宽高比设置的不对, 会发生图片失真的情况

解决: 宽高只设置一个, 另外一个自适应

## 2.链接

(1) 语法 <a href="#"></a>

<a href="http://www.tmooc.cn/">Tmooc</a>

属性:

href:链接的路径

target:指定打开链接的方式

取值: \_self 默认值, 在当前页面标签页红打开新链接

\_blank 在新的标签页打开链接

(2) 其它的表现形式

<!-- 下载 -->

<a href="1.zip">下载</a>

<!-- 新建邮件 -->

<a href="mailto:邮箱">新建邮件</a>

<!-- 执行 js 脚本 -->

<a href="javascript:js 脚本">执行 js</a>

<!-- 返回当前页面的顶部 -->

<a href="#">返回顶部</a>

### (3) 锚点

锚点就是网页中的一个记号	
定义锚点 H4 方式	定义锚点 H5 方式
使用 a 标签的属性 name 定义锚点	任意标签的 id 属性定义锚点
<a name="锚点名称"></a>	<any id="id"></any>
链接到锚点	链接到锚点
使用 a 标签链接锚点, #锚点名称	使用 a 标签链接锚点, #id
<a href="#锚点名称">回到锚点</a>	<a href="#id">回到锚点</a>

## 二.表格

按照一定结构显示数据, 单元格按照从左往右, 从上到下的方式排列表格, 先全部读取到内存中, 再一次性渲染到页面, 表格的大小是由内容决定的, 也可以用于页面的布局(尽量不要使用)

### 1.表格语法

<table>

<tr>

<td></td>

<td></td>

</tr>

</table>

## 2.属性

### 1.table 标签的属性

border:设置边框

width: 设置表格的宽

height:设置表格的高

可以设置宽高

内容小于宽高的, 按宽高尺寸

内容大于宽高, 按内容改变尺寸

align: 设置表格的对齐方式(table 自己的水平位置) 取值: left/center/right

bgcolor:背景颜色

cellpadding:设置单元格内边距(边框到内容的距离, 与 div 不同)

cellspacing:设置单元格外边距(边框与边框之间的距离)

### 2.tr 的属性

(1) align 设置当前行内容的水平对齐方式(设置表格内容的对齐方式)

(2) valign 设置当前行内容的垂直对齐方式

取值: top/middle/bottom

(3) bgcolor 当前行的背景颜色

### 3.td 的属性 (th 加粗居中)

width

height

align

valign

bgcolor

colspan

rowspan

传统的表格, 每一行的列数都是统一的(每一行的列数相同)。一行中所有单元格的高度都是相同的。每行中相同列的宽度是一样的, 默认以最宽的单元格位置

## 3.不规则表格的使用

### 1.跨列

colspan=?

从指定的单元格位置处开始, 横向向右合并?个单元格(包含自己)

被合并掉的单元格, 要删除

### 2.跨行

rowspan=?

从指定的单元格位置处开始, 纵向向下合并?个单元格(包含自己)

被合并掉的单元格, 要删除

## 4.table 中的可选标记

### (1) 表格的标题

<caption> </caption>

如果设置标题, 必须将<caption>放在<table>之后

### (2) 行/列的标题

<th> </th> 代替 <td> </td>

加粗, 居中

## 5.表格的复杂应用

### (1) 行分组

可以将连续几个行, 划分到一个组中, 进行统一管理

表头行 <thead> </thead> 表格最上面一行的标题放在thead中

表主体 <tbody> </tbody> 允许将若干放在tbody中进行统一管理

表尾行 <tfoot> </tfoot> 表格中最后一行放在tfoot中

### (2) 表格的嵌套

一张表格中, 所有的嵌套的表格, 只能放在td中

## 三.列表(重点\*\*\*)

默认按照从上到下的顺序来显示数据, 并且可以在数据前添加标识

使用列表布局(去掉标识, 从上到下, 从左到右排列)

## 1.列表的组成

(1) 有序列表<ol> </ol> order list

(2) 无序列表<ul> </ul> unordered list

列表项 <li> </li> list item

## 2.列表属性

### 1.有序列表的属性 ol

type 设置标识符的类型

取值: 1 默认值, 数字

a A 字母

i I 罗马数字

start 指定起始编号

取值: 无单位数字

### 2.无序列表的属性 ul

type: 设置标识

取值 disc 默认值 实心圆

circle 空心圆

square 实心方块

none 无



### 3.列表的嵌套

所有的嵌套必须只能写在 `<li></li>` 中，不会报错但是不合语义。

## HTMLday03

### 一.列表

#### 1.定义列表

##### 1.作用

做名词解释使用  
长由于图文混排

```
<dl>
  <dt>需要被解释说明的名字</dt>
  <dd>对名词的解释内容</dd>
</dl>
```

#### 二.结构标记

##### 1.结构标签的作用

代替 `div`，用于描述整个网页的结构，提升标记的语义

##### 2.常用的结构标记

<code>&lt;header&gt;&lt;/header&gt;</code>	定义网页或者某个区域的头部内容
<code>&lt;nav&gt;&lt;/nav&gt;</code>	定义网页的导航链接部分
<code>&lt;section&gt;&lt;/section&gt;</code>	定义网页的主体内容
<code>&lt;aside&gt;&lt;/aside&gt;</code>	定义网页的侧边栏
<code>&lt;footer&gt;&lt;/footer&gt;</code>	定义网页底部内容，多用于版权，解释说明，备案号
<code>&lt;article&gt;&lt;/article&gt;</code>	定义与文字相关的内容 论坛，帖子，微博条目，用户评论

### 三.表单(重点\*\*\*\*\*&难点\*\*\*)

#### 1.作用

- (1) 提供可视化的输入控件
- (2) 收集用户填写信息并提交给服务器

#### 2.表单的组成部分

- (1) 前端部分  
提供表单的控件，与用户做交互的可视化控件
- (2) 后端部分  
对提交的数据进行处理(接口)

### 3.表单

```
<form></form>
```

#### 属性

##### (1) action

定义表单被提交时发送的动作  
通常是定义服务器处理数据的地址，接口(url)  
默认，提交给本页地址

##### (2) method

指定表单数据提交的方式(方法)  
**a.get(默认值)**  
明文提交，提交时内容会显示在浏览器地址栏上  
提交的数据有大小限制，2kb  
向服务器要数据的时候使用 **get**  
**b.post**  
提交数据无大小限制  
隐式提交，提交的数据不会显示在地址栏中  
把数据传递给服务器的时候，使用 **post**

**c.delete**  
**d.put**

##### (3) enctype 编码方式

设置表单允许将什么样的数据提交给服务器  
**a.提交普通字符，不能有特殊符号**  
**enctype="text/plain"**  
**b.允许将任意字符提交给服务器**  
**application/x-www-form-urlencoded**  
**c.允许将文件提交给服务器**  
**multipart/form-data**

#### 4.表单控件

能够与用户进行交互的可视化元素。共四类：

**input** 元素  
**textarea** 多行文本域  
下拉选择框  
其它元素

##### (1) input 元素

在页面中提供各种各样的输入控件，通过 **type** 属性来设置输入控件的类型  
`<input>`或者`<input/>`  
**type** 设置输入控件的类型

**name** 为控件定义名称，提供给服务端使用(必须值)

**value** 控件的值，提交给服务端使用

**disabled** 禁用控件，不能操作也不能提交  
无值属性

**type** 取值

a. 文本框和密码框

**type**="text" 默认值

**type**="password"

属性:

① **maxlength** 限制输入的最大字符数

② **readonly** 只读，只能看不能写，但允许被提交，无值属性

③ **placeholder** 占位符，默认显示在控件上文本，只要得到焦点，文本就无效了，此文本不能提交

b. 按钮

**type**="submit" 提交按钮

将表单中的数据，提交给服务器

**type**="reset" 重置按钮

让表单内容恢复到初始化状态

**type**="button" 无功能按钮，可调用 js

c. 单选按钮和复选按钮

① 单选按钮

**type**="radio"

② 多选按钮

**type**="checkbox"

**name** 属性: 除了定义控件名称，还起到分组的效果，必须有 **value**，不然提交的值是 on

**checked**: 设置默认选中项，无值属性

**checkbox** 的 **name** 要写成数组 **hobby[]**

d. 隐藏域

**type**="hidden"

想提交给服务器，但是不想展示给用户看的数据，使用隐藏域

e. 文件选择框

**type**="file"

注意: **method**="post"

**enctype**="multipart/form-data"

**multiple** 无值属性，可以同时上传多个文件

需要 **name** 属性值为数组

(2) **textarea** 文本域

允许录入多行数据的文本框

**<textarea>****</textarea>**

**name** 控件的名称

**readonly** 只读

**cols** 指定文本域的列数，以英文单词为准，中文减半

**rows** 指定文本域的行数，

(3) 下拉选择器框

**<select>** 表示一个下拉选

**<option>****</option>** 下拉选中的一个选项

.....

**</select>**

注意:

当 **option** 没有 **value** 属性时，提交的时候 **select** 的 **value** 为选中 **option** 的内容。

当 **option** 有 **value**，提交时，**select** 的 **value** 为选中 **option** 的 **value**。

**select** 属性

**name** 选项框的名称

**size** 默认为 1，定义显示选项的数量。

如果值大于 1 的数字。下拉选表现为滚动列表

**multiple** 无值属性，可以多选，但是 **name** 要用数组

设置了 **multiple** 之后，就算 **size**="1" 也是滚动列表

**option** 的属性

**value**

**selected** 设置默认选中项

(4) 其它元素

1. **label**

文本与表单控件的关联

**<label>****</label>**

属性:

**for**: 要与 **label** 中文字发生关联的 **form** 控件的 **id**

2. 为控件分组

**<fieldset>****</fieldset>** 为控件分组

**<legend>****</legend>** 分组标题

3. 浮动框架

允许在一个网页中，引入另外一个网页

<code>&lt;iframe&gt;&lt;/iframe&gt;</code> 属性 <code>src="url"</code> <code>width="100%"</code> <code>height="200px"</code> 高度存在问题, 需要使用 <code>js.dom</code> 来解决 <code>scrolling="no"</code> 不显示滚动条 <code>frameborder="0"</code> 不显示边框
--

#### 附加: H5 新表单元素

什么是新表单元素 在 HTML5 版本中新提出来表单控件
<b>1.电子邮件</b> <code>&lt;input type="email"&gt;</code> 表单提交时, 会验证数据是否符合 email 的规范(@以及@后面有内容)
<b>2.搜索类型</b> <code>&lt;input type="search"&gt;</code> 提供快速清除文本框的功能
<b>3.url 类型</b> <code>&lt;input type="url"&gt;</code> 提交时, 验证数据是否符合 url 的规范, 绝对路径的规范 验证 <code>http://</code> 以及后面必须有内容
<b>4.电话号码</b> <code>&lt;input type="tel"&gt;</code> 在移动设备中, 显示拨号键的效果
<b>5.数字类型</b> <code>&lt;input type="number"&gt;</code> 只能接受数字, 并且提供了箭头灵活的调整数字 属性 <code>value</code> 指定数字 <code>min</code> 能接收提交的最小值 <code>max</code> 能接收提交的最大值 <code>step</code> 步数, 每次点击箭头, 变化的数量
<b>6.范围类型</b> <code>&lt;input type="range"&gt;</code> 提供一个滑块组件, 允许用户指定选取的范围 属性 <code>value</code> 指定初始数字 <code>min</code> 范围最小值 <code>max</code> 范围最大值 <code>step</code> 步数

<b>7.颜色拾取器</b> <code>&lt;input type="color"&gt;</code> 提供一个颜色的拾取器
<b>8.日期类型</b> <code>&lt;input type="date"&gt;</code> 提供了一个选择日期的控件, 可以选择年/月/日
<b>9.月份类型</b> <code>&lt;input type="month"&gt;</code> 提供了一个选择月份的控件, 可以选择年/月
<b>10.周类型</b> <code>&lt;input type="week"&gt;</code> 提供了一个选择周的控件, 可以选择年/周