互联网原理

HTML：制作网页。

互联网原理：上网即请求数据。

在浏览器输入一个网址，浏览器会通过http协议向服务器发送请求，服务器响应之后，讲文件传输到浏览器端，过程中有实际的物理文件的传输。

数据：文字、图片、视频、音频等。

1. 服务器：

本身也是一台计算机。

服务器：server。主要用于存储网页文件和数据

数据上传和更改是在本地计算机上完成，通过一些管理软件远程控制服务器。ftp。

1. 浏览器：
   1. 网页请求和接收的客户端。
   2. 浏览器：browser。客户端渲染网页使用的软件。
   3. 浏览器的作用：发起http请求，接收服务器传输的文件。
   4. 任何网站在打开第二次的时候速度会加快，因为这个网站的大部分文件已经存储在临时本地文件中
2. http协议：超文本传输协议
   1. http包含请求和响应两部分。
   2. 请求：request，浏览器根据网址向对应的服务器发送请求。
   3. 响应：response，服务器根据请求响应文件，将页面传输给客户端，在浏览器中进行网页的渲染。
   4. 浏览一个网页的时候，http是多个请求同时发送

Html基础概念：

<h1></h1>一级标题

<h2></h2>二级标题

<p></p>段落

真正控制文本样式的是css

HTML骨架

1. html：最根本的标签，表示整个的网页
2. head：网页的头部，里面都是一些网页的配置，除了title标签，其他的涅日荣在浏览器上都是不可见的。
3. title：网页的标题
4. body：用户能够看到的网页区域

DTD：doctype definition。文档类型定义。定义我们这个html文件使用的是哪一个版本的html规范

html、css、js规范的维护和制定的组织W3C。

中文网站：

HTML1.0-HTML5。

HTML4.0版本，将结构和样式进行了分离。常用的HTML4.01版本

XHTML1.0在4.01版本基础进行严格化，和扩展升级，严格标签必须小写，属性必须用双引号包裹，借宿标签的关闭符号/必须写。

strict 不可以用font等废弃标签，不可使用框架集

transitional，可以使用font等废弃标签，不可使用框架集

frameset 可以使用框架集

transitional：

学习过程中最常用的是XHTML1.0 transitional

html:xt 使用的是XHTML1.0 transitional

html:xs 使用的是XHTML1.0 strict

发展到HTML5版本时，进行了大幅度更新升级，取消了3个小版本

HTML5的DTD进行了极大的简化：<!DOCTYPE html>

html标签是一个双标签。

双标签：有开始标签和结束标签，必须是成对出现的。

html标签有两个属性

xmlns规定了我们的标签使用的是哪种命名规范。是xml 的 namespace的缩写，用户浏览器不一致，必须规范什么是标签，什么是内容。标签具有什么语义。

lang：规定我们html文件标签都必须使用英文。

head标签也是一对双标签，里面是对我们网页进行的一个配置。

meta标签：单标签

我们常用的字符集：

国际标准字库：utf-8，包括世界上所有人类的语言，中文占三个字节。

国家标准字库：gb2312/gbk，包括所有的简体字，大部分繁体，一些特殊符号，片假名，中文占两个字符

这两种字库的使用情况：

1. 网页中有大量的中文，同时要求加载速度，使用gbk字符集
2. 制作外文网站，有少数民族语言的网站，或不要求家在速度，使用utf-8

注意：规定的字符集必须与编辑软件的保存字符集类型一致，否则会乱码。

title标签：是一个双标签，内部书写的是网页的标题

搜索引擎优化：SEO。提升我们搜索排名。

title里的文字会作为关键首先被搜索引擎抓取。

html的语法：

1. html对换行与空格不敏感。对标签之间的嵌套敏感
   1. 为了保证我们代码的可读性强，需要对我们的标签进行合理的缩进
2. 空白折叠现象
   1. 在普通文本中，多个空格，缩进，换行，在浏览器中只显示一个空格（即使文本内部区域）
3. html标签
   1. 标签必须写在一对尖括号里<>
   2. 双标签必须成对出现
      1. 如果书写不完整，会影响后面的标签。
   3. 结束标签必须写关闭符号/。
   4. 标签分类：
      1. 容器级：内部还可以放置嵌套任意类型的标签，甚至是容器级标签
      2. 文本级：只能放文字、图片、表单元素，废弃的文字修饰标签等。不能放容器级标签。
4. 标签属性
   1. 每一个标签都有自己特定的属性。
   2. 属性都有属性值，属性名k(key)，属性值v(value)，键值对写法
      1. 属性书写在标签的起始标签内部，与标签名之间用空格隔开，属性与属性之间也需要用空格隔开
      2. 键值对书写格式：k=”v”。中间不能有空格。
      3. 属性值必须是用双引号包裹，XHMTL严格要求使用双引号。
      4. 标签属性值可以有一个或多个，多个属性值之间用空格隔开，全部必须在引号内部。
5. html标签
   1. headline，标题的缩写，容器级标签
   2. h系列
   3. h系列标签分为了六个的等级，表示一到六级标题。
   4. 约定俗成，一般一个页面里面h1标签的权重是最高的，最便于搜索引擎搜索的，一个页面只写一个h1标签，一般将这个标签给logo
   5. 没有嵌套关系
6. P标签
   1. Paragraph，段落的意思，文本级标签。给我们文本添加一个段落的语义。
   2. 注意：一个p标签只能作用于一个段落，多个段落要用多个标签分别标记
7. img标签
   1. image，图片的意思。表示引入一张图片，本身相当于一个特殊的文本，
   2. img标签是一个单标签。
   3. img标签属性：
      1. src：source，资源，表示图片的来源、路径。必须写的属性
      2. width:宽度。单位是px。可以省略单位。单独设置宽度，高度会等比例变化
      3. height:高度。单位是px。可以省略单位。单独设置宽度，宽度会等比例变化
      4. 同时设置高宽，任意变形
      5. title：设置提示文本
      6. alt：设置图像没有找到时候的替换文本。
8. 路径：
   1. 路径就是我们查找相关文件的位置的方法。
   2. 查找方式有两种：相对路径、绝对路径。
      1. 相对路径：查找文件时，从html文件本身出发，找文件的位置
         1. 同级查找：直接书写文件名字（包括文件名和拓展名）
         2. 子级查找：进入文件查找过程。书写对应的文件夹的名字，加/表示进入文件夹，然后直接书写指定的文件名。
         3. 上级查找：../书写文件名字。
         4. 不能跨盘进行查找
      2. 绝对路径
         1. 从盘符出发寻找目标文件，以http://开头也是绝对路径

http:协议: hypertext transfer protocol，超文本协议

快捷键：

Html:xt🡪tab/ctrl+E 生成基本骨架

标签名🡪tab 快速生成标签

ctrl+滚轮上下滚动 调整文字大小

按住滚轮拖动 编辑多行

ctrl+shift+k/ctrl+x 删除当前行

ctrl+shift+d 复制当前行

a标签

anchor, 锚的意思。超级连接，可以跳转到相应的网站

1. 是一个双标签
2. A的属性：
   1. href:超文本引用。属性值就是我们要跳转的网站
   2. target：目标的意思。设置我们的跳转后的网站是否要在新窗口打开，target=“\_blank”.
   3. Title:设置跳转显示文本
3. A的页面内锚点跳转（通过id，href=”#idKey”）
   1. 给对应位置的标签添加一个id属性，属性值就是id名。
   2. 在需要添加锚点的位置，加入一个a标签，不用书写href属性，添加一个name，命名与id一样
   3. 将href属性值书写为#id名，#name名
   4. Id名和name的名字在网页里必须是唯一的

列表：

1. 将一些内容或者样式类似相近，相关的内容一起书写。
2. 有三种列表：无序列表、有序列表、定义列表。

无序列表：

1. 无序列表作用：给我们添加一个无序列表的语义
   1. <ul><li></li></ul>
2. 无序列表是一组标签，ul和li这两个标签必须成对出现，
   1. ul：unordered list，无序列表。
   2. li：list item， 列表项
3. ul和li之间是嵌套的关系，ul里可以嵌套一个或多个li标签，
   1. 快捷键：ul>li\*5.
4. li标签是一个经典的容器级标签，内部可以防止任何其他的标签，甚至还可以放ul>li,
5. 无序列表并不会帮我们添加小圆点等样式
6. 无序列表里的项并没有先后顺序之分

有序列表

1. 与无序列表相似
2. 有序列表的作用：添加有序列表的语义，搭建列表结构。
3. 也是一组标签，ol和li。
   1. ol：ordered list，有序的列表。
   2. li：list item。 列表项。
4. ol和li必须同时出现，之间是嵌套关系ol>li
5. ol只能嵌套li标签，li标签不能脱离ol单独出现
6. li是一个容器及标签，里面可以放其他标签，甚至是ol、ul。
7. ol默认有一个数字前缀，内置的一些预设样式，并不是ol的功能，只能同css样式调整。

定义列表：

1. 有三个标签参与的，dl，dt，dd。
2. 有自己定义的主题，定义的解释。
3. 定义列表的作用：给我们添加一个列表的结构，里面有定义主题，对主题的解释。
4. 标签解释：
   1. dl：definition list。定义列表。
   2. dt：defination title。定义标题。
   3. dd：definition description。定义说明、解释。解释的是前面最近的这个dt。
5. 这一组标签需要同时出现，之间有一个嵌套关系。Dl里面嵌套dt和dd
   1. Dl>dt+dd.
6. 每一个dt后面可以跟一个或者多个dd，表示对主题的多层说明
7. dt后面也可以不跟dd，只有主题，没有解释。
8. dt和dd都是容器级标签。
9. 列表的使用：根据实际情况，根据结构语义来选择使用哪种列表

div和span标签

1. div和span的认识：
   1. 通常称为“盒子“。div是大的范围，span是小的跨度。
   2. div：division。范围、区域、分割。并没有什么特殊的语义。经常用来布局。
   3. div里经常放置一些具有某些特殊功能、相似类型的标签。这就是布局的过程。
   4. div是一个典型容器级标签，可以放置任何的标签。
   5. span：小区域、小跨度。常用于小范围调整布局。
   6. span是一个文本级标签，不能盛放容器级标签。
   7. span在p标签内：一般认为p>span>a。
2. 简单的div+css布局，最开始使用表格布局，结构和样式不分离。
   1. div+css：结构和样式分离

表格：

1. 表格基础
   1. table、tr、td。
   2. 表格的标签之间是嵌套的关系。table>tr>td。
   3. table：表格。
   4. tr：table rows。行
   5. td：table dock。单元格
   6. 如果我们需要添加表头的语义，th标签表示表头单元格
2. 合并单元格
   1. 单元格标签：td、th
   2. 有两个属性用来合并单元格：
   3. rowspan：合并行单元格。
   4. colspan：合并列方向单元格。
   5. 属性值：是一些数字，是几表示合并几个单元格。
   6. 方法：先将所有tr写完，对每一个tr内部的td单独书写
3. 表格分区
   1. 标题：caption。
   2. 表头：thead。
   3. 主体：tbody。

表单：

表单就是网页上可以提供用户输入和选择的一些控件。

1. form
   1. 我们所有的表单元素需要卸载form标签内部，不是一个结构性标签，而是一个功能性的标签。
   2. 功能：规定我们提交的数据发送到哪里，发送的方法是哪种。
   3. method：方法，提交数据的方法，属性值get，post。
   4. action：提交的位置。
2. 文本框
   1. 允许用户输入文本。
   2. input：输入。
   3. 有一个属性：type。根据属性值不同，input标签代表的是不同的表单元素
   4. 表单元素：类似一些特殊文本。
   5. 文本框input标签type属性值为text。value属性值是没输入之前的默认的文本
3. 密码框
   1. type属性值：password。
4. 按钮
   1. 有三种：普通按钮，提交按钮，重置按钮
   2. 普通：button
   3. 提交：submit
   4. 重置：reset
5. 单选框
   1. type：radio。收音机的意思。同一组单选框只能选择一个
   2. 如果不设置name属性值，不知道哪些单选框是一组，不能做到单选的效果，name属性值必须一致
6. 多选框
   1. type：checkbox。可以在同一组内进行多选。
7. 文本域
   1. 标签：textarea。可以输入多行文本。两个属性值，规定显示多少行多少列文字
   2. rows：属性值是数字，数字表示有多少行
   3. cols：属性值是数字，表示每一行有多少个字节。
8. 下拉菜单
   1. 是一组标签，必须同时出现，有嵌套关系
   2. select：选择。option：选项。
      1. select>option.
   3. 实际工作中，一些下拉菜单都是用无序列表写的

一般喜欢给单选框和多选框绑定文字：就是可以点击相关内容，也选择选框。提升用户体验。

方法：将文字内容卸载一个label标签内部，让lable标签的for属性值和input标签的id属性值一致。

html杂项：

1. 注释：里面的内容是给程序员看的。在浏览器上不会渲染
2. 作用：清晰标注，有一部分代码不想显示，希望保留。
3. 书写：<!—注释内容-->
   1. Sublime ：ctrl+/
4. 字符实体：
   1. 我们的一些特殊符号，并不能直接在文本内容里书写。
   2. Html提供了一套书写特殊符号的替代符号，就是字符实体，或转义字符

<小于号：&lt; less than

>大于号：&gt; greater than

无换行空格：&nbsp; non-breaking space

版权©：&copy; copy right

Css概念：

1. css:cascading style sheets 层叠式样式表。规定了html标签在网页上的显示样式。
   1. html4的时候，w3c组织将html的结构和样式做了分离
   2. 前端三层：
      1. Html 结构层，打架网页的整体结构。
      2. Css 样式层，装饰页面。
      3. Javascript 行为层，一些页面交互效果。
   3. 这三层标准都是w3c组织来维护。
   4. Css发展到现在，最新版本是css3，最常用的是css2.1
2. Css作用
   1. Css两个重要的概念：层叠式，样式。
   2. 样式：html标签在页面的显示效果。
   3. 某一个标签又什么样式，直接将对应的属性及属性值罗列出来。
   4. Css样式设置的时候，有两个关键：选择器，样式表。
   5. Css作用细化：
      1. 给文本添加文字显示样式
      2. 给盒子添加属性进行结构布局
3. Css几个小属性
   1. 文本样式
      1. 字体、颜色、大小
         1. 大小：font-size。本身是一个符合属性font；里面的单一属性，需要用font单一属性名。属性值，以像素为单位。
         2. 颜色：color。属性值有几种选择方案：十六进制，RGB，rgba，英文颜色名。
         3. 字体：font-family。复合属性的一个单一属性。中文字体和英文字体。
         4. 书写的时候将英文字体写在前面，中文字体在后面。
         5. 英文字体常用：Arial，consolas。
         6. 中文字体：默认字体为宋体，常用字体有宋体、微软雅黑，一般写字体时，标注好备选字体。
         7. 字体用引号包裹，字体之间用逗号给开，表示或。
         8. 为了追求加载速度，将一些中文字体名字写成英文表示法。
            1. 微软雅黑：Microsoft Yahei
            2. 宋体：SimSun
         9. 盒子实体化基本属性
         10. 实体化：将盒子给宽、高、背景色、边框。
         11. 宽度：width。属性值是像素为单位。
         12. 高度：height。属性值也是像素为单位。
         13. 背景色：background-color。属性值就是颜色色值。
         14. 边框：border。复合属性，有多个属性值，属性值之间用空格隔开。边框的宽度、颜色、线的类型。
         15. 实现边框：solid。
4. Css的书写位置
   1. 根据书写的css属性位置不同，将我们的样式表分为三种：行内式，内嵌式，外链式。
   2. 行内式：
      1. Css属性写在标签的一个style属性里，可以书写他的所有css样式。
   3. 内嵌式：
      1. 写在html文件里的，head标签内有一对style标签，是一个双标签，我们所有的css样式属性都写在双标签内部。
      2. Style标签有个属性type，属性值“text/css”，表示我们内部写的是纯文本样式代码。
5. Css的注意事项：
   1. Html的属性和属性值，键值对：k=‘v’
   2. Css的属性和属性值，键值对：k：v；
   3. 多个属性之间必须用分号分隔。不写出错。
   4. 内嵌式。所有的样式必须写在一对大括号里。
   5. Css样式，对于空格，换行，缩进不敏感
   6. 合理换行，提高可读性。
   7. 为了提高加载速度，需要将所有的空格，缩进，换行删除，即压缩代码。

基础选择器

1. 选择器：指的是我们选择需要添加样式的标签元素的模式
2. 标签选择器：通过标签名直接选择相应的标签。标签是什么我们就用什么选择。
3. 用途，默认样式的清除
4. Id选择器
   1. 通过标签的id属性值来选择相应的标签。
   2. 选择器书写：#开头，后面紧跟id名。中间没有任何空格。
   3. 由于我们的id页面唯一，id选择器只能选中一个元素。
   4. 标签都有id属性，属性值也就是id名，命名需要遵循一个规则：
      1. 必须以字母开头（严格区分大小写，A和a表示的不一样），后面可以是数字、下划线或者横线。
      2. 一个页面不允许出现两个相同的id名，即使是不同类型的标签。
5. Id选择器劣势：只能选中一个元素。如果有相同的样式，只能分开命id名，多次书写相同属性。
6. 标签选择器劣势：会选中所有的元素，不能进行单独或者部分控制。
7. Id常用场景：并不是用来添加样式，留给js添加行为。

类选择器

1. 通过标签的clss属性来选择这个标签。
2. 选择器写法：开头，后面class属性值。中间不能有空格。
3. Class属性，值可以不唯一。类选择器选中的是相同的class 名的元素。
4. 标签的class属性，属性值命名规则与id一样。
5. 优点：可以选择一部分标签，添加相同的样式。
6. 根据类的特性，我们要习惯使用原子类。原子类设置的原则，里面只设置单一的css属性，某一个标签需要添加这个属性，可以直接添加这个原子类的类名
7. 一个标签的class可以有多个属性值，有多个类名。多个类名之间要用空格隔开
8. 类选择器好用，多用于css样式书写。
9. 类上样式，id上行为。

通配符\*

可以选中包括body在内的所有标签，现在不常用，效率不高。

高级选择器

1. 基础选择器并不能满足我们所有的需求，在基础选择器上进行了一些延申。
2. 三种：后代选择器、交集选择器、并集选择器。
   1. 后代选择器：又叫做包含选择器。通过标签之间的嵌套关系、层级关系，限定我们大体范围，在范围内具体查找相关元素。
      1. 后代选择器：根据嵌套关系，空格左侧是右侧的祖先元素。
      2. 后代选择器空格两边只要是后代关系都可以，不一定是父子关系
      3. 一般的开始的祖先元素都不使用标签选择器。
      4. 可以根据嵌套关系来确定最终元素
   2. 交集选择器：
      1. 既满足条件1也满足条件2
      2. 两种或者以上选择器同时存在一个标签上
      3. 写法：选择器之间直接连接，没有任何符号。
      4. 交集使用的基础选择器，可以都是类选择器，可以标签和类的混写。
      5. 交集的连续书写。
      6. 兼容问题：IE6不支持类的连续交集。
   3. 并集选择器
      1. 有很多元素可能都有相同css样式，我们可以通过并集选择器将这些标签写在一起，统一设置css样式，前面所有的选择器都可以作为并集里的一项，在选择器中间用逗号隔开。

继承性和层叠性

1. 继承性
   1. 有一些属性给祖先元素设置了，后代元素会继承
   2. Css继承性：会继承文字的样式，不继承盒子的样式
2. 层叠性：
   1. 我们学了多种选择器，同一个标签可以用多个选择器选中。
   2. 问题：多个选择器选中一个元素，相同的样式听？
   3. 涉及到一个选择器权重问题：
      1. 标签、类、id权重逐渐增高
      2. 通配符权重最低
   4. 到底一个标签的样式听谁的：
      1. 选择器都选中了元素。
         1. 首先比较权重，权重大的层叠权重小的
         2. 计算权重的方法：数基础选择器的个数，依次比较id个数🡪类的个数🡪标签的个数。
         3. 如果权重一样，比较是书写顺序，谁写在后面，听谁的
      2. 如果没有选中元素，样式靠继承。
         1. 第一种：祖先元素距离目标元素距离不同，继承的样式，谁近，听谁的
         2. 第二种情况：如果距离相同，去比较权重。
         3. 第三种情况：如果距离相同，权重相同，谁在后，听谁的
3. 关键字 ！important
   1. 给单一属性添加，会给这个属性的权重提升到最大。前提：不适用于就近原则。

Css属性

1. 主要分类：文本、盒模型、背景、浮动、定位
2. 文本
   1. Color：字体颜色
      1. 色值：十六进制、RGB、rgba、颜色名。
      2. 颜色使用：背景色、边框色等。
      3. Font-family:字体
         1. 中文：微软雅黑、宋体
         2. 英文：Arial、Consola
      4. Font-size：字号
         1. 文字大小。自己设置一个通用字体大小，如果不设置，网页会有一个默认的字体大小。
         2. 谷歌浏览器：最小是显示12px。如果文字设置小于12px，则会自动显示12px；
      5. Line-height 行高
         1. 一行文字实际占有的高度，单位：像素或者百分比
         2. 文字在自己的行高内，垂直居中
      6. Font-weight：字体加粗
         1. 属性值：数字、单词
         2. Font复合属性，至少写三个，字体大小，行高，字体
      7. Font-style：字体样式
         1. 设置字体倾斜样式或者正常样式。
         2. 取值
   2. 文本
      1. Text-align：对齐
         1. 对齐方式：水平左对齐、水平右对齐、水平居中
         2. 对应的属性值：left、right、center
         3. 默认文本都是左对齐：left；
      2. Text-indent：文本缩进
         1. 设置的是文本的首行缩进。单位：em，px，百分比
         2. 如果是百分比，依据的是父盒子宽度的百分之多少。
      3. Text-decoration：文本修饰
         1. 对于文本整体样式的一个修饰效果
            1. None、overline、line-through、underline；

盒模型

1. 盒模型，也叫做框模型
2. 一个盒子他的内容包含了五部分：宽度、高度、内边距、边框、外边距