2020/8/16 第一节

（一）

权限修饰符 静态类型 返回值类型(基本/引用) 方法名 参数列表0/N

Public [static] Boolean zengjia （）

private

protected

default

{方法：具体实现

Return true;

}

Viod是不需要返回的

Web/IE request serxlet dao()方法

浏览器 容器

（二）

数组 不可改变

List 弹性 可增减 查询费时

类名 User 首字母大写

（三）固定长度 有序 连续 相同数据类型 可用For循环

数据类型[ ] 数组名= new 数据类型 [ 默认长度];

数据类型[ ] 数组名= { 元素1，元素2，………};

数据类型[ ] 数组名= new 数据类型 [ ]{ 元素1，元素2，…… };

2020/9/6第四节

2020/9/13第五节

2020/10/11第六节

Collection下的两个接口

List接口 Set接口

List实现类 ArryList LinkedList Vector（可以重复，有序排列）

Set实现类 HashSet TreeSet （存放无序，不可重复）

运用多态形式

Arrylist 查询效率高，删除和新建效率低 占连续内存

List list = new ArryList­（）；

添加元素

List.add(1);

List.add(“test”);

List.add(new Date());

List.add(true);

List.(null);

元素个数size();

包含

List.contains(“test”);

删除 下行上移

List.remove(1);

判断是否为空

list.isEmpty();

清空

list.clear()；

返回元素位置 没有找到返回-1

List.indexof(“true”)

上处细节 上面是布尔 此处是字符串

增强for循环

// for循环第一种常规方式

for (int i = 0; i < listBook.size(); i++) {

System.out.println(listBook.get(i));

}

// for循环第二种加强方式

For （数据类型（泛型） 临时变量：循环对象）{}

多于在显示数据

for (Book b : listBook) {

System.out.println(b);

}

Linkedlist:多首尾操作， 不占连续内存

listBook.add此时发现没有 是子类独有

需要向下造型

LinkedList <Book>ll = (LinkedList)listBook;

Set:存放无序 不可重复

只能用加强版的for循环（因为没序号）

重复标准 ：比较虚地址 和 String中的eq比较内容，一句总结 根据eq内容比较

Treeset可以实现有序，hashset是无序的

Map：也是个接口 是以键值对key和valua来储存

HashMap， TreeMap HashTable

**2020/10/18 第7课 IO流**

1，定义文件对象

二，递归

三，字节流 input/output

四,字符流 可以读一串

**2020/10/25第8节 多线程**

1. 进程：一个执行的应用程序
2. 线程：程序的执行流，内部细节同时运行
3. 时间轮转片，3心2意

Thread类

常见实现方式

1. 继承thread类
2. 1

//Thread.currentThread().getName() 当前线程的名字

4．线程的休眠sleep

.equalsIgnoreCase是忽略大小写的完全匹配

**2020/11/1 管第一课**

**HTML：**

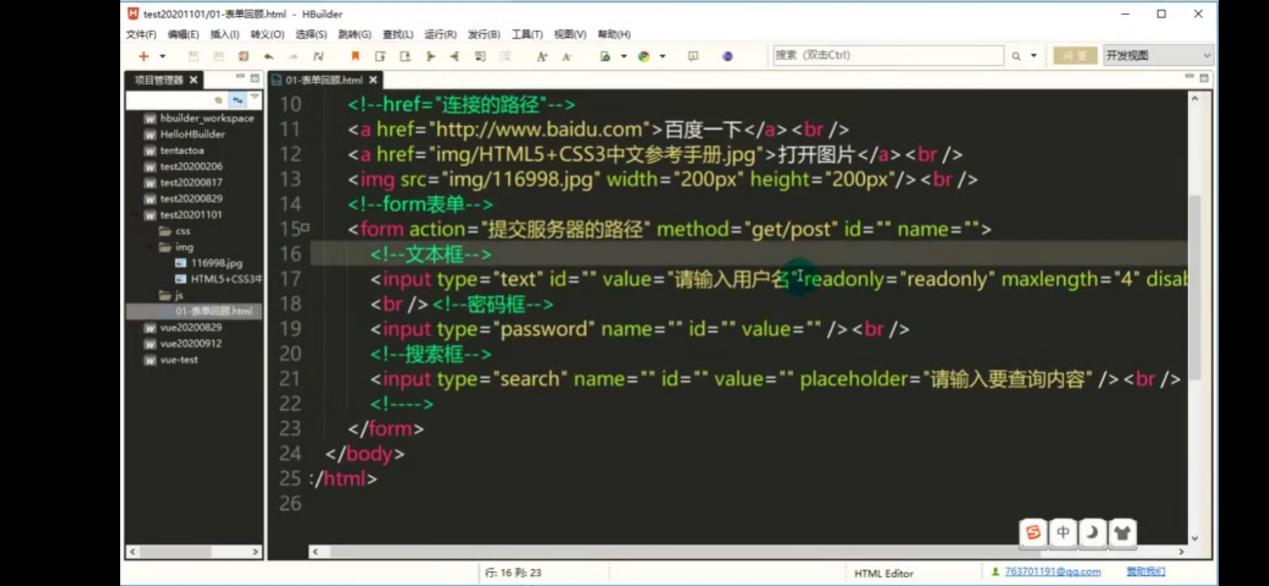
1. **a标签超链接（网址，图片）**

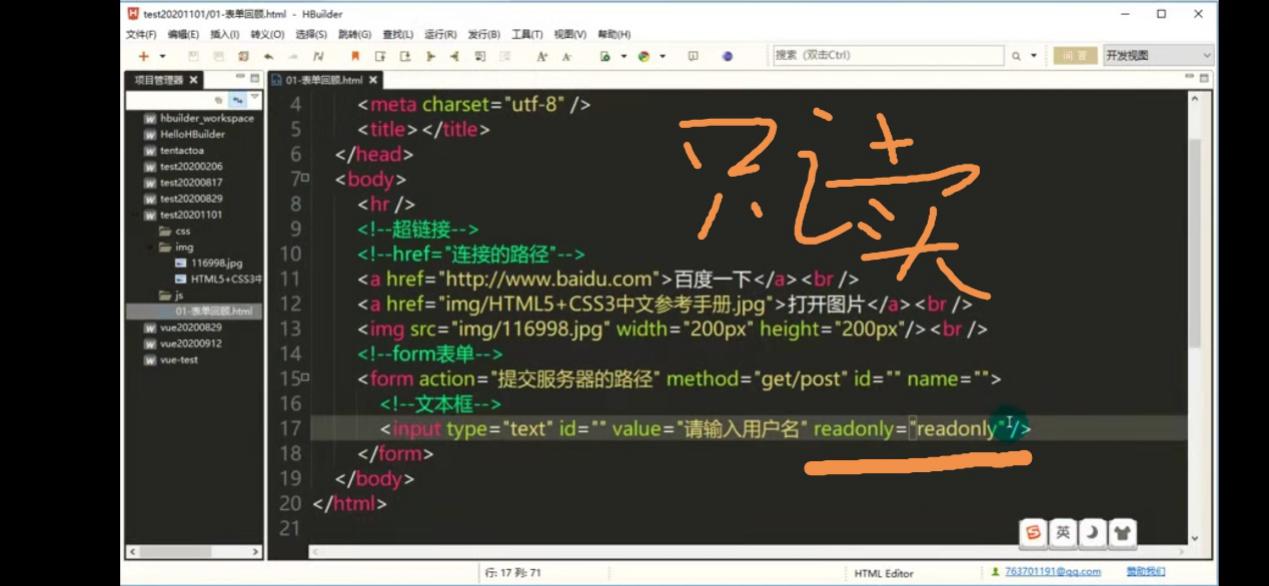
**img图片标签**

**form表单（文本框，密码框，搜索框）**

**只读readonly的设置**

**失效" disabled="disabled"的设置**



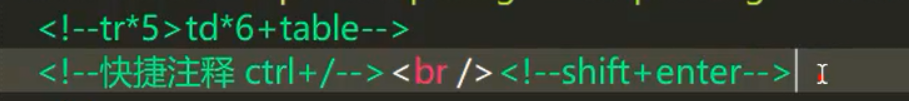


**2. 快捷键：**

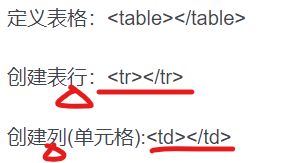
**快捷注释ctrl+/**

**换行 shift+enter**

**快捷生成多个行和列 tr\*5或者td\*6+table**



**3.table表格**

****



****

1. **无序 <ul >**

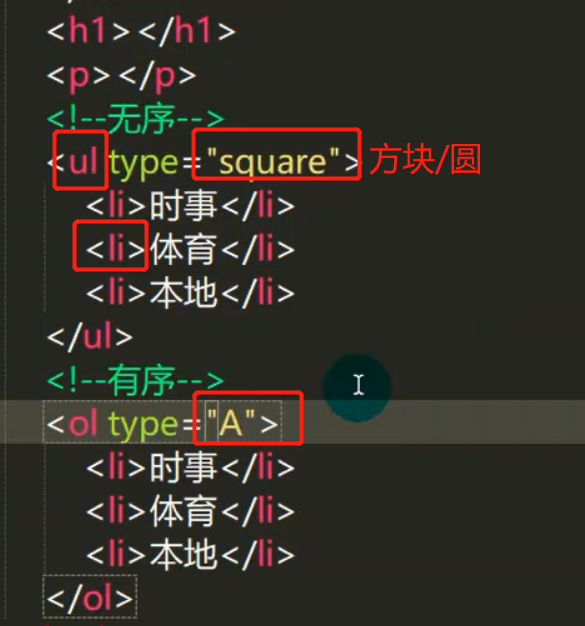
**<li></li>**

**</ul>**

**有序 <ol>**

**<li></li>**

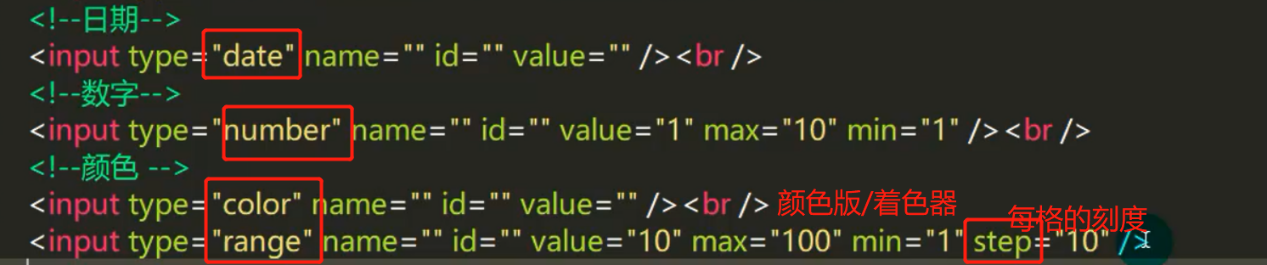
**</ol>**



**5.input标签type**

**（日期，数字，颜色，rang）**

Range对象代表页面上一段连续的区域，通过Range对象可以获取或者修改页面上任何区域的内容。也可以通过Range的方法进行复制和移动页面任何区域的元素。

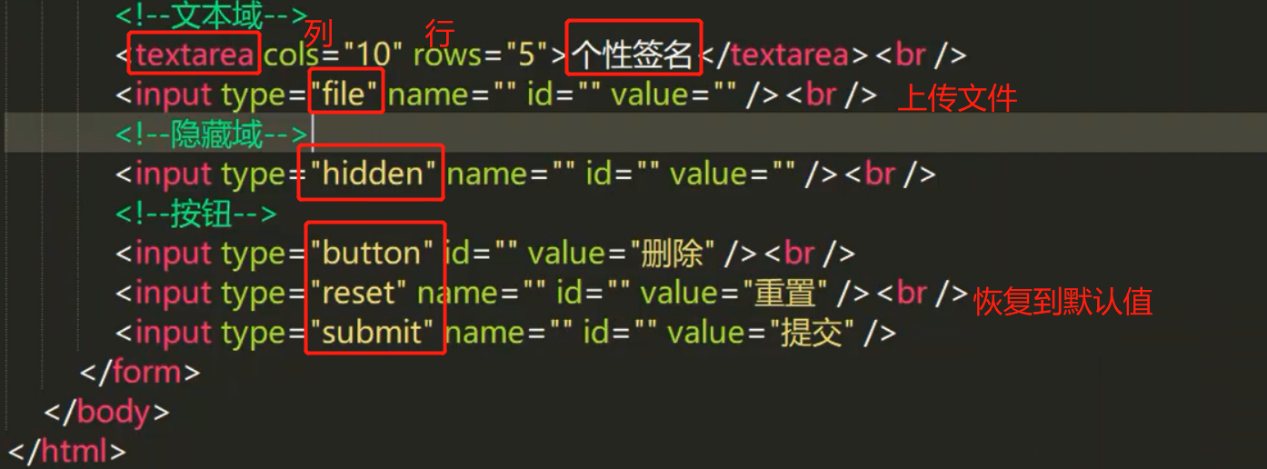


**Date和number的显示图**



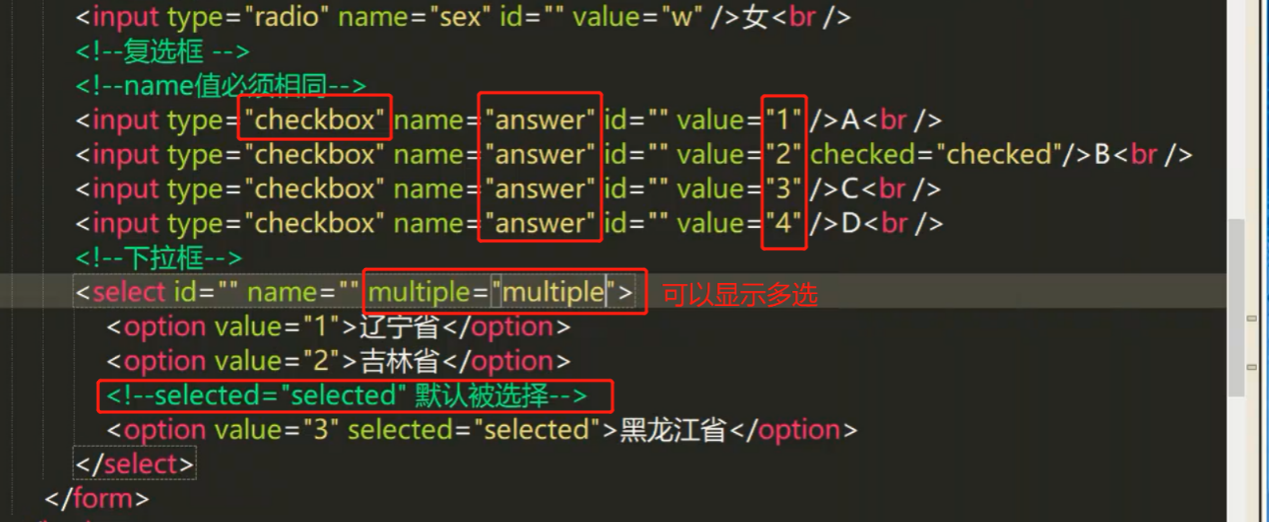
**6.文本域**

**textarea 个性签名，file上传文件，hidden隐藏域，按钮（button，reset重置，submit提交）**



**7.复选框：checkbox 可多选的话 multiple**

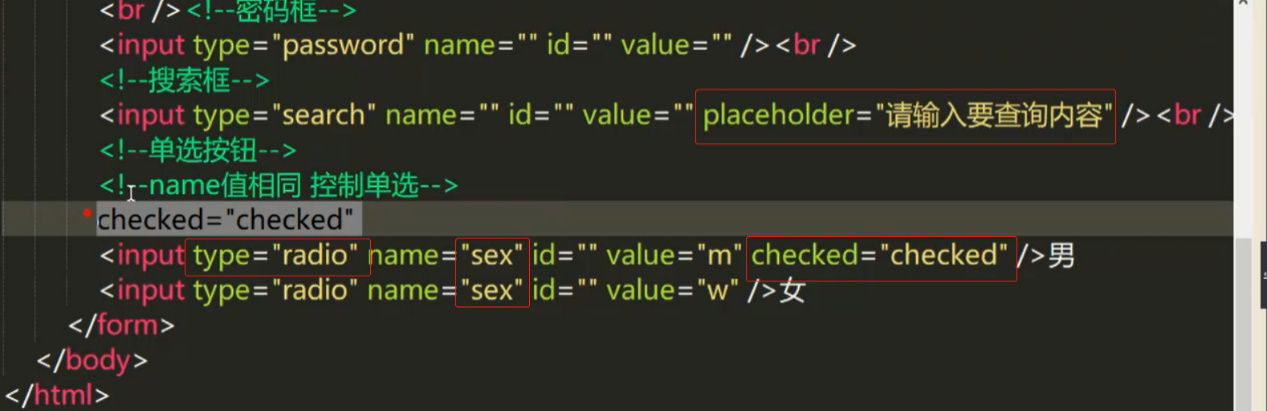
**下拉框：select 可多选的话 multiple selected=“selected”默认被选择**



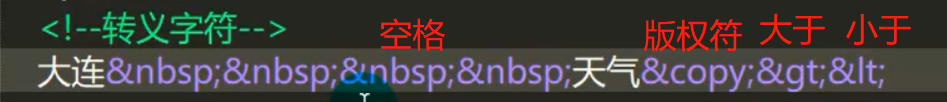
**8.密码框：input type=“password”**

**搜索框：search placeholder属性提供可描述输入字段预期值的提示信息（hint）**

**单选按钮：type=“radio” 注意：名字值必须相同，默认选择checked=“checked“**



**9.转义字符：空格，版权符，大于，小于**



**二，css样式的引入**

**最常用：外部引入**

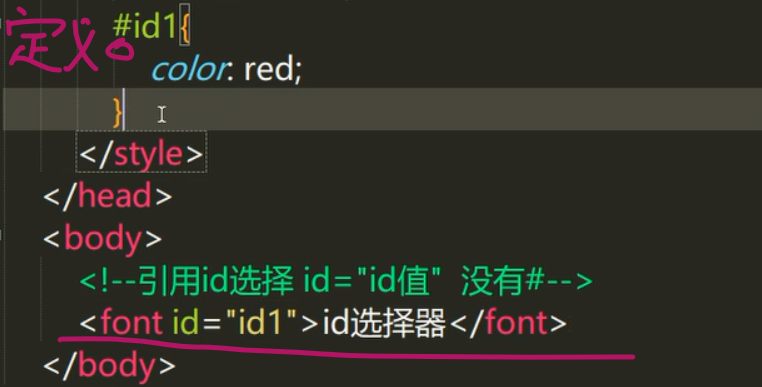
**内嵌/内联**

**头部引入**

**当发生冲突时 style>头部引入内联>外部引入 作用域最近的优先**

**CSS有3个选择器：1. id选择器 2. 标签选择器 3. 类选择器**

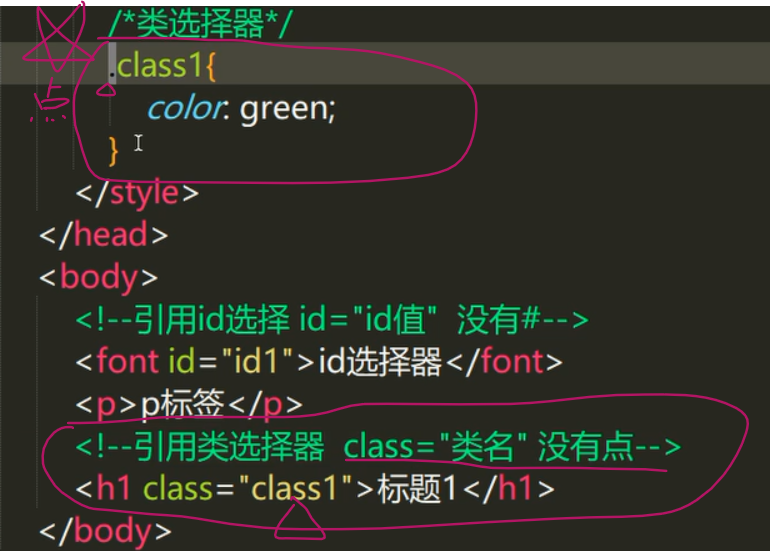
**Id选择器 定义时前边加#表示id,但是引用时前面不用加#号**

****

**标签选择器**

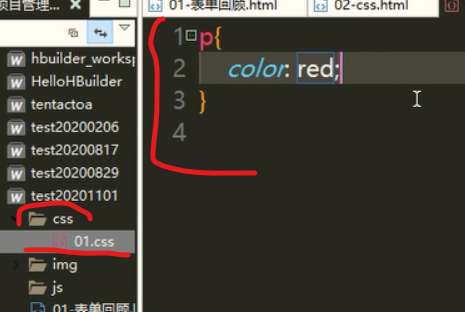
****

**类选择器**

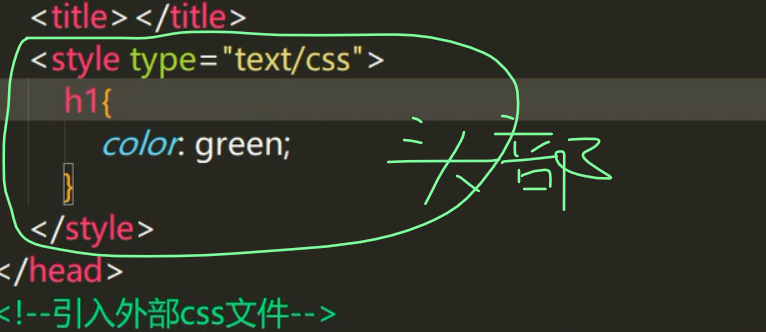
****

**外部引入**

****

****

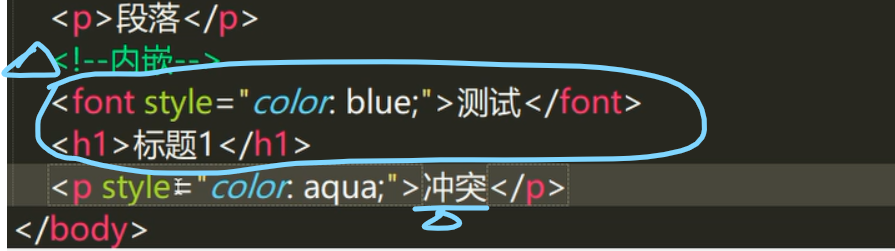
**头部引入**

****

**头部引用的语法**

****

**内嵌引入**

****