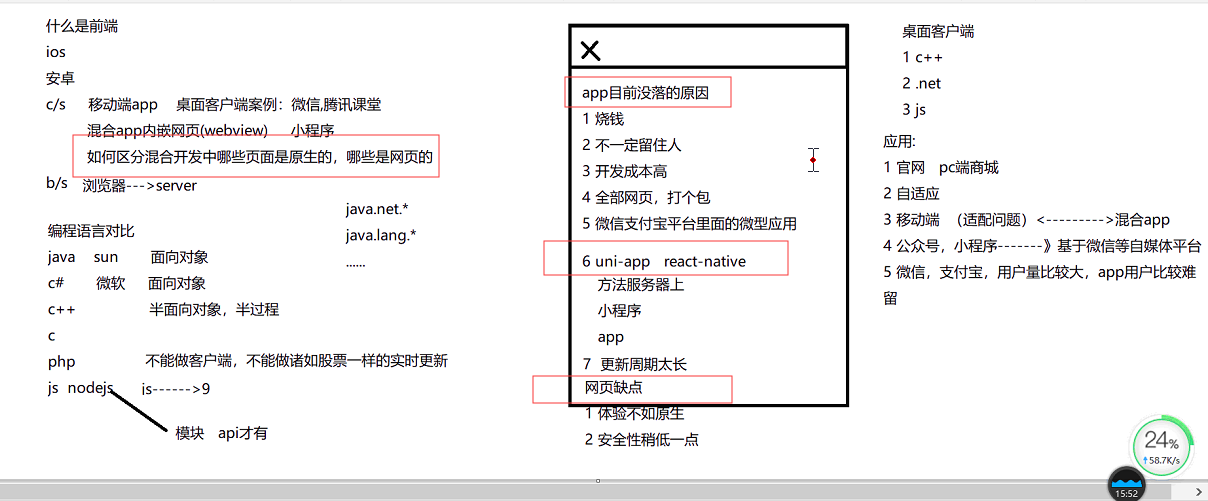
# 响应式

## 媒体查询

Vscode

混合app内嵌网页



**块元素**：h1~h6、p、div、ul、ol、dl、table、from、hr

**行内块**：span、a、br、img、input、select、i、em、del、u、strong、sup（上标）、sub（下标）

## 如何区分开发中那些页面是原生的，那些是网页？

### 看断网情况

* 1. 网络断掉之后，显示正常的是原生写的；显示404或错误的页面是html页面

### 看边界布局（打开开发者选项中的显示布局边界）

* 1. 若左上角显示有“X”就是网页
  2. 布局密密麻麻的就是原生控件

### 看复制文章的提示，需要通过对比才能得出结果

* 1. 长按页面可以复制，若出现文字选择、粘贴功能的是H5页面，否则是原生的页面

### 看加载的方式

* 1. 若打开新页面的导航栏下面有一条加载线的就是H5页面；若没有就是原生

### 看app顶部 导航栏是否会有关闭的操作

### 判断页面下拉刷新的时候(前提是有下拉刷新的功能)

### 下拉页面的时候显示网址提供方的一定是H5



<https://blog.csdn.net/detergent/article/details/79021500>

## 原生语言对比：

Java：sun 面向对象 大型项目 提供大量API

C#：微软 面向对象

C++： 半面向对象，办过程

C：

Php：不能做客户端，不能做诸如

微小宝：了解

# \*\*\*浏览器打开HTML中间发生的事情

### 1.输入网址

### 2.向服务端发送请求

### 3.服务器接收请求并查询浏览器想要的数据，返回给浏览器

### 4.浏览器拿到数据展示页面

### 5、HTTP协议：

# \*\*\*\*\*\*HTML5和HTML区别，

### 文档的类型声明不同

### 语义结构不同

(1) html没有结构语义化标签（不知道那个是开头哪个是结尾）

(2)H5添加许多具有语义化的标签，式结构清晰，提高代码的可读性；header、footer

### 绘图区别

1. html：指可伸缩矢量图形，用于定义网络的基于矢量的图形
2. H5的canvas元素使用脚本(通常使用js)在网页上绘制图像，可以控制画布的每一个像素

# \*\*\*\*\*\*H5新增特性

1. 语义标签化
2. 表单新特性
3. 视频(video)和音频(audio)
4. Canvas画布
5. 地理定位
6. 拖放API
7. webWorker
8. webStorage
9. webSocket
10. 本地存储(localStorage)，离线存储(sessionStorage)

# CSS3新增特性

1. 颜色：新增rgba、hsla模式
2. 文字阴影（text-shadow）
3. 边框：圆角（border-radius）、边框阴影：（box-shadow）
4. 盒子模型：（box-sizing）
5. 渐变
6. 过渡：（transition）
7. 2D转换：transform：translate(x,y)、rotate(x,y)、scale(x,y)、shew(x,y)

# \*\*\*\*\*\*都是什么标签没有结束标签

1. img、link、input、hr、br、meta、、、
2. 块元素
3. 内敛块元素(行内块)：不像块元素，不会独占一行，
   1. 元素：img......
4. 区别：两者都可以设宽高
5. 行内元素

# 行内块元素和块元素的区别

**块级元素**会独占一行

**行内块**：不单独占一行，会自带间隙

**行内元素**不会独占一行,相邻的行内元素会排列在同一行里,直到一行排不下,才会换行,其宽度随元素的内容而变化

text-align这个属性在水平方向上对行内（inline），行内块(inline-block)以及文字都起作用（行内元素或行内块元素设置水平居中用text-align属性）

# 行内元素和行内块垂直居中

**行内元素**：垂直居中用line-height属性

**块元素：**（父级高度-自己高度）/2：设置成内间距，父级设置 box-sizing:border-box，子级设置成margin：0 auto;是设置成水平居中

**行内块**：给父级设置宽高，然后使行高==高，让子级设置成行内块，在用vertical-align：midden配合行高使用（设置垂直居中，不能line-height会出问题，应该用vertical-align.多个行内块元素并排，同时设置居中）

## 块元素水平居中

1. 第一种：
   1. 设置每一个子块元素的的宽高，然后用float浮动
   2. 给父级定义一个总宽度，处理塌陷，然后使用margin：0 auto，水平居中
2. 第二种：块元素转成行内块元素(**会有默认的边距**)
   1. 设置每一个子块元素的宽高，然后设置成行内块元素（display：inline-block）
   2. 设置父级塌陷，然后水平居中（text-align：center）
3. Flex布局

display: flex;

justify-content: center;

align-items: center;

## margin塌陷：

子级设置一个margin-top之后父级也会随之变化，需要给父级设置一下padding-top

**行内元素垂直居中：**用高===行高

**行内块垂直居中：**用行高+vertical-align：middle实现

## 行内元素(内联元素)：排列在一行

span i a strong...

1. 水平居中
2. 垂直居中

## 布局使用：

header、footer、setion、nav、ai、

## 清除表格(table)间隙：border-collapse：collapse

## 常见标签：

块：div、、、、、

如何证明input是什么元素(块、行、行内块)

设置宽高，用一个对比一下，若两个在一行

margin是对于块元素而言

text-align是对行内块使用

# Table布局

## 绝对定位：



登录、注册、详情页、列表页

Window下六大属性

## 在一个页面中垂直居中

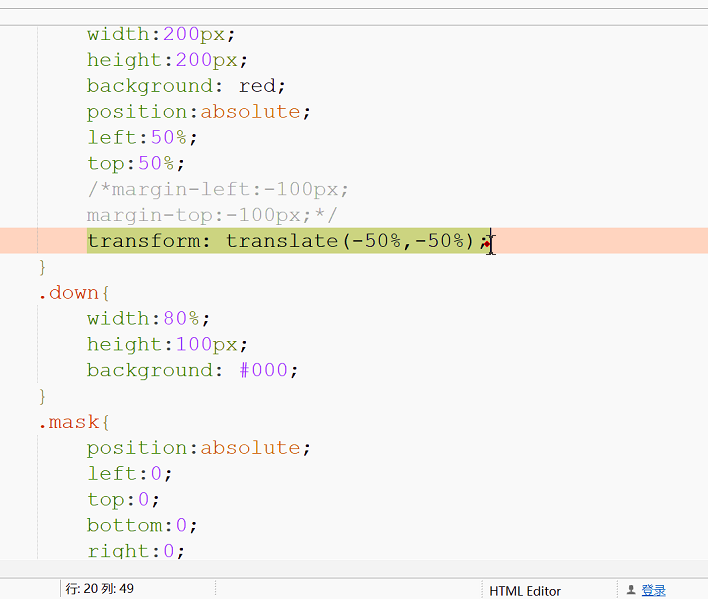
用position：absolute

Left:50%

Top:50%

Margin-top:-100px;//原有块的一半

Margin-left:-100px;



# flex



### flex-wrap:wrap//换行

### flex-direction:column//行、列

# a标签中写javascript：；是取消默认行为

# from表单取消默认行为：e.perventDefault()

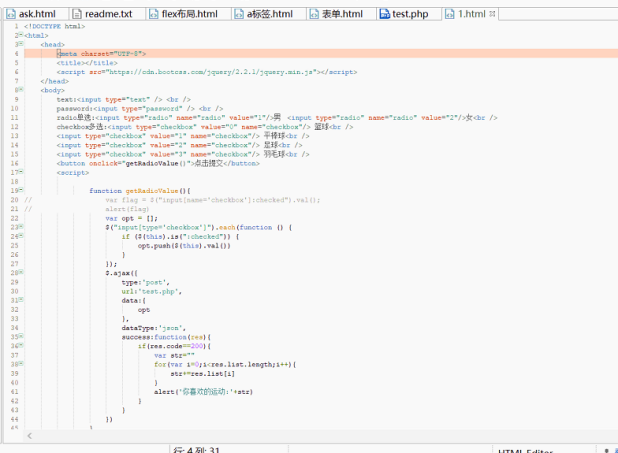
## 取消表单的默认行为

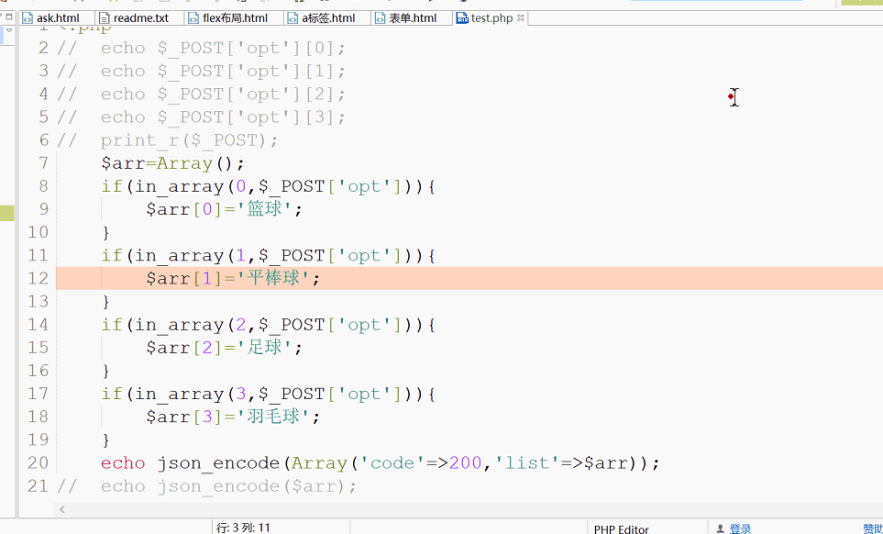


**php获取数据**



**从前台输入到后台获取，然后前台输出**

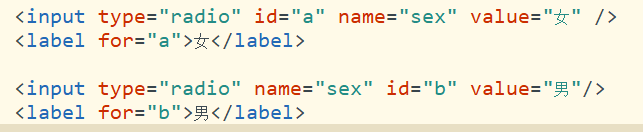




## 表单的类型

text、password、radio、checkbox、button、submit、reset....

## label标签--快速获取焦点



**<label for=“id值”>男</label>**

iframe写的后台

## 阿里图标：基础库中需要写一个iconfont