

# rem

---

## 1. 媒体查询

### 1.1.1. 引入资源

## 2. less基础

### 2.1. less变量

### 2.2. less编译

### 2.3. less嵌套

### 2.4. less运算

### 2.5. 将common的less 导入到 index.less中

## 3. rem适配方案

# 1. 媒体查询

媒体查询 ( **Media Query** ) 是CSS3新语法。

- 使用 @media 查询，可以针对不同的媒体类型定义不同的样式
- **@media** 可以针对不同的屏幕尺寸设置不同的样式

语法规范：

```
@media mediatype and|not|only (media feature) {  
    CSS-Code;  
}
```

- 用 @media 开头 注意@符号
- mediatype 媒体类型
- 关键字 and not only
- media feature 媒体特性 必须有小括号包含

## 1. mediatype 查询类型

将不同的终端设备划分成不同的类型，称为媒体类型

值	解释说明
all	用于所有设备
print	用于打印机和打印预览
screen	用于电脑屏幕，平板电脑，智能手机等

## 3. 媒体特性

每种媒体类型都具体各自不同的特性，根据不同媒体类型的媒体特性设置不同的展示风格。我们暂且了解三个。  
注意他们要加小括号包含

值	解释说明
width	定义输出设备中页面可见区域的宽度
min-width	定义输出设备中页面最小可见区域宽度
max-width	定义输出设备中页面最大可见区域宽度

```
/* 这句话的意思就是： 在我们屏幕上 并且 最大的宽度是 800像素 设置我们想要的样式 */  
/* max-width 小于等于800 */  
/* 媒体查询可以根据不同的屏幕尺寸在改变不同的样式 */
```

```
@media screen and (max-width: 800px) {  
  body {  
    background-color: pink;  
  }  
}
```

```
@media screen and (max-width: 500px) {  
  body {  
    background-color: purple;  
  }  
}
```

根据不同的条件匹配不同的样式

```

/* 2. 小于 540px 页面的背景颜色变为蓝色 */

@media screen and (max-width: 539px) {
    body {
        background-color:  blue;
    }
}

/* 3. 540 ~ 970 我们的页面颜色改为 绿色 */

@media screen and (min-width: 540px) and (max-width: 969px) {
    body {
        background-color:  green;
    }
}

/* 4. 大于等于970 我们页面的颜色改为 红色 */

@media screen and (min-width: 970px) {
    body {
        background-color:  red;
    }
}

```

从小到大写，重复的地方会去覆盖，故可以简化：

```

}

/* 3. 540 ~ 970 我们的页面颜色改为 绿色 */
/* @media screen and (min-width: 540px) and (max-width: 969px) {
    body {
        background-color: green;
    }
} */

@media screen and (min-width: 540px) {
    body {
        background-color:  green;
    }
}

/* 4. 大于等于970 我们页面的颜色改为 红色 */

@media screen and (min-width: 970px) {
    body {
        background-color:  red;
    }
}

```

screen 和 and 不可以省略

```

/* 从小到大顺序 */

@media screen and (min-width: 320px) {
  html {
    font-size: 50px;
  }
}

@media screen and (min-width: 640px) {
  html {
    font-size: 100px;
  }
}

.top {
  height: 1rem;
  font-size: .5rem;
  background-color: green;
  color: #fff;
  text-align: center;
}

```

### 1.1.1. 引入资源

```

</style>
/* 当我们屏幕大于等于 640px以上的，我们让div 一行显示2个 */
/* 当我们屏幕小于640 我们让div一行显示一个 */
/* 一个建议： 我们媒体查询最好的方法是从小到大 */
</style>
<link rel="stylesheet" href="style320.css" media="screen and (min-width: 320px)">
<link rel="stylesheet" href="style640.css" media="screen and (min-width: 640px)">

```

## 2. less基础

后缀名为less

## 2.1. less变量

变量是指没有固定的值，可以改变的。因为我们CSS中的一些颜色和数值等经常使用。

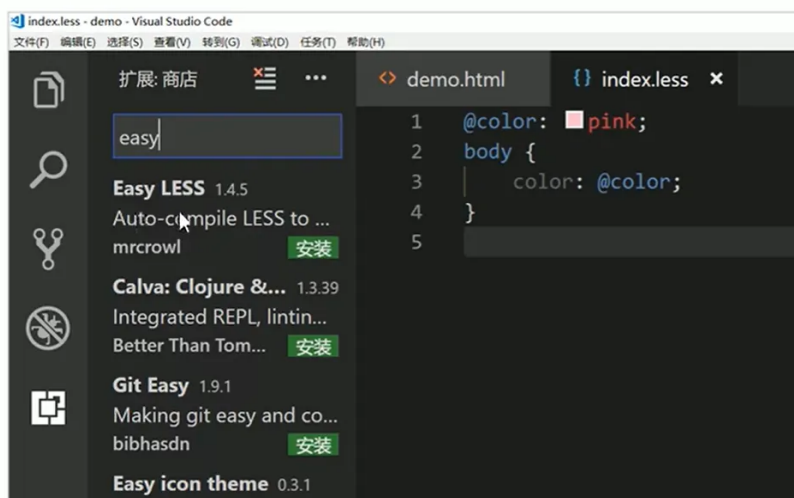
```
@变量名: 值;
```

```
2  @color: pink;
3  body {
4      background-color: @color;
5  }
```

## 2.2. less编译

### vocode Less 插件 ★

Easy LESS 插件用来把less文件编译为css文件  
安装完毕插件，重新加载下vscode。  
只要保存一下Less文件，会自动生成CSS文件。





## 2.3. less嵌套

在less中写：

子元素的样式直接写到父元素中

```

1  .header {
2      width: 200px;
3      height: 200px;
4      background-color:  pink;
5      a {
6          color:  red;
7      }
8  }

```

半年多没玩游戏了  
????????????  
👍要是离家出走了呢

如果要写伪类元素 or 伪元素 要加上一个 &符号

```

8      2. 如果有伪类 伪元素
9      &:hover {
10         color: blue;
11     }
12 }
13 }
14 .nav {
15     .logo {
16         color:  green;
17     }
18     &::before {
19         content: "";
20     }

```

## 2.4. less运算

## 3.7 Less 运算 ★

### 注意：

- 乘号 ( `*` ) 和除号 ( `/` ) 的写法
- 运算符中间左右有个空格隔开 `1px + 5`
- 对于两个不同的单位的值之间的运算，运算结果的值取第一个值的单位
- 如果两个值之间只有一个值有单位，则运算结果就取该单位

## 2.5. 将common的less 导入到 index.less中

2. 将刚才设置好的 common.less 引入到 index.less 里面 语法如下：

```
// 在 index.less 中导入 common.less 文件
@import "common"
```

## 3. rem适配方案

### 技术方案1

- less
- 媒体查询
- rem

### 技术方案2（推荐）

- flexible.js
- rem

pink 插件使用： 如何修改默认的那个html文字大小

还要注意要设置：

```
5  /* 如果我们的屏幕超过了 750px 那么我们就按照 750设计稿来走 不会让我们页面超过750px */
6
7
8  @media screen and (min-width: 750px) {
9      html {
10         font-size: 75px!important;
11     }
12 }
```

单纯用插件 要保证宽度超过750px html的字体大小是固定的

swiper的使用

下载css js文件 引入 css js 写入div script文本

新增左右箭头： 先在script里面 然后在div中加结构