

响应式开发

原理

使用媒体查询针对不同宽度的设备进行布局和样式设置，从而达到适配不同设备的目的

| 设备划分 | 尺寸区间 |
|--------------|---------------------|
| 超小屏幕（手机） | < 768px |
| 小屏设备（平板） | >= 768px ~ < 992px |
| 中等屏幕（桌面显示器） | >= 992px ~ < 1200px |
| 宽屏设备（大桌面显示器） | >= 1200px |

分类

响应式布局容器

一般会在所有元素外设置一个父容器以控制元素的显示
原理就是在不同设备下，通过媒体查询来改变这个父容器的大小，然后在改变元素的排列和大小

- 超小屏幕（手机，小于 768px）：设置宽度为 100%
- 小屏幕（平板，大于等于 768px）：设置宽度为 750px
- 中等屏幕（桌面显示器，大于等于 992px）：宽度设置为 970px
- 大屏幕（大桌面显示器，大于等于 1200px）：宽度设置为 1170px

尺寸划分

为什么设置的宽度都比设备最大尺寸要小呢？ 因为一般的页面两侧会留白

响应式导航栏案例

```
<div class="container">
  <ul>
    <li>导航栏</li>
    <li>导航栏</li>
    <li>导航栏</li>
    <li>导航栏</li>
    <li>导航栏</li>
    <li>导航栏</li>
    <li>导航栏</li>
    <li>导航栏</li>
  </ul>
</div>
```

html

```
.container{
  width: 750px;
  margin:0 auto;
}
.container ul li{
  float:left;
  width: 93.75px;
  height:30px;
  background-color: green;
}
@media screen and (max-width:767px){
  .container{
    width: 100%;
  }
  .container ul li{
    width: 33.33%;
  }
}
```

CSS

```
@media screen and (max-width: 767px) {
  .container{
    width:100%;
  }
}
@media screen and (min-width: 768px) {
  .container{
    width:750px;
  }
}
@media screen and (min-width: 992px) {
  .container{
    width:970px;
  }
}
@media screen and (min-width: 1200px) {
  .container{
    width:1170px;
  }
}
```

代码演示