

# 任务1 首页响应式轮播制作

## 1.1 预期效果

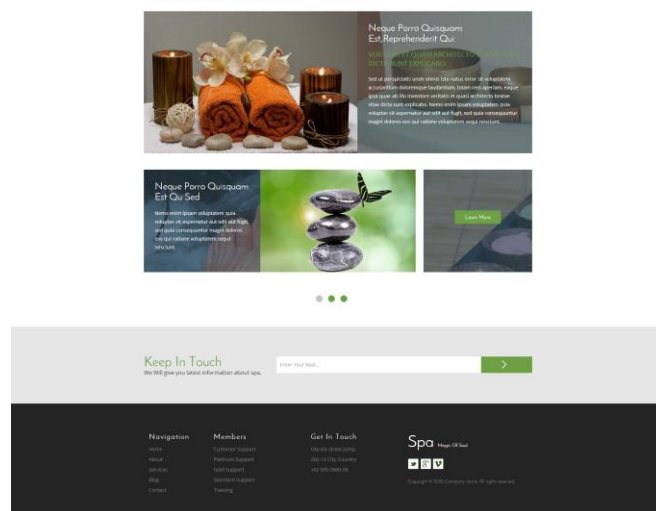


图 1-1 在大屏幕显示器上 (≥1200px) 效果图

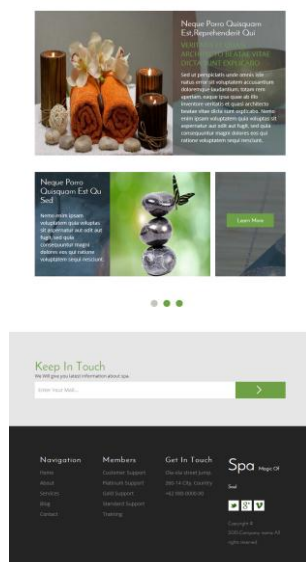


图 1-2 在中等屏幕显示器上 (1200px~992px) 效果图



图 1-3 在小屏幕显示器上(992px~768px)效果图



图 1-4 在超小屏幕显示器上(&lt;320px)效果图



## 1.2 掌握技能

表 1-1 任务说明

任务名称	子任务	任务内容
首页响应式轮播及 keep-in-touch 部分的制作	1	在 PC 端编写首页轮播的 html+css
	2	按屏幕从大到小进行屏幕的适配并编写轮播的 js
	3	编写 keep-in-touch 部分并做屏幕适配

## 1.3 任务分解

### 1. 在 PC 端编写首页轮播的 html+css

(1) 在 js 目录中创建 index.js 并在 index.html 中引入 index.js

(2) 继续在 index.html 里面建立放置轮播的容器, 并设置 bootstrap 的基本的标签结构

```
<div class="banner_wrap">
  <div class="container">
    <div class="banner">
      <ul class="banner_list">
        <li class="banner_item">
          <!-- 放置轮播的内容-->
        </li>
      <div class="clearfix"></div>
    </ul>
  </div>
</div>
<div id="footer"></div>
<script>
  $("#indexHeader").Load("header.html");
  $("#footer").Load("footer.html");
</script>
```

1、创建一个 .banner\_wrap 的 div 盒子包裹住整体的轮播内容并创建 .container 的 div



盒子（在 bootstrap 为栅格系统的容器）

2、引入公共的尾部，写法如上

2、在里面创建一个 .banner 的 div 放置轮播的列表 .banner\_list 里面 .banner\_item 为轮播的实际内容（由于每个轮播的实际内容及布局都一样，所有这里我们只写一个，其余粘贴复制 4 个就行）

（3）设置轮播实际内容的基本标签结构

```
<div class="banner_top">
  <div class="top_left">
    
  </div>
  <div class="top_right">
    <h3>Neque Porro Quisquam Est,Reprehenderit Qui</h3>
    <h4>VERITATIS ET QUASI ARCHITECTO BEATAE VITAE DICTA SUNT
    EXPLICABO</h4>
    <p>Sed ut perspiciatis unde omnis iste natus error sit voluptatem
    accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab
    illo inventore veritatis et quasi architecto beatae vitae dicta sunt
    explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit
    aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem
    sequi nesciunt.</p>
  </div>
  <div class="clearfix"></div>
</div>
<div class="banner_bottom">
  <div class="btm_left">
    <h3>Neque Porro Quisquam Est Qu Sed</h3>
    <p>Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit
    aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem
    sequi nesciunt. </p>
  </div>
  <div class="btm_left1">
    
  </div>
  <div class="btm_right">
    <div class="more">
      <a href="#">Learn More</a>
    </div>
  </div>
</div>
```



```

</div>
<div class="clearfix"></div>
</div>

```

这部分写在上面的 li.banner\_item 中。单个轮播分为上部分内容和下部分内容，上部分内容又分为左右两部分，下部分内容分为左右中三部分，按照这个关系写轮播的标签结构

(4) 设置首页轮播模在 PC 端的样式

```

.banner_wrap {
    padding: 5em 0 10em;
    position: relative;
}

.banner {
    width: 100%;
    overflow: hidden;
    position: relative;
}

.banner .banner_list {
    width: 100%;
    zoom: 1;
    /*position: absolute;*/
}

.banner .banner_list li.banner_item {
    float: left;
    width: 1140px;
}

.banner_top .top_left {
    float: left;
    width: 55%;
}

.banner_top .top_left img {
    width: 100%;
    display: block;
}

```

```
}  
  
.banner_top .top_right {  
    float: right;  
    width: 45%;  
    background: url(../images/7.jpg) no-repeat 0px 0px;  
    display: block;  
    min-height: 417px;  
    padding: 2em;  
}  
  
.banner_top .top_right h3 {  
    text-transform: capitalize;  
    color: #fff;  
    font-size: 1.8em;  
    margin: 0;  
    font-family: 'Josefin Sans', sans-serif;  
}  
  
.banner_top .top_right h4 {  
    text-transform: uppercase;  
    font-size: 20px;  
    margin: 1em 0;  
    line-height: 1.3em;  
    color: #6ea145;  
}  
  
.banner_top .top_right p {  
    font-size: 14px;  
    color: #fff;  
    margin: 0;  
    line-height: 1.8em;  
}  
  
.banner_bottom {  
    margin: 3em 0 0;  
}  
  
.banner_bottom .btm_Left {  
    float: left;  
    width: 30%;  
    background: url(../images/8.jpg) no-repeat 0px 0px;
```



```
display: block;
min-height: 301px;
padding: 2em;
}

.banner_bottom .btm_left h3 {
  text-transform: capitalize;
  color: #fff;
  font-size: 1.8em;
  margin: 0;
  font-family: 'Josefin Sans', sans-serif;
}

.banner_bottom .btm_left p {
  font-size: 14px;
  color: #fff;
  margin: 1.5em 0 0em;
  line-height: 1.8em;
}

.banner_bottom .btm_left1 {
  float: left;
  width: 40%;
}

.banner_bottom .btm_left1 img {
  display: block;
  width: 100%;
}

.banner_bottom .btm_right {
  float: right;
  width: 28%;
  background: url(../images/10.jpg) no-repeat 0px 0px;
  display: block;
  min-height: 301px;
  padding: 8em 0 0;
  text-align: center;
}
```

这部分的样式写在 index.css 中



- 1、设置轮播包裹.banner\_wrap 距上方和下方的部分的距离这里我们用 padding 撑开
- 2、设置轮播盒子.banner 的属性并设置它的定位方式为 relative 方便轮播列表.banner\_list 的移动。设置 overflow:hidden;使超出部分隐藏掉
- 3、设置.banner\_list 的宽度为 1000% (以免宽度不够), 定位方式为 absolute 在这里我们先注释掉, 为了防止我们静态页面显示不出来 (在父级元素为相对定位并设有超出隐藏且不设置高度时, 子元素定位方式设置为 absolute 后会脱离文档流, 撑不开父级的高度, 造成子元素不显示)。
- 4、设置轮播中上部分内容的样式 (里面的大宽度利用百分比设置, 方便一会设置不同屏幕的适配)
- 5、设置轮播中下部分内容的样式 (通样里面的大宽度利用百分比设置)

#### (5) 设置轮播焦点的标签结构

```
<ul class="banner_nav">
  <li class="active">1</li>
  <li class="">2</li>
  <li class="">3</li>
</ul>
```

轮播小点的标签结构写在.banner 的 div 的下面

#### (6) 设置轮播焦点的样式

```
.banner_nav {
  position: absolute;
  left: 46.3%;
  bottom: 7%;
}
.banner_nav li {
  margin: 0 6px;
  display: inline-block;
  zoom: 1;
  width: 20px;
  height: 20px;
  background: #6EA145;
  cursor: pointer;
  text-indent: -9999px;
  border-radius: 15px;
```





```

}
.banner_nav li.active {
    background: #C6C5C0;
    cursor: default;
}

```

这部分的样式继续写在 index.css 中

- 1、设置 .banner\_nav 的定位方式为 absolute 并调整它的位置
- 2、设置 li 的 display 为 inline-block 让它变为行块元素并设置它们的距离和小点的样式
- 3、设置当前显示的 .active 的样式

## 2. 按屏幕从大到小进行屏幕的适配并编写轮播的 js

- (1) 利用 js 设置不同分辨率下的宽度值

```

$(function () {
    /*轮播 */
    var banner_list = $(".banner_list");
    var dot = $(".banner_nav li");
    var bannerItemW = 0;
    var bannerItemH = 0;
    var banner = $("div.banner");
    var banner_item = $("li.banner_item");
    var timer=null;
    var num = 0;
    setBannerWidth();//页面首次加载时 初始化 每个部分的 宽度好高度值
    $(window).resize(function () { //窗口改变时 重置各个部分的宽度 高度值
        setBannerWidth()
    });
    /* 初始化时或窗口改变时设置 重置页面各个部分的宽度值 */
    function setBannerWidth() {
        var winW = $(window).width();//获取窗口的宽度
        if (winW >= 1183) { //设置每个轮播块的宽度值
            banner_item.width(1140)
        } else if (winW < 1183 && winW >= 975) {
            banner_item.width(940)
        } else if (winW < 975 && winW >= 768) {
            banner_item.width(720)
        } else if (winW < 768) {

```



```

        banner_item.width(winW - 30)
    }
    bannerItemW = banner_item.width();
    bannerItemH = banner_item.height();
    banner.css({height: bannerItemH + "px"}); //设置轮播窗口的高度值
    banner_list.css({left: 0 + "px"}); //设置轮播列表的位置
}
})

```

这部分写在 index.js 中

- 1、统一声明要用到的变量和获取要用到的 dom 节点
- 2、写一个 setBannerWidth 的函数用来设置轮播的宽度和高度
- 3、setBannerWidth 函数的调用分为页初次加载时和窗口改变时

(2) 在小于等于 1200px 的屏上调整轮播中的细节位置

```

@media (max-width: 1200px) {
    .banner_top .top_left img {
        min-height: 426px;
    }
    .banner_bottom .btm_left1 img {
        min-height: 301px;
    }
    .banner_top .top_right h3, .banner_bottom .btm_left h3 {
        font-size: 1.5em;
    }
}

```

在小于等于 1200px 的屏上是轮播的图片会出现错位，用 min-height 来调整图片的高度值并调整 h3 标签内的字体大小

(3) 在小于等于 991px 的屏上设置

```

@media (max-width: 991px) {
    .banner_top .top_left img{
        min-height:417px;
    }
    .banner_top .top_right {
        padding: 1.5em;
    }
    .banner_top .top_right h4 {

```



```

        margin: 0.5em 0;
    }
    .banner_top .top_right p, .banner_bottom .btm_left p {
        line-height: 1.5em;
    }
    .banner_bottom .btm_left {
        padding: 1em;
    }
}

```

在小于等于 991px 的屏上调整轮播内的细节样式问题

(4) 在小于等于 768px 的屏上设置

```

@media (max-width: 991px) {
    .banner_top .top_left {
        float: none;
        width: 100%;
    }
    .banner_top .top_left img {
        min-height: initial;
    }
    .banner_top .top_right {
        float: none;
        width: 100%;
        background: url(../images/7.jpg) no-repeat 0px 0px;
        background-size: cover;
        -webkit-background-size: cover;
        -moz-background-size: cover;
        -o-background-size: cover;
        -ms-background-size: cover;
        min-height: 300px;
    }
    .banner_bottom .btm_left {
        float: none;
        width: 100%;
        background: url(../images/8.jpg) no-repeat 0px 0px;
        background-size: cover;
        -webkit-background-size: cover;
        -moz-background-size: cover;
        -o-background-size: cover;
        -ms-background-size: cover;
        min-height: 165px;
    }
}

```



```

}
.banner_bottom .btm_left1 {
    float: none;
    width: 100%;
}
.banner_bottom .btm_right {
    float: none;
    width: 100%;
    background: url(../images/10.jpg) no-repeat 0px 0px;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    -ms-background-size: cover;
    min-height: 150px;
    padding: 4em 0 0;
}
.banner_wrap {
    padding: 3em 0 6em;
}
.banner_nav {
    left: 41.3%;
    bottom: 2%;
}
}

```

在小于等于 768px 的屏上时轮播上下部分中的内容变成单行显示，所以要清除掉浮动，然后设置宽的为 100%，背景是图片的还要设置 `background:cover;`，让背景图片保持图片本身的宽高比例，将图片缩放到正好完全覆盖定义背景的区域。调整其他细节样式。

(5) 在小于等于 480px 的屏上位置

```

@media (max-width: 480px) {
    .banner_bottom .more a {
        font-size: 15px;
        padding: 8px 20px;
    }
    .banner_top .top_right h3, .banner_bottom .btm_left h3 {
        font-size: 1.2em;
    }
    .banner_top .top_right h4 {
        font-size: 16px;
    }
}

```



```

}
.banner_top .top_right {
    padding: 1em 1.5em;
}
.banner_bottom .btm_Left1 {
    margin: 1em 0 0;
}
.banner_wrap {
    padding: 1em 0 4em;
}
.banner_nav li a {
    width: 15px;
    height: 15px;
}
.banner_nav {
    left: 38.3%;
    bottom: 1%;
}
}

```

在小于等于 480px 的屏上是调整轮播整体的变形和其他小细节

(6) 在小于等于 320px 的屏上位置

```

@media (max-width: 320px) {
    .banner_wrap {
        padding: 0em 0 3em;
    }
    .banner_top .top_right h4 {
        font-size: 13px;
        margin: .5em 0;
    }
    .banner_top .top_right {
        padding: 1em;
    }
    .banner_top .top_right p, .banner_bottom .btm_Left p {
        font-size: 13px;
    }
    .banner_bottom .btm_Left p {
        margin: 0.5em 0 0em;
    }
}

```



```
.banner_bottom .btm_left {
    padding: 1em;
}
.banner_top .top_right h3, .banner_bottom .btm_left h3 {
    font-size: 1em;
}
.banner_bottom .btm_right {
    min-height: 90px;
    padding: 2em 0 0;
}
.banner_nav {
    left: 31.3%;
    bottom: 0.5%;
}
.banner_nav li a {
    width: 10px;
    height: 10px;
}
}
```

在小于等于 320px 的屏上是调整各个部分的细节的位置关系和字体的大小

(7) 轮播的 js 代码部分

```
dot.click(function () { // 点的点击事件
    clearInterval(timer);
    num = $(this).index();
    moveBanner(num);
    timer=setInterval(function(){automatic()},3000);
});

timer=setInterval(function(){automatic()},3000);

function moveBanner(num) { // 设置轮播的动作
    dot.eq(num).addClass("active").siblings().removeClass("active")
    banner_list.animate({left: -bannerItemW * num + "px"},1000);
}

function automatic() { // 自动轮播的函数
    num++;
    if (num==3){
        banner_list.animate({left: -bannerItemW * num + "px"},1000)
        .animate({left: 0 + "px"},0);
    }
}
```



```

    num=0;

    dot.eq(0).addClass("active").siblings().removeClass("active")
  }else {
    moveBanner(num)
  }
}

```

这部分写早 index.js 里面接着前面向下写

- 1、编写轮播的运动函数 moveBanner () 在这里面设置点和轮播的关系
- 2、编写自动轮播的函数 automatic() 在这里面设置自动轮播的关系
- 3、设置轮播焦点的点击事件，调用轮播的运动函数，清除自动轮播的函数，在几秒后在设置自动轮播
- 4、第一次加载时设置自动轮播
- 5、解开 (4) 里 .banner\_list 中的定位属性 position: absolute;

### 3. 编写 keep-in-touch 部分并做屏幕适配

(1) 设置 keep-in-touch 的基本标签结构

```

<div class="keep_in_touch">
  <div class="container">
    <div class="row">
      <div class="col-lg-4 touch_text">
        <h3>Keep In Touch</h3>
        <p>We Will give you latest information about spa.</p>
      </div>
      <div class="col-lg-8 touch_input">
        <input type="text" value="Enter Your Mail..." itemscope="Enter Your Mail..." onfocus="this.value = ' '; onblur="if (this.value == ' '){this.value = 'Enter Your Mail...';}" required="">
        <input type="submit" value=" ">
      </div>
      <div class="clearfix"> </div>
    </div>
  </div>
</div>

```



这部分写在 index.html 中的轮播的下面

- 1、编写基本的标签结构和栅格系统的结构标签
- 2、input 输入框上面的文字利用获取焦点和失去焦点的事件配合来实现

(2) 编写 keep-in-touch 的基本样式

```
.touch_text h3 {
    font-family: 'Josefin Sans', sans-serif;
    font-size: 2.5em;
    color: #6ea145;
    margin: 0;
}

.touch_input{
    margin: 0.5em 0 0 0;
}

.touch_textp {
    font-size: 14px;
    color: #999;
    margin: 0;
}

.touch_input input[type="text"] {
    width: 80%;
    background: #fff;
    border: none;
    font-size: 14px;
    color: #A2A2A2;
    outline: none;
    padding: 14px 10px;
    float: left;
}

.touch_input input[type="submit"] {
    width: 20%;
    float: left;
    background: url(../images/1.png) no-repeat 56px 11px #6ea145;
    outline: none;
    border: none;
    padding: 13px 0;
    text-align: center;
    transition: .5s all;
    -webkit-transition: .5s all;
    -moz-transition: .5s all;
```





```

-o-transition: .5s all;
-ms-transition: .5s all;
}
.touch_input input[type="submit"]:hover {
  background: url(../images/1.png) no-repeat 56px 11px #c6c5c0;
}
.keep_in_touch{
  padding: 5em 0;
  background-color: #E6E6E6;
}

```

这部分继续写在 index.css 中

- 1、设置 .keep\_in\_touch 距其他部分的距离用 padding 撑开并设置背景颜色
- 2、设置各个部分的样式

(3) 在 bootstrap 定义的栅格系统上进行适配

```

<div class="col-lg-4 col-md-4 col-sm-12 col-xs-12 touch_text">
  <!-- 内容部分已省略-->
</div>
<div class="col-lg-8 col-md-8 col-sm-12 col-xs-12 touch_input">
  <!-- 内容部分已省略-->
</div>

```

拖动浏览器屏幕到 bootstrap 定义的栅格系统的屏幕宽度，为 .row 下面的直接子元素设置上该屏幕宽度下列的类前缀并调整直接子元素的列数

(4) 在小于等于 1024px 的屏上调整

```

@media (max-width: 1024px) {
  .touch_text p {
    font-size: 13px;
  }
  .touch_text h3 {
    font-size: 2.2em;
  }
}

```

调整文字的字体大小

(5) 在小于等于 768px 的屏上调整



```
@media (max-width: 768px) {
  .keep_in_touch{
    padding: 3em 0;
  }
  .touch_text{
    text-align: center;
    margin: 0 0 1em;
  }
}
```

调整 .keep\_in\_touch 距外面部分的距离通过 padding 来调整

在小于等于 768px 的屏上时这个模块分为上下两部分，上部分的文字设置为居中对齐

(6) 在小于等于 480px 的屏上调整

```
@media (max-width: 480px) {
  .touch_text h3 {
    font-size: 1.5em;
  }
  .keep_in_touch{
    padding: 2em 0;
  }
  .touch_input input[type="submit"] {
    background: url(../images/1.png) no-repeat 30px 11px #6ea145;
  }
  .touch_input input[type="submit"]:hover {
    background: url(../images/1.png) no-repeat 30px 11px #c6c5c0;
  }
}
```

调整各个部分的位置关系和字体大小还有按钮的背景图片的位置

(7) 在小于等于 320px 的屏上调整

```
@media (max-width: 320px) {
  .touch_text p {
    font-size: 12px;
  }
  .touch_text h3 {
    font-size: 1.2em;
  }
  .keep_in_touch{
```



```
padding: 1em 0;
}
.touch_input input[type="text"] {
    font-size: 13px;
    padding: 10px 10px;
}
.touch_input input[type="submit"] {
    padding: 8px 0;
    background: url(../images/1.png) no-repeat 14px 7px #6ea145;
}
.touch_input input[type="submit"]:hover {
    background: url(../images/1.png) no-repeat 14px 7px #c6c5c0;
}
.get_in_touch h3 {
    letter-spacing: 0px;
    font-size: 1.1em;
}
}
```

调整各个部分的位置关系和字体的大小还有按钮背景图片的位置

## 1.4 参考资料

无

## 1.5 扩展练习

教师自行补充