



任务1 首页 banner 轮播图(8 小时)

1.1 预期效果



图 1-1 导航+banner 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
banner	1	创建相关文件和目录结构
	2	轮播布局页面编写
	3	图片运动效果
	4	点击切换效果



1.3 任务分解

1. 创建相关文件和目录结构

(1) 用 DW 软件, 新建 H5 格式的 Html 文件, 保存到 guanwan 文件夹下, 取名 index.html, (此项目我们将头部和底部写成独立的文件, 以便各个页面公用)

(2) 用 DW 软件, 新建 css 文件, 保存到 css 文件夹下, 取名 index.css

(3) 用 DW 软件, 新建 js 文件, 保存到 js 文件夹下, 取名 index.js

(4) 修改 title 及写入搜索关键字

```
<title>北京卓新思创科技有限公司</title>
```

```
<meta name="keywords" content=" 小鸟掌学, 企业培训平台, 全民全学、全民全校、全民全智, 微课堂, 微学习, 企业定制培训">
```

```
<meta name="description" content=" “小鸟掌学” 云平台是一套完整的专家知识、学习管理、监督考核系统。首先根据企业和员工的特性, 抓取学习需求, 然后从“平台学习库、行业学习库和企业内训库”中调出教学资源, 按学习计划“推送”任务给每一位员工, 并对学习过程进行监管, 对学习效果进行评估, 根据学习情况调整学习计划, 不断优化员工的职业生涯">
```

Title 是文档要显示的标题

Meta 标签 中的 keywords 有助于浏览器抓取, 优化搜索

Meta 标签 中的 description 是搜索出来显示的简略结果

(5) 在 css 文件夹中添加 Animate.css 文件

由于页面用到了 Animate.css, 所以需要在 css 文件夹中添加 Animate.css 文件。到示例代码的 css 文件中拷贝此文件到项目的 css 文件夹下

(6) 引入 css, js

```
<link href="css/common.css" type="text/css" rel="stylesheet" />
<link href="css/index.css" type="text/css" rel="stylesheet" />
<link href="css/animate.css" type="text/css" rel="stylesheet" />
```

```
<script src="js/jquery-1.8.3.min.js"></script>
<script src="js/move.js"></script>
```



```
<script src="js/index.js"></script>
```

Animate.css 是一个 css 特效库，轮播图效果编写的时候会用到

(7) 在 index.js 中编写代码，引入头部

```
$(function(){
    $("#header").Load("header.html");
})
```

jQuery 的 load () 方法可以直接加载页面，但需要放在服务器上才有用，火狐浏览器比较特殊，允许直接加载，以后的页面查看，请使用火狐浏览器。

2. 编写 banner 的 html、css

(1) 写入 banner 包裹层的 html

```
<div class="banner_wrap">
    <div class="bchange banner_change">
    </div>
    <div class="banner_one banner_01">
    </div>
</div>
```

banner_wrap 是 banner 包裹层

bchange 是切换按钮的包裹层

banner_one 是一个轮播的包裹层

(2) 在 index.css 中设置 banner_wrap 和 banner_one 样式

```
.banner_wrap{
    height:593px;
    position:relative;
    min-width:1100px;
```



```

}
.banner_one{
    height:593px;
    position:absolute;
    top:0;
    left:0;
    width:100%;
}
.banner_01{
    background:url(../images/banner01_bg.jpg) no-repeat top center;
}
    
```

.banner_wrap 设置高度，最小宽度，并设置 position:relative 以让子集可以相对于它定位。

banner_one 设置高度宽度和定位，（此处因为设置了定位，需要设置 width:100%，才能让其宽度和父级一样宽）

设置上 banner_01 的背景

(3) 在 .banner_01 中写入第一个轮播的 html

```

<div class="banner_one_contents">
    
    
    
</div>
    
```

一个轮播包含三个小图。

(4) 设置第一个轮播的样式

```

.banner_one_contents{
    width:1100px;
    margin:0 auto;
    position:relative;
}
.banner_one_contents *{
    display:none;
}
    
```



```

}
.banner_01 .image01{
    position:absolute;
    top:204px;
    left:343px;
}
.banner_01 .image02{
    position:absolute;
    top:300px;
    left:375px;
}
.banner_01 .image03{
    position:absolute;
    top:354px;
    left:374px;
}
    
```

.banner_one_contents 设置宽度和 Margin 值以居中，设置 position:relative 让图片可以相对于其定位

.banner_one_contents *将所有的图片隐藏，设置的意义为等后面设置动画的时候避免闪烁，（此处先不设置，等编写完成在设置）

设置各小图片的定位位置。

(5) 另两个轮播图的 html

```

<div class="banner_one banner_02" style="display:none;">
    <div class="banner_one_contents">
        
        
        
    </div>
</div>

<div class="banner_one banner_01 banner_03" style="display:none;">
    <div class="banner_one_contents">
        
        
        
    </div>
    
```



</div>

- 1、修改相应的图片链接。
- 2、banner_03 和 banner_01 类似，可以使用 banner_01 的样式，所以也以上此类
- 3、这两个 banner 需设置隐藏（先不设置，等样式设置完，再设置隐藏）

(6) 另两个轮播图设置样式

```
.banner_02{
    background:url(../images/banner02_bg.jpg) no-repeat top center;
}

.banner_02 .image01{
    position:absolute;
    top:177px;
    left:61px;
}

.banner_02 .image02{
    position:absolute;
    top:70px;
    left:261px;
}

.banner_02 .image03{
    position:absolute;
    top:195px;
    left:270px;
}

.banner_03 .image01{
    top:189px;
    left:384px;
}

.banner_03 .image02{
    top:298px;
    left:397px;
}
```

设置 banner 的背景。

设置各小图片的定位位置。



(7) 在 .bchange 中写入轮播按钮

```
<div class="bchange banner_change">
  <span class="prev"></span>

  <div class="middle_points">
    //这里还需要添加位置点的代码
  </div>

  <span class="next"></span>
</div>
```

Prev 上一个。

middle_points 包含中间的位置显示小点

next 下一个

(8) 设置样式

```
.bchange{
  position:absolute;
  left:50%;
  margin-left:-108px;
  bottom:40px;
  z-index:20;
  height:29px;
  width:216px;
  background:url(../images/next&prev_line.png) no-repeat left center;
}
.bchange > span{
  position:absolute;
  top:0;
  height:29px;
  width:29px;
  background:url(../images/prev_jiantou.png) no-repeat;
  cursor:pointer;
}
.bchange > .next{
  right:0;
  background:url(../images/next_jiantou.png) no-repeat;
}
/*关于 next. prev 的以下代码需复制到 css 最底部，以便公用 */
```



```
.next:hover{
    background:url(../images/next_jiantou_hover.png);
}
.prev:hover{
    background:url(../images/prev_jiantou_hover.png);
}
.prev:hover, .next:hover{
    animation:1s linear 0s normal none 1 running su-pudding;
}
.bchange .middle_points{
    margin:0 auto;
    width:45px;
    height:14px;
    background:url(../images/next&prev_mbg01.png) no-repeat;
    margin-top:7px;
}
```

.bchange 设置定位和 margin-left 值以居中显示, 设置 z-index 以浮动到轮播图上边

.bchange > span 和 .bchange > .next 设置向前、向后按钮样式

.next:hover / .prev:hover 设置鼠标移上时变换背景效果

用 animation 设置跳动的动画效果, 采用的运动形式是 su-pudding (此效果是一个弹性伸缩的运动 keyframes, 代码在下一步给, 大家复制放到 css 底部即可)

Ps: next:hover / .prev:hover 的代码也放到页面底部, 后面也有同样布局的箭头用到此效果, 以便公用 (如果放到前面, 根据 css 叠层样式规则, 会被覆盖)

.middle_points 设置宽度和 margin 以居中 (此处的宽度如果因为轮播图数量变化而引起调整, 需要用 js 去动态设置, 此处先写三个的宽度)

设置背景是左边的圆角, 设置 margin-top 值以设置上部距离

(9) @keyframes su-pudding 代码

```
0% {
    transform: scale(1.2, 0.8);
}
1% {
    transform: scale(1.18, 0.82);
}
```




```

2% {
    transform: scale(1.16, 0.84);
}
3% {
    transform: scale(1.13, 0.87);
}
4% {
    transform: scale(1.1, 0.9);
}
5% {
    transform: scale(1.07, 0.93);
}
6% {
    transform: scale(1.04, 0.96);
}
7% {
    transform: scale(1.01, 0.99);
}
8% {
    transform: scale(0.99, 1.01);
}
9% {
    transform: scale(0.97, 1.03);
}
10% {
    transform: scale(0.95, 1.05);
}
11% {
    transform: scale(0.94, 1.06);
}
12% {
    transform: scale(0.93, 1.07);
}
13% {
    transform: scale(0.93, 1.07);
}
14% {
    transform: scale(0.93, 1.07);
}
15% {
    transform: scale(0.93, 1.07);
}
    
```



```

16% {
  transform: scale(0.94, 1.06);
}
17% {
  transform: scale(0.94, 1.06);
}
18% {
  transform: scale(0.95, 1.05);
}
19% {
  transform: scale(0.96, 1.04);
}
20% {
  transform: scale(0.98, 1.02);
}
21% {
  transform: scale(0.99, 1.01);
}
22% {
  transform: scale(1, 1);
}
23% {
  transform: scale(1, 1);
}
24% {
  transform: scale(1.01, 0.99);
}
25% {
  transform: scale(1.02, 0.98);
}
26% {
  transform: scale(1.02, 0.98);
}
27% {
  transform: scale(1.02, 0.98);
}
28% {
  transform: scale(1.03, 0.97);
}
29% {
  transform: scale(1.03, 0.97);
}

```



```

30% {
  transform: scale(1.02, 0.98);
}
31% {
  transform: scale(1.02, 0.98);
}
32% {
  transform: scale(1.02, 0.98);
}
33% {
  transform: scale(1.02, 0.98);
}
34% {
  transform: scale(1.01, 0.99);
}
35% {
  transform: scale(1.01, 0.99);
}
36% {
  transform: scale(1.01, 0.99);
}
37% {
  transform: scale(1, 1);
}
38% {
  transform: scale(1, 1);
}
39% {
  transform: scale(1, 1);
}
40% {
  transform: scale(0.99, 1.01);
}
41% {
  transform: scale(0.99, 1.01);
}
42% {
  transform: scale(0.99, 1.01);
}
43% {
  transform: scale(0.99, 1.01);
}
    
```



```
44% {
    transform: scale(0.99, 1.01);
}
45% {
    transform: scale(0.99, 1.01);
}
46% {
    transform: scale(0.99, 1.01);
}
47% {
    transform: scale(0.99, 1.01);
}
48% {
    transform: scale(0.99, 1.01);
}
49% {
    transform: scale(1, 1);
}
}
```

这是一个写好的运动 keyframes 效果，包含 50 帧的宽高改变，大家复制到 css 中即可使用。不必自己编写。

(10) .middle_points 内部 html

```
<div class="middle_wrap">
    <div class="middle">
        <span class="now"></span>
        <span></span>
        <span class="last" ></span>
    </div>
</div>
```

(11) 设置样式

```
.bchange .middle_wrap{
```



```
background:url(../images/next&prev_mbg03.png) no-repeat right;
height:14px;
}
.bchange .middle{
background:url(../images/next&prev_mbg02.png) repeat -x;
height:14px;
margin:0 6px;
}
.bchange .middle span{
display:block;
float:left;
width:6px;
height:6px;
margin-right:7px;
margin-top:4px;
background:url(../images/next&prev_others.png);
cursor:pointer;
}
.bchange .middle span.last {
margin-right:0px;
}
.bchange .middle .now{
background:url(../images/next&prev_now.png);
}
```

.middle_wra 设置高度，背景为右边的圆角。

.middle 设置高度和 margin 值，背景。

里面的小点儿标签 span，设置响应样式，最后一个 span.last 让右边额 margin 值清零 (说明：对于一般浏览器，可使用 last-child 选择器设置，但在 ie8 及以下是不管用的，为了兼容处理，我们这里采用了对最后一个 span 使用独立的 class 设置)。当前个 .now 设置不同的背景。

轮播图的 html，css 编写结束。

3. 图片的运动效果

(1) 在 index.js 的 \$(function() {}) 中写入代码，获取 banner 全部的 DOM

```
(function(){
```



```
var oDiv = $('.banner_wrap');
var oPrev = oDiv.find(".prev");
var oNext = oDiv.find(".next");
var aSpan = oDiv.find(".middle span");
var animateFater = null;
var nowIndex = 0;
//本效果后续代码写在此处
})();
```

为了避免和其他代码之间变量的影响，这里将轮播的 js 写成函数自执行的形式

```
(function() {
//此处执行代码

})();
```

oDiv 是最外面的 banner 包裹层 DOM

oPrev 是前一页 DOM

oNext 是下一页 DOM

aSpan 是中间显示位置的小点 DOM

animateFater 代表需要执行 css3 动画的父级，这里先声明，以后再赋值

nowIndex 代表当前要显示的 banner 的 index ，初始值为 0；

(2) 执行第一个 banner 进入时的运动

```
animateFater = $(".banner_one").eq(0);
animateImage();

//图片运动的方法
function animateImage(){
    animateFater.find(".image01").show().addClass("animated
fadeInLeft");
    setTimeout(function(){
        animateFater.find(".image02").show().addClass("animated
bounceInRight");
        animateFater.find(".image03").show().addClass("animated
fadeIn");
    },300);
}
```



animateFater 赋值为第一个 banner

调用执行一下图片运动方法 animateImage();

下面编写图片运动方法:

```
teFater.find(".image01").show().addClass("animated fadeInLeft");
```

表示找到里面的第一个图片, 显示, 并添加 css3 运动类 (此类为 animate.css 中设置好的运动类, 只要添加类, 就能执行运动)

setTimeout(function() {}, 300); 的意义为, 等第一个图片动画开始执行 300 毫秒以后, 在执行 image02 和 image03 的动画, 这样让动画显得更有层次。

4. 点击切换效果

(1) 点击下一页时的 js

```
oNext.click(function(){
    if( nowIndex < $(".banner_change .middle span").length-1 ){
        nowIndex++;
    }else{
        nowIndex = 0;
    }
    bannerAnimate();
});
```

这里主要设置 nowIndex 的值, 如果没到最后一个, 就让 nowIndex 自加一, 如果到了, 就执行 else 里代码, 让 nowIndex 变为第一个, 以实现循环。

然后执行 banner 切换的方法, bannerAnimate 下一步的时候再编写。

(2) bannerAnimate 方法

```
function bannerAnimate(){
    $(".banner_change .middle span").removeClass("now");
    $(".banner_change .middle span").eq(nowIndex).addClass("now");
    $(".banner_one").fadeOut(200);
    animateFater = $(".banner_one").eq(nowIndex);
    animateFater.fadeIn(200);
    animateImage()
```



```
}
```

- 1、span 小点的当前项，设置当前项。now
- 2、隐藏 banner，显示需要显示的 banner，并把需要显示的 banner 赋值给 animateFater，以便后面的 animateImage() 方法使用，最后调用 animateImage() 实现中间图片的运动效果。

(3) 点击上一页的 js

```
oPrev.click(function(){
    if( nowIndex >0 ){
        nowIndex--;
    }else{
        nowIndex = $(".banner_change .middle span").length-1
    }
    bannerAnimate();
});
```

和点击下一页类似。当没到第一页时，nowIndex 自减，当小于第一页时，让 nowIndex 设置为最后一页。

(4) 点击中间的切换小点 js

```
aSpan.click(function(){
    nowIndex = $(this).index();
    bannerAnimate();
})
```

将当前 span 的 index 赋值给 nowIndex，执行 banner 运动函数即可。

Banner 的 js 运动效果编写完毕！



1.4 参考资料

无

1.5 扩展练习

教师自行补充