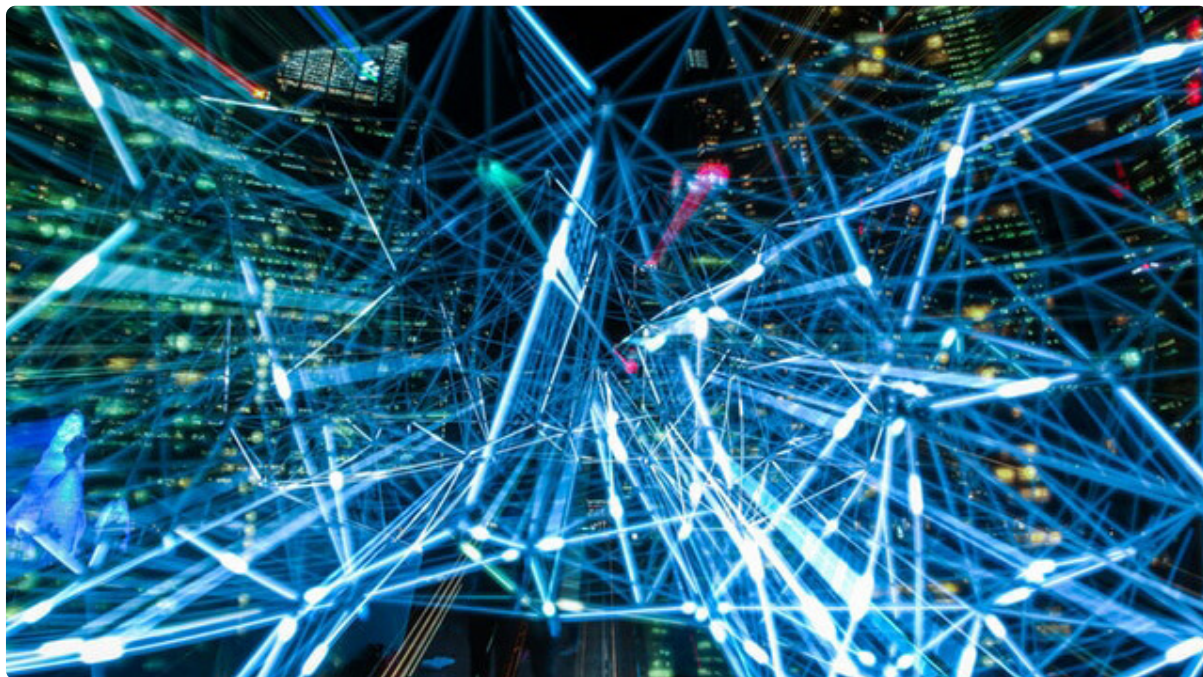


39 列表页的设计与开发

更新时间：2019-08-26 17:04:51



“

我们活着不能与草木同腐，不能醉生梦死，枉度人生，要有所作为。

——方志敏

”

这一节我们来实现列表页，这个页面的作用就是对同一类的多个信息进行展示。以我们这个网站用到的列表页为例，他就是从某一个角度来展示一系列文章用的。这个角度可以通过关键词搜索的相关文章，也可能是官方推荐的一系列文章，还可以是某个专题分类下的文章。虽然划分的角度不一样，但对于列表页来说需要做的是没有太大差异的。下面就是我们这一节要实现的列表页的效果：



这个列表页分为四部分内容，最上面是和首页相同的搜索功能，下面是用来排序的两个按钮，再下面就是列表的内容区，最后也是和首页相同的底部导航。下来我们就实现一下这个列表页的效果。

文件的建立

首先建立 `/page/list.html` 文件。因为搜索和底部导航条我们在首页的实现时已经用过一次，所以这里直接把这两个组件随着文件一起建立出来，只需要把导航选中的样式和导航链接修改一下即可。这个 `HTML` 文件的基础结构如下：

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,initial-scale=1.0,user-scalable=no">
    <link rel="stylesheet" href="../../dist/tuitui-ui.min.css">
    <link rel="stylesheet" href="/list.css">
    <title>推推优品-发现</title>
  </head>
  <body class="page-list">
    <!-- 底部导航 -->
    <div class="tt-navbar">
      <a class="navbar-item" href="/index.html">
        <i class="fa fa-home icon"></i>
        <span class="name">首页</span>
      </a>
      <a class="navbar-item active">
        <i class="fa fa-search icon"></i>
        <span class="name">发现</span>
      </a>
      <a class="navbar-item" href="/mine.html">
        <i class="fa fa-user-o icon"></i>
        <span class="name">我的</span>
      </a>
    </div>
    <!-- 内容区 -->
    <div class="tt-content">
      <!-- 带suggest的搜索框 -->
      <div class="tt-search">
        <!-- 搜索框 -->
        <form class="tt-search-form" action="#">
          <div class="tt-search-input-wrap">
            <i class="fa fa-search tt-search-icon"></i>
            <input type="text" class="tt-search-input" placeholder="搜索" autocomplete="off" required/>
            <i class="fa fa-close tt-search-clear"></i>
          </div>
          <span class="tt-search-cancel">取消</span>
        </form>
        <!-- 搜索建议 -->
        <ul class="tt-search-suggest">
          <li class="tt-suggest-item">手机</li>
          <li class="tt-suggest-item">iPhone XS Max</li>
          <li class="tt-suggest-item">华为P30</li>
          <li class="tt-suggest-item">小米 MIX3</li>
          <li class="tt-suggest-item">诺基亚1110</li>
        </ul>
      </div>
    </div>
  </body>
</html>

```

下来再创建一个空的 CSS 文件 /page/list.css:

```

/*
 * @Author: Rosen
 * @Date: 2019-08-25 10:30:04
 * @Last Modified by: Rosen
 * @Last Modified time: 2019-08-25 12:07:13
 */

/* 列表页样式 */

```

这样基础文件就建立好了，我们下来要实现的就是排序的两个按钮和列表两部分内容了。

排序按钮的实现

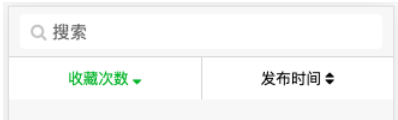
这里要实现的排序按钮是页面里比较个性的样式，所以我们要自己去实现，但是两个按钮可以以样式库里的按钮样式作为基础。我们先在 **HTML** 文件中建立出这个排序按钮的结构，在搜索组件后面追加下面的代码：

```
<!-- 排序功能区 -->
<div class="sort-wrap">
  <a class="tt-btn sort-btn active">
    收藏次数
    <!-- 没有排序时使用的图标 -->
    <!-- <i class="fa fa-sort"></i> -->
    <!-- 升序排列时使用的图标 -->
    <!-- <i class="fa fa-sort-asc"></i> -->
    <!-- 降序排列时使用的图标 -->
    <i class="fa fa-sort-desc"></i>
  </a>
  <a class="tt-btn sort-btn">
    发布时间
    <i class="fa fa-sort"></i>
  </a>
</div>
```

这里的结构实际上就是一个容器里装了两个按钮，每个按钮里有个表示排序方式的图标。这里在第一个按钮里把三种排序方式要用到的图标都给了出来，这样 **JS** 工程师在修改按钮状态的时候，就会更方便了。另外最外层会有一个 **active** 类，用来控制按钮是不是在激活状态。这些样式也都比较简单了，应该可以直接写代码了：

```
/* 排序按钮的容器 */
.page-list .sort-wrap{
  display: flex;
}
/* 排序按钮 */
.page-list .sort-wrap > .sort-btn{
  flex: 1;
  border-radius: 0;
  border: none;
  border-top: 1px solid #ddd;
  background: #fff;
}
/* 激活状态下的排序按钮 */
.page-list .sort-wrap > .sort-btn.active{
  color: #09bb07;
}
.page-list .sort-wrap > .sort-btn + .sort-btn{
  border-left: 1px solid #ddd;
}
```

这样排序按钮的样式就实现了：



这里要注意的是，我们需要覆盖样式库里的组件样式的时候，最好在组件上新添加一个类进行样式覆盖，而不是直接使用样式库里类的命名，这样能减少同一个页面内因多处使用同一个组件而造成的样式冲突。

列表的实现

这里的列表会用到复杂列表组件，和首页里的复杂列表是一样的。不过在列表页，是会有分页的，所以要在这个列表里加上加载相关的提示信息。这里我们就还会用到加载提示的组件，包括加载中提示、下拉加载更多提示和没有更多数据的提示。这里把两种组件的代码直接放在刚完成的排序功能区的后面即可，在 **tt-content** 的最后添加下列代码：

```

<!-- 列表内容 -->
<div class="tt-panel-body no-padding">
  <div class="tt-loading-inline">
    <span class="tt-loading-info">
      <i class="fa fa-circle-o-notch fa-spin tt-loading-icon"></i>
      用力加载中
    </span>
  </div>
  <!-- 文章列表 -->
  <div class="tt-list">
    <a href="/article.html" class="tt-list-item">
      <div class="item-img-wrap">
        
      </div>
      <div class="item-content-wrap">
        <h1 class="item-title">大美新疆 | 玩遍这片大好山河，这些装备必不可少</h1>
        <p class="item-desc">发布时间：2019-8-18</p>
        <p class="item-desc">阅读1923次 | 收藏398次</p>
      </div>
    </a>
    <a href="/article.html" class="tt-list-item">
      <div class="item-img-wrap">
        
      </div>
      <div class="item-content-wrap">
        <h1 class="item-title">大美新疆 | 玩遍这片大好山河，这些装备必不可少</h1>
        <p class="item-desc">发布时间：2019-8-18</p>
        <p class="item-desc">阅读1923次 | 收藏398次</p>
      </div>
    </a>
    <a href="/article.html" class="tt-list-item">
      <div class="item-img-wrap">
        
      </div>
      <div class="item-content-wrap">
        <h1 class="item-title">大美新疆 | 玩遍这片大好山河，这些装备必不可少</h1>
        <p class="item-desc">发布时间：2019-8-18</p>
        <p class="item-desc">阅读1923次 | 收藏398次</p>
      </div>
    </a>
  </div>
  <div class="tt-loading-inline">
    <span class="tt-loading-info">
      上拉加载更多
      <i class="fa fa-long-arrow-up fa-vibrate-y tt-loading-icon"></i>
    </span>
  </div>
  <div class="tt-loading-inline">
    <span class="tt-loading-info">
      我可是有底线的
    </span>
  </div>
</div>

```

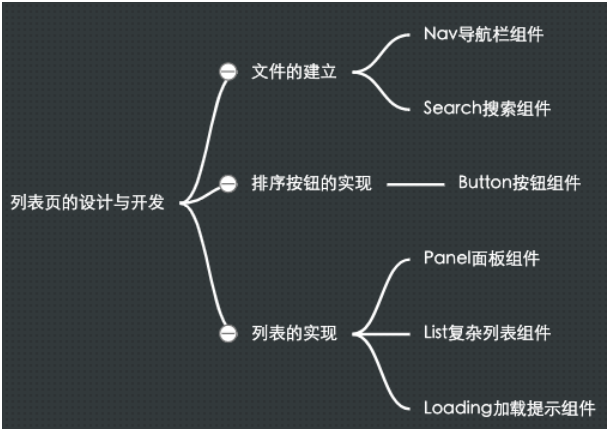
这里我们用 **panel** 组件套住了整个列表相关的内容，列表里每一个项都链接到了 **detail.html** 页面，是我们下一节将要实现的文章详情页。这段代码添加上以后，页面的样式就是下面这样：



在做页面切图的时候，是可以直接把这样的样式直接交给 JS 工程师的。有些组件在不遮挡其他内容的时候，我们可以不用隐藏掉，这样 JS 工程师在拿到 HTML 和 CSS 文件的时候，会更容易找到各个组件的位置。

结语

到这里我们列表页的样式也完成了。这一节需要单独实现的是一个排序的功能区，其他的样式都可以直接从样式库里取得。这一节的内容结构如下：



我们这一节的内容就到这里，同学们可以访问【[列表页在线预览](#)】来查看这一节的演示效果。下一节进行文章详情页的设计和开发。

}

