# JavaWeb

## day26\_综合案例2

### 0.1学习目标

* 能够完成分类数据展示功能
* 能够完成分页展示\_分页数据展示功能

### 0.2 学习指南

* 完成baseServlet抽取
* 完成分类数据展示
* 完成旅游线路分页展示

### 第一堂课

本节知识点

1.1BaseServlet抽取

1.2BaseServlet抽取\_UserServlet&页面路径改写

本节目标

完成BaseServlet抽取以及页面路径改写

#### 项目导入

##### 1.1.1知识概述

public class BaseServlet extends HttpServlet {

@Override

protected void service(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

//System.out.println("baseServlet的service方法被执行了...");

//完成方法分发

//1.获取请求路径

String uri = req.getRequestURI(); // /travel/user/add

System.out.println("请求uri:"+uri);// /travel/user/add

//2.获取方法名称

String methodName = uri.substring(uri.lastIndexOf('/') + 1);

System.out.println("方法名称："+methodName);

//3.获取方法对象Method

//谁调用我？我代表谁

System.out.println(this);//UserServlet的对象cn.itcast.travel.web.servlet.UserServlet@4903d97e

try {

//获取方法

Method method = this.getClass().getMethod(methodName, HttpServletRequest.class, HttpServletResponse.class);

//4.执行方法

//暴力反射

//method.setAccessible(true);

method.invoke(this,req,resp);

} catch (NoSuchMethodException e) {

e.printStackTrace();

} catch (IllegalAccessException e) {

e.printStackTrace();

} catch (InvocationTargetException e) {

e.printStackTrace();

}

}

/\*\*

\* 直接将传入的对象序列化为json，并且写回客户端

\* @param obj

\*/

public void writeValue(Object obj,HttpServletResponse response) throws IOException {

ObjectMapper mapper = new ObjectMapper();

response.setContentType("application/json;charset=utf-8");

mapper.writeValue(response.getOutputStream(),obj);

}

/\*\*

\* 将传入的对象序列化为json，返回

\* @param obj

\* @return

\*/

public String writeValueAsString(Object obj) throws JsonProcessingException {

ObjectMapper mapper = new ObjectMapper();

return mapper.writeValueAsString(obj);

}

##### 1.1.2视频详情



##### 1.1.3总结与补充

无

##### 1.1.4课堂提问与练习

无

##### 1.1.5习题答案

无

##### 1.1.6视频缺陷

无

##### 1.1.7视频扩展

无

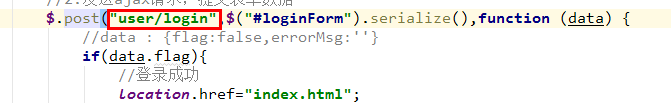
#### 1.2BaseServlet抽取\_UserServlet&页面路径改写

##### 1.2.1知识概述

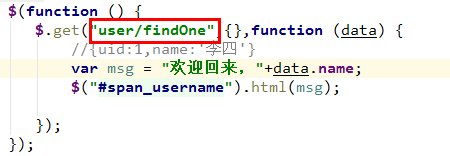
register.html



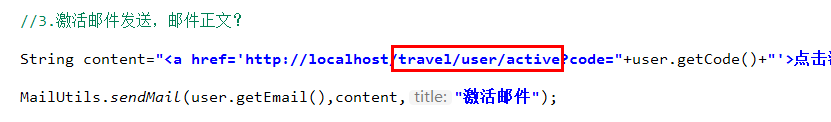
login.html



header.html



UserServiceImpl发送邮件



##### 1.2.2视频详情



##### 1.2.3总结与补充

为了避免端口冲突尽量不要使用8080端口

##### 1.2.4课堂提问与练习

无

##### 1.2.5习题答案

无

##### 1.2.6视频缺陷

无

##### 1.2.7视频扩展

无

### 第二堂课

本节知识点：

2.1分类数据展示功能\_分析

2.2分类数据展示功能\_实现\_后台代码

2.3分类数据展示功能\_实现\_前台代码

2.4分类数据展示功能\_缓存优化\_分析

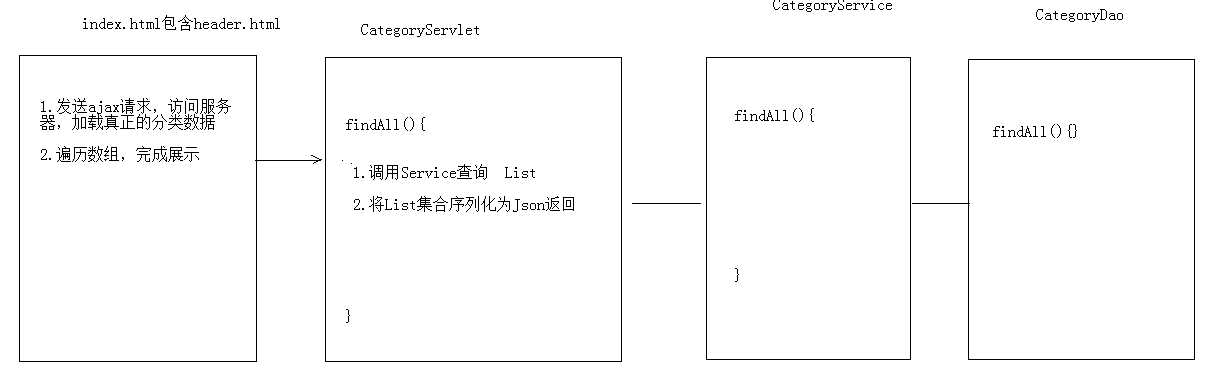
目标：

1、能够完成分类数据展示前后台代码

2、能够理解缓存优化

#### 分类数据展示功能\_分析

##### 2.1.1知识概述



##### 2.1.2视频详情



##### 2.1.3总结与补充

##### 2.1.4课堂提问与练习

无

##### 2.1.5习题答案

无

##### 2.1.6视频缺陷

无

##### 2.1.7视频扩展

无

#### 2.2分类数据展示功能\_实现\_后台代码

##### 2.2.1知识概述

CategoryServlet

@WebServlet(**"/category/\*"**)  
**public class** CategoryServlet **extends** BaseServlet {  
  
 **private** CategoryService **service** = **new** CategoryServiceImpl();  
  
 /\*\*  
 \* 查询所有  
 \* **@param *request*** \* **@param *response*** \* **@throws** ServletException  
 \* **@throws** IOException  
 \*/  
 **public void** findAll(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {  
 //1.调用service查询所有  
 List<Category> cs = **service**.findAll();  
 //2.序列化json返回  
 /\* ObjectMapper mapper = new ObjectMapper();  
 response.setContentType("application/json;charset=utf-8");  
 mapper.writeValue(response.getOutputStream(),cs);\*/  
 writeValue(cs,response);  
  
 }  
  
}

CategoryService

**public class** CategoryServiceImpl **implements** CategoryService {  
  
 **private** CategoryDao **categoryDao** = **new** CategoryDaoImpl();  
  
 @Override  
 **public** List<Category> findAll() {  
 **return categoryDao**.findAll();  
 }  
}

CategoryDao

**public class** CategoryDaoImpl **implements** CategoryDao {  
  
 **private** JdbcTemplate **template** = **new** JdbcTemplate(JDBCUtils.*getDataSource*());  
  
 @Override  
 **public** List<Category> findAll() {  
 String sql = **"select** *\** **from tab\_category "**;  
 **return template**.query(sql,**new** BeanPropertyRowMapper<Category>(Category.**class**));  
 }  
}

在BaseServlet中封装了序列化json的方法

/\*\*  
 \* 直接将传入的对象序列化为json，并且写回客户端  
 \* **@param *obj*** \*/  
**public void** writeValue(Object obj,HttpServletResponse response) **throws** IOException {  
 ObjectMapper mapper = **new** ObjectMapper();  
 response.setContentType(**"application/json;charset=utf-8"**);  
 mapper.writeValue(response.getOutputStream(),obj);  
}  
  
/\*\*  
 \* 将传入的对象序列化为json，返回  
 \* **@param *obj*** \* **@return** \*/  
**public** String writeValueAsString(Object obj) **throws** JsonProcessingException {  
 ObjectMapper mapper = **new** ObjectMapper();  
 **return** mapper.writeValueAsString(obj);  
}

##### 2.2.2视频详情



##### 2.2.3总结与补充

##### 2.2.4课堂提问与练习

无

##### 2.2.5习题答案

无

##### 2.2.6视频缺陷

无

##### 2.2.7视频扩展

无

#### 2.3分类数据展示功能\_实现\_前台代码

##### 2.3.1知识概述

hader.html加载后，发送ajax请求，请求category/findAll

//查询分类数据  
 **$**.get(**"category/findAll"**,{},**function** (data) {  
 //[{cid:1,cname:国内游},{},{}]  
 **var** lis = **'<li class="nav-active"><a href="index.html">首页</a></li>'**;  
 //遍历数组,拼接字符串(<li>)  
 **for** (**var** i = 0; i < data.**length**; i++) {  
 **var** li = **'<li><a href="route\_list.html">'**+data[i].cname+**'</a></li>'**;  
  
 lis += li;  
   
 }  
 //拼接收藏排行榜的li,<li><a href="favoriterank.html">收藏排行榜</a></li>  
   
 lis+= **'<li><a href="favoriterank.html">收藏排行榜</a></li>'**;  
  
 //将lis字符串，设置到ul的html内容中  
 **$**(**"#category"**).html(lis);  
 });

##### 2.3.2视频详情



##### 2.3.3总结与补充

##### 2.3.4课堂提问与练习

无

##### 2.3.5习题答案

无

##### 2.3.6视频缺陷

无

##### 2.3.7视频扩展

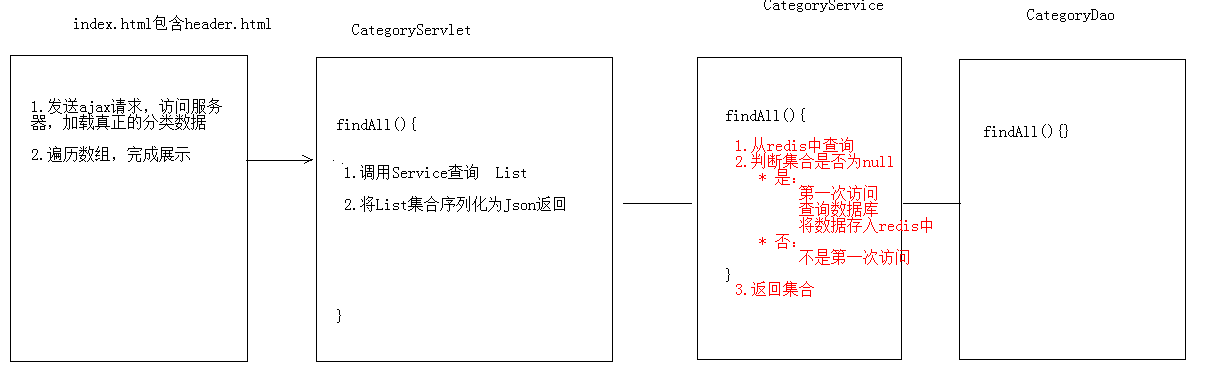
无

#### 2.4分类数据展示功能\_缓存优化\_分析

##### 2.4.1知识概述

分析发现，分类的数据在每一次页面加载后都会重新请求数据库来加载，对数据库的压力比较大，而且分类的数据不会经常产生变化，所有可以使用redis来缓存这个数据。

分析：



##### 2.4.2视频详情



##### 2.4.3总结与补充

##### 2.4.4课堂提问与练习

无

##### 2.4.5习题答案

无

##### 2.4.6视频缺陷

无

##### 2.3.7视频扩展

无

### 第三堂课

本节知识点

3.1分类数据展示功能\_缓存优化\_代码实现

3.2旅游线路分页展示\_类别id传递

本节目标

1. 完成缓存优化代码
2. 完成旅游线路分页展示\_类别id传递

#### 3.1分类数据展示功能\_缓存优化\_代码实现

##### 3.1.1知识概述

   期望数据中存储的顺序就是将来展示的顺序，使用redis的sortedset

@Override  
**public** List<Category> findAll() {  
 //1.从redis中查询  
 //1.1获取jedis客户端  
 Jedis jedis = JedisUtil.*getJedis*();  
 //1.2可使用sortedset排序查询  
 Set<String> categorys = jedis.zrange(**"category"**, 0, -1);  
 List<Category> cs = **null**;  
 //2.判断查询的集合是否为空  
 **if** (categorys == **null** || categorys.size() == 0) {  
  
 System.***out***.println(**"从数据库查询...."**);  
 //3.如果为空,需要从数据库查询,在将数据存入redis  
 //3.1 从数据库查询  
 cs = **categoryDao**.findAll();  
 //3.2 将集合数据存储到redis中的 category的key  
 **for** (**int** i = 0; i < cs.size(); i++) {  
  
 jedis.zadd(**"category"**, cs.get(i).getCid(), cs.get(i).getCname());  
 }  
 } **else** {  
 System.***out***.println(**"从redis中查询....."**);  
  
 //4.如果不为空,将set的数据存入list  
 cs = **new** ArrayList<Category>();  
 **for** (String name : categorys) {  
 Category category = **new** Category();  
 category.setCname(name);  
 cs.add(category);  
  
 }  
 }  
  
  
 **return** cs;  
}

##### 3.1.2视频详情



##### 3.1.3总结与补充

##### 3.1.4课堂提问与练习

##### 3.1.5习题答案

##### 3.1.6视频缺陷

无

##### 3.1.7视频扩展

无

#### 3.2旅游线路分页展示\_类别id传递

##### 3.2.1知识概述

Redis中查询score（cid）

**public class** CategoryServiceImpl **implements** CategoryService {  
  
 **private** CategoryDao **categoryDao** = **new** CategoryDaoImpl();  
  
 @Override  
 **public** List<Category> findAll() {  
 //1.从redis中查询  
 //1.1获取jedis客户端  
 Jedis jedis = JedisUtil.*getJedis*();  
 //1.2可使用sortedset排序查询  
 //Set<String> categorys = jedis.zrange("category", 0, -1);  
 //1.3查询sortedset中的分数(cid)和值(cname)  
 Set<Tuple> categorys = jedis.zrangeWithScores(**"category"**, 0, -1);  
  
 List<Category> cs = **null**;  
 //2.判断查询的集合是否为空  
 **if** (categorys == **null** || categorys.size() == 0) {  
  
 System.***out***.println(**"从数据库查询...."**);  
 //3.如果为空,需要从数据库查询,在将数据存入redis  
 //3.1 从数据库查询  
 cs = **categoryDao**.findAll();  
 //3.2 将集合数据存储到redis中的 category的key  
 **for** (**int** i = 0; i < cs.size(); i++) {  
  
 jedis.zadd(**"category"**, cs.get(i).getCid(), cs.get(i).getCname());  
 }  
 } **else** {  
 System.***out***.println(**"从redis中查询....."**);  
  
 //4.如果不为空,将set的数据存入list  
 cs = **new** ArrayList<Category>();  
 **for** (Tuple tuple : categorys) {  
 Category category = **new** Category();  
 category.setCname(tuple.getElement());  
 category.setCid((**int**)tuple.getScore());  
 cs.add(category);  
  
 }  
 }  
  
  
 **return** cs;  
 }  
}

页面传递cid

header.html传递cid

**var** li = **'<li><a href="route\_list.html?cid='**+data[i].cid+**'">'**+data[i].cname+**'</a></li>'**;

获取cid

**$**(**function** () {  
 **var** search = ***location***.search;  
 //alert(search);//?id=5  
 // 切割字符串，拿到第二个值  
 **var** cid = search.split(**"="**)[1];  
});

##### 3.2.2视频详情



##### 3.2.3总结与补充

##### 3.2.4课堂提问与练习

无

##### 3.2.5习题答案

无

##### 3.2.6视频缺陷

无

##### 3.2.7视频扩展

无

### 第四堂课

本节知识点：

4.1旅游线路分页展示\_分页数据展示\_分析

4.2旅游线路分页展示\_分页数据展示\_代码实现\_Servlet

4.3旅游线路分页展示\_分页数据展示\_代码实现\_Dao

4.4旅游线路分页展示\_分页数据展示\_代码实现\_Service

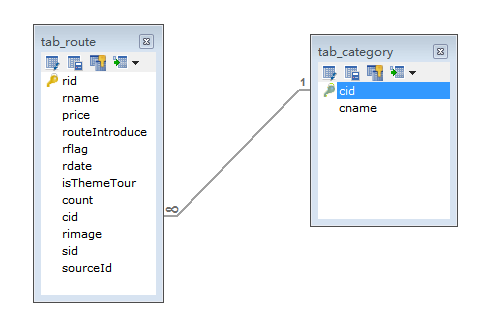
本节目标 ：

完成旅游线路分页数据展示

#### 4.1旅游线路分页展示\_分页数据展示\_分析

##### 4.1.1知识概述

点击了不同的分类后，将来看到的旅游线路不一样的。通过分析数据库表结构，发现，旅游线路表和分类表时一个多对一的关系



查询不同分类的旅游线路sql

Select \* from tab\_route where cid = ?;

##### 4.1.2视频详情



##### 4.1.3总结与补充

##### 4.1.4课堂提问与练习

无

##### 4.1.5习题答案

无

##### 4.1.6视频缺陷

无

##### 4.1.7视频扩展

无

#### 4.2旅游线路分页展示\_分页数据展示\_代码实现\_Servlet

##### 4.2.1知识概述

* 1. RouteServlet

@WebServlet(**"/route/\*"**)  
**public class** RouteServlet **extends** BaseServlet {  
  
 **private** RouteService **routeService** = **new** RouteServiceImpl();  
  
 /\*\*  
 \* 分页查询  
 \* **@param *request*** \* **@param *response*** \* **@throws** ServletException  
 \* **@throws** IOException  
 \*/  
 **public void** pageQuery(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {  
 //1.接受参数  
 String currentPageStr = request.getParameter(**"currentPage"**);  
 String pageSizeStr = request.getParameter(**"pageSize"**);  
 String cidStr = request.getParameter(**"cid"**);  
  
 **int** cid = 0;//类别id  
 //2.处理参数  
 **if**(cidStr != **null** && cidStr.length() > 0){  
 cid = Integer.*parseInt*(cidStr);  
 }  
 **int** currentPage = 0;//当前页码，如果不传递，则默认为第一页  
 **if**(currentPageStr != **null** && currentPageStr.length() > 0){  
 currentPage = Integer.*parseInt*(currentPageStr);  
 }**else**{  
 currentPage = 1;  
 }  
  
 **int** pageSize = 0;//每页显示条数，如果不传递，默认每页显示5条记录  
 **if**(pageSizeStr != **null** && pageSizeStr.length() > 0){  
 pageSize = Integer.*parseInt*(pageSizeStr);  
 }**else**{  
 pageSize = 5;  
 }  
  
 //3. 调用service查询PageBean对象  
 PageBean<Route> pb = **routeService**.pageQuery(cid, currentPage, pageSize);  
  
 //4. 将pageBean对象序列化为json，返回  
 writeValue(pb,response);  
  
 }  
  
}

##### 4.2.2视频详情



##### 4.2.3总结与补充

##### 4.2.4课堂提问与练习

##### 4.2.5习题答案

##### 4.2.6视频缺陷

无

##### 4.2.7视频扩展

无

#### 4.3 旅游线路分页展示\_分页数据展示\_代码实现\_Dao

##### 4.3.1知识概述

1. RouteDao

**public class** RouteDaoImpl **implements** RouteDao {  
 **private** JdbcTemplate **template** = **new** JdbcTemplate(JDBCUtils.*getDataSource*());  
  
 @Override  
 **public int** findTotalCount(**int** cid) {  
 String sql = **"select** *count***(***\****) from tab\_route where cid = ?"**;  
 **return template**.queryForObject(sql,Integer.**class**,cid);  
 }  
  
 @Override  
 **public** List<Route> findByPage(**int** cid, **int** start, **int** pageSize) {  
 String sql = **"select** *\** **from tab\_route where cid = ? limit ? , ?"**;  
  
 **return template**.query(sql,**new** BeanPropertyRowMapper<Route>(Route.**class**),cid,start,pageSize);  
 }  
}

##### 4.3.2视频详情



##### 4.3.3总结与补充

##### 4.3.4课堂提问与练习

无

。

##### 4.3.5习题答案

无

##### 4.3.6视频缺陷

无

##### 4.3.7视频扩展

无

#### 4.4旅游线路分页展示\_分页数据展示\_代码实现\_Service

##### 4.4.1知识概述

1. RouteService

**public class** RouteServiceImpl **implements** RouteService {  
 **private** RouteDao **routeDao** = **new** RouteDaoImpl();  
 @Override  
 **public** PageBean<Route> pageQuery(**int** cid, **int** currentPage, **int** pageSize) {  
 //封装PageBean  
 PageBean<Route> pb = **new** PageBean<Route>();  
 //设置当前页码  
 pb.setCurrentPage(currentPage);  
 //设置每页显示条数  
 pb.setPageSize(pageSize);  
   
 //设置总记录数  
 **int** totalCount = **routeDao**.findTotalCount(cid);  
 pb.setTotalCount(totalCount);  
 //设置当前页显示的数据集合  
 **int** start = (currentPage - 1) \* pageSize;//开始的记录数  
 List<Route> list = **routeDao**.findByPage(cid,start,pageSize);  
 pb.setList(list);  
  
 //设置总页数 = 总记录数/每页显示条数  
 **int** totalPage = totalCount % pageSize == 0 ? totalCount / pageSize :(totalCount / pageSize) + 1 ;  
 pb.setTotalPage(totalPage);  
  
  
 **return** pb;  
 }  
}

##### 4.4.2视频详情



##### 4.4.3总结与补充

##### 4.4.4课堂提问与练习

无

。

##### 4.4.5习题答案

无

##### 4.4.6视频缺陷

无

##### 4.4.7视频扩展

无

### 第五堂课

本节知识点：

5.1旅游线路分页展示\_分页数据展示\_前台代码\_页码展示

5.2旅游线路分页展示\_分页数据展示\_前台代码\_数据展示

本节目标 ：

实现旅游线路分页展示\_分页数据展示\_前台代码\_页码与数据展示

#### 旅游线路分页展示\_分页数据展示\_前台代码\_页码展示

##### 5.1.1知识概述

**$**(**function** () {  
 /\* var search = location.search;  
 //alert(search);//?id=5  
 // 切割字符串，拿到第二个值  
 var cid = search.split("=")[1];\*/  
 //获取cid的参数值  
 **var** cid = *getParameter*(**"cid"**);  
 //获取rname的参数值  
 **var** rname = *getParameter*(**"rname"**);  
 //判断rname如果不为null或者""  
 **if**(rname){  
 //url解码  
 rname = ***window***.*decodeURIComponent*(rname);  
 }  
  
 //当页码加载完成后，调用load方法，发送ajax请求加载数据  
 *load*(cid,**null**,rname);  
});  
  
**function** *load*(cid ,currentPage,rname){  
 //发送ajax请求，请求route/pageQuery,传递cid  
 **$**.get(**"route/pageQuery"**,{**cid**:cid,**currentPage**:currentPage,**rname**:rname},**function** (pb) {  
 //解析pagebean数据，展示到页面上  
  
 //1.分页工具条数据展示  
 //1.1 展示总页码和总记录数  
 **$**(**"#totalPage"**).html(pb.totalPage);  
 **$**(**"#totalCount"**).html(pb.totalCount);  
  
 /\*  
 <li><a href="">首页</a></li>  
 <li class="threeword"><a href="#">上一页</a></li>  
 <li class="curPage"><a href="#">1</a></li>  
 <li><a href="#">2</a></li>  
 <li><a href="#">3</a></li>  
 <li><a href="#">4</a></li>  
 <li><a href="#">5</a></li>  
 <li><a href="#">6</a></li>  
 <li><a href="#">7</a></li>  
 <li><a href="#">8</a></li>  
 <li><a href="#">9</a></li>  
 <li><a href="#">10</a></li>  
 <li class="threeword"><a href="javascript:;">下一页</a></li>  
 <li class="threeword"><a href="javascript:;">末页</a></li>  
  
  
 \*/  
 **var** lis = **""**;  
  
 **var** fristPage = **'<li onclick="javascipt:***load***('**+cid+**',1,\''**+rname+**'\')"><a href="javascript:void(0)">首页</a></li>'**;  
  
 //计算上一页的页码  
 **var** beforeNum = pb.**currentPage** - 1;  
 **if**(beforeNum <= 0){  
 beforeNum = 1;  
 }  
  
 **var** beforePage = **'<li onclick="javascipt:***load***('**+cid+**','**+beforeNum+**',\''**+rname+**'\')" class="threeword"><a href="javascript:void(0)">上一页</a></li>'**;  
  
 lis += fristPage;  
 lis += beforePage;  
 //1.2 展示分页页码  
 /\*  
 1.一共展示10个页码，能够达到前5后4的效果  
 2.如果前边不够5个，后边补齐10个  
 3.如果后边不足4个，前边补齐10个  
 \*/  
  
 // 定义开始位置begin,结束位置 end  
 **var** begin; // 开始位置  
 **var** end ; // 结束位置  
  
  
 //1.要显示10个页码  
 **if**(pb.totalPage < 10){  
 //总页码不够10页  
  
 begin = 1;  
 end = pb.totalPage;  
 }**else**{  
 //总页码超过10页  
  
 begin = pb.**currentPage** - 5 ;  
 end = pb.**currentPage** + 4 ;  
  
 //2.如果前边不够5个，后边补齐10个  
 **if**(begin < 1){  
 begin = 1;  
 end = begin + 9;  
 }  
  
 //3.如果后边不足4个，前边补齐10个  
 **if**(end > pb.totalPage){  
 end = pb.totalPage;  
 begin = end - 9 ;  
 }  
 }  
  
  
 **for** (**var** i = begin; i <= end ; i++) {  
 **var** li;  
 //判断当前页码是否等于i  
 **if**(pb.**currentPage** == i){  
  
 li = **'<li class="curPage" onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
  
 }**else**{  
 //创建页码的li  
 li = **'<li onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
 }  
 //拼接字符串  
 lis += li;  
 }  
  
  
  
  
  
 /\* for (var i = 1; i <= pb.totalPage ; i++) {  
 var li;  
 //判断当前页码是否等于i  
 if(pb.currentPage == i){  
  
 li = '<li class="curPage" onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
  
 }else{  
 //创建页码的li  
 li = '<li onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
 }  
 //拼接字符串  
 lis += li;  
 }\*/  
 **var** lastPage = **'<li class="threeword"><a href="javascript:;">末页</a></li>'**;  
 **var** nextPage = **'<li class="threeword"><a href="javascript:;">下一页</a></li>'**;  
  
 lis += nextPage;  
 lis += lastPage;  
  
  
 //将lis内容设置到 ul  
 **$**(**"#pageNum"**).html(lis);  
  
  
  
 /\*  
 <li>  
 <div class="img"><img src="images/04-search\_03.jpg" alt=""></div>  
 <div class="text1">  
 <p>【减100元 含除夕/春节出发】广州增城三英温泉度假酒店/自由行套票</p>  
 <br/>  
 <p>1-2月出发，网付立享￥1099/2人起！爆款位置有限，抢完即止！</p>  
 </div>  
 <div class="price">  
 <p class="price\_num">  
 <span>&yen;</span>  
 <span>299</span>  
 <span>起</span>  
 </p>  
 <p><a href="route\_detail.html">查看详情</a></p>  
 </div>  
 </li>  
  
 \*/  
  
 //2.列表数据展示  
 **var** route\_lis = **""**;  
  
 **for** (**var** i = 0; i < pb.list.**length**; i++) {  
 //获取{rid:1,rname:"xxx"}  
 **var** route = pb.list[i];  
  
 **var** li = **'<li>\n'** +  
 **' <div class="img"><img src="'**+route.rimage+**'" style="width: 299px;"></div>\n'** +  
 **' <div class="text1">\n'** +  
 **' <p>'**+route.**rname**+**'</p>\n'** +  
 **' <br/>\n'** +  
 **' <p>'**+route.routeIntroduce+**'</p>\n'** +  
 **' </div>\n'** +  
 **' <div class="price">\n'** +  
 **' <p class="price\_num">\n'** +  
 **' <span>&yen;</span>\n'** +  
 **' <span>'**+route.price+**'</span>\n'** +  
 **' <span>起</span>\n'** +  
 **' </p>\n'** +  
 **' <p><a href="route\_detail.html">查看详情</a></p>\n'** +  
 **' </div>\n'** +  
 **' </li>'**;  
 route\_lis += li;  
 }  
 **$**(**"#route"**).html(route\_lis);  
  
 //定位到页面顶部  
 ***window***.scrollTo(0,0);  
 });  
  
}

##### 5.1.2视频详情



##### 5.1.3总结与补充

##### 5.1.4课堂提问与练习

无

##### 5.1.5习题答

无

##### 5.1.6视频缺陷

无

##### 5.1.7视频扩展

无

#### 旅游线路分页展示\_分页数据展示\_前台代码\_数据展示

##### 5.2.1知识概述

**$**(**function** () {  
 /\* var search = location.search;  
 //alert(search);//?id=5  
 // 切割字符串，拿到第二个值  
 var cid = search.split("=")[1];\*/  
 //获取cid的参数值  
 **var** cid = *getParameter*(**"cid"**);  
 //获取rname的参数值  
 **var** rname = *getParameter*(**"rname"**);  
 //判断rname如果不为null或者""  
 **if**(rname){  
 //url解码  
 rname = ***window***.*decodeURIComponent*(rname);  
 }  
  
 //当页码加载完成后，调用load方法，发送ajax请求加载数据  
 *load*(cid,**null**,rname);  
});  
  
**function** *load*(cid ,currentPage,rname){  
 //发送ajax请求，请求route/pageQuery,传递cid  
 **$**.get(**"route/pageQuery"**,{**cid**:cid,**currentPage**:currentPage,**rname**:rname},**function** (pb) {  
 //解析pagebean数据，展示到页面上  
  
 //1.分页工具条数据展示  
 //1.1 展示总页码和总记录数  
 **$**(**"#totalPage"**).html(pb.totalPage);  
 **$**(**"#totalCount"**).html(pb.totalCount);  
  
 /\*  
 <li><a href="">首页</a></li>  
 <li class="threeword"><a href="#">上一页</a></li>  
 <li class="curPage"><a href="#">1</a></li>  
 <li><a href="#">2</a></li>  
 <li><a href="#">3</a></li>  
 <li><a href="#">4</a></li>  
 <li><a href="#">5</a></li>  
 <li><a href="#">6</a></li>  
 <li><a href="#">7</a></li>  
 <li><a href="#">8</a></li>  
 <li><a href="#">9</a></li>  
 <li><a href="#">10</a></li>  
 <li class="threeword"><a href="javascript:;">下一页</a></li>  
 <li class="threeword"><a href="javascript:;">末页</a></li>  
  
  
 \*/  
 **var** lis = **""**;  
  
 **var** fristPage = **'<li onclick="javascipt:***load***('**+cid+**',1,\''**+rname+**'\')"><a href="javascript:void(0)">首页</a></li>'**;  
  
 //计算上一页的页码  
 **var** beforeNum = pb.**currentPage** - 1;  
 **if**(beforeNum <= 0){  
 beforeNum = 1;  
 }  
  
 **var** beforePage = **'<li onclick="javascipt:***load***('**+cid+**','**+beforeNum+**',\''**+rname+**'\')" class="threeword"><a href="javascript:void(0)">上一页</a></li>'**;  
  
 lis += fristPage;  
 lis += beforePage;  
 //1.2 展示分页页码  
 /\*  
 1.一共展示10个页码，能够达到前5后4的效果  
 2.如果前边不够5个，后边补齐10个  
 3.如果后边不足4个，前边补齐10个  
 \*/  
  
 // 定义开始位置begin,结束位置 end  
 **var** begin; // 开始位置  
 **var** end ; // 结束位置  
  
  
 //1.要显示10个页码  
 **if**(pb.totalPage < 10){  
 //总页码不够10页  
  
 begin = 1;  
 end = pb.totalPage;  
 }**else**{  
 //总页码超过10页  
  
 begin = pb.**currentPage** - 5 ;  
 end = pb.**currentPage** + 4 ;  
  
 //2.如果前边不够5个，后边补齐10个  
 **if**(begin < 1){  
 begin = 1;  
 end = begin + 9;  
 }  
  
 //3.如果后边不足4个，前边补齐10个  
 **if**(end > pb.totalPage){  
 end = pb.totalPage;  
 begin = end - 9 ;  
 }  
 }  
  
  
 **for** (**var** i = begin; i <= end ; i++) {  
 **var** li;  
 //判断当前页码是否等于i  
 **if**(pb.**currentPage** == i){  
  
 li = **'<li class="curPage" onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
  
 }**else**{  
 //创建页码的li  
 li = **'<li onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
 }  
 //拼接字符串  
 lis += li;  
 }  
  
  
  
  
  
 /\* for (var i = 1; i <= pb.totalPage ; i++) {  
 var li;  
 //判断当前页码是否等于i  
 if(pb.currentPage == i){  
  
 li = '<li class="curPage" onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
  
 }else{  
 //创建页码的li  
 li = '<li onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
 }  
 //拼接字符串  
 lis += li;  
 }\*/  
 **var** lastPage = **'<li class="threeword"><a href="javascript:;">末页</a></li>'**;  
 **var** nextPage = **'<li class="threeword"><a href="javascript:;">下一页</a></li>'**;  
  
 lis += nextPage;  
 lis += lastPage;  
  
  
 //将lis内容设置到 ul  
 **$**(**"#pageNum"**).html(lis);  
  
  
  
 /\*  
 <li>  
 <div class="img"><img src="images/04-search\_03.jpg" alt=""></div>  
 <div class="text1">  
 <p>【减100元 含除夕/春节出发】广州增城三英温泉度假酒店/自由行套票</p>  
 <br/>  
 <p>1-2月出发，网付立享￥1099/2人起！爆款位置有限，抢完即止！</p>  
 </div>  
 <div class="price">  
 <p class="price\_num">  
 <span>&yen;</span>  
 <span>299</span>  
 <span>起</span>  
 </p>  
 <p><a href="route\_detail.html">查看详情</a></p>  
 </div>  
 </li>  
  
 \*/  
  
 //2.列表数据展示  
 **var** route\_lis = **""**;  
  
 **for** (**var** i = 0; i < pb.list.**length**; i++) {  
 //获取{rid:1,rname:"xxx"}  
 **var** route = pb.list[i];  
  
 **var** li = **'<li>\n'** +  
 **' <div class="img"><img src="'**+route.rimage+**'" style="width: 299px;"></div>\n'** +  
 **' <div class="text1">\n'** +  
 **' <p>'**+route.**rname**+**'</p>\n'** +  
 **' <br/>\n'** +  
 **' <p>'**+route.routeIntroduce+**'</p>\n'** +  
 **' </div>\n'** +  
 **' <div class="price">\n'** +  
 **' <p class="price\_num">\n'** +  
 **' <span>&yen;</span>\n'** +  
 **' <span>'**+route.price+**'</span>\n'** +  
 **' <span>起</span>\n'** +  
 **' </p>\n'** +  
 **' <p><a href="route\_detail.html">查看详情</a></p>\n'** +  
 **' </div>\n'** +  
 **' </li>'**;  
 route\_lis += li;  
 }  
 **$**(**"#route"**).html(route\_lis);  
  
 //定位到页面顶部  
 ***window***.scrollTo(0,0);  
 });  
  
}

##### 5.2.2视频详情



##### 5.2.3总结与补充

##### 5.2.4课堂提问与练习

##### 5.2.5习题答案

##### 5.2.6视频缺陷

无

##### 5.2.7视频扩展

无

### 第六堂课

本节知识点：

6.1旅游线路分页展示\_分页数据展示\_前台代码\_异步加载数据

6.2旅游线路分页展示\_分页数据展示\_前台代码\_页码处理

本节目标 ：

1. 完成旅游线路分页展示\_分页数据展示\_前台代码\_异步加载数据与页码处理

#### 6.1旅游线路分页展示\_分页数据展示\_前台代码\_异步加载数据

##### 6.1.1知识概述

**$**.get(**"route/pageQuery"**,{**cid**:cid,**currentPage**:currentPage,**rname**:rname},**function** (pb) {  
 //解析pagebean数据，展示到页面上  
  
 //1.分页工具条数据展示  
 //1.1 展示总页码和总记录数  
 **$**(**"#totalPage"**).html(pb.totalPage);  
 **$**(**"#totalCount"**).html(pb.totalCount);  
  
 /\*  
 <li><a href="">首页</a></li>  
 <li class="threeword"><a href="#">上一页</a></li>  
 <li class="curPage"><a href="#">1</a></li>  
 <li><a href="#">2</a></li>  
 <li><a href="#">3</a></li>  
 <li><a href="#">4</a></li>  
 <li><a href="#">5</a></li>  
 <li><a href="#">6</a></li>  
 <li><a href="#">7</a></li>  
 <li><a href="#">8</a></li>  
 <li><a href="#">9</a></li>  
 <li><a href="#">10</a></li>  
 <li class="threeword"><a href="javascript:;">下一页</a></li>  
 <li class="threeword"><a href="javascript:;">末页</a></li>  
  
  
 \*/  
 **var** lis = **""**;  
  
 **var** fristPage = **'<li onclick="javascipt:***load***('**+cid+**',1,\''**+rname+**'\')"><a href="javascript:void(0)">首页</a></li>'**;  
  
 //计算上一页的页码  
 **var** beforeNum = pb.**currentPage** - 1;  
 **if**(beforeNum <= 0){  
 beforeNum = 1;  
 }  
  
 **var** beforePage = **'<li onclick="javascipt:***load***('**+cid+**','**+beforeNum+**',\''**+rname+**'\')" class="threeword"><a href="javascript:void(0)">上一页</a></li>'**;  
  
 lis += fristPage;  
 lis += beforePage;  
 //1.2 展示分页页码  
 /\*  
 1.一共展示10个页码，能够达到前5后4的效果  
 2.如果前边不够5个，后边补齐10个  
 3.如果后边不足4个，前边补齐10个  
 \*/  
  
 // 定义开始位置begin,结束位置 end  
 **var** begin; // 开始位置  
 **var** end ; // 结束位置  
  
  
 //1.要显示10个页码  
 **if**(pb.totalPage < 10){  
 //总页码不够10页  
  
 begin = 1;  
 end = pb.totalPage;  
 }**else**{  
 //总页码超过10页  
  
 begin = pb.**currentPage** - 5 ;  
 end = pb.**currentPage** + 4 ;  
  
 //2.如果前边不够5个，后边补齐10个  
 **if**(begin < 1){  
 begin = 1;  
 end = begin + 9;  
 }  
  
 //3.如果后边不足4个，前边补齐10个  
 **if**(end > pb.totalPage){  
 end = pb.totalPage;  
 begin = end - 9 ;  
 }  
 }  
  
  
 **for** (**var** i = begin; i <= end ; i++) {  
 **var** li;  
 //判断当前页码是否等于i  
 **if**(pb.**currentPage** == i){  
  
 li = **'<li class="curPage" onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
  
 }**else**{  
 //创建页码的li  
 li = **'<li onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
 }  
 //拼接字符串  
 lis += li;  
 }  
  
  
  
  
  
 /\* for (var i = 1; i <= pb.totalPage ; i++) {  
 var li;  
 //判断当前页码是否等于i  
 if(pb.currentPage == i){  
  
 li = '<li class="curPage" onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
  
 }else{  
 //创建页码的li  
 li = '<li onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
 }  
 //拼接字符串  
 lis += li;  
 }\*/  
 **var** lastPage = **'<li class="threeword"><a href="javascript:;">末页</a></li>'**;  
 **var** nextPage = **'<li class="threeword"><a href="javascript:;">下一页</a></li>'**;  
  
 lis += nextPage;  
 lis += lastPage;  
  
  
 //将lis内容设置到 ul  
 **$**(**"#pageNum"**).html(lis);  
  
  
  
 /\*  
 <li>  
 <div class="img"><img src="images/04-search\_03.jpg" alt=""></div>  
 <div class="text1">  
 <p>【减100元 含除夕/春节出发】广州增城三英温泉度假酒店/自由行套票</p>  
 <br/>  
 <p>1-2月出发，网付立享￥1099/2人起！爆款位置有限，抢完即止！</p>  
 </div>  
 <div class="price">  
 <p class="price\_num">  
 <span>&yen;</span>  
 <span>299</span>  
 <span>起</span>  
 </p>  
 <p><a href="route\_detail.html">查看详情</a></p>  
 </div>  
 </li>  
  
 \*/  
  
 //2.列表数据展示  
 **var** route\_lis = **""**;  
  
 **for** (**var** i = 0; i < pb.list.**length**; i++) {  
 //获取{rid:1,rname:"xxx"}  
 **var** route = pb.list[i];  
  
 **var** li = **'<li>\n'** +  
 **' <div class="img"><img src="'**+route.rimage+**'" style="width: 299px;"></div>\n'** +  
 **' <div class="text1">\n'** +  
 **' <p>'**+route.**rname**+**'</p>\n'** +  
 **' <br/>\n'** +  
 **' <p>'**+route.routeIntroduce+**'</p>\n'** +  
 **' </div>\n'** +  
 **' <div class="price">\n'** +  
 **' <p class="price\_num">\n'** +  
 **' <span>&yen;</span>\n'** +  
 **' <span>'**+route.price+**'</span>\n'** +  
 **' <span>起</span>\n'** +  
 **' </p>\n'** +  
 **' <p><a href="route\_detail.html">查看详情</a></p>\n'** +  
 **' </div>\n'** +  
 **' </li>'**;  
 route\_lis += li;  
 }  
 **$**(**"#route"**).html(route\_lis);  
  
 //定位到页面顶部  
 ***window***.scrollTo(0,0);  
 });  
  
}

##### 6.1.2视频详情



##### 6.1.3总结与补充

##### 6.1.4课堂提问与练习

##### 6.1.5习题答案

##### 6.1.6视频缺陷

无

##### 6.1.7视频扩展

无

#### 6.2登录功能\_姓名提示

##### 6.2.1知识概述

**$**(**function** () {  
 /\* var search = location.search;  
 //alert(search);//?id=5  
 // 切割字符串，拿到第二个值  
 var cid = search.split("=")[1];\*/  
 //获取cid的参数值  
 **var** cid = *getParameter*(**"cid"**);  
 //获取rname的参数值  
 **var** rname = *getParameter*(**"rname"**);  
 //判断rname如果不为null或者""  
 **if**(rname){  
 //url解码  
 rname = ***window***.*decodeURIComponent*(rname);  
 }  
  
 //当页码加载完成后，调用load方法，发送ajax请求加载数据  
 *load*(cid,**null**,rname);  
});  
  
**function** *load*(cid ,currentPage,rname){  
 //发送ajax请求，请求route/pageQuery,传递cid  
 **$**.get(**"route/pageQuery"**,{**cid**:cid,**currentPage**:currentPage,**rname**:rname},**function** (pb) {  
 //解析pagebean数据，展示到页面上  
  
 //1.分页工具条数据展示  
 //1.1 展示总页码和总记录数  
 **$**(**"#totalPage"**).html(pb.totalPage);  
 **$**(**"#totalCount"**).html(pb.totalCount);  
  
 /\*  
 <li><a href="">首页</a></li>  
 <li class="threeword"><a href="#">上一页</a></li>  
 <li class="curPage"><a href="#">1</a></li>  
 <li><a href="#">2</a></li>  
 <li><a href="#">3</a></li>  
 <li><a href="#">4</a></li>  
 <li><a href="#">5</a></li>  
 <li><a href="#">6</a></li>  
 <li><a href="#">7</a></li>  
 <li><a href="#">8</a></li>  
 <li><a href="#">9</a></li>  
 <li><a href="#">10</a></li>  
 <li class="threeword"><a href="javascript:;">下一页</a></li>  
 <li class="threeword"><a href="javascript:;">末页</a></li>  
  
  
 \*/  
 **var** lis = **""**;  
  
 **var** fristPage = **'<li onclick="javascipt:***load***('**+cid+**',1,\''**+rname+**'\')"><a href="javascript:void(0)">首页</a></li>'**;  
  
 //计算上一页的页码  
 **var** beforeNum = pb.**currentPage** - 1;  
 **if**(beforeNum <= 0){  
 beforeNum = 1;  
 }  
  
 **var** beforePage = **'<li onclick="javascipt:***load***('**+cid+**','**+beforeNum+**',\''**+rname+**'\')" class="threeword"><a href="javascript:void(0)">上一页</a></li>'**;  
  
 lis += fristPage;  
 lis += beforePage;  
 //1.2 展示分页页码  
 /\*  
 1.一共展示10个页码，能够达到前5后4的效果  
 2.如果前边不够5个，后边补齐10个  
 3.如果后边不足4个，前边补齐10个  
 \*/  
  
 // 定义开始位置begin,结束位置 end  
 **var** begin; // 开始位置  
 **var** end ; // 结束位置  
  
  
 //1.要显示10个页码  
 **if**(pb.totalPage < 10){  
 //总页码不够10页  
  
 begin = 1;  
 end = pb.totalPage;  
 }**else**{  
 //总页码超过10页  
  
 begin = pb.**currentPage** - 5 ;  
 end = pb.**currentPage** + 4 ;  
  
 //2.如果前边不够5个，后边补齐10个  
 **if**(begin < 1){  
 begin = 1;  
 end = begin + 9;  
 }  
  
 //3.如果后边不足4个，前边补齐10个  
 **if**(end > pb.totalPage){  
 end = pb.totalPage;  
 begin = end - 9 ;  
 }  
 }  
  
  
 **for** (**var** i = begin; i <= end ; i++) {  
 **var** li;  
 //判断当前页码是否等于i  
 **if**(pb.**currentPage** == i){  
  
 li = **'<li class="curPage" onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
  
 }**else**{  
 //创建页码的li  
 li = **'<li onclick="javascipt:***load***('**+cid+**','**+i+**',\''**+rname+**'\')"><a href="javascript:void(0)">'**+i+**'</a></li>'**;  
 }  
 //拼接字符串  
 lis += li;  
 }  
  
  
  
  
  
 /\* for (var i = 1; i <= pb.totalPage ; i++) {  
 var li;  
 //判断当前页码是否等于i  
 if(pb.currentPage == i){  
  
 li = '<li class="curPage" onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
  
 }else{  
 //创建页码的li  
 li = '<li onclick="javascipt:load('+cid+','+i+')"><a href="javascript:void(0)">'+i+'</a></li>';  
 }  
 //拼接字符串  
 lis += li;  
 }\*/  
 **var** lastPage = **'<li class="threeword"><a href="javascript:;">末页</a></li>'**;  
 **var** nextPage = **'<li class="threeword"><a href="javascript:;">下一页</a></li>'**;  
  
 lis += nextPage;  
 lis += lastPage;  
  
  
 //将lis内容设置到 ul  
 **$**(**"#pageNum"**).html(lis);  
  
  
  
 /\*  
 <li>  
 <div class="img"><img src="images/04-search\_03.jpg" alt=""></div>  
 <div class="text1">  
 <p>【减100元 含除夕/春节出发】广州增城三英温泉度假酒店/自由行套票</p>  
 <br/>  
 <p>1-2月出发，网付立享￥1099/2人起！爆款位置有限，抢完即止！</p>  
 </div>  
 <div class="price">  
 <p class="price\_num">  
 <span>&yen;</span>  
 <span>299</span>  
 <span>起</span>  
 </p>  
 <p><a href="route\_detail.html">查看详情</a></p>  
 </div>  
 </li>  
  
 \*/  
  
 //2.列表数据展示  
 **var** route\_lis = **""**;  
  
 **for** (**var** i = 0; i < pb.list.**length**; i++) {  
 //获取{rid:1,rname:"xxx"}  
 **var** route = pb.list[i];  
  
 **var** li = **'<li>\n'** +  
 **' <div class="img"><img src="'**+route.rimage+**'" style="width: 299px;"></div>\n'** +  
 **' <div class="text1">\n'** +  
 **' <p>'**+route.**rname**+**'</p>\n'** +  
 **' <br/>\n'** +  
 **' <p>'**+route.routeIntroduce+**'</p>\n'** +  
 **' </div>\n'** +  
 **' <div class="price">\n'** +  
 **' <p class="price\_num">\n'** +  
 **' <span>&yen;</span>\n'** +  
 **' <span>'**+route.price+**'</span>\n'** +  
 **' <span>起</span>\n'** +  
 **' </p>\n'** +  
 **' <p><a href="route\_detail.html">查看详情</a></p>\n'** +  
 **' </div>\n'** +  
 **' </li>'**;  
 route\_lis += li;  
 }  
 **$**(**"#route"**).html(route\_lis);  
  
 //定位到页面顶部  
 ***window***.scrollTo(0,0);  
 });  
  
}

##### 6.2.2视频详情



##### 6.2.3总结与补充

##### 6.2.4课堂提问与练习

##### 6.2.5习题答案

##### 6.2.6视频缺陷

无

##### 6.2.7视频扩展

无