航线月度销售预案

系统设计

## 月度淡旺季划分

### 流程设计

* 通过输入月份，始发地，达到地，经停地（可选），航班号（可选）查询相关的数据，生成淡旺季划分表格
* 在淡旺季表格中输入季节划分描述，本月运力，往返程的同比均值偏离幅度，生成淡旺季划分点和对应月度的同比环比往返数据，以不同背景色区分淡旺季。
* 根据往返程的同比和环比数据填写当前月份中每一段时间的淡旺季走向。
* 填写完毕后点击保存，将数据保存到数据库中。

### 流程实现

* 前台页面airline.jsp 提交查询条件，调用后台AirlineAction类中的方法getMonthData(MonthSalesPlanQuery query)获取输入月份相关的同比和环比数据，返回页面airline.jsp生成如下表格



* 在表格中填入相应的淡旺季划分描述，本月运力，往返偏离值幅度，触发前台事件，调用后台AirlineAction类中的方法getMonthData(MonthSalesPlanQuery query)，获取当前输入月份的同比环比的月份的每一天的数据，生成往返的同比环比表格，表格中根据不同背景颜色区分淡旺季。如下图：





* 根据往返同比和环比数据填写淡旺季划分的日期区间



* 点击保存，调用后台AirlineAction类中的方法saveMonthParam(MonthParameterQuery query)保存相应的淡旺季划分数据和时间区间

### 使用数据表

z\_airdata

t\_divide\_season(淡旺季划分参数表)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 数据长度 | 注释 |
| Divide\_Id | Int | 10 | 主键，自增 |
| Divide\_Month | Varchar | 10 | 淡旺季划分月份 |
| Dpt\_AirPt\_Cd | Varchar | 10 | 起始机场 |
| Arrv\_Airpt\_Cd | Varchar | 10 | 到达机场 |
| Flt\_Nbr | Varchar | 20 | 航班号 |
| Flt\_Rte\_Cd | Varchar | 10 | 航线 |
| Days | Int | 11 | 当月天数 |
| Yoy\_Trans | Decimal | 10,0 | 同比运力 |
| Qoq\_Trans | Decimal | 10,0 | 环比运力 |
| Curr\_Trans | Decimal | 10,0 | 当月运力 |
| Avg\_GO\_Cust | Int | 10 | 往程平均客量 |
| Avg\_Return\_Cust | Int | 10 | 返程平均客量 |
| Avg\_Go\_Dis | Decimal | 10,0 | 往程偏移均值幅度 |
| Avg\_Return\_Dis | Decimal | 10,0 | 返程偏移均值幅度 |
| Description | Varchar | 100 | 淡旺季描述 |

t\_divide\_time(淡旺季划分日期区间表)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 数据长度 | 注释 |
| Divide\_Id | Int | 10 | 主键，自增 |
| Season\_Id | Int | 10 | 外键淡旺季划分表 |
| Flt\_Direct | Varchar | 10 | 航线方向 |
| Start\_Date | Varchar | 10 | 区间起始日期 |
| End\_Date | Varchar | 10 | 区间截止日期 |
| Ref\_Sale\_Start\_Date | Varchar | 10 | 对应同比销售起始日期 |
| Ref\_Sale\_End\_Date | Varchar | 10 | 对应同比销售截止日期 |
| Nature | Varchar | 10 | 区间淡旺季性质 |

## 销售预案

### 流程设计

* 通过输入年份，始发地，达到地，经停地（可选），航班号（可选）查询相关的数据，生成销售预案表格
* 在预案表格中输入协议，散客数量，散客均折，最低起售，散客调控周期，团队数量，折扣，团队调控周期，班次，预估虚耗位置及处理方案
* 填写完毕后，保存，每一行都对应生成一条销售预案数据。

### 流程实现

* 在销售预案页面上，输入对应的数据后，点击查询，调用后台AirlineAction类中的方法getMonthSalesPlanParam(MonthSalesPlanQuery query)，得到参数相对应的淡旺季划分数据，返回页面上生成如下表格：





* 向表格中的空白输入框中填写入对应的数据，生成完整表格



* 当表格空白输入框都有数据之后，点击保存，页面提交表格中的销售预案数据，后台AirlineAction类中方法saveSalePlanData(SalePlanData data)将数据保存到数据库中。

### 使用数据表格

t\_divide\_season(淡旺季划分参数表), t\_divide\_time(淡旺季划分日期区间表),

t\_monthsaleplan

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 数据长度 | 注释 |
| Id | Int | 10 | 主键，自增 |
| Time\_id | Int | 10 | 对应季节划分区间id |
| Agreement | Varchar | 10 | 协议 |
| FIT\_Nbr | Int | 10 | 散客数量 |
| All\_fold | Decimal | 10,2 | 均折 |
| Lowest\_Sale | Decimal | 10,2 | 最低仓位起售 |
| Group\_Nbr | Int | 10 | 团队数量 |
| FIT\_Sale\_Cycle | Int | 10 | 散客销售调控周期 |
| Discount\_Return | Decimal | 10,2 | 往返折扣 |
| Group\_Sale\_Cycle | Int | 10 | 团队销售调控周期 |
| Est\_Posi\_Nbr | Int | 10 | 班次 |
| Est\_Dispose\_Plan | Varchar | 100 | 虚耗处置方案 |
| Frequency | Int | 10 | 每班虚耗位数量 |

## 舱位设置

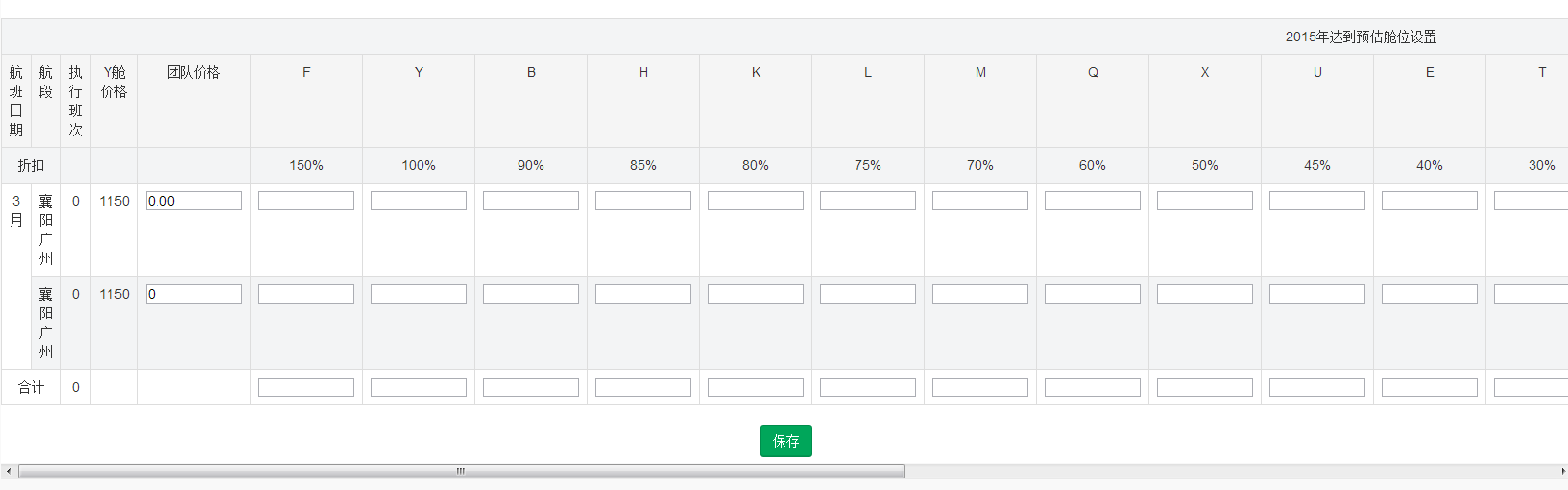
### 流程设计

* 通过输入年份，始发地，达到地，经停地（可选），航班号（可选）查询相关的数据，生成前两年的同比均班营收表，环比均班营收表，销售预估表，舱位设置表。
* 根据往年的数据和销售预估数据，填写对应舱位设置表的各种舱位数量。
* 每个舱位的数量设置好之后，自动计算小时收入，补贴还有客座率等，最后点击保存，将当前月份的舱位设置数据填写入数据库中。

### 流程实现

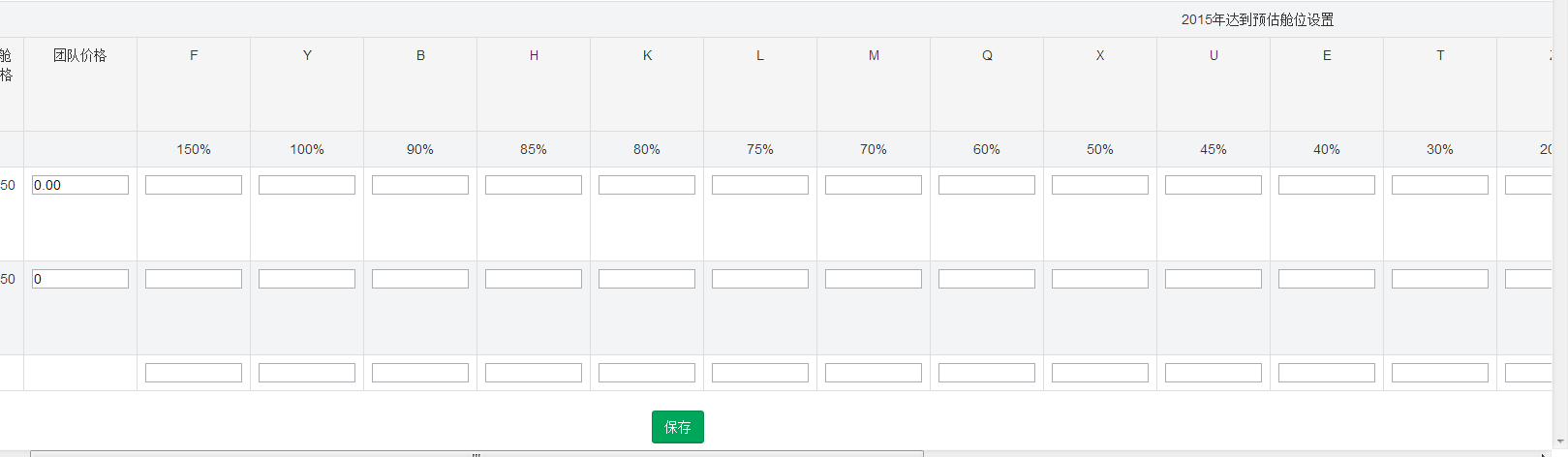
* 在销售预案模块中切换到舱位设定选项卡，数据相关参数值，点击查询，调用后台AirlineAction类中的方法getCabinSeatSetData(MonthSalesPlanQuery query)获取当前输入月份的三年前和两年前的同比数据，当前月份的环比数据和当前月份的销售预案数据，返回到前台页面，生成相应的数据表格。







* 向表格中的空白输入框中填写入对应的数据，生成完整表格





* 点击保存，调用后台AirlineAction类中的方法saveCabinSeatSetData(CabinSeatSetQuery query)将数据保存到数据库中。

### 使用数据表格

Z\_airdata,

t\_monthsaleplan（月度销售预案表），

t\_cabinseatset（舱位设置表）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 数据长度 | 注释 |
| Id | Int | 10 | 主键，自增 |
| Time\_Id | Int | 10 | 划分区间id |
| Grp\_Prc | Decimal | 10,2 | 团队价格 |
| Two\_Tak\_Ppt | Int | 10 | 两舱比例 |
| Ful\_Pce\_Ppt | Int | 10 | 100% |
| Nne\_Dnt\_Ppt | Int | 10 | 90% |
| Eht\_Five\_Dnt\_Ppt | Int | 10 | 85% |
| Eht\_Dnt\_Ppt | Int | 10 | 80% |
| Sen\_Five\_Dnt\_Ppt | Int | 10 | 75% |
| Sen\_Dnt\_Ppt | Int | 10 | 70% |
| Six\_Dnt\_Ppt | Int | 10 | 60% |
| Fve\_Dnt\_Ppt | Int | 10 | 50% |
| Fur\_Fve\_Dnt\_Ppt | Int | 10 | 45% |
| Fur\_Dnt\_Ppt | Int | 10 | 40% |
| Thr\_Dnt\_Ppt | Int | 10 | 30% |
| Two\_Dnt\_Ppt | INT | 10 | 20% |
| Grp\_Nbr | INT | 10 | 团队数量 |

## 城市信息管理

### 流程设计

#### 1 查询

* 前台页面输入要查询的城市名称（可以不输入，查询所有城市），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 新增

* 前台页面点击新增，进入城市新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 修改

* 在列表页面选择任意一条数据，点击修改，进入城市修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 2 流程实现

#### 查询

* 进入城市信息列表页面cityinfo.jsp，在条件输入框中输入需要查询的城市名称（模糊查询）



* 点击查询按钮调用后台的CityInfoAction类中的方法search（String cityName）.
* 在方法search(String cityName)调用业务逻辑类CityInfoService的方法search(String cityName)。
* 类CityInfoServiceImpl实现了CityInfoService中的方法search(String cityName),利用mybatis的持久化操作将满足查询条件的数据封装成List<CityInfo>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 新增

* 进入城市信息列表页面cityinfo.jsp，点击页面中的添加按钮，进入添加页面。





* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的CityInfoAction类中的方法add（CityInfo info）接收前台提交数据，并封装成对象.
* 在方法add(CityInfo info)调用业务逻辑类CityInfoService的方法add(CityInfo info)。
* 类CityInfoServiceImpl实现了CityInfoService中的方法add(CityInfo info),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 修改

* 进入城市信息列表页面cityinfo.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的CityInfoAction类中的方法update（CityInfo info）.
* 在方法update（CityInfo info）调用业务逻辑类CityInfoService的方法update（CityInfo info）。
* 类CityInfoServiceImpl实现了CityInfoService中的方法update（CityInfo info）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 批量删除

* 进入城市信息列表页面cityinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的CityInfoAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类CityInfoService的方法batchdel（int[] ids）。
* 类CityInfoServiceImpl实现了CityInfoService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

T\_cityinfo(城市信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 数据长度 | 注释 |
| Id | Int | 10 | 主键，自增 |
| Cityname | Varchar | 20 | 城市名称 |
| Economy | Bigint | 18 | 经济 |
| Population | Int | 10 | 人口数量 |
| Percapita | Decimal | 10,2 | 人均收入 |
| Structure | Varchar | 50 | 结构 |
| Traffic | Varchar | 50 | 交通 |
| Area | Decimal | 10,2 | 面积 |
| Popedom | Varchar | 100 | 辖区 |

## 机场建设进度管理

### 1流程设计

#### 1 查询

* 前台页面输入要查询的机场名称（可以不输入，查询所有机场），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入机场进度新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入机场进度修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入机场进度信息列表页面display.jsp，在条件输入框中输入需要查询的机场名称（模糊查询）



* 点击查询按钮调用后台的BuidScheduleAction类中的方法search（String airportname）.
* 在方法search（String airportname）调用业务逻辑类BuidScheduleService的方法search（String airportname）。
* 类BuidScheduleServiceImpl实现了BuidScheduleService中的方法search（String airportname）,利用mybatis的持久化操作将满足查询条件的数据封装成List<BuidSchedule>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入机场进度信息列表页面display.jsp，点击页面中的添加按钮，进入添加页面。





* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的BuidScheduleAction类中的方法add（CityInfo buid）接收前台提交数据，并封装成对象.
* 在方法add(CityInfo buid)调用业务逻辑类CityInfoService的方法add(BuidSchedule buid)。
* 类CityInfoServiceImpl实现了CityInfoService中的方法add(BuidSchedule buid),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入机场进度信息列表页面display.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的BuidScheduleAction类中的方法update（BuidSchedule buid）.
* 在方法update（BuidSchedule buid）调用业务逻辑类BuidSchedule Service的方法update（BuidSchedule info）。
* 类BuidSchedule ServiceImpl实现了BuidSchedule Service中的方法update（BuidSchedule buid）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入机场进度信息列表页面display.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的BuidScheduleAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类BuidSchedule Service的方法batchdel（int[] ids）。
* 类BuidSchedule ServiceImpl实现了BuidSchedule Service中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

t\_buidschedule (机场建设进度信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 数据长度 | 注释 |
| Id | Int | 10 | 主键，自增 |
| Airport\_name | Varchar | 20 | 机场名称 |
| init\_buid\_date | Varchar | 20 | 初建时间 |
| accept\_date | Varchar | 20 | 验收时间 |
| test\_flight\_date | Varchar | 20 | 试飞时间 |
| planed\_ship\_date | Varchar | 20 | 计划通航时间 |
| revised\_date | Varchar | 20 | 校飞时间 |
| destination | Varchar | 50 | 计划通航城市 |
| target\_throughput | Int | 10 | 计划吞吐量 |
| awarding\_policy | Varchar | 100 | 奖励政策 |

## 机场信息管理

### 1流程设计

#### 1 查询

* 前台页面输入要查询的机场名称（可以不输入，查询所有机场），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入机场新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入机场修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入机场信息列表页面display.jsp，在条件输入框中输入需要查询的机场名称（模糊查询）



* 点击查询按钮调用后台的AirportInfoAction类中的方法search（String info）.
* 在方法search(String info)调用业务逻辑类AirportInfoService的方法search(String info)。
* 类AirportInfoServiceImpl实现了AirportInfoService中的方法search(String info),利用mybatis的持久化操作将满足查询条件的数据封装成List<AirportInfo>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入机场信息列表页面display.jsp，点击页面中的添加按钮，进入添加页面。





* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的AirportInfoAction类中的方法add（AirportInfo info）接收前台提交数据，并封装成对象.
* 在方法add(AirportInfo info)调用业务逻辑类AirportInfoService的方法add(AirportInfo info)。
* 类AirportInfoServiceImpl实现了AirportInfoService中的方法add(AirportInfo info),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入机场信息列表页面display.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的AirportInfoAction类中的方法update（AirportInfo info）.
* 在方法update（AirportInfo info）调用业务逻辑类AirportInfoService的方法update（AirportInfo info）。
* 类AirportInfoServiceImpl实现了AirportInfoService中的方法update（AirportInfo info）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入机场信息列表页面display.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的AirportInfoAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类AirportInfoService的方法batchdel（int[] ids）。
* 类AirportInfoServiceImpl实现了AirportInfoService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

t\_airpotinfo(机场信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| **Id** | **Int** | **10** | **主键，自增** |
| **airport\_name** | **Varchar** | **20** | **机场名称** |
| **runway\_length** | **Int** | **5** | **跑道长度** |
| **runway\_width** | **Int** | **3** | **跑道宽度** |
| **airport\_level** | **Varchar** | **10** | **机场等级** |
| **fire\_rating** | **Varchar** | **10** | **消防等级** |
| **can\_stop\_type** | **Varchar** | **50** | **可停降机型** |
| **airport\_type** | **Varchar** | **20** | **机场类型** |
| **climatic\_changes** | **Varchar** | **50** | **气候变化规律** |
| **clearance\_condition** | **Varchar** | **50** | **净空条件** |

## 机场交通工具信息管理

### 1流程设计

#### 1 查询

* 前台页面输入要查询的机场名称（可以不输入，查询所有机场附近的交通工具），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入机场交通工具新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入机场交通工具修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入机场交通工具信息列表页面display.jsp，在条件输入框中输入需要查询的机场名称（模糊查询）



* 点击查询按钮调用后台的TransportInfoAction类中的方法search（String transportname）.
* 在方法search(String transportname)调用业务逻辑类TransportInfoService的方法search(String transportname)。
* 类TransportInfoServiceImpl实现了TransportInfoService中的方法search(String transportname),利用mybatis的持久化操作将满足查询条件的数据封装成List<TransportInfo>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入机场交通工具信息列表页面display.jsp，点击页面中的添加按钮，进入添加页面。



* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的TransportInfo Action类中的方法add（TransportInfo info）接收前台提交数据，并封装成对象.
* 在方法add(TransportInfo info)调用业务逻辑类TransportInfoService的方法add(TransportInfo info)。
* 类TransportInfoServiceImpl实现了TransportInfoService中的方法add(TransportInfo info),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入机场交通工具信息列表页面display.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的TransportInfoAction类中的方法update（TransportInfo info）.
* 在方法update（TransportInfo info）调用业务逻辑类TransportInfo Service的方法update（TransportInfo info）。
* 类TransportInfo ServiceImpl实现了TransportInfo Service中的方法update（TransportInfo info）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入机场交通工具信息列表页面transportinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的TransportInfoAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类TransportInfoService的方法batchdel（int[] ids）。
* 类TransportInfoServiceImpl实现了TransportInfoService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

T\_ transportinfo(机场交通工具信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| **Id** | **Int** | **10** | **主键，自增** |
| **transport\_name** | **Varchar** | **20** | **交通工具名称** |
| **transport\_type** | **Varchar** | **50** | **交通工具类型** |
| **airport\_name** | **Varchar** | **20** | **附近机场名称** |
| **hyperbolic** | **Varchar** | **20** | **运行线路** |
| **running\_time** | **Varchar** | **20** | **发车时间** |
| **off\_running\_time** | **Varchar** | **20** | **收班时间** |
| **sigle\_trip\_time** | **Int** | **10** | **单程所需时间** |
| **interval\_time** | **Int** | **10** | **发车间隔时间** |

## 基地信息管理

### 1流程设计

#### 1 查询

* 前台页面输入要查询的基地名称（可以不输入，查询所有基地），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入基地新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入基地修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入基地信息列表页面shipinfo.jsp，在条件输入框中输入需要查询的基地名称（模糊查询）



* 点击查询按钮调用后台的BaseAction类中的方法search（String basename）.
* 在方法search(String basename)调用业务逻辑类BaseService的方法search(String basename)。
* 类BaseServiceImpl实现了BaseService中的方法search(String basename),利用mybatis的持久化操作将满足查询条件的数据封装成List<Base>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入基地信息列表页面shipinfo.jsp，点击页面中的添加按钮，进入添加页面。





* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的BaseAction类中的方法add（Base base）接收前台提交数据，并封装成对象.
* 在方法add(Base base)调用业务逻辑类BaseService的方法add(Base base)。
* 类BaseServiceImpl实现了BaseService中的方法add(Base base),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入基地信息列表页面shipinfo.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的BaseAction类中的方法update（Base base）.
* 在方法update（Base base）调用业务逻辑类BaseService的方法update（Base base）。
* 类BaseServiceImpl实现了BaseService中的方法update（Base base）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入基地信息列表页面shipinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的BaseAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类BaseService的方法batchdel（int[] ids）。
* 类BaseServiceImpl实现了BaseService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

T\_base(基地信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| Id | Int | 10 | 主键，自增 |
| **base\_name** | **Varchar** | **10** | **基地名称** |
| **plane\_stop\_position** | **Varchar** | **30** | **飞机停场位置** |
| **cityname** | **Varchar** | **20** | **城市名称** |
| **plane\_count** | **Int** | **10** | **飞机数量** |
| **plane\_type** | **Varchar** | **20** | **飞机机型** |
| **flying\_routes** | **Varchar** | **20** | **在飞航线** |

## 驻外营业部管理

### 1流程设计

#### 1查询

* 前台页面输入要查询的驻外营业部名称（可以不输入，查询所有驻外营业部），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入驻外营业部新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入驻外营业部修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入驻外营业部信息列表页面shipinfo.jsp，在条件输入框中输入需要查询的营业部名称（模糊查询）



* 点击查询按钮调用后台的SaleDptInfoAction类中的方法search（String dptname）.
* 在方法search(String dptname)调用业务逻辑类SaleDptInfoService的方法search(String dptname)。
* 类SaleDptInfoServiceImpl实现了SaleDptInfoService中的方法search(String dptname),利用mybatis的持久化操作将满足查询条件的数据封装成List<SaleDptInfo>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入驻外营业部信息列表页面shipinfo.jsp，点击页面中的添加按钮，进入添加页面。



* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的SaleDptInfoAction类中的方法add（SaleDptInfo info）接收前台提交数据，并封装成对象.
* 在方法add(SaleDptInfo info)调用业务逻辑类SaleDptInfoService的方法add(SaleDptInfo info)。
* 类SaleDptInfoServiceImpl实现了SaleDptInfoService中的方法add(SaleDptInfo info),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入驻外营业部信息列表页面shipinfo.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的SaleDptInfoAction类中的方法update（SaleDptInfo info）.
* 在方法update（SaleDptInfo info）调用业务逻辑类CityInfoService的方法update（SaleDptInfo info）。
* 类SaleDptInfoServiceImpl实现了SaleDptInfoService中的方法update（SaleDptInfo info）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入驻外营业部信息列表页面shipinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的SaleDptInfoAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类SaleDptInfoService的方法batchdel（int[] ids）。
* 类SaleDptInfoServiceImpl实现了SaleDptInfoService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

T\_saledptinfo (驻外营业部信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| **Id** | **Int** | **10** | **主键，自增** |
| **dpt\_name** | **Varchar** | **20** | **部门名称** |
| **address** | **Varchar** | **50** | **地址** |
| **linkman** | **Varchar** | **20** | **联系人** |
| **contact\_way** | **Varchar** | **20** | **联系方式** |
| **whether\_local\_base** | **Varchar** | **10** | **是否同城基地** |
| **map\_display** | **Varchar** | **10** | **地图显示** |

## 机队管理

### 1流程设计

#### 1查询

* 前台页面输入要查询的机队名称（可以不输入，查询所有机队），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入机队新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入机队修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入机队信息列表页面shipinfo.jsp，在条件输入框中输入需要查询的机队名称（模糊查询）



* 点击查询按钮调用后台的PlaneTeamAction类中的方法search（String teamname）.
* 在方法search(String teamname)调用业务逻辑类PlaneTeamService的方法search(String teamname)。
* 类PlaneTeamServiceImpl实现了PlaneTeamService中的方法search(String teamname),利用mybatis的持久化操作将满足查询条件的数据封装成List<PlaneTeam>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入机队信息列表页面shipinfo.jsp，点击页面中的添加按钮，进入添加页面。





* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的PlaneTeamAction类中的方法add（PlaneTeam team）接收前台提交数据，并封装成对象.
* 在方法add(PlaneTeam team)调用业务逻辑类PlaneTeamService的方法add(PlaneTeam team)。
* 类PlaneTeamServiceImpl实现了PlaneTeamService中的方法add(PlaneTeam team),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入机队信息列表页面shipinfo.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的PlaneTeamAction类中的方法update（PlaneTeam team）.
* 在方法update（PlaneTeam team）调用业务逻辑类PlaneTeamService的方法update（PlaneTeam team）。
* 类PlaneTeamServiceImpl实现了PlaneTeamService中的方法update（PlaneTeam team）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入机队信息列表页面shipinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的PlaneTeamAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类PlaneTeamService的方法batchdel（int[] ids）。
* 类PlaneTeamServiceImpl实现了PlaneTeamService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

T\_planeteam(机队信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| **Id** | **Int** | **10** | **主键，自增** |
| **team\_name** | **Varchar** | **20** | **机队名称** |
| **plane\_count** | **Int** | **10** | **飞机数量** |
| **plane\_type** | **Varchar** | **20** | **飞机机型** |
| **plane\_age** | **Int** | **10** | **飞机机龄** |
| **flyer\_count** | **Int** | **10** | **飞行员数量** |
| **team\_level** | **Varchar** | **20** | **机队等级** |

## 时刻资源管理

### 1流程设计

#### 1查询

* 前台页面输入要查询的航站名称（可以不输入，查询所有航站的时刻资源信息），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入航站时刻资源新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入航站时刻资源修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入航站时刻资源信息列表页面shipinfo.jsp，在条件输入框中输入需要查询的航站名称（模糊查询）



* 点击查询按钮调用后台的TimeResourceAction类中的方法search（String terminal）.
* 在方法search(String terminal)调用业务逻辑类TimeResourceService的方法search(String terminal)。
* 类TimeResourceServiceImpl实现了TimeResourceService中的方法search(String terminal),利用mybatis的持久化操作将满足查询条件的数据封装成List<TimeResource>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入航站时刻资源信息列表页面shipinfo.jsp，点击页面中的添加按钮，进入添加页面。



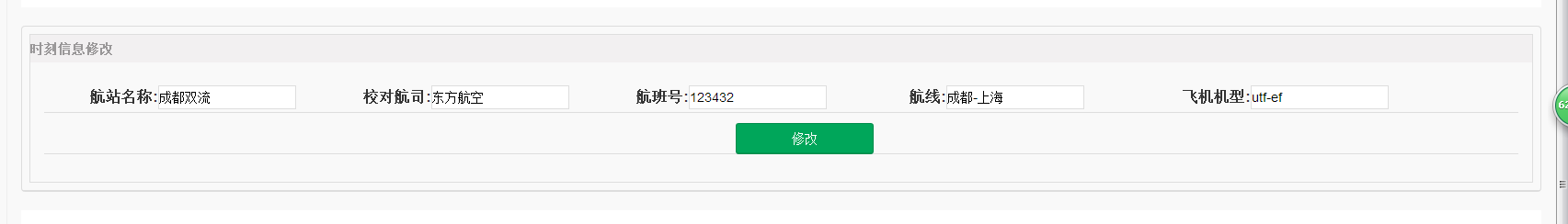


* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的TimeResourceAction类中的方法add（CityInfo resource）接收前台提交数据，并封装成对象.
* 在方法add(TimeResource resource)调用业务逻辑类TimeResourceService的方法add(TimeResource resource)。
* 类TimeResourceServiceImpl实现了TimeResourceService中的方法add(TimeResource resource),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入航站时刻资源信息列表页面shipinfo.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的TimeResourceAction类中的方法update（TimeResource resource）.
* 在方法update（TimeResource resource）调用业务逻辑类TimeResourceService的方法update（TimeResource resource）。
* 类TimeResourceServiceImpl实现了TimeResourceService中的方法update（TimeResource info）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入航站时刻资源信息列表页面shipinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的TimeResourceAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类TimeResourceService的方法batchdel（int[] ids）。
* 类TimeResourceServiceImpl实现了TimeResourceService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

t\_time\_resource (时刻资源表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| **Id** | **Int** | **10** | **主键，自增** |
| **terminal** | **Varchar** | **20** | **航站名称** |
| **proofread\_airline** | **Varchar** | **20** | **校对航司** |
| **flight\_nbr** | **Varchar** | **20** | **航班号** |
| **air\_route** | **Varchar** | **20** | **航线** |
| **plane\_type** | **Varchar** | **20** | **飞机机型** |

## 在飞航线管理

### 1 流程设计

#### 1查询

* 前台页面输入要查询的航线（可以不输入，查询所有航线），提交至后台。
* 后台接收前台输入的查询条件，通过相关数据连接，查询数据库中数据。
* 将查询得到结果集处理，返回前台页面，生成对应列表。

#### 2新增

* 前台页面点击新增，进入航线新增页面
* 在新增页面输入框中录入相应的数据，点击添加按钮，提交页面数据。
* 后台将接收到的数据通过添加的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回添加结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 3修改

* 在列表页面选择任意一条数据，点击修改，进入航线修改页面
* 在修改页面输入框中修改相应的数据，点击修改按钮，提交页面数据。
* 后台将接收到的数据通过修改的业务逻辑存入数据库，数据库返回操作数据库的结果。
* 后台返回修改结果，在页面作出相应的提示。（成功则返回到列表页面，失败则不进行页面跳转）

#### 4批量删除

* 前台列表页面选中一条或者多条数据，点击批量删除，调用后台。
* 后台将接收到的id字符串，分割成id数组，调用业务逻辑执行相应的删除操作，数据库返回删除操作的结果。
* 后台返回删除结果，在页面作出相应的提示。

### 流程实现

#### 1查询

* 进入航线信息列表页面shipinfo.jsp，在条件输入框中输入需要查询的航线（模糊查询）



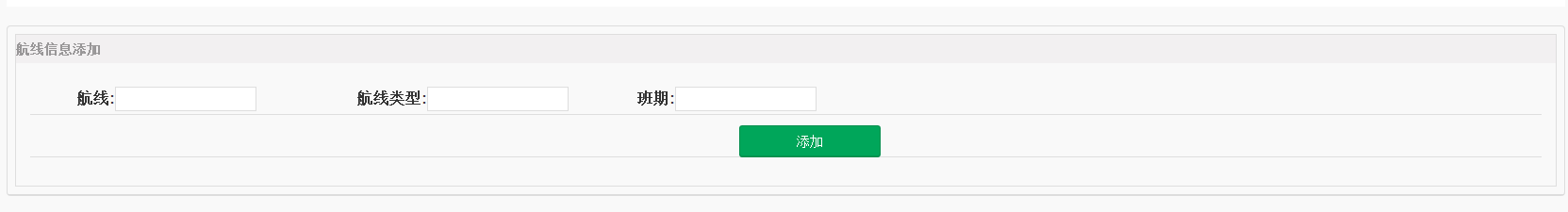
* 点击查询按钮调用后台的FlyingAirRouteAction类中的方法search（String airroute）.
* 在方法search(String airroute)调用业务逻辑类FlyingAirRouteService的方法search(String airroute)。
* 类FlyingAirRouteServiceImpl实现了FlyingAirRouteService中的方法search(String airroute),利用mybatis的持久化操作将满足查询条件的数据封装成List<FlyingAirRoute>存入到Map中返回前台页面。
* 页面解析后台返回数据，生成列表。



#### 2新增

* 进入航线列表页面shipinfo.jsp，点击页面中的添加按钮，进入添加页面。





* 在输入框中录入相应的数据，点击添加按钮，调用后台服务。
* 后台的FlyingAirRouteAction类中的方法add（FlyingAirRoute info）接收前台提交数据，并封装成对象.
* 在方法add(FlyingAirRoute info)调用业务逻辑类FlyingAirRouteService的方法add(FlyingAirRoute info)。
* 类FlyingAirRouteServiceImpl实现了FlyingAirRouteService中的方法add(FlyingAirRoute info),利用mybatis的持久化操作将数据插入数据库中，返回操作结果到页面。
* 页面解析后台返回结果, 弹出提示框，显示操作结果,进行页面跳转。

#### 3修改

* 进入航线信息列表页面shipinfo.jsp，选择一条数据，点击修改，进入修改页面。





* 点击查询按钮调用后台的FlyingAirRouteAction类中的方法update（FlyingAirRoute info）.
* 在方法update（FlyingAirRoute info）调用业务逻辑类FlyingAirRouteService的方法update（FlyingAirRoute info）。
* 类FlyingAirRouteServiceImpl实现了FlyingAirRouteService中的方法update（FlyingAirRoute info）,利用mybatis的持久化操作将修改的值更新到数据库，返回操作结果到前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果,进行页面跳转。

#### 4批量删除

* 进入航线信息列表页面shipinfo.jsp，选择一条或几条数据，点击批量删除



* 点击批量删除按钮调用后台的FlyingAirRouteAction类中的方法batchdel（String ids）.
* 在方法batchdel（int[] ids）调用业务逻辑类FlyingAirRouteService的方法batchdel（int[] ids）。
* 类FlyingAirRouteServiceImpl实现了FlyingAirRouteService中的方法batchdel（int[] ids）,利用mybatis的持久化操作将满足主键id为数组中的数据删除，最后将操作结果返回前台页面。
* 页面解析后台返回数据，弹出提示框，显示操作结果。

### 3 使用数据表格

T\_flyingairroute(在飞航线信息表)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **数据长度** | **注释** |
| **Id** | **Int** | **10** | **主键，自增** |
| **air\_route** | **Varchar** | **20** | **航线** |
| **air\_route\_type** | **Varchar** | **20** | **航线类型** |
| **schedule** | **Varchar** | **20** | 班期 |