|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| Finance. checkUser | 语法 | public boolean isValid (UserVo user); |
| 前置条件 | User的工号不为null |
| 后置条件 | 如果该财务人员具有高级权限，返回true，否则返回false |
| Finance. createDebitnote | 语法 | Public DebitVO createDebitNote(DebitVO debit) |
| 前置条件 | Debit符合规范 |
| 后置条件 | 返回给界面完整的付款单信息 |
| Finance. saveDebitnote | 语法 | Public Boolean saveDebitnote(DebitVO debit) |
| 前置条件 | 无 |
| 后置条件 | 持久化更新涉及的领域对象的数据 |
| Finance. show Debitnote | 语法 | public ArrayList<debitnoteVO> showDebitnote (Calendar date, long department) |
| 前置条件 | Department编号存在 |
| 后置条件 | 返回符合该时间段内所有收款单信息 |
| Finance.getDebitSum | 语法 | public long getDebitSum(Calendar date) |
| 前置条件 | debitnote 不为空 |
| 后置条件 | 返回该列表中收款单的收款总和 |
| Finance. show DebitnoteList | 语法 | public ArrayList<debitnoteVO> showDebitnoteList (Calendar date) |
| 前置条件 | 无 |
| 后置条件 | 返回该时间段各个营业厅的收款单 |
| Finance. showRent | 语法 | public RentVO showRent(int sum, int year) |
| 前置条件 | 无 |
| 后置条件 | 返回完整的租金付款单信息 |
| Finance. saveRent | 语法 | public boolean saveRent(RentVO rent) |
| 前置条件 | 已确认该付款单 |
| 后置条件 | 结束此次管理，持久化更新设计的领域对象的数据 |
| Finance. showFreight | 语法 | Public FreightVO showFreight() |
| 前置条件 | 无 |
| 后置条件 | 返回当前的运费付款单 |
| Finance. changeFreight | 语法 | Public boolean ­­changeFreight(int freight) |
| 前置条件 | 无 |
| 后置条件 | 如果修改成功，返回true，否则返回false |
| Finance. saveFreight | 语法 | Public boolean saveFreight(FreightVO freight) |
| 前置条件 | 已确认运费付款单 |
| 后置条件 | 结束此次管理，持久化更新涉及的领域对象的数据 |
| Finance. showSalary | **语法** | public pesonVO showSalary(long person) |
| 前置条件 | 员工id符合输入规则 |
| 后置条件 | 返回该员工的工资信息 |
| Finance. showSalarySum | 语法 | Public SalaryVO showSalarySum(int year, int month) |
| 前置条件 | 无 |
| 后面条件 | 返回该月的工资付款单信息 |
| Finance. showEarning | 语法 | Public EarningVO showEarning() |
| 前置条件 | 无 |
| 后置条件 | 返回截止到当前日期的经营情况表 |
| Finance. exportEarning | 语法 | Public Boolean saveEarning(EarningVO earnings) |
| 前置条件 | 无 |
| 后置条件 | 如果导出成功，返回true，否则返回false |
| Finance. addAccount | 语法 | Public Boolean addAccount(AccountVO account) |
| 前置条件 | AccoutVO不为null |
| 后置条件 | 如果增加成功，返回true， 否则返回false |
| Finance. showAccount | 语法 | Public AccountPO showAccount(long id) |
| 前置条件 | 输入的id符合规范 |
| 后置条件 | 返回查询的账户信息 |
| Finance. deleteAccount | 语法 | Public Boolean deleteAccount(long id) |
| 前置条件 | 输入的id符合规范 |
| 后置条件 | 如果删除成功返回true，否则返回false |
| Finance. changeAccount | 语法 | Public Boolean changeAccount(AccountVO account) |
| 前置条件 | Account不为null |
| 后置条件 | 如果修改成功返回true，否则返回false |
| Finance. addInitialInfo | 语法 | Public Boolean addInitialInfo(InitialInforVO initial); |
| 前置条件 | 输入的期初建账信息符合规范 |
| 后置条件 | 增加期初建账的持久化对象 |
| Finance. addPriceStrategy | 语法 | Public Boolean addPriceStrategy (PriceStrategyVO priceStrategy); |
| 前置条件 | 输入的价格策略信息都符合规范 |
| 后置条件 | 增加价格策略的持久化对象 |
| FinanceDataService  .updatePriceStrategy | 语法 | Public Boolean updatePriceStrategy(PriceStrategyVO priceStrategy); |
| 前置条件 | 修改过的运费策略信息都符合规范 |
| 后置条件 | 查找到相应的运费策略并修改 |
| FinanceDataService  .findPriceStrategy | 语法 | Public PriceStrategyVO findPriceStrategy(); |
| 前置条件 | 无 |
| 后置条件 | 查找到运费策略 |
| FinanceDataService  .addSalaryStrategy | 语法 | Public boolean addSalaryStrategy(SalaryStrategyVO salaryStrategy); |
| 前置条件 | 传入的薪水策略符合规范 |
| 后置条件 | 增加薪水策略持久化对象 |
| FinanceDataService  .findSalaryStrategy | 语法 | Public SalaryStrategyVO findSalaryStrategy(); |
| 前置条件 | 无 |
| 后置条件 | 返回薪水策略 |
| FinanceDataService  .updateSalaryStrategy | 语法 | Public boolean updateSalaryStrategy(SalaryStrategyVO salaryStrategy); |
| 前置条件 | 传入的薪水策略符合规范 |
| 后置条件 | 查找薪水策略并修改 |
| 需要的服务（需接口） | | |
| 服务名 | 服务 | |
| FinanceDataService  .addReceipt(ReceiptPo Receipt): boolean | 增加收款单的单一持久化对象 | |
| FinanceDataService  .findReceipt(Calendar date):List<ReceiptPo> | 根据输入的时间查找收款单的单一持久化对象，返回所有符合条件的收款单的列表 | |
| FinanceDataService  .deleteReceipt(long id):boolean | 根据输入的id查找收款单并删除该单一持久化对象 | |
| FinanceDataService  .updateReceipt(ReceiptPo receipt):boolean | 根据输入的收款单id查找收款单并用传入的收款单替换该单一持久化对象 | |
| FinanceDataService  .addPayment(PaymentPo payment):Boolean | 增加新的付款单单一持久化对象 | |
| FinanceDataService  .findPayment(Calendar date):List<PaymentPo> | 根据输入的date查找符合条件的付款单单一持久化对象 | |
| FinanceDataService  .deletePayment(long id):boolean | 根据输入的id查找到相应的单一持久化对象并删除 | |
| FinanceDataService  .updatePayment(PaymentPo Payment):boolean | 根据传入的PaymentPo的id查找到该单一持久化对象并替换 | |
| FinanceDataService  .addFreight(FreightPo freight):boolean | 增加新的freight单一持久化对象 | |
| FinanceDataService  .findFreight(Calendar date):List<FreightPo> | 根据输入的日期查找符合条件的单一持久化对象 | |
| FinanceDataService  .deleteFreight(long id):boolean | 按id查找相应的FreghtPo结果并删除 | |
| FinanceDataService  .updateFreight(FreightPo freight):boolean | 查找到相应的FreightPo并用传入的Freight替换 | |
| FinanceDataService  .findEarning(Calendar date):List<EarningPo> | 按date进行查找返回符合条件的成本收益表单一持久化对象 | |
| FinanceDataService  .deleteEarning(long id):boolean | 按id查找相应的持久化对象并删除 | |
| FinanceDataService  .updateEarning(EarningPo earning) | 查找到相应的EarningPo并用传入的Earning替换 | |
| FinanceDataService  .addAccount(AccountPo account):boolean | 增加新的银行账户单一持久化对象 | |
| FinanceDataService  .findAccount(long id):AccountPo | 按id进行查找返回相应的AccountPo结果 | |
| FinanceDataService  .deleteAccount(long id):boolean | 按id查找到相应的AccountPo单一持久化对象结果并删除 | |
| FinanceDataService  .updateAccount(AccountPo account):boolean | 按id查找到AccountPo的单一持久化对象并替换 | |
| FinanceDataService  .addInitialInfo(InitialInfoPo InitialInfo):Boolean | 增加新的期初建账持久化对象 | |
| FinanceDataService  .addPriceStrategy(PriceStrategyPo PriceStrategy):boolean | 增加新的运费策略持久化对象 | |
| FinanceDataService  .updatePriceStrategy(PriceStrategyPo PriceStrategy):boolean | 修改运费策略持久化对象 | |
| FinanceDataService  .findPriceStrategy():PriceStrategyPo | 查看运费策略持久化对象 | |
| FinanceDataService  .updateSalaryStrategy(SalaryStrategyPO):boolean | 修改工资策略持久化对象 | |
| FinanceDataService  .findSalaryStrategy():SalaryStrategyPO | 查看工资策略持久化对象 | |
| FinanceDataService  .addSalaryStrategy(SalaryStrategyPO salaryStrategy):boolean | 增加工资策略持久化对象 | |