**易事装中间层接口定义**

**接口技术：**

接口是使用HTTP来实现，接口是以POST或GET参数到指定URL后服务器返回Json数据包。

**接口说明：**

以下接口省略前段URL如：  
http://192.168.0.199/Services/Unit/Get  
直接去掉前面一段使用Unit/Get来进行说明

2、以下所有查询接口都有pageSize（分页大小）、pageNum（第几页）参数，默认pageSize=0，pageNum=0

# 目录

[目录 1](#_Toc21581)

[1. 数据库基表定义 3](#_Toc27348)

[1.1. 数据库表 3](#_Toc3864)

[1.2. 数据库表基本表 4](#_Toc13810)

[1.3. 数据库带地址信息的表 4](#_Toc25829)

[2. 数据库表结构 5](#_Toc4664)

[2.1. 模块功能表 5](#_Toc11691)

[2.2. 员工模块功能权限表 5](#_Toc15165)

[2.3. 公司信息表 6](#_Toc21561)

[2.4. 部门表 7](#_Toc24541)

[2.5. 职位表 8](#_Toc9756)

[2.6. 员工表 8](#_Toc4361)

[2.7. 单位表 10](#_Toc27572)

[2.8. 商品类型表 11](#_Toc19471)

[2.9. 供应商资料表 11](#_Toc20952)

[2.10. 仓库登记表 12](#_Toc9252)

[2.11. 商品资料表 13](#_Toc14779)

[2.12. 销售价格管理（商品资料子表） 15](#_Toc27956)

[2.13. 库存表 15](#_Toc26874)

[2.14. 库存初始化表 16](#_Toc6729)

[2.15. 库存警告设置表 16](#_Toc8940)

[3. 接口 17](#_Toc21506)

[3.1. 查询地址信息 17](#_Toc15795)

[3.2. 添加公司信息 17](#_Toc12276)

[3.3. 修改公司信息 18](#_Toc6307)

[3.4. 删除公司信息 18](#_Toc30703)

[3.5. 查询公司信息 18](#_Toc18159)

[3.6. 添加部门、分店信息 19](#_Toc10357)

[3.7. 修改部门、分店信息 19](#_Toc6339)

[3.8. 删除部门、分店信息 19](#_Toc18409)

[3.9. 查询部门、分店信息 19](#_Toc15925)

[3.10. 查询部门、分店分页信息 20](#_Toc1091)

[3.11. 添加职位架构信息 20](#_Toc23219)

[3.12. 修改职位架构信息 20](#_Toc13128)

[3.13. 删除职位架构信息 20](#_Toc21711)

[3.14. 查询职位架构信息 21](#_Toc29076)

[3.15. 查询职位架构分页信息 21](#_Toc19878)

[3.16. 添加人事档案 21](#_Toc16129)

[3.17. 修改人事档案 22](#_Toc17452)

[3.18. 删除人事档案 22](#_Toc11050)

[3.19. 查询人事档案 22](#_Toc19488)

[3.20. 查询人事档案分页信息 22](#_Toc31516)

[3.21. 员工登录 23](#_Toc15746)

[3.22. 员工修改密码 23](#_Toc14704)

[3.23. 员工退出登录 23](#_Toc10422)

[3.24. 员工查看自己信息 24](#_Toc27752)

[3.25. 员工查看自己权限 24](#_Toc31927)

[3.26. 员工修改自己基本信息 24](#_Toc18885)

[3.27. 添加计量单位 24](#_Toc30741)

[3.28. 修改计量单位 25](#_Toc15981)

[3.29. 删除计量单位 25](#_Toc642)

[3.30. 查询计量单位 25](#_Toc4120)

[3.31. 查询计量单位分页信息 25](#_Toc15429)

[3.32. 添加商品类别 26](#_Toc20898)

[3.33. 修改商品类别 26](#_Toc14798)

[3.34. 删除商品类别 26](#_Toc26293)

[3.35. 查询商品类别 26](#_Toc21807)

[3.36. 查询商品类别分页信息 27](#_Toc31200)

[3.37. 添加供应商资料 27](#_Toc3167)

[3.38. 修改供应商资料 27](#_Toc8286)

[3.39. 删除供应商资料 28](#_Toc23133)

[3.40. 查询供应商资料 28](#_Toc279)

[3.41. 查询供应商资料分页信息 28](#_Toc10935)

[3.42. 添加仓库登记 28](#_Toc16542)

[3.43. 修改仓库登记 29](#_Toc16898)

[3.44. 删除仓库登记 29](#_Toc16960)

[3.45. 查询仓库登记 29](#_Toc2875)

[3.46. 查询仓库登记分页信息 29](#_Toc3674)

[3.47. 添加商品资料 30](#_Toc31034)

[3.48. 修改商品资料 30](#_Toc28818)

[3.49. 删除商品资料 30](#_Toc25746)

[3.50. 查询商品资料 31](#_Toc13481)

[3.51. 查询商品资料分页信息 31](#_Toc20882)

[3.52. 添加销售价格管理（商品资料子表） 31](#_Toc24313)

[3.53. 修改销售价格管理（商品资料子表） 32](#_Toc10062)

[3.54. 删除销售价格管理（商品资料子表） 32](#_Toc26086)

[3.55. 查询销售价格管理（商品资料子表） 32](#_Toc26571)

[3.56. 查询销售价格管理（商品资料子表）分页信息 32](#_Toc23492)

[3.57. 添加库存表 33](#_Toc15054)

[3.58. 修改库存表 33](#_Toc13571)

[3.59. 删除库存表 33](#_Toc26587)

[3.60. 查询库存表 33](#_Toc29853)

[3.61. 查询库存表分页信息 34](#_Toc354)

[3.62. 添加仓库初始化 34](#_Toc6431)

[3.63. 修改仓库初始化 34](#_Toc21045)

[3.64. 删除仓库初始化 34](#_Toc14015)

[3.65. 查询仓库初始化 35](#_Toc26545)

[3.66. 查询仓库初始化分页信息 35](#_Toc20315)

[3.67. 添加库存警告设置表 35](#_Toc9802)

[3.68. 修改库存警告设置表 36](#_Toc16227)

[3.69. 删除库存警告设置表 36](#_Toc856)

[3.70. 查询库存警告设置表 36](#_Toc3818)

[3.71. 查询库存警告设置表分页信息 36](#_Toc5712)

# 数据库基表定义

## 数据库表

/// <summary>

/// 数据库表

/// </summary>

public class TbId

{

public Guid Id ;

}

## 数据库表基本表

/// <summary>

/// 数据库表基本表

/// </summary>

public class TbBase:TbId

{

public DateTime ModDate;

public DateTime UploadDate;

public bool Del ;

public bool TrashFlag ;

public string ExHost ;

}

## 数据库带地址信息的表

public class TbLocation:TbBase

{

/// <summary>

/// 地址,具体到街道

/// </summary>

public string Addr ;

/// <summary>

/// 位置，具体第几号第几房

/// </summary>

public string Location ;

/// <summary>

/// 地址代码,具体到街道的地址代码

/// </summary>

public string AddrCode ;

/// <summary>

/// 纬度

/// </summary>

public double Lng ;

/// <summary>

/// 经度

/// </summary>

public double Lat ;

}

# 数据库表结构

## 模块功能表

/// <summary>

/// 模块功能

/// </summary>

public class ModFunc : TbBase

{

/// <summary>

/// 功能所属类全称

/// </summary>

public virtual string Cls { get; set; }

/// <summary>

/// 功能函数名称,可以存储多个用|隔开,如 |Add|Update|Del|Get|

/// </summary>

public virtual string FuncNames { get; set; }

}

## 员工模块功能权限表

/// <summary>

/// 员工模块功能权限,在页面的Add、Update、Del等功能时做控制

/// </summary>

public class EmpModFunc : TbBase

{

/// <summary>

/// 绑定员工

/// </summary>

public virtual Employee Emp { get; set; }

/// <summary>

/// 绑定模块功能

/// </summary>

public virtual ModFunc ModFunc { get; set; }

/// <summary>

/// 功能函数名称,可以存储多个用|隔开,如 |Add|Update|Del|Get|

/// </summary>

public virtual string FuncNames { get; set; }

}

## 公司信息表

/// <summary>

/// 公司信息

/// </summary>

public class Company : TbLocation

{

/// <summary>

/// 公司名称

/// </summary>

public string CnName ;

/// <summary>

/// 英文名

/// </summary>

public string EnName ;

/// <summary>

/// 公司的英文简称。

/// </summary>

public string CompCode ;

/// <summary>

/// 公司简称。

/// </summary>

public string CompShortTitle ;

/// <summary>

/// 最高职位代码

/// </summary>

public string TopPosCode ;

/// <summary>

/// 最高职位名称

/// </summary>

public string TopPosName ;

/// <summary>

/// 邮编

/// </summary>

public string Zipcode ;

/// <summary>

/// 电话

/// </summary>

public string Tel ;

/// <summary>

/// 传真

/// </summary>

public string Fax ;

/// <summary>

/// 电子邮箱

/// </summary>

public string Mail ;

/// <summary>

/// 网站

/// </summary>

public string Web ;

}

## 部门表

/// <summary>

/// 部门

/// </summary>

public class Department : TbLocation

{

/// <summary>

/// 上一级部门ID

/// </summary>

public Guid PId ;

/// <summary>

/// 代码

/// </summary>

public string Code ;

/// <summary>

/// 名称

/// </summary>

public string Name ;

public string Phone1;

public string Phone2;

public string Phone3;

public string ContactName;

public string ContactPhone;

public string DeptType;

public int OrderID ;

}

## 职位表

/// <summary>

/// 职位

/// </summary>

public class Position : TbBase

{

/// <summary>

/// 上一级部门ID

/// </summary>

public Guid PId ;

/// <summary>

/// 代码

/// </summary>

public string Code ;

/// <summary>

/// 名称

/// </summary>

public string Name ;

public Department Dept ;

public int OrderId ;

}

## 员工表

/// <summary>

/// 员工

/// </summary>

public class Employee : TbLocation

{

/// <summary>

/// 所属部门

/// </summary>

public Department Dept ;

/// <summary>

/// 所属职位

/// </summary>

public Position Pos ;

////分店

//public Shop Shop ;

/// <summary>

/// 员工编号

/// </summary>

public string Code ;

/// <summary>

/// 名称

/// </summary>

public string EmpName ;

public string NickName;

/// <summary>

/// 密码

/// </summary>

public string PwdWeb ;

/// <summary>

/// =0 离职，=1 入职，=2 试用，=3 转正，=4 在职

/// </summary>

public string State;

/// <summary>

/// 身份证

/// </summary>

public string IDCode ;

/// <summary>

/// 籍贯

/// </summary>

public string Native;

/// <summary>

/// 根据身份证信息自动计算,性别，false为女,true为男

/// </summary>

public string Sex ;

/// <summary>

/// 民族

/// </summary>

public string Nation;

public string Education;

public string Institution;

/// <summary>

/// 根据身份证信息自动计算,生日

/// </summary>

public DateTime Birthday ;

public DateTime ComeDate;

public DateTime StaffDate;

public DateTime LeaveDate;

public string Tel;

public string Mobile;

public string Tel1;

public string Tel2;

public string OtherAddress;

public string Resume;

public string EMail;

public string QQ;

public string IfPrincipal;

public string IfSysUser;

public string IfWork;

public string OneCardCode;

public DateTime OnDuty;

public DateTime OffDuty;

}

## 单位表

/// <summary>

/// 单位

/// </summary>

public class Unit : TbBase

{

/// <summary>

/// 单位名称

/// </summary>

public string UnitName ;

/// <summary>

/// 单位描述

/// </summary>

public string UnitDesc ;

}

## 商品类型表

/// <summary>

/// 商品类型

/// </summary>

public class GoodsType : TbBase

{

/// <summary>

/// 上级ID

/// </summary>

public Guid PId ;

/// <summary>

/// 单位描述

/// </summary>

public string UnitDesc ;

/// <summary>

/// 树状结构的线性编码,采用形如：001001001的结构来编码

/// </summary>

public string Xpath ;

/// <summary>

/// 类型编号

/// </summary>

public string GoodsTypeCode ;

/// <summary>

/// 类型名

/// </summary>

public string GoodsTypeName ;

}

## 供应商资料表

/// <summary>

/// 供应商资料

/// </summary>

public class VendorInfo : TbLocation

{

/// <summary>

/// 代码

/// </summary>

public string VendorCode ;

/// <summary>

/// 名称

/// </summary>

public string VendorName ;

/// <summary>

/// 联系人

/// </summary>

public string ContactName ;

/// <summary>

/// 电话

/// </summary>

public string Telephone;

/// <summary>

/// 传真

/// </summary>

public string Fax;

/// <summary>

/// 移动电话

/// </summary>

public string Mobile;

/// <summary>

/// 电子邮件

/// </summary>

public string Email;

/// <summary>

/// QQ

/// </summary>

public string QQ;

/// <summary>

/// 产地

/// </summary>

public string VendorArea;

/// <summary>

/// 供应商网址

/// </summary>

public string WebSite ;

}

## 仓库登记表

/// <summary>

/// 仓库

/// </summary>

public class Storehouse : TbLocation

{

/// <summary>

/// 仓库编号

/// </summary>

public string StoreCode ;

/// <summary>

/// 仓库名称

/// </summary>

public string StoreName ;

/// <summary>

/// 管理员

/// </summary>

public string Manager ;

/// <summary>

/// 部门

/// </summary>

public Department Dept ;

/// <summary>

/// 是否有负库存。0：否；1：允许负库存。

/// </summary>

public string IsNegative ;

}

## 商品资料表

/// <summary>

/// 商品资料表

/// </summary>

public class GoodsInfo : TbBase

{

/// <summary>

/// 商品编号

/// </summary>

public string GoodsCode ;

/// <summary>

/// 商品名称

/// </summary>

public string GoodsName ;

/// <summary>

/// 商品条码

/// </summary>

public string BarCode ;

/// <summary>

/// 商品类别ID。

/// </summary>

public GoodsType GoodsType ;

/// <summary>

/// 型号

/// </summary>

public string ModelNum ;

/// <summary>

/// 计量单位

/// </summary>

public string UnitName ;

/// <summary>

/// 包装数量/规格

/// </summary>

public string Spec ;

/// <summary>

/// 是否使用,是否使用。0：使用；1：停用。

/// </summary>

public string IsUse ;

/// <summary>

/// 销售价格

/// </summary>

public double SellPrice ;

/// <summary>

/// 安装价格

/// </summary>

public double SetupPrice ;

/// <summary>

/// 商品图片值

/// </summary>

public byte[] Photo ;

/// <summary>

/// 销售折扣

/// </summary>

public double Discount ;

/// <summary>

/// 供应商主键

/// </summary>

public VendorInfo Vendor ;

}

## 销售价格管理（商品资料子表）

/// <summary>

/// 销售价格管理（商品资料子表）

/// </summary>

public class SellPriceInfo : TbBase

{

public Storehouse Storehouse;

public GoodsInfo Goods;

public double SellPrice;

public double SetupPrice;

}

## 库存表

/// <summary>

/// 库存表

/// </summary>

public class Stock : TbBase

{

/// <summary>

/// 仓库

/// </summary>

public virtual Storehouse Storehouse { get; set; }

/// <summary>

/// 商品

/// </summary>

public virtual GoodsInfo Goods { get; set; }

/// <summary>

/// 数量

/// </summary>

public virtual double Quantity { get; set; }

}

## 库存初始化表

/// <summary>

/// 库存初始化表

/// </summary>

public class StockInit : TbBase

{

/// <summary>

/// 仓库

/// </summary>

public virtual Storehouse Storehouse { get; set; }

/// <summary>

/// 商品

/// </summary>

public virtual GoodsInfo Goods { get; set; }

/// <summary>

/// 数量

/// </summary>

public virtual double Quantity { get; set; }

}

## 库存警告设置表

/// <summary>

/// 库存警告设置表（库存数量在上限与下限之间时，销售时，报警提示；当达到下限时，通过ifStopSell判断是否停售，有停售提示）

/// </summary>

public class StorageAlarm : TbBase

{

/// <summary>

/// 仓库

/// </summary>

public virtual Storehouse Storehouse { get; set; }

/// <summary>

/// 商品

/// </summary>

public virtual GoodsInfo Goods { get; set; }

/// <summary>

/// 上限数量

/// </summary>

public virtual double UpperLimit { get; set; }

/// <summary>

/// 下限数量

/// </summary>

public virtual double LowerLimit { get; set; }

/// <summary>

/// 当达到下限时，是否停售商品

/// </summary>

public virtual string IfStopSell { get; set; }

}

# 接口

## 查询地址信息

接口：

AddrCode/Search

参数：

code 地址码，如查询省份则为空，查询广东省下的所有市则为44

返回：

AddrCode数组(Json)

## 添加公司信息

接口：

Company/Add

参数：

Company类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改公司信息

接口：

Company/Update

参数：

Company类结构

返回：

true 保存成功

## 删除公司信息

接口：

Company/Del

参数：

Company类结构

返回：

true 保存成功

## 查询公司信息

接口：

Company/Get

参数：

空

返回：

Company对象数组Json

## 添加部门、分店信息

接口：

Department/Add  
  
参数：

Department类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改部门、分店信息

接口：

Department/Update  
  
参数：

Department类结构

返回：

true 保存成功

## 删除部门、分店信息

接口：

Department/Del  
  
参数：

Department类结构

返回：

true 删除成功

## 查询部门、分店信息

接口：

Department/Get  
  
参数：

pId 上级部门的ID

code 部门代码

name 部门名

addr 部门地址

返回：

Department对象数组(Json)

## 查询部门、分店分页信息

接口：

Department/GetPageCount  
  
参数：

pId 上级部门的ID

code 部门代码

name 部门名

addr 部门地址

返回：

分页数

## 添加职位架构信息

接口：

Position/Add  
  
参数：

Position类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改职位架构信息

接口：

Position/Update  
  
参数：

Position类结构

返回：

true 保存成功

## 删除职位架构信息

接口：

Position/Del  
  
参数：

Position类结构

返回：

true 删除成功

## 查询职位架构信息

接口：

Position/Get  
  
参数：

pId 上级职位的ID

code 职位代码

name 职位名

返回：

Position对象数组(Json)

## 查询职位架构分页信息

接口：

Position/GetPageCount  
  
参数：

pId 上级职位的ID

code 职位代码

name 职位名

返回：

分页数

## 添加人事档案

接口：

Employee/Add  
  
参数：

Employee类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改人事档案

接口：

Employee/Update  
  
参数：

Employee类结构

返回：

true 保存成功

## 删除人事档案

接口：

Employee/Del  
  
参数：

Employee类结构

返回：

true 删除成功

## 查询人事档案

接口：

Employee/Get  
  
参数：

code 员工工号

empName 员工名称

nickName 昵称

返回：

Employee对象数组(Json)

## 查询人事档案分页信息

接口：

Employee/GetPageCount  
  
参数：

code 员工工号

empName 员工名称

nickName 昵称

返回：

分页数

## 员工登录

接口：

Employee/Login  
  
参数：

Employee类结构 只用到Code与PwdWeb字段

返回：

true/false

## 员工修改密码

接口：

Employee/UpdatePwd  
  
参数：

oldPwd 旧密码

newPwd 新密码

返回：

true/false

## 员工退出登录

接口：

Employee/Logout  
  
参数：

无

返回：

true/false

## 员工查看自己信息

接口：

Employee/GetMySelf  
  
参数：

无

返回：

Employee类结构对象（json）

## 员工查看自己权限

接口：

Employee/GetMyModFunc  
  
参数：

无

返回：

EmpModFunc对象数组（json）

## 员工修改自己基本信息

接口：

Employee/UpdateMySelf  
  
参数：

Employee类结构

返回：

true/false

## 添加计量单位

接口：

Unit/Add  
  
参数：

Unit类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改计量单位

接口：

Unit/Update  
  
参数：

Unit类结构

返回：

true 保存成功

## 删除计量单位

接口：

Unit/Del  
  
参数：

Unit类结构

返回：

true 删除成功

## 查询计量单位

接口：

Unit/Get  
  
参数：

unitName 单位名称

返回：

Unit对象数组(Json)

## 查询计量单位分页信息

接口：

Unit/GetPageCount  
  
参数：

unitName 单位名称

返回：

分页数

## 添加商品类别

接口：

GoodsType/Add  
  
参数：

GoodsType类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改商品类别

接口：

GoodsType/Update  
  
参数：

GoodsType类结构

返回：

true 保存成功

## 删除商品类别

接口：

GoodsType/Del  
  
参数：

GoodsType类结构

返回：

true 删除成功

## 查询商品类别

接口：

GoodsType/Get  
  
参数：

pId 上一级商品类别

unitDesc 单位描述

goodsTypeCode 类型编码

goodsTypeName 类型名称

返回：

GoodsType对象数组(Json)

## 查询商品类别分页信息

接口：

GoodsType/GetPageCount  
  
参数：

pId 上一级商品类别

unitDesc 单位描述

goodsTypeCode 类型编码

goodsTypeName 类型名称

返回：

分页数

## 添加供应商资料

接口：

VendorInfo/Add  
  
参数：

VendorInfo类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改供应商资料

接口：

VendorInfo/Update  
  
参数：

VendorInfo类结构

返回：

true 保存成功

## 删除供应商资料

接口：

VendorInfo/Del  
  
参数：

VendorInfo类结构

返回：

true 删除成功

## 查询供应商资料

接口：

VendorInfo/Get  
  
参数：

vendorCode 供应商代码

vendorName 供应商名称

返回：

VendorInfo对象数组(Json)

## 查询供应商资料分页信息

接口：

VendorInfo/GetPageCount  
  
参数：

vendorCode 供应商代码

vendorName 供应商名称

返回：

分页数

## 添加仓库登记

接口：

Storehouse/Add  
  
参数：

Storehouse类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改仓库登记

接口：

Storehouse/Update  
  
参数：

Storehouse类结构

返回：

true 保存成功

## 删除仓库登记

接口：

Storehouse/Del  
  
参数：

Storehouse类结构

返回：

true 删除成功

## 查询仓库登记

接口：

Storehouse/Get  
  
参数：

storeCode 仓库代码

storeName 仓库名称

deptID 部门ID

返回：

Storehouse对象数组(Json)

## 查询仓库登记分页信息

接口：

Storehouse/GetPageCount  
  
参数：

storeCode 仓库代码

storeName 仓库名称

deptID 部门ID

返回：

分页数

## 添加商品资料

接口：

GoodsInfo/Add  
  
参数：

GoodsInfo类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改商品资料

接口：

GoodsInfo/Update  
  
参数：

GoodsInfo类结构

返回：

true 保存成功

## 删除商品资料

接口：

GoodsInfo/Del  
  
参数：

GoodsInfo类结构

返回：

true 删除成功

## 查询商品资料

接口：

GoodsInfo/Get  
  
参数：

goodsCode 商品资料代码

goodsName 商品资料名称

barCode 商品条码

goodsTypeID 商品类型ID

modelNum 型号

返回：

GoodsInfo对象数组(Json)

## 查询商品资料分页信息

接口：

GoodsInfo/GetPageCount  
  
参数：

goodsCode 商品资料代码

goodsName 商品资料名称

barCode 商品条码

goodsTypeID 商品类型ID

modelNum 型号

返回：

分页数

## 添加销售价格管理（商品资料子表）

接口：

SellPriceInfo/Add  
  
参数：

SellPriceInfo类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改销售价格管理（商品资料子表）

接口：

SellPriceInfo/Update  
  
参数：

SellPriceInfo类结构

返回：

true 保存成功

## 删除销售价格管理（商品资料子表）

接口：

SellPriceInfo/Del  
  
参数：

SellPriceInfo类结构

返回：

true 删除成功

## 查询销售价格管理（商品资料子表）

接口：

SellPriceInfo/Get  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

SellPriceInfo对象数组(Json)

## 查询销售价格管理（商品资料子表）分页信息

接口：

SellPriceInfo/GetPageCount  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

分页数

## 添加库存表

接口：

Stock/Add  
  
参数：

Stock类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改库存表

接口：

Stock/Update  
  
参数：

Stock类结构

返回：

true 保存成功

## 删除库存表

接口：

Stock/Del  
  
参数：

Stock类结构

返回：

true 删除成功

## 查询库存表

接口：

Stock/Get  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

Stock对象数组(Json)

## 查询库存表分页信息

接口：

Stock/GetPageCount  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

分页数

## 添加仓库初始化

接口：

StockInit/Add  
  
参数：

StockInit类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改仓库初始化

接口：

StockInit/Update  
  
参数：

StockInit类结构

返回：

true 保存成功

## 删除仓库初始化

接口：

StockInit/Del  
  
参数：

StockInit类结构

返回：

true 删除成功

## 查询仓库初始化

接口：

StockInit/Get  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

StockInit对象数组(Json)

## 查询仓库初始化分页信息

接口：

StockInit/GetPageCount  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

分页数

## 添加库存警告设置表

接口：

StorageAlarm/Add  
  
参数：

StorageAlarm类结构

返回：

"478FF722-746B-4CDD-B91E-574B9B2265C7" 保存后的ID

## 修改库存警告设置表

接口：

StorageAlarm/Update  
  
参数：

StorageAlarm类结构

返回：

true 保存成功

## 删除库存警告设置表

接口：

StorageAlarm/Del  
  
参数：

StorageAlarm类结构

返回：

true 删除成功

## 查询库存警告设置表

接口：

StorageAlarm/Get  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

StorageAlarm对象数组(Json)

## 查询库存警告设置表分页信息

接口：

StorageAlarm/GetPageCount  
  
参数：

storehouseID 仓库ID

goodsID 商品ID

返回：

分页数