项目第十天 统计分析

课程安排：

采购单药品明细采购金额总计

本系统统计分析需求：

交易明细查询

按XXXX(药品、医院、供货商)分类统计

统计分析功能开发：

多表关联查询统计速度很慢

将多表关联数据聚合到一张表中查询。

了解统一统计分析解决方案

开发交易明细查询

开发按XXX分类统计

介绍DWR的使用。

# 采购单药品明细采购金额总计

## 需求

用户在创建采购单，页面上明显统计出采购单中所有药品采购总量、采购金额总量。



总计范围：符合查询条件的所有记录的采购量、采购金额。

## 实现

dao

统计符合查询条件的采购单药品明细的采购量、采购金额。

实现思路：

对符合查询条件的采购单明细列表的采购量、采购金额求总和，注意不是对一页的数据求总计 。

select sum(nvl(yycgdmx.cgl, 0)) cgl, sum(nvl(yycgdmx.cgje, 0)) cgje

from yycgdmx2014 yycgdmx, yycgd2014 yycgd, useryy, ypxx, usergys

where yycgdmx.yycgdid = yycgd.id

and yycgd.useryyid = useryy.id

and yycgdmx.ypxxid = ypxx.id

and yycgdmx.usergysid = usergys.id

*---指定采购单的id*

and yycgdmx.yycgdid='2014101220'

mapper.xml

<!-- 采购单明细总计（采购量、采购金额） -->

<select id=*"findYycgdmxListSum"* parameterType=*"yycg.business.pojo.vo.YycgdQueryVo"*

resultType=*"yycg.business.pojo.vo.YycgdmxCustom"*>

select sum(nvl(yycgdmx.cgl, 0)) cgl, sum(nvl(yycgdmx.cgje, 0)) cgje

from yycgdmx${businessyear} yycgdmx, yycgd${businessyear} yycgd, useryy, ypxx, usergys

where yycgdmx.yycgdid = yycgd.id

and yycgd.useryyid = useryy.id

and yycgdmx.ypxxid = ypxx.id

and yycgdmx.usergysid = usergys.id

<!-- 只查询某个采购单下药品明细 -->

<include refid=*"query\_yycgdmx\_where"* />

<!-- 采购单查询条件 -->

<include refid=*"query\_yycgd\_where"* />

<!-- 药品查询条件 -->

<include refid=*"yycg.business.dao.mapper.YpxxMapperCustom.query\_ypxx\_where"* />

</select>

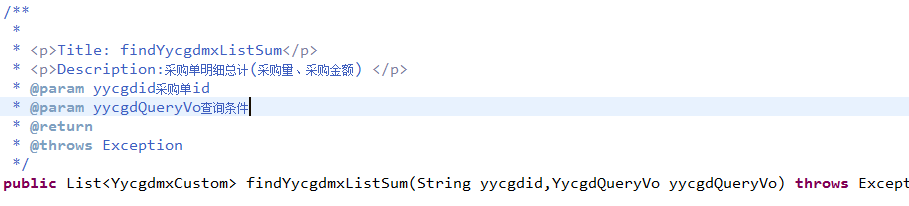
Mapper.java



service

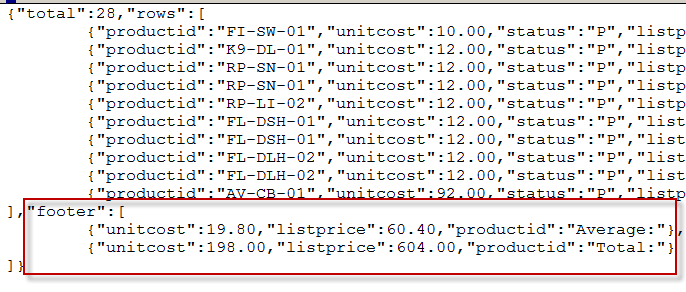
接口功能：采购单明细总计

接口参数：采购单id、查询条件



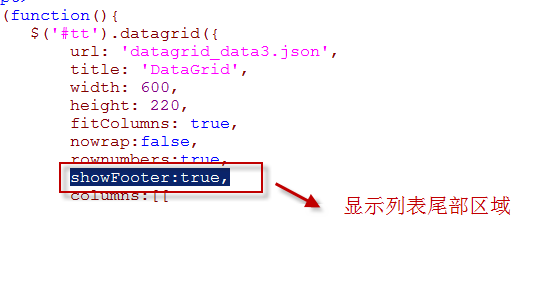
在datagrid中展示总计行

Datagird加载 的json中定义footer



Footer：在列表的最下边显示一块儿区域。

在加载 datagrid中指定显示footer



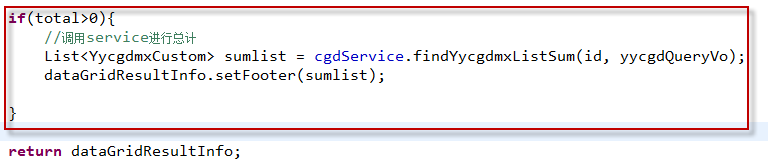
action

在采购单修改action方法中，在采购单明细结果集json方法中，添加取总计结果集代码，将总计结果集添加datagrid的footer中。

实现方法：

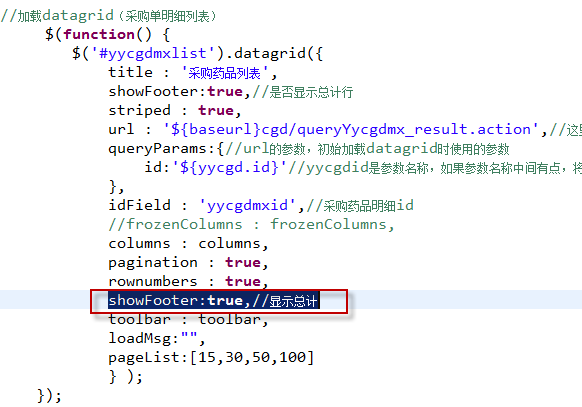
在统一数据列表结果类(DataGridResultInfo)添加属性footer。

修改：CgdAction.java中queryYycgdmx\_result方法。



页面

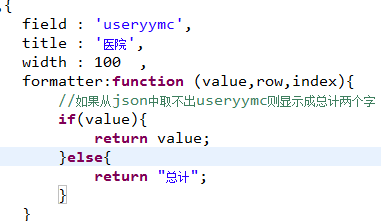
在加载明细列表的方法中，添加：



在datagrid中对总计行的显示样式和普通行的显示样式不一样，如何区别？

使用普通行中不为空的字段(列)，作为区别的标记。

例如：



# 交易明细查询

## 需求

医院、供货商、监管单位通过交易明细查询功能，查询出指定时间段内历史 交易清单。

监管单位，查询目的是为了监督。

医院、供货商查询目的，查询与自己相关交易信息。



## 分析

多表查询

主查询表：yycgdmx（采购单明细表）

关联查询表： useryy(内链接)、yycgd(内链接)、usergys(内链接)、ypxx(内链接)、入库信息表（外链接）、退货信息（外链接）、结算 信息(外链接)

select

yycgdmx.\*,

yycgdrk.rkl,

yycgdrk.rkje,

yythdmx.thl,

yythdmx.thje,

yyjsdmx.jsl,

yyjsdmx.jsje

from (

select

useryy.mc useryymc,

yycgd.bm yycgdbm,

yycgd.id yycgdid,

usergys.mc usergysmc,

yycgdmx.ypxxid,

ypxx.\*,

yycgdmx.cgl,

yycgdmx.cgje

from yycgdmx2014 yycgdmx, yycgd2014 yycgd, useryy,usergys,ypxx

where yycgdmx.yycgdid = yycgd.id

and yycgd.useryyid = useryy.id

and yycgdmx.usergysid = usergys.id

and yycgdmx.ypxxid = ypxx.id

)yycgdmx

*--外链接入库信息*

left join yycgdrk2014 yycgdrk on yycgdrk.yycgdid = yycgdmx.yycgdid and yycgdrk.ypxxid = yycgdmx.ypxxid

*--外链接退货信息*

left join yythdmx2014 yythdmx on yythdmx.yycgdid = yycgdmx.yycgdid and yythdmx.ypxxid = yycgdmx.ypxxid

*--外链接结算信息*

left join yyjsdmx2014 yyjsdmx on yyjsdmx.yycgdid = yycgdmx.yycgdid and yyjsdmx.ypxxid = yycgdmx.ypxxid

分析数据量：

采购明细表：上千万

入库信息：上千万

退货信息： 少

结算信息：上千万

三张大数据量的表关联查询，如果作统计，在三张表关联的基础上实现，速度慢，。

设想：如果将四张表的数据全部放在一张表，单表查询速度提高很大。

数据聚合意义

数据聚合的意义：

案例：

统计网站页面的访问量。

统计每一年每一个页面访问量。

实现方法：

数据采集：

完成原始数据（统计的来源数据）采集。

当用户点击页面时，需要记录用户点击信息（用户客户端ip、访问时间、访问页面的地址）

用户客户端ip：统计出用户来源于哪个地区

访问时间：统计出网站访问时段（比如哪些时间段访问量大）

访问页面的地址：统计出用户的喜好，哪些页面用户访问多。

如果对原始数据直接进行统计，速度很慢，因为原始数据量很大，所以说要对原始进行数据分析，分析后进行数据聚合。

数据聚合结果：

按分钟聚合：

将原始数据按分钟聚合后存储到单独的分钟聚合表：

聚合时执行一次小范围的统计。

时间（年、月、日、时、分）、XXX页面、访问量（这一分钟的访问总量）

时间（年、月、日、时、分）、XXX页面、访问量（这一分钟的访问总量）

按小时聚合：

将原始数据按小时聚合后存储到单独的小时聚合表：

从分钟聚合表中执行一次小范围的统计。

时间（年、月、日、时）、XXX页面、访问量（这一小时的访问总量）

时间（年、月、日、时）、XXX页面、访问量（这一小时的访问总量）

….

按天聚合：

近年聚合：

通常情况下，单独开发统计系统，后台开始定时任务进行专门数据采集、数据分析、数据聚合。

统计分析数据库：

数据聚合表

…

统计分析（系统、模块）

数据分析、数据聚合

数据采集

数据采集

原始数据库B

原始数据库A

本系统数据聚合方法

目标：将采购单明细信息、入库信息、退货信息、结算 信息聚合到一张表中，提高查询统计的速度。

设想：在采购单明细表中添加入库信息、退货信息。。。。字段？

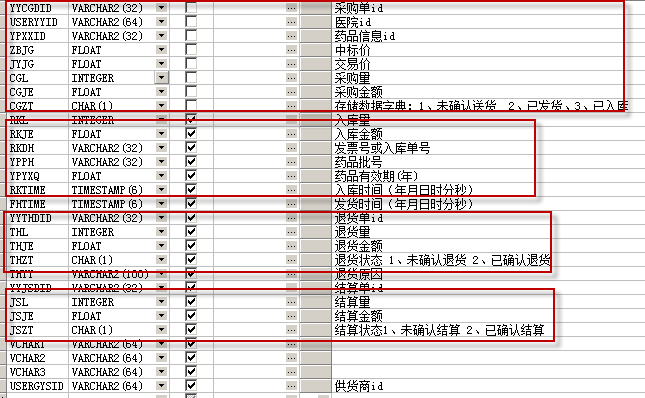
这种方法不合理，将统计分析业务功能和采购业务功能使用一张表存储信息，耦合性较强，可扩展性较差。

采用方案：

新创建一张交易明细表YYBUSINESS+4位年份：

记录采购明细、入 库信息、退货信息、结算 信息。

唯一约束：采购单id和药品id



聚合方法如下：

交易明细表：

YYBUSINESS

采购明细聚合：

* 聚合时机:

在数据不再修改时进行聚合。

在采购单审核通过后进行数据聚合。

* 聚合方法：

将采购单明细记录全部插入到交易明细表。

是修改service?通过触发器进行数据同步？

如果触发器：

在采购单基本表中创建更新触发器：

If(采购单状态为审核通过){（就是业务逻辑）

insert 交易明细表（。。。）

}

**如果触发器有业务逻辑，不推荐使用触发器。**

修改service实现数据聚合：

修改采购单审核service方法：

如果审核通过，将采购单明细记录全部插入到交易明细表。

采购明细

数据聚合

插入操作

入库信息聚合：

* 聚合时机

医院执行入库提交时进行数据聚合。

* 聚合方法

采用触发器：

在入库信息表创建插入触发器，内容是向交易明细表更新记录（入库量、入库金额、入库单号等入库信息）。

create or replace trigger yycgdrk2014\_insert after insert on yycgdrk2014 for each row

declare

begin

update Yybusiness2014 t

set t.rkl = :new.rkl,

t.rkje = :new.rkje,

t.ypph = :new.ypph,

t.cgzt = :new.cgzt,

t.rkdh = :new.rkdh,

t.ypyxq = :new.ypyxq,

t.rktime = :new.rktime

where t.yycgdid = :new.yycgdid

and t.ypxxid = :new.ypxxid;

end yycgdrk2014\_insert;

数据聚合

更新操作

入库信息：

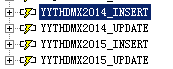
数据聚合

交易明细表：

YYBUSINESS

退货信息聚合：

使用触发器：



结算信息聚合：

使用触发器：



使用触发器进行数据同步

目标表

原始表

如果原始表和目标表在一个数据库中，建议使用触发器进行数据同步：

原始表插入时向目标表插入

原始表修改时更新目标表记录

原始表删除时删除目标表记录

注意：如果触发器仅仅数据同步，建议使用触发器，如果触发器中有业务逻辑，不建议使用触发器。

:new:在触发器中的一个关键字，表示新记录。

在insert和update触发器中使用。

在delete触发器中:old，:old表示旧数据。

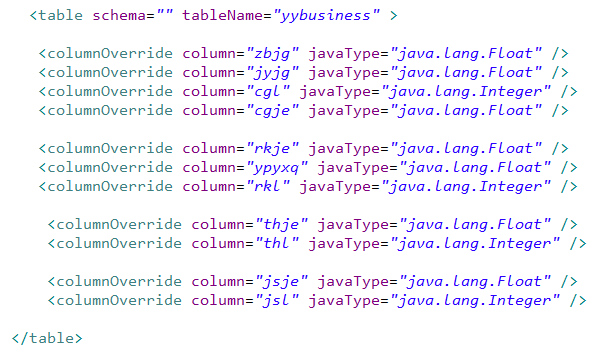
## 实现

数据聚合实现

第一步：采购明细的聚合

采购单明细表-----》交易明细表

* 交易明细表逆向工程



* 修改service saveYycgdCheckStatus方法

在审核通过时，将采购单明细记录插入到交易明细表中。



第二步：测试数据聚合结算

创建一个采购单，审核采购单通过、供货商受理、入库，流程结束后看交易表中记录是否聚合成功。。

### dao

监管单位查询：

查询管理地区内医院采购明细信息。

医院查询：

查询自己的采购明细信息

供货商查询：

与本供货商相关的采购明细信息

sql：

主查询表：交易明细表

关联查询表：医院信息、供货商信息、采购单主信息。。

select useryy.id useryyid,

useryy.mc useryymc,

yycgd.bm yycgdbm,

yycgd.id yycgdid,

usergys.id usergysid,

usergys.mc usergysmc,

yycgdmx.ypxxid,

ypxx.id,

ypxx.bm,

ypxx.mc,

ypxx.jx,

ypxx.gg,

ypxx.zhxs,

ypxx.scqymc,

ypxx.spmc,

ypxx.jyzt,

(select info

from dictinfo

where ypxx.jyzt = dictcode

and typecode = '003') jyztmc,

yycgdmx.cgl,

yycgdmx.cgje,

yycgdmx.rkl,

yycgdmx.rkje,

yycgdmx.thl,

yycgdmx.thje,

yycgdmx.jsl,

yycgdmx.jsje

from yybusiness2014 yycgdmx, yycgd2014 yycgd, useryy, usergys, ypxx

where yycgdmx.yycgdid = yycgd.id

and yycgd.useryyid = useryy.id

and yycgdmx.usergysid = usergys.id

and yycgdmx.ypxxid = ypxx.id

*--监管单位查询管理地区内医院采购明细信息*

and useryy.id in (

*--管理地区内医院*

select id from useryy where dq like '1.1.%')

*--医院查询自己的采购明细信息*

and useryy.id = '1f8b098b-067e-11e3-8a3c-0019d2ce5116'

*--供货商查询： 与本供货商相关的采购明细信息*

and usergys.id = '5197cdd2-08cf-11e3-8a4f-60a44cea4388'



service

接口功能：医院、监管单位、供货商查询交易明细

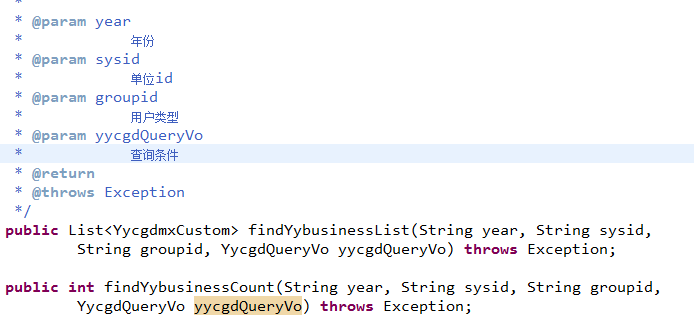
查询内容如下：

监管单位查询：查询管理地区内医院采购明细信息。

医院查询：查询自己的采购明细信息

供货商查询：与本供货商相关的采购明细信息

接口参数：年份、查询条件，单位id、用户类型（1：卫生局 2:卫生院 3：卫生室 4:供货商）



action

参考查询页面模版代码实现。

调试

分别以医院、供货商、监管单位登陆，操作交易明细列表。

测试前，准备测试数据（以不同区域的医院创建采购单）

# 分类统计

## 按药品分类统计

需求：

按药品统计指定时间段时采购量、采购金额、入库量、入库金额。。。。



分类统计：统计数据来源？原始数据是什么？

统计数据来源：交易明细列表，在交易明细的基础进行分类统计。

## 按供货商统计

需求：

按供货商统计指定时间段时采购量、采购金额、入库量、入库金额。。。。



## 按医院统计

需求：

按医院统计指定时间段时采购量、采购金额、入库量、入库金额。。。。



## 统计数据导出

建议：所有的统计功能，添加导出功能，将统计数据以excel方式导出。

## 按药品分类统计实现

dao

sql:

select

business.id,

business.bm,

business.mc,

sum(nvl(business.cgl,0))cgl,

sum(nvl(business.cgje,0))cgje,

sum(nvl(business.rkl,0))rkl,

sum(nvl(business.rkje,0))rkje,

sum(nvl(business.thl,0))thl,

sum(nvl(business.thje,0))thje,

sum(nvl(business.jsl,0))jsl,

sum(nvl(business.jsje,0))jsje

from (

select useryy.id useryyid,

useryy.mc useryymc,

yycgd.bm yycgdbm,

yycgd.id yycgdid,

usergys.id usergysid,

usergys.mc usergysmc,

yycgdmx.ypxxid,

ypxx.id,

ypxx.bm,

ypxx.mc,

ypxx.jx,

ypxx.gg,

ypxx.zhxs,

ypxx.scqymc,

ypxx.spmc,

ypxx.jyzt,

(select info

from dictinfo

where ypxx.jyzt = dictcode

and typecode = '003') jyztmc,

(select info

from dictinfo

where typecode = '011'

and dictcode = yycgdmx.cgzt) cgztmc,

yycgdmx.cgl,

yycgdmx.cgje,

yycgdmx.rkl,

yycgdmx.rkje,

yycgdmx.thl,

yycgdmx.thje,

yycgdmx.jsl,

yycgdmx.jsje

from yybusiness2014 yycgdmx, yycgd2014 yycgd, useryy, usergys, ypxx

where yycgdmx.yycgdid = yycgd.id

and yycgd.useryyid = useryy.id

and yycgdmx.usergysid = usergys.id

and yycgdmx.ypxxid = ypxx.id

*--监管单位查询管理地区内医院采购明细信息*

and useryy.id in (

*--管理地区内医院*

select id from useryy where dq like '1.1.%')

*--医院查询自己的采购明细信息*

and useryy.id = '1f8b098b-067e-11e3-8a3c-0019d2ce5116'

*--供货商查询： 与本供货商相关的采购明细信息*

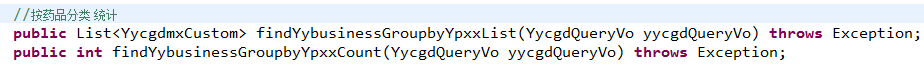
and usergys.id = '5197cdd2-08cf-11e3-8a4f-60a44cea4388'

)business

*--分类统计*

*--按药品统计*

group by business.id,business.bm,business.mc



service

接口功能：医院、监管单位、供货商按药品统计

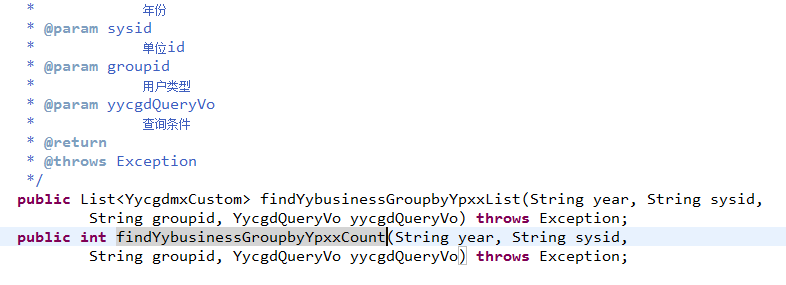
统计内容如下：

监管单位：对管理地区内医院采购明细进行统计。

医院：统计自己的采购明细统计

供货商：与本供货商相关的采购明细统计

接口参数：年份、查询条件，单位id、用户类型（1：卫生局 2:卫生院 3：卫生室 4:供货商）



action

参考查询页面模版代码实现。

# dwr使用方法

## 什么是dwr?

DWR（Direct Web Remoting）是一个用于改善web页面与Java类交互的远程服务器端Ajax开源框架,可以帮助开发人员开发包含AJAX技术的网站.它可以允许在浏览器里的代码使用运行在WEB服务器上的JAVA函数,就像它就在浏览器里一样。

## Dwr调用方法

普通ajax调用方法如下：

客户端(页面dwr的请求代码)

客户端(页面jsp，ajax的请求代码)

表现层（request,response）

业务层(service，业务方法)

Dwr调用：

业务层(service，业务方法)

Flex:用于flash编程，实现页面/视图层的开发 ，开发出的效果和网页样式不一样，更接近windows桌面的样式，开发的成果就是一个flash文件。

Flex实现了视图层：

Flex视图(UI)

Dwr请求

业务层(service，业务方法)

## Dwr的配置

Jar包：

如果是maven工程加入以下依赖：

<!-- dwr -->

<dependency>

<groupId>org.directwebremoting</groupId>

<artifactId>dwr</artifactId>

<version>3.0.M1</version>

</dependency>

如果不是maven工程则需要加入; dwr-3.0.M1.jar

Dwr servlet

在web.xml中加入：

<servlet>

<servlet-name>dwr-invoker</servlet-name>

<servlet-class>uk.ltd.getahead.dwr.DWRServlet</servlet-class>

<!-- 是否允许调试，如果要在浏览器中调试则必须设置为true -->

<init-param>

<param-name>debug</param-name>

<param-value>true</param-value>

</init-param>

<!-- 如果允许跨域请求，则必须将此值设置为false，默认值为true -->

<init-param>

<param-name>crossDomainSessionSecurity</param-name>

<param-value>false</param-value>

</init-param>

<init-param>

<param-name>allowScriptTagRemoting</param-name>

<param-value>true</param-value>

</init-param>

</servlet>

<servlet-mapping>

<servlet-name>dwr-invoker</servlet-name>

<url-pattern>/dwr/\*</url-pattern>

</servlet-mapping>

访问/dwr/交给dwr来处理。

在WEB-INF下配置dwr.xml文件：

配置dwr服务端哪些方法允许页面ajax调用。

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE dwr PUBLIC "-//GetAhead Limited//DTD Direct Web Remoting 2.0//EN" "http://getahead.ltd.uk/dwr/dwr30.dtd">

<dwr>

<allow>

<create creator=*"spring"* javascript=*"dwrService"*>

<param name=*"beanName"* value=*"dwrService"* />

</create>

</allow>

</dwr>

javascript=*"dwrService"*:

如果要调用dwr.xml配置的service(这里配置是名称为：*dwrService的bean*)的方法，需要在页面中引入*dwrService.js*

*Js是地址是：*

<script type='text/javascript' src='工程路径/dwr/interface/ dwrService.js'></script>

/dwr/interface/ dwrService.js交给dwr的servlet进行处理，servlet会根据dwr.xml中配置，生成一个js文件的内容。

<param name=*"beanName"* value=*"dwrService"* />：

Spring容器中名称为：*dwrService的bean*

## Dwr测试

调用yycg.base.service.impl.DwrServiceImpl里边的方法。

由于yycg.base.service.impl.DwrServiceImpl使用spring管理，在spring容器中配置：

<!-- dwrService -->

<bean id=*"dwrService"* class=*"yycg.base.service.impl.DwrServiceImpl"*/>

在页面中加入dwrService的js引用，可以调用上边dwrService这个bean中的方法。

<script type='text/javascript' src='工程路径/dwr/interface/ dwrService.js'></script>

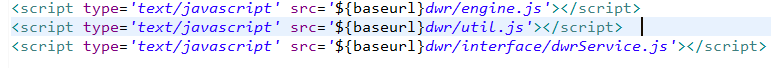
如果要调用其它bean的方法，还引入:

<script type='text/javascript' src='工程路径/dwr/interface/ XXXXXXXX.js'></script>

都 必须引用dwr的引擎配置js:

<script type=*'text/javascript'* src=*'*${baseurl}*dwr/engine.js'*></script>

<script type=*'text/javascript'* src=*'*${baseurl}*dwr/util.js'*></script>



测试如下：

调用无参数方法：

//dwrService是加载js的名称

//testdwr是dwrService对应的bean的方法名

//callback：回调方法

dwrService.testdwr({

callback:**function**(data) {

alert(data);

}});

调用有参数方法：

dwrService.testdwr2('张三',{

callback:**function**(data) {

alert(data);

}});

## Dwr在本系统的应用

1. 在年份下框显示数据，通过dwr来获取

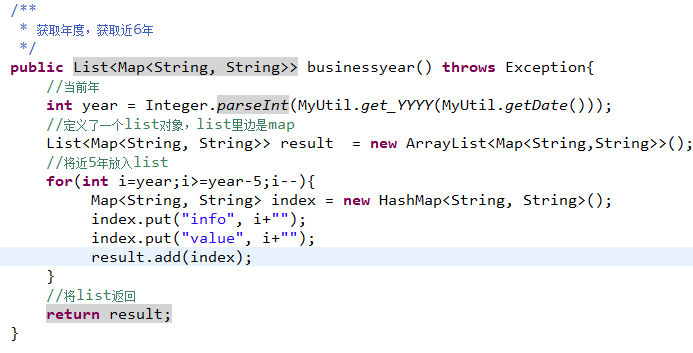
需求，显示近6年的年份。

需要在dwrService写一个方法：

获取近6年的年份

客户端通过dwr调用该方法，得到近6年的年份，在页面向年份下拉框设置6年的年份。

如果通过dwr方法，在action方法无需调用service获取数据，通过model传到页面。

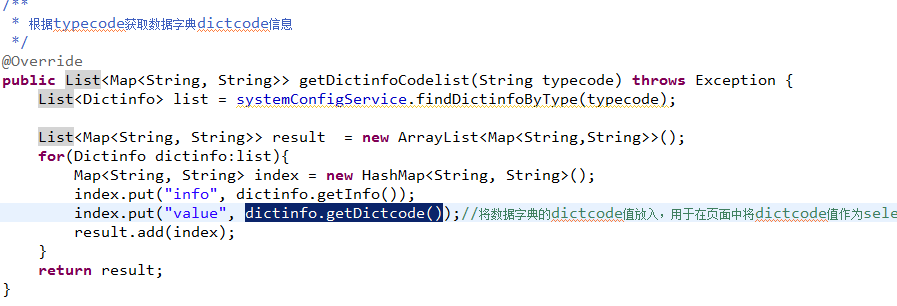


客户端调用上边的方法，将list转成json在客户端解析。

1. 在所有数据字典下拉框, 通过dwr来获取数据字典信息

针对业务代码：

编写根据typecode获取业务代码及代码对应的名称

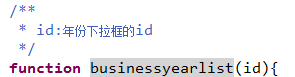


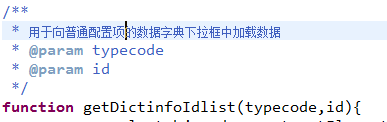
针对普通配置项：

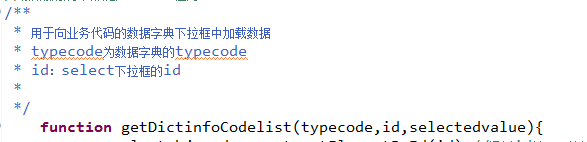
编写根据typecode获取普通配置项id及对应的名称



客户端使用固定的三个js方法，解析上边三个方法返回数据，解析完数据，将数据填充到select下拉框。







# 作业

## 交易明细查询添加总计

效果图：



## 按供货商统计

## 按医院统计