

内容

1. 开发环境的准备	2
2. DB 环境的准备	4
3. 菜单	5
4. 多层对话框	6
5. 动态画面	7
6. 日历	7
7. 防止重复登录的画面锁定	9
8. 独自开发的输入检查	11
8. 1 输入项目检查	11
8. 2 上传文件检查	12
8. 3 画像文字认证	13
9. 画面迁移	14
10. 数据映射	15
11. Token 检查	17
12. 自作的数据库	17
12. 1 DB 的连接池管理	17
12. 2 动态 XML 的 SQL 文解析	19
12. 3 Access 类的 DB 操作	21
12. 4 Transact 类的 DB 操作	22
12. 5 支持五种类型的数据库	24
13. 文件上传	25
13. 1 单个文件上传 (Post 方式)	25
13. 2 多个文件上传 (Ajax 方式)	26
14. Excel 文件输出	27
15. PDF 文件及条形码的输出	29
16. 检索和分页	29
17. 多国语言	33
18. 和宁静 DB 工具配合	35

宁静框架软件用户手册

宁静框架软件(njweb)利用最新的 J2EE 技术, 紧密连接各个部品, 能够有效地提高 Web 软件项目的开发效率。

下面通过利用宁静框架软件开发的实例 (njsam) 来进行详细说明, 这里也推荐用户通过对实例 (njsam) 的改造来进行实际开发。

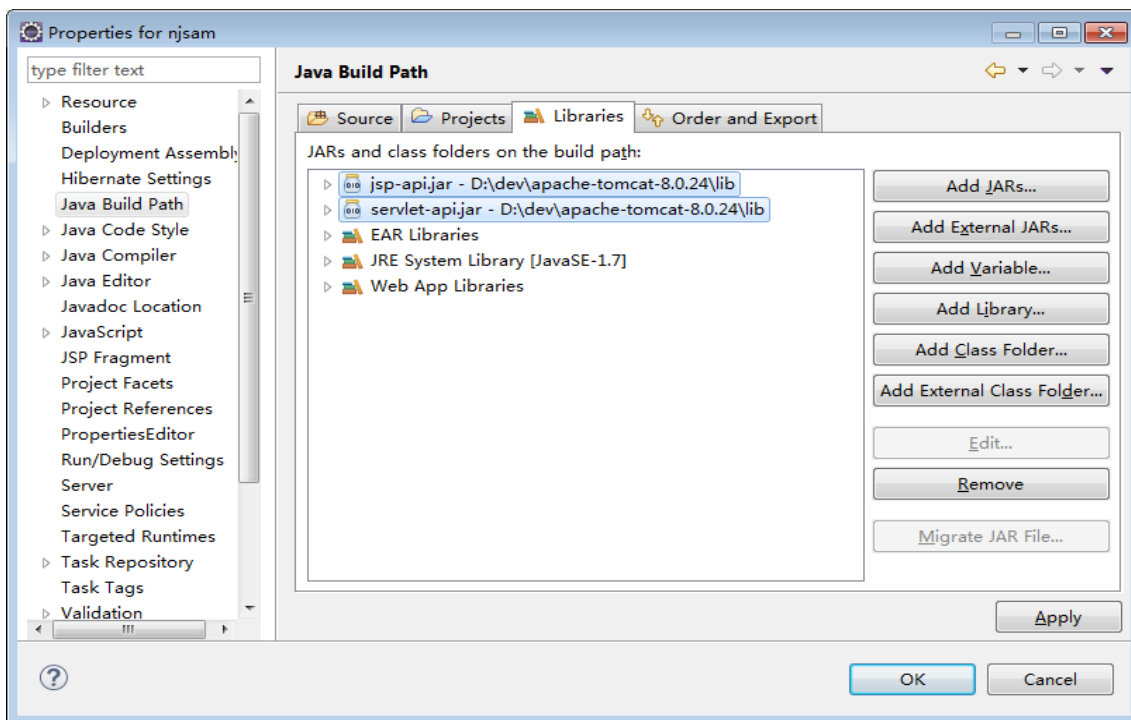
1. 开发环境的准备

1) 宁静框架软件在以下的外部环境及版本下确认通过。

- Eclipse Indigo Service Release 2 / Mars Release (4.5.0)
- JDK 1.7 / 1.8
- Tomcat 7 / 8
- PostgreSQL 9 x64
- MySql 5.6
- SqlServer 2008 R2
- Oracle 11g
- DB2 Express-C 10.1

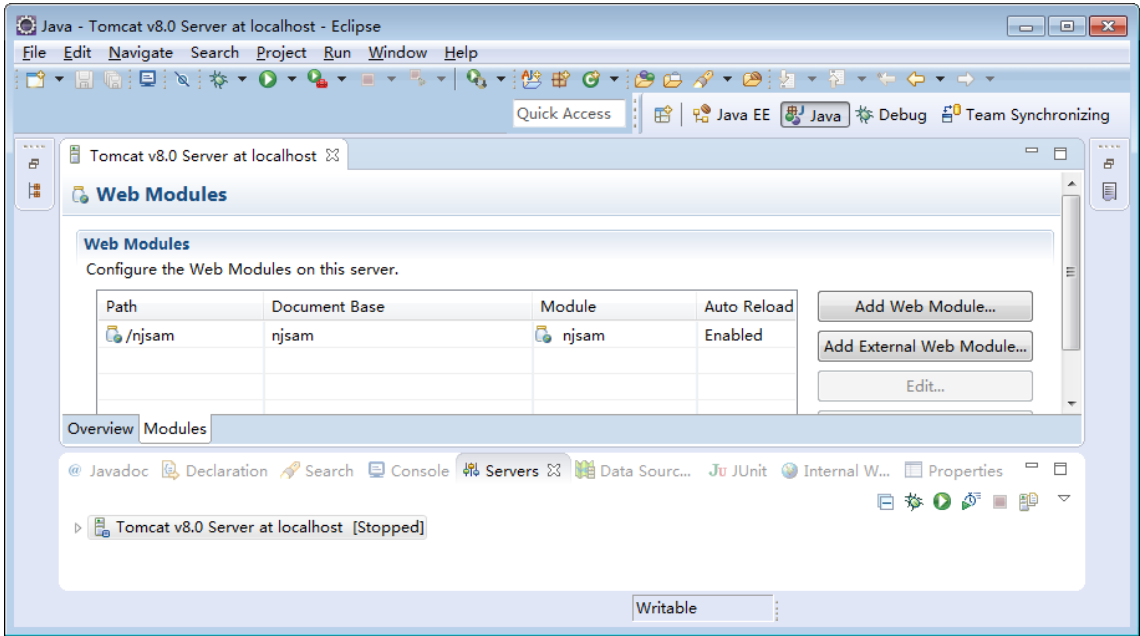
2) 用上面的版本或比上面更高版本构筑外部环境, 解压 njweb.1.00_chinese.zip 文件, 将里面的 njsam 目录作为 Eclipse 的已经存在的项目, 导入到 Eclipse 开发环境中。

导入后, 对 jsp-api.jar 和 servlet-api.jar 的所在目录进行变更, 变更到本地 Tomcat 的 lib 目录下。(图 1-1 Jar 文件所在目录变更)



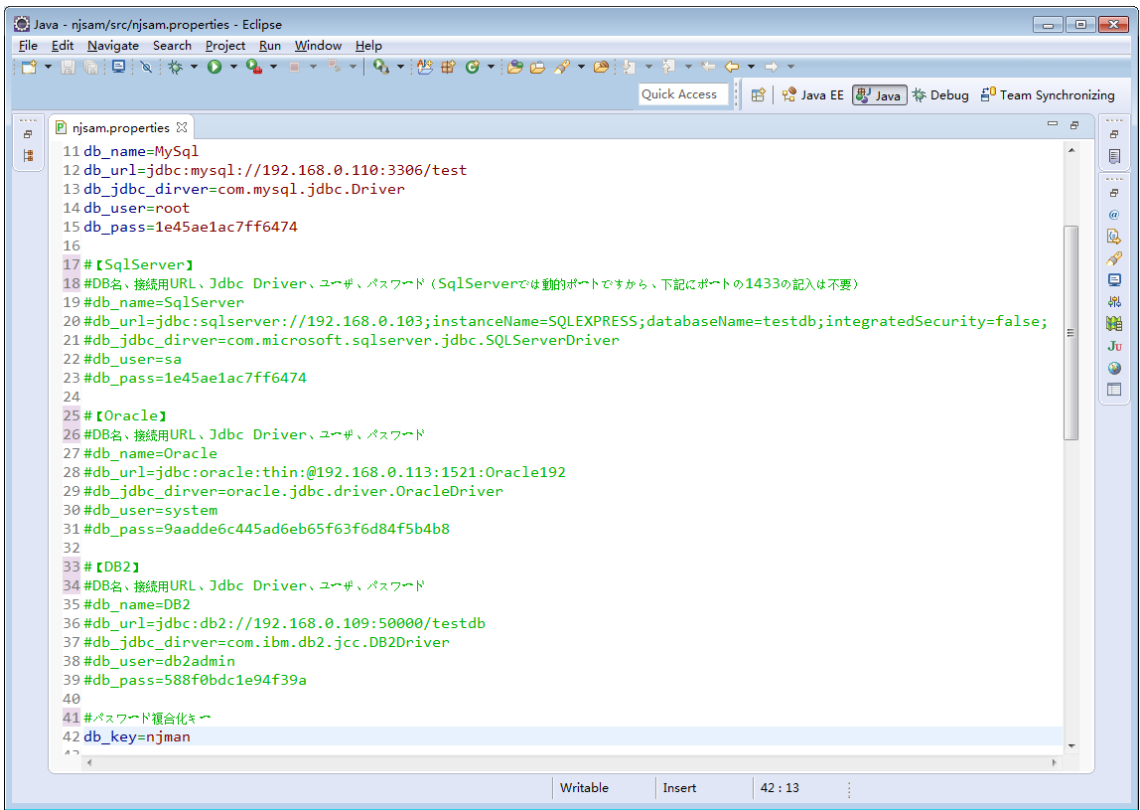
3) 对 njsam 进行重新编译，并在 Tomcat 的模块中加入 njsam 项目。

(图 1－2 Web 模块添加)



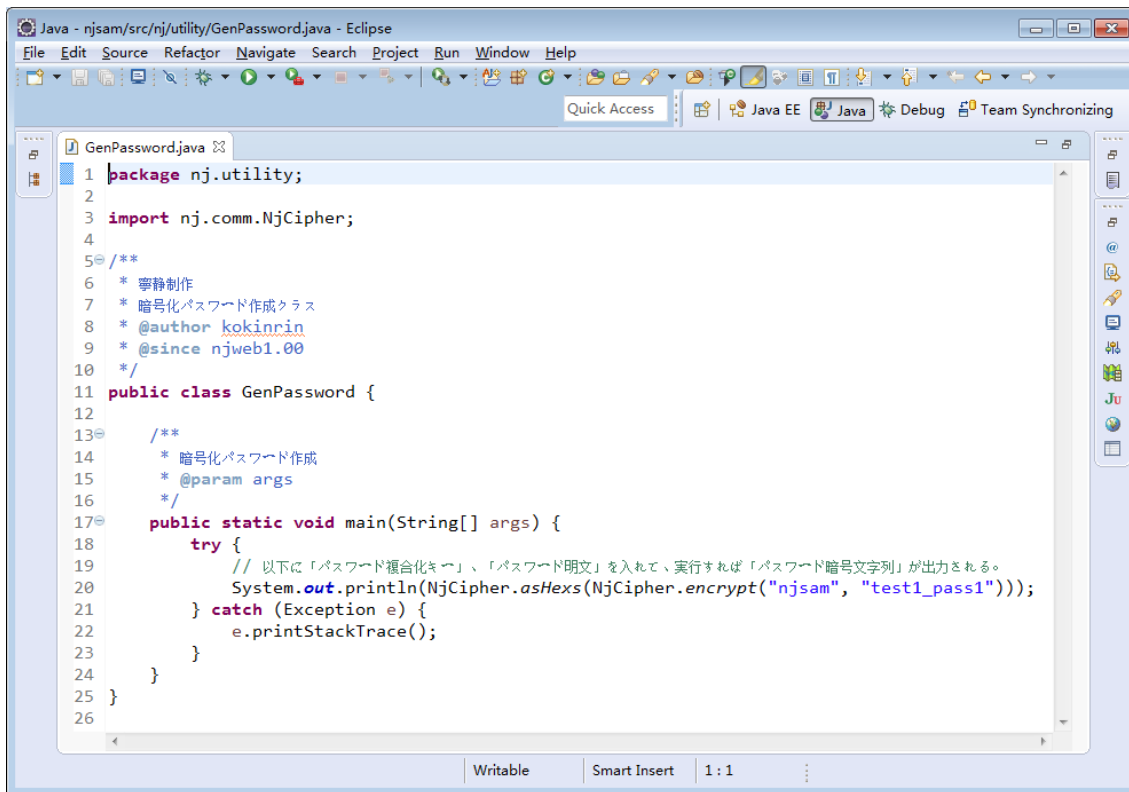
4) 根据实际利用，修改设定文件中的 D B 定义。

(图 1－3 DB 定义变更)



5) 上面的 db_pass 的值是加密后的数据库连接密码, 加密时需要上面的 db_key 的值为关键字利用, 为了取得这个加密后的数据库密码, 需要执行下图中的程序, 执行前将 db_key 的值和数据库连接密码的明文写到程序中, 执行后加密后的数据库连接密码会在控制台输出, 复制后放到定义文件中。

(图 1 - 4 加密后的数据库连接密码取得)



2. DB 环境的准备

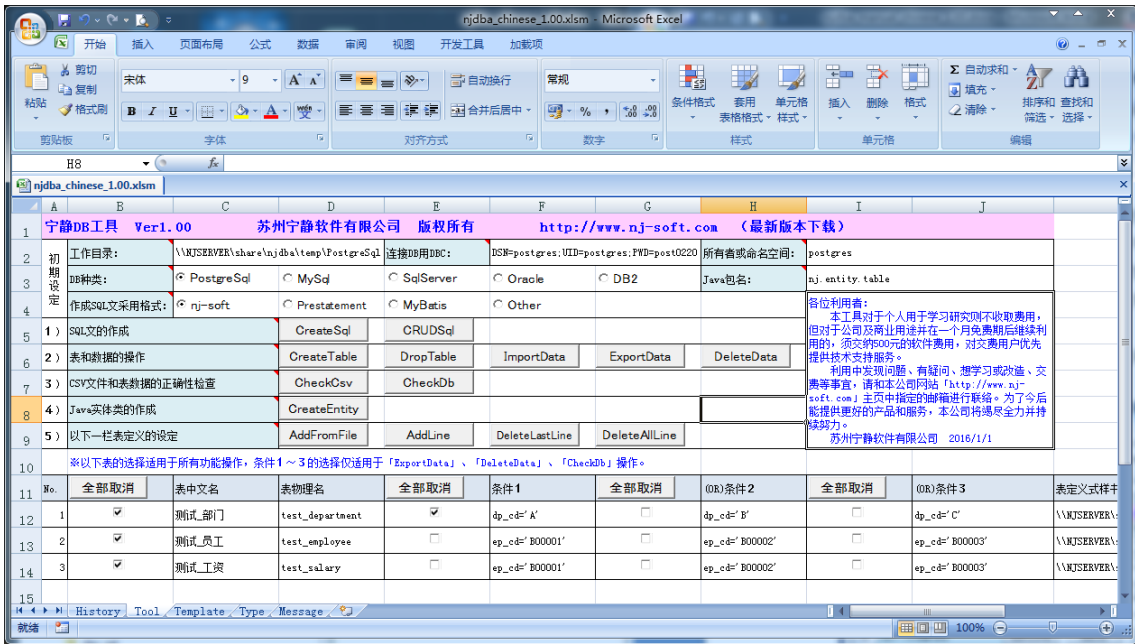
宁静框架软件支持 PostgreSQL、MySQL、SqlServer、Oracle、DB2 等五种数据库。用户需根据实际的数据库进行建表和数据的导入。

建表和数据的导入可以通过宁静 DB 工具进行, 解压后的 njweb_chinese_1.00.zip 中, 利用 table 目录下的 DB 定义书和 work\Data 目录下的 csv 数据, 由宁静 DB 工具创建数据表和导入数据。

※关于宁静 DB 工具的详细, 请从本公司的网站下载宁静 DB 工具后参照。

<http://www.nj-soft.com/njweb/Download.action?baseInfo.menuId=3030&baseInfo.curPage=0>

(图 2 — 1 宁静 DB 管理工具印象)



3. 菜单

宁静框架软件利用 jQuery，可以作成横式或竖式的多层菜单，还可以表示现在位置的进度条。如果设定权限则可以控制菜单项的有效和无效。

(图 3 — 1 菜单例子)



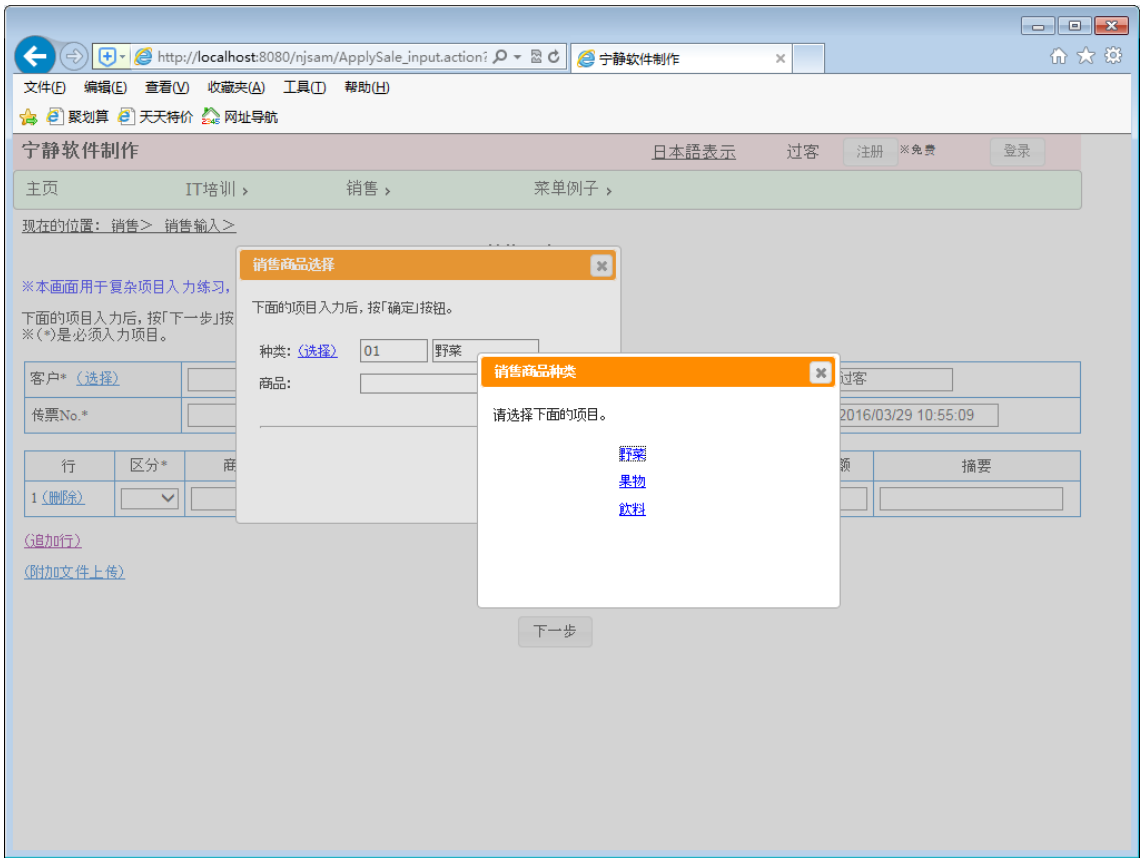
※关于菜单的作成，可以参照以下源代码。

```
/njsam/src/nj/action/ext/Help.java
/njsam/WebContent/jsp/include/header.jsp
/njsam/WebContent/jsp2/include/header.jsp
```

4. 多层对话框

宁静框架软件通过 jQuery，可以控制多层对话框。下面的例子是销售输入画面的商品代码的「(选择)」点击后，销售商品选择对话框显示，再点击对话框的种类的「(选择)」，销售商品种类对话框显示。

(图 4 - 1 多层对话框)



※关于多层对话框的作成，可以参照以下源代码。

```
/njsam/WebContent/jsp/sale/sale-input.jsp
/njsam/WebContent/jsp2/sale/sale-input.jsp
/njsam/src/nj/action/ApplySale.java
```

5. 动态画面

宁静框架软件利用 jQuery，可以作成动态画面。下面的例子通过「(追加行)」到第 4 行，然后再删除第 2 行得到的结果。(图 5 - 1 动态画面)

← → + http://localhost:8080/njsam/ApplySale_input.action? 宁静软件制作 ×

文件(F) 编辑(E) 查看(V) 收藏夹(A) 工具(T) 帮助(H)

★ 聚划算 天天特价 网站导航

宁静软件制作 日本語表示 过客 注册 ※免费 登录

主页 IT培训 > 销售 > 菜单例子 >

现在的位置: 销售 > 销售输入 >

销售入力

※本画面用于复杂项目入力练习, 请自由利用。

下面的项目入力后, 按「下一步」按钮。

※(*)是必须入力项目。

客户* (选择)	0101 東京A社	担当者	过客
传票No.*	111111	订购日時	2016/03/29 10:55:09

行	区分*	商品代码*	商品名	销售数量	单位	销售单价	销售金额	摘要
1 (删除)	销售 ▾	0101 (选择)	人参	1	盒	10	10	一盒
3 (删除)	销售 ▾	0102 (选择)	キャベツ	3	个	3	9	
4 (删除)	购销 ▾	0103 (选择)	白菜	4	个	4	16	

(追加行)

(附加文件上传)

下一步

※关于动态画面的作成，可以参照以下源代码。

```
/njsam/WebContent/jsp/sale/sale-input.jsp
/njsam/WebContent/jsp2/sale/sale-input.jsp
/njsam/src/nj/action/ApplySale.java
```

6. 日历

画面的 jsp 文件中，只需要把日历部品包括进去，用于日期选择输入的日历就可以表示出来。下面的例是「希望开始日*」点击后，日语和中国语的日历表示。

(图 6－1 日本語日历)

希望開始日* ※YYYYMM/DDで入力してください。

週に受講回数* 数字を入力してください。

希望時間帯*

日	月	火	水	木	金	土
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

木曜日> ☐ 指定なし ☐ 10:15-12:15 ☐ 14:30-16:30 ☐ 18:30-20:30

金曜日> ☐ 指定なし ☐ 10:15-12:15 ☐ 14:30-16:30 ☐ 18:30-20:30

(图 6－2 中国語日历)

Email*

照片 浏览...

希望开始日* ※请按YYYYMM/DD格式输入。

每周听课回数* 输入半角数字。

希望时间段*

日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

星期四> ☐ 不指定 ☐ 10:15-12:15 ☐ 14:30-16:30 ☐ 18:30-20:30

※关于日历的作成，可以参照以下源代码。

/nj sam/WebContent/jsp/train/train-input.jsp

/nj sam/WebContent/jsp2/train/train-input.jsp

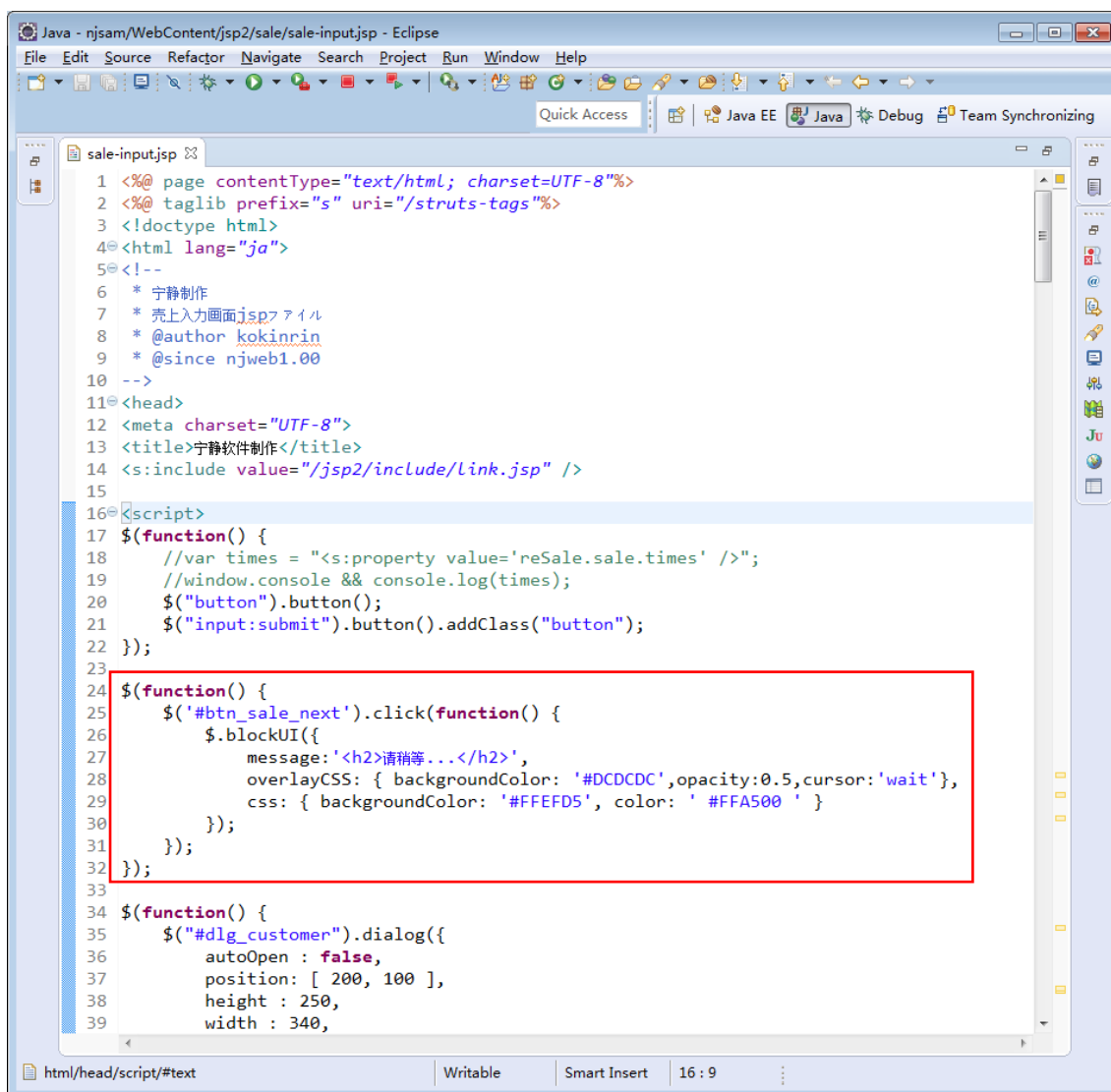
7. 防止重复登录的画面锁定

宁静框架软件在点击按钮或链接进行提交或检索时，可以对画面进行锁定，以防止多重提交或多重登录。实现程序简洁明了。

(图 7 - 1 画面锁定)



(图 7 - 2 画面锁定实现)



```

1 <%@ page contentType="text/html; charset=UTF-8"%>
2 <%@ taglib prefix="s" uri="/struts-tags"%>
3 <!doctype html>
4 <html lang="ja">
5 <!--
6  * 宁静制作
7  * 壳上入力画面jspファイル
8  * @author kokinnrin
9  * @since njweb1.00
10 -->
11 <head>
12 <meta charset="UTF-8">
13 <title>宁静软件制作</title>
14 <s:include value="/jsp2/include/Link.jsp" />
15
16 <script>
17 $(function() {
18     //var times = "<s:property value='reSale.sale.times' />";
19     //window.console && console.log(times);
20     $("button").button();
21     $("input:submit").button().addClass("button");
22 });
23
24 $(function() {
25     $('#btn_sale_next').click(function() {
26         $.blockUI({
27             message: '<h2>请稍等...</h2>',
28             overlayCSS: { backgroundColor: '#DCDCDC',opacity:0.5,cursor:'wait'},
29             css: { backgroundColor: '#FFEDF5', color: ' #FFA500 ' }
30         });
31     });
32 });
33
34 $(function() {
35     $('#dlg_customer').dialog({
36         autoOpen : false,
37         position: [ 200, 100 ],
38         height : 250,
39         width : 340,

```

※关于防止重复登录的画面锁定的作成，可以参照以下源代码。

/njsam/WebContent/jsp/sale/sale-input.jsp

/njsam/WebContent/jsp2/sale/sale-input.jsp

8. 独自开发的输入检查

宁静框架宁静包含独自开发的输入检查，出错信息的表示以及出错项目的背景色处理等，画面表示简洁明了。以下对输入项目、上传文件、和画像文字认证分别进行说明。

8. 1 输入项目检查

输入有误、或者必须项目没有输入等，出错信息和出错项目如下图。因为输入检查函数已经提供，程序实装只需要调用相应的函数就可以了。（图 8 - 1 输入项目检查）

http://localhost:8080/njsam/ApplyTrain_input.acti... 宁静软件制作

文件(F) 编辑(E) 查看(V) 收藏夹(A) 工具(T) 帮助(H)

用户名(拼音)是必须输入项目。
性别是必须输入项目。
入力的日期希望开始日格式不正，请用YYYY/MM/DD格式入力。
入力的Email地址格式不正。
请在每周听课回数里入力半角数字。
希望时间段是必须输入项目。

用户名(汉字)	<input type="text"/>
用户名(拼音)*	<input type="text"/>
性别*	<input type="radio"/> 男 <input type="radio"/> 女
电话号码	<input type="text"/>
Email*	<input type="text" value="aaa"/>
照片	<input type="button" value="浏览..."/>
希望开始日*	<input type="text" value="2012"/> ※请按YYYY/MM/DD格式入力。
每周听课回数*	<input type="text" value="e"/> ※请入力半角数字。
希望时间段*	※请尽量多选你可以出席的时间段。
	星期日> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
	星期一> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
	星期二> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
	星期三> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
	星期四> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
	星期五> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
	星期六> <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30
希望参加讲习的场所	※默认的讲习场所是南船桥的VISIT2楼，如果你希望别的场所的话，请在下面输入。 <input type="text"/>

※关于输入项目检查的作成，可以参照以下源代码。

```
/njsam/src/nj/action/ApplyTrain.java
/njsam/src/nj/action/chk/Check.java
/njsam/WebContent/jsp/train/train-input.jsp
/njsam/WebContent/jsp2/train/train-input.jsp
```

11

8. 2 上传文件检查

对于上传文件，可以检查文件类型和文件大小。下面的例子是照片项目中输入的不是图像文件时的出错情况。

(图 8 - 2 上传文件检查)

宁静软件制作

日本語表示 过客 注册 ※免费 登录

主页 IT培训 > 销售 > 菜单例子 >

现在的位置: IT培训 > IT培训申请 >

IT培训申请入力

※本画面用于复杂项目入力练习, 请自由利用。

※培训采用1对1方式, 从基础知识开始, 到宁静开发框架进行详细讲解。

※讲习中用到的程序全部免费提供。

※讲习的详细内容请参考本网站的[培训内容]。

※讲习中可以免费做成较小的Web项目或网站。

※日本SAKURA的VPS申请, DB构筑, 电子邮箱服务器, 网站构筑等都可以免费代理。

※电子邮箱的Web管理机能(Email追加, 删除, 密码的变更)也免费提供。

入力的文件类型不被许可。

用户名(汉字)	<input type="text"/>
用户名(拼音)*	<input type="text" value="aa"/>
性别*	<input checked="" type="radio"/> 男 <input type="radio"/> 女
电话号码	<input type="text"/>
Email*	<input type="text" value="aa@bb.cc"/>
照片	<input type="text"/> 浏览...
希望开始日*	<input type="text" value="2016/03/30"/> ※请按YYYY/MM/DD格式入力。
每周听课回数*	<input type="text" value="1"/> ※请入力半角数字。
希望时间段*	※请尽量多选你可以出席的时间段。 星期日 > <input checked="" type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30 星期一 > <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30 星期二 > <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30 星期三 > <input type="checkbox"/> 不指定 <input type="checkbox"/> 10:15-12:15 <input type="checkbox"/> 14:30-16:30 <input type="checkbox"/> 18:30-20:30

※关于上传文件检查的作成, 可以参照以下源代码。

/njsam/src/nj/action/inter/InterUpload.java

8. 3 图像文字认证

图像文字认证主要为了预防机器人或工具程序等自动执行登录。下面的例子是认证文字输入有误时的出错情况。(图 8 - 3 图像文字认证)



※关于图像文字认证的作成，可以参照以下源代码。

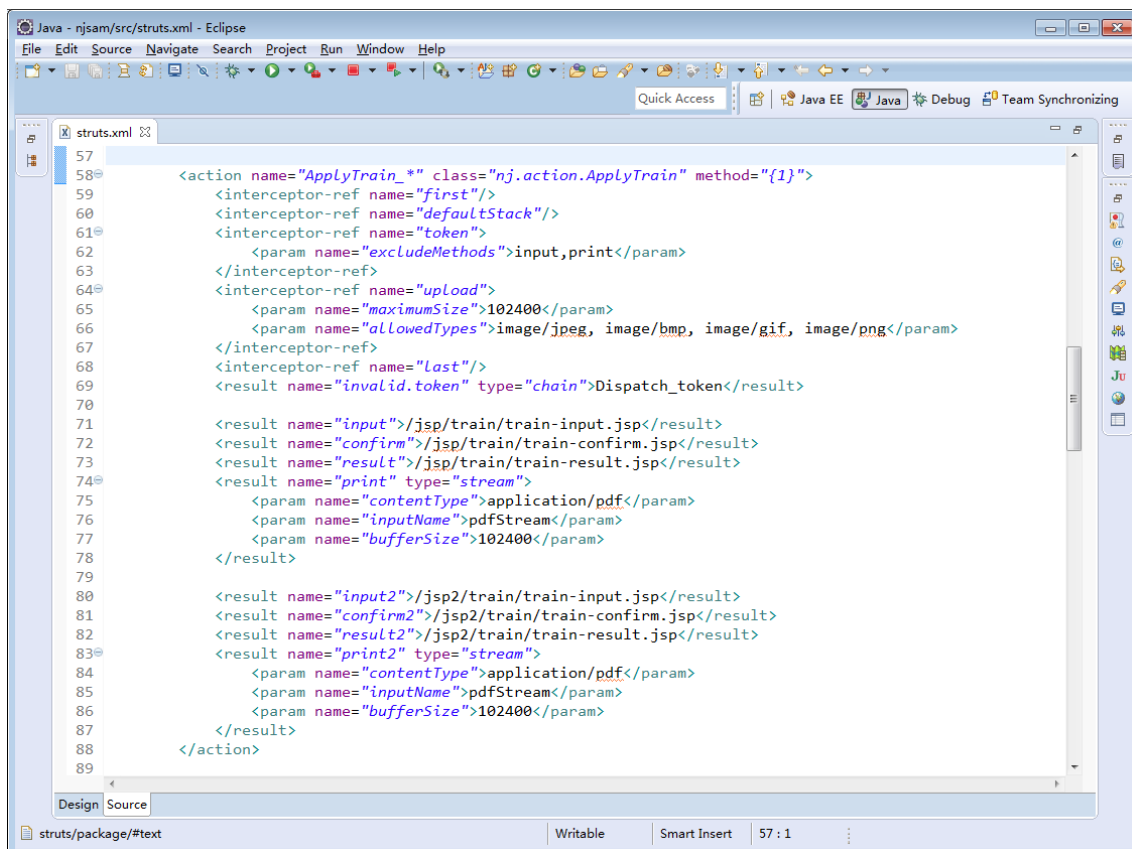
```
/njsam/src/nj/action/ajax/GetCaptcha.java
/njsam/src/nj/action/ApplyTrain.java
/njsam/src/nj/action/chk/Check.java
/njsam/WebContent/jsp/train/train-confirm.jsp
/njsam/WebContent/jsp2/train/train-confirm.jsp
```

9. 画面迁移

宁静框架软件采用 Struts2 的迁移机能来实现画面迁移。下面的设定例子是日语和中文的 IT 培训申请的各个画面的迁移设定。

IT 培训申请入力（包含文件上传检查、Token 接续检查等）←→IT 培训申请入力确认→IT 培训申请入力结果→IT 培训申请入力结果 PDF 输出。

（图 9－1 画面迁移设定）



※关于画面迁移的作成，可以参照以下源代码。

/njsam/src/struts.xml

/njsam/src/ajax.xml

1 0 . 数据映射

宁静框架软件采用 Struts2 的 OGNL (Object-Graph Navigation Language) 机能，实现画面数据和 Java 实体类（包含字典类型和数组类型数据）的映射，即画面提交时程序无需将画面数据设定到实体类，反之画面显示时程序也无需将实体类数据设定到画面，由映射机能根据名字空间自动实现。下面的例子是销售入力画面的各个项目和对应的 Java 实体类。

(图 1 0 - 1 数据映射画面)

http://localhost:8080/njsam/ApplySale_input.action

宁静软件制作

文件(E) 编辑(E) 查看(V) 收藏夹(A) 工具(T) 帮助(H)

宁静软件制作 日本語表示 system 注册 ※免费 登录

主页 IT培训 > 销售 > 菜单例子 >

现在的位置: 销售> 销售入力>

销售入力

※本画面用于复杂项目入力练习, 请自由利用。

下面的项目入力后, 按「下一步」按钮。

※(*)是必须入力项目。

客户* (选择)	0101	東京A社	担当者	过客
传票No.*	111111		订购日時	2016/03/31 08:52:02

行	区分*	商品代码*	商品名	销售数量	单位	销售单价	销售金额	摘要
1 (删除)	销售 ▼	0101 (选择)	人参	1	盒	100	100	
2 (删除)	销售 ▼	0102 (选择)	キャベツ	2	顆	200	400	
3 (删除)	赊销 ▼	0103 (选择)	白菜	3	顆	300	900	

(追加行)

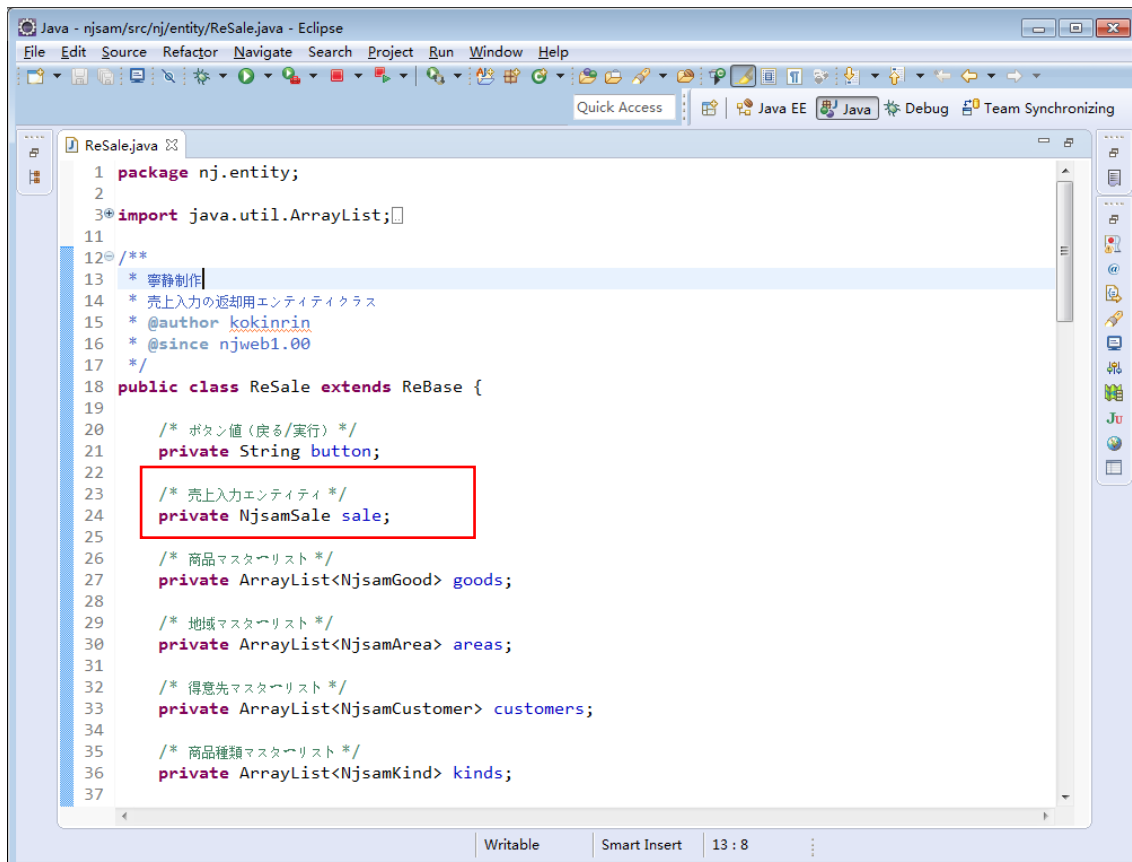
(附加文件上传)

1 注文書 1.pdf (删除)

2 注文書 2.pdf (删除)

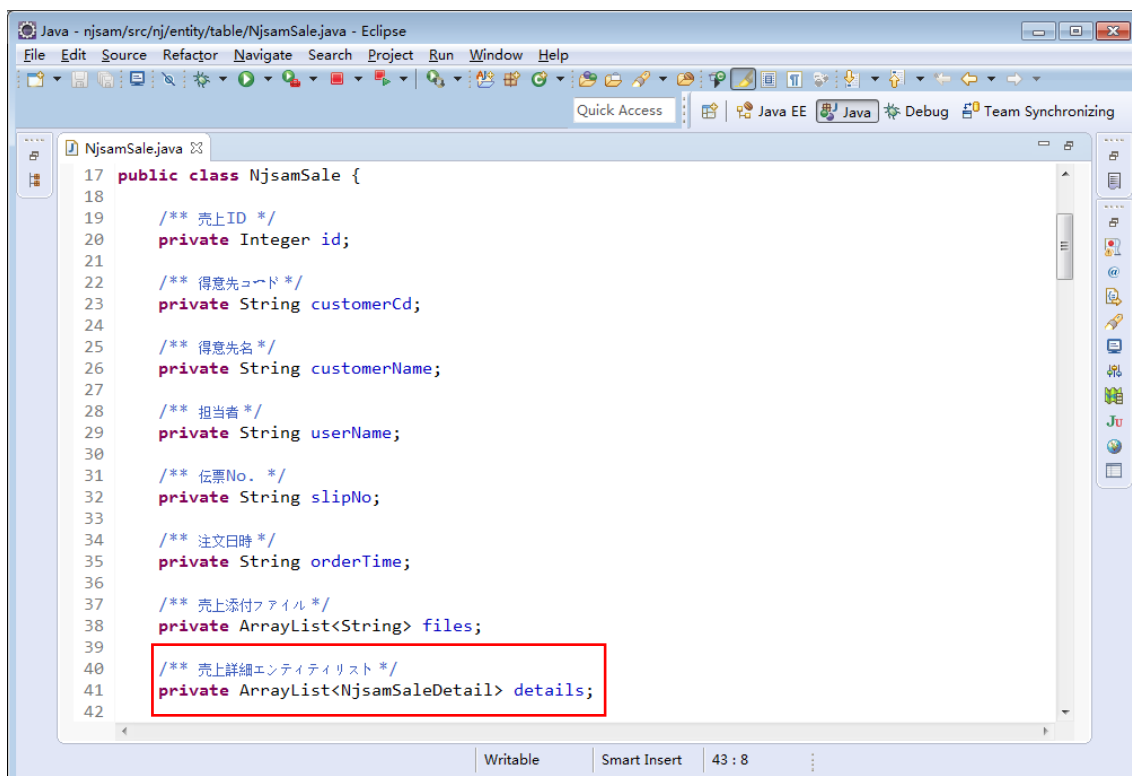
下一步

(图 1 0 - 2 数据映射实体类)



```

1 package nj.entity;
2
3 import java.util.ArrayList;
4
5 /**
6  * 零售制作
7  * 売上入力のリクエストエンティティクラス
8  * @author kokinnin
9  * @since njweb1.00
10 */
11
12 public class ReSale extends ReBase {
13
14     /** ボタン値 (戻る/実行) */
15     private String button;
16
17     /** 売上入力エンティティ */
18     private NjsamSale sale;
19
20     /** 商品マスターリスト */
21     private ArrayList<NjsamGood> goods;
22
23     /** 地域マスターリスト */
24     private ArrayList<NjsamArea> areas;
25
26     /** 得意先マスターリスト */
27     private ArrayList<NjsamCustomer> customers;
28
29     /** 商品種類マスターリスト */
30     private ArrayList<NjsamKind> kinds;
31
32 }
    
```



```

17 public class NjsamSale {
18
19     /** 売上ID */
20     private Integer id;
21
22     /** 得意先コード */
23     private String customerCd;
24
25     /** 得意先名 */
26     private String customerName;
27
28     /** 担当者 */
29     private String userName;
30
31     /** 伝票No. */
32     private String slipNo;
33
34     /** 注文日時 */
35     private String orderTime;
36
37     /** 売上添付ファイル */
38     private ArrayList<String> files;
39
40     /** 売上詳細エンティティリスト */
41     private ArrayList<NjsamSaleDetail> details;
42
43 }
    
```

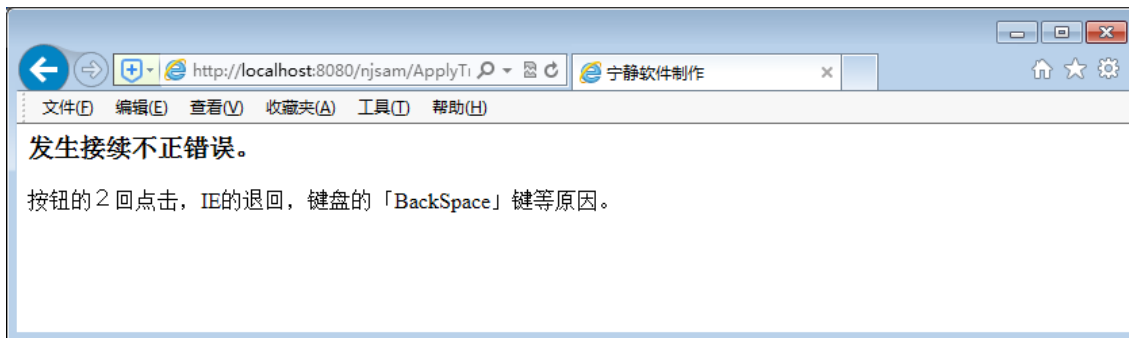

※关于数据映射，可以参照以下源代码。

```
/njsam/src/nj/entity/ReSale.java  
/njsam/src/nj/entity/table/NjsamSale.java  
/njsam/src/nj/entity/table/NjsamSaleDetail.java  
/njsam/src/nj/action/ApplySale.java  
/njsam/WebContent/jsp/sale/sale-input.jsp  
/njsam/WebContent/jsp2/sale/sale-input.jsp
```

1 1 . Token 检查

Token 检查用于预防多重登录和接续不正等问题。宁静框架软件采用了 Struts2 的 Token 检查机能，只要在设定文件中设置即可利用。下面的例子是在 IT 培训申请入力结果画面，点击浏览器的刷新（F5）时的出错画面。

（图 1 1 - 1 Token 检查）



※关于 Token 检查，可以参照以下源代码。

```
/njsam/src/nj/action/ApplyTrain.java  
/njsam/src/struts.xml
```

1 2 . 自作的数据源

宁静框架软件提供自作的数据源，包括和 DB 连接的连接池管理、动态 SQL 文管理、SQL 文的分页查询等。

1 2 . 1 DB 的连接池管理

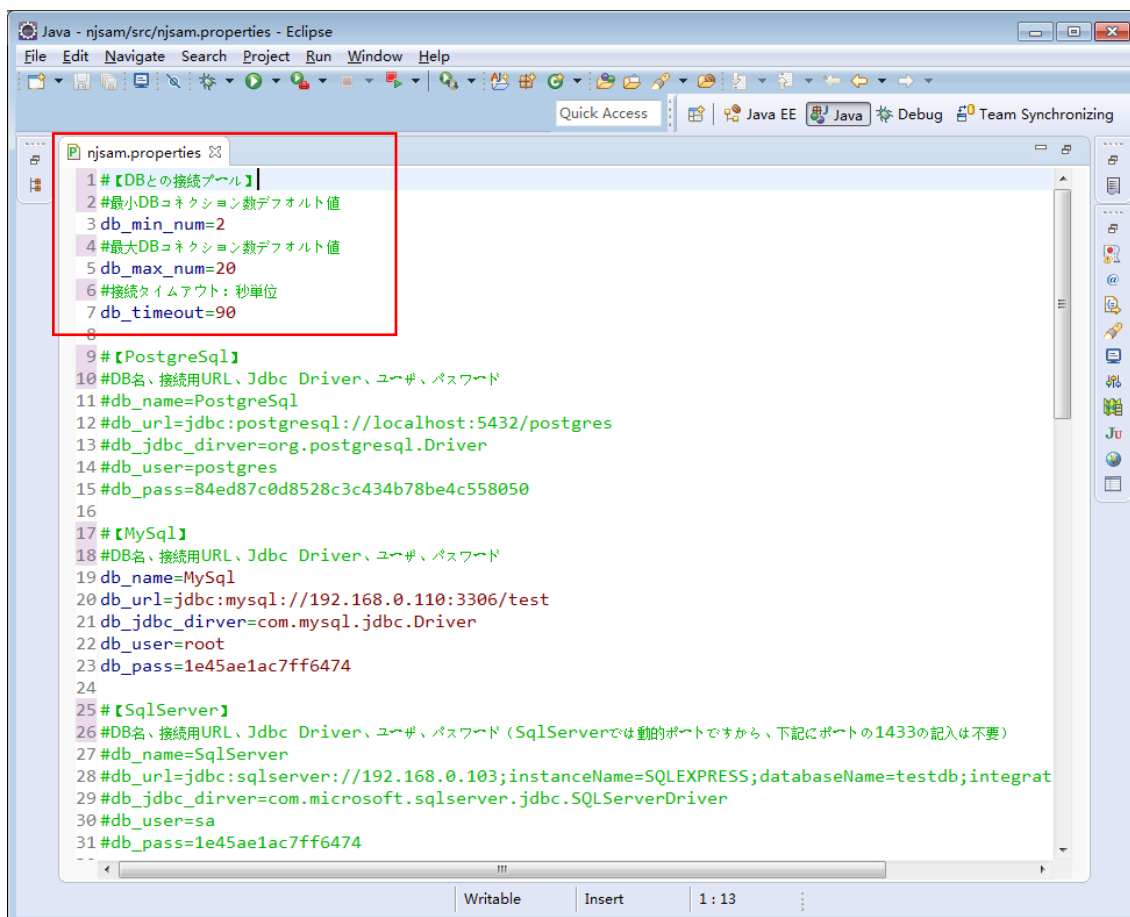
按下图设定 DB 的连接池管理参数后，连接池的管理就可以自动地进行。Web 服务器启动时，设定的最小数的 DB 连接就会被作成，连接池管理就进入等待利用申请状态。

利用后的连接的关闭（返还）发生遗漏时，连接池管理会根据设定的最多占用时间，由监视线程进行事务回滚，并关闭（返还）该连接。当然本框架软件的 DB 存取机能中提供了完善的关闭（返还）机能。另外在 Web 项目开发中，系统会以请求（Request）单位，在每个请求的最后自动关闭（返还）利用中的各个连接。

连接池中连接不够用时，连接池管理会根据请求动态增加，但如果达到设定的最大连接

数时，连接池管理将不再增加新的连接，而是等待利用中的连接返还后，再分配给利用申请等待中的第一个请求。最后在 Web 服务器停止时，所有利用中的连接都将被执行事务回滚后，和未利用的连接一起被关闭。

(图 1 2 - 1 连接池)



```

1 #【DBとの接続プール】
2 #最小DBコネクション数デフォルト値
3 db_min_num=2
4 #最大DBコネクション数デフォルト値
5 db_max_num=20
6 #接続タイムアウト: 秒単位
7 db_timeout=90
8
9 #【PostgreSQL】
10 #DB名、接続用URL、Jdbc Driver、ユーザ、パスワード
11 #db_name=PostgreSQL
12 #db_url=jdbc:postgresql://localhost:5432/postgres
13 #db_jdbc_driver=org.postgresql.Driver
14 #db_user=postgres
15 #db_pass=84ed87c0d8528c3c434b78be4c558050
16
17 #【MySQL】
18 #DB名、接続用URL、Jdbc Driver、ユーザ、パスワード
19 db_name=MySQL
20 db_url=jdbc:mysql://192.168.0.110:3306/test
21 db_jdbc_driver=com.mysql.jdbc.Driver
22 db_user=root
23 db_pass=1e45ae1ac7ff6474
24
25 #【SqlServer】
26 #DB名、接続用URL、Jdbc Driver、ユーザ、パスワード (SqlServerでは動的ポートですから、下記にポートの1433の記入は不要)
27 #db_name=SqlServer
28 #db_url=jdbc:sqlserver://192.168.0.103;instanceName=SQLEXPRESS;databaseName=testdb;integrat
29 #db_jdbc_driver=com.microsoft.sqlserver.jdbc.SQLServerDriver
30 #db_user=sa
31 #db_pass=1e45ae1ac7ff6474

```

※关于连接池管理的设定，可以参照以下源代码。

/njsam/src/njsam.properties

(这里的 njsam 是项目名 (Context root 名)，用户需根据实际开发的项目名来修改。)

※详细请参照以下的 njcomm-1.00 API 式样。

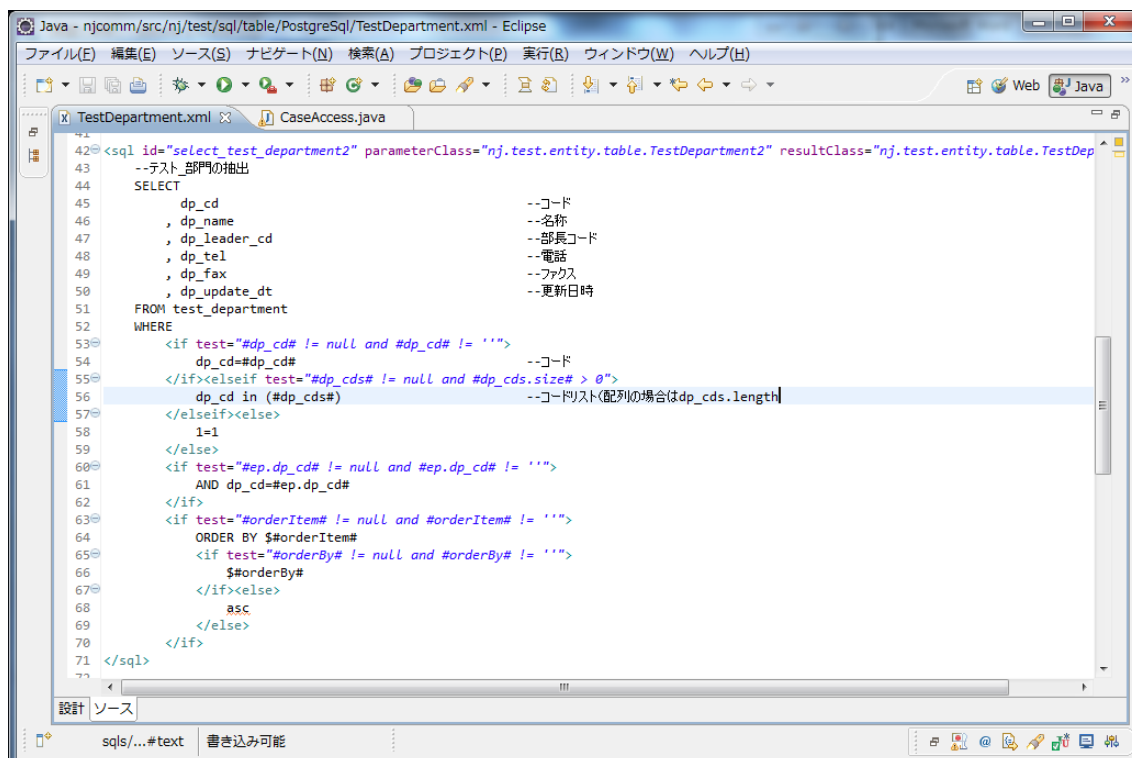
解压文件 njweb_chinese_1.00.zip 中的 njcomm-api (用 index.html 启动)。

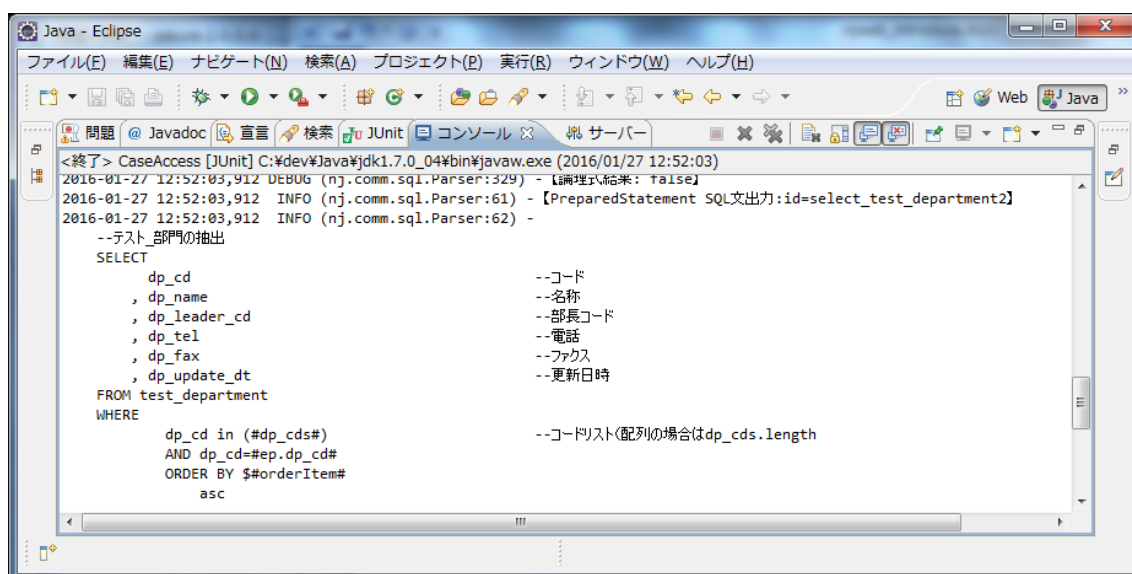
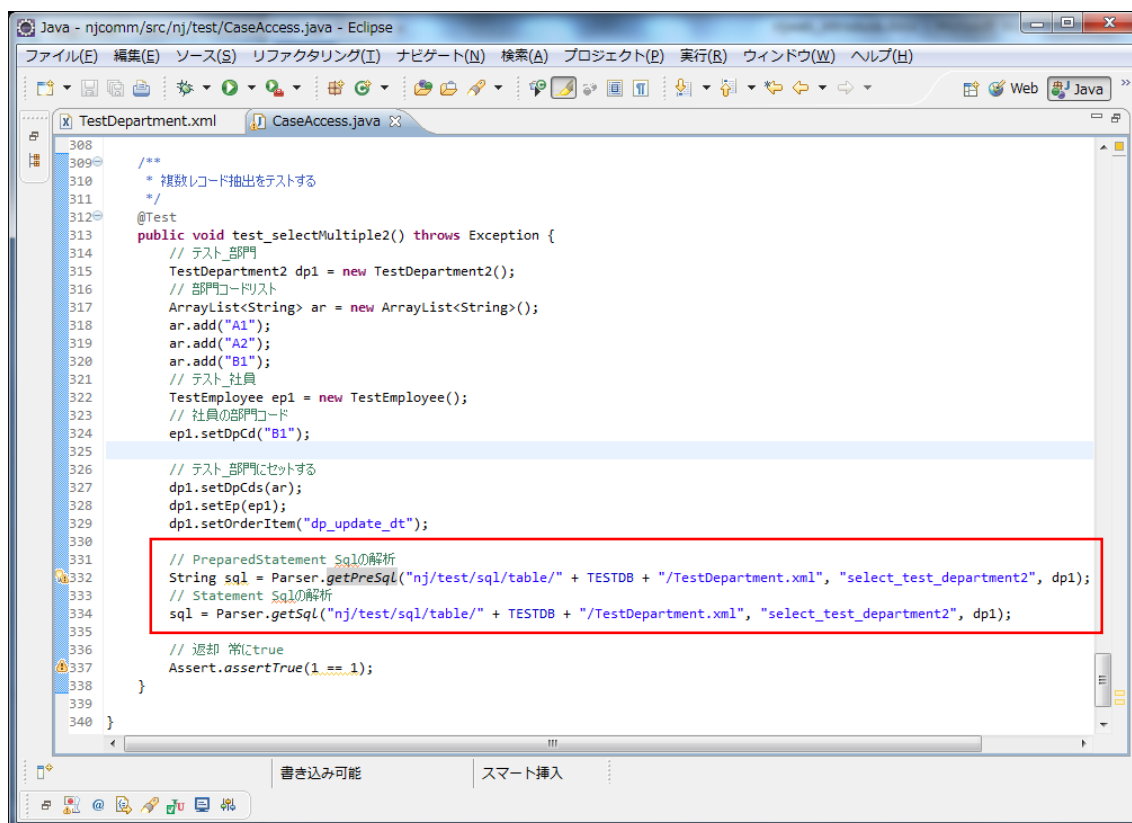
1 2. 2 动态 XML 的 SQL 文解析

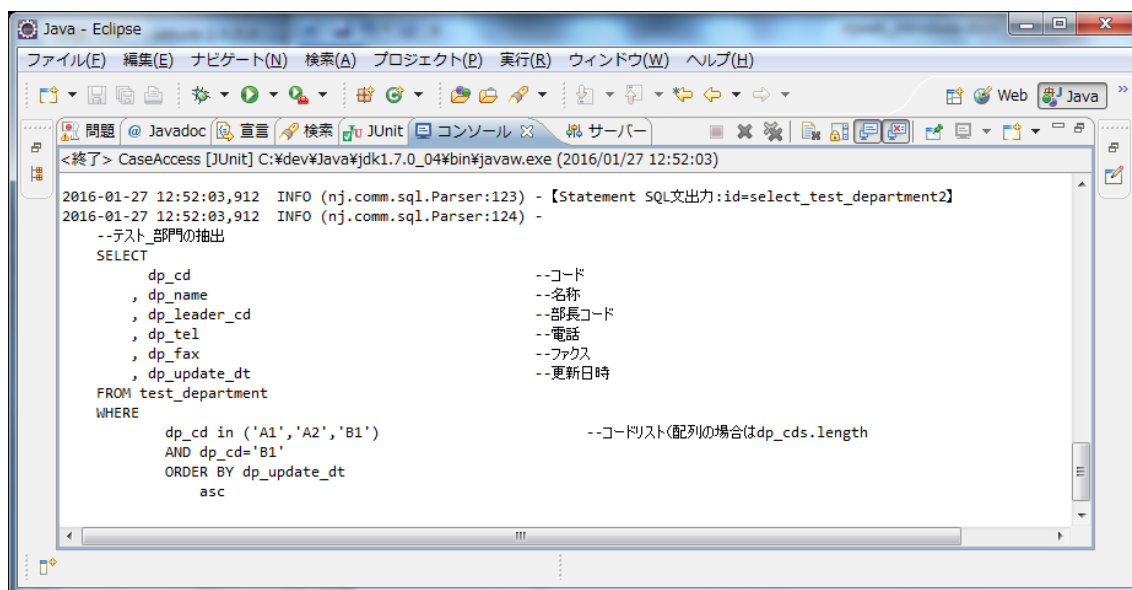
根据动态 XML 的 SQL 文的 test 条件,可以输出 PreparedStatement 或 StatementSQL 文。下面的例子是根据定义的「测试_部门」的 XML 文、和「TestDepartment」实体类的属性值,进而判断 test 条件,来输出 PreparedStatement 或 Statement 的 SQL 文。

PreparedStatement 的 SQL の場合,在绑定参数时,首先将「#FIELD#」转换成「?」后,再在「?」上绑定「#FIELD#」对应的 Java 实体类的属性值。

(图 1 2 - 2 XML 的 SQL 文解析)







※详细请参照以下的 njcomm-1.00 API 式样。

解压文件 njweb_chinese_1.00.zip 中的 njcomm-api (用 index.html 启动)。

1 2 . 3 Access 类的 DB 操作

Access 类中将 Statement 的 SQL 作为参数, 在执行 SQL 后, 立即关闭 (返还) 连接。除去二进制文字列字段, 其它参数都事先代入到 SQL 文中, 所以 SQL 文可以带着参数输出到控制台或日志文件中。对于极端追求 SQL 性能的情况, 可以利用 PreparedStatement 的 SQL 的 Transact 类。

有二进制文字列字段的情况, 本框架软件解析的 Statement 的 SQL 中也保留「?」, 执行时自动把二进制字段的值代入。

多条更新 SQL 需要在同一个事务中执行的情况, 可以用下面的「updates」函数, 或利用管理事务的 Transact 类进行更新操作。

Access 类提供了以下操作 DB 的函数。

- 1) DB 的数据插入 (带重复主键判定)

```
public static int insert(String sql, byte[]... bss) throws Exception
```

- 2) DB 的数据更新 (不带重复主键判定)

```
public static int update(String sql, byte[]... bss) throws Exception
```

- 3) DB 的多条数据插入或数据更新 (带重复主键判定)

```
public static int updates(String[] sqls, Integer[] effects, byte[][]... bsss) throws Exception
```

- 4) DB 中检索 1 条记录

```
public static Object selectSingle(String sql, String entityName) throws Exception
```

5) DB 中检索多条记录

```
public static Object selectMultiple(String sql, String entityName) throws
Exception
```

6) DB 中分页检索

```
public static void selectPage(String sql, IntSelPageselPage) throws
Exception
```

7) 检索结果输出到 CSV 文件

```
public static Integer writeCsvFile(String sql, String path, String charset,
String head) throws Exception
```

※详细请参照以下的 njcomm-1.00 API 式样。

解压文件 njweb_chinese_1.00.zip 中的 njcomm-api (用 index.html 启动)。

1 2 . 4 Transact 类的 DB 操作

在管理事务的 Transact 类的 DB 操作前，先要取得和 DB 的连接 (Transact.getInstance(baseInfo))，再将 PreparedStatement 的 SQL 和包含各个参数值的实体类作为参数进行 DB 操作，操作完毕后，Application 应用的场合需要关闭 (返还) 连接，Web 应用的场合在请求 (Request) 的最后由系统自动关闭 (返还) 连接，编写程序时不需要写关闭 (返还) 连接的代码。

Transact 类提供了以下 DB 操作函数。

1) 实例的取得

```
it Transact 接口 (对于同一个 Transact 接口，取得的是同一个 DB 连接。)
public static synchronized Transact getInstance(IntTransact it) throws
Exception
```

2) 提交事务

```
public void commit() throws Exception
```

3) 回滚事务

```
public void rollback() throws Exception
```

4) 提交事务后，把和 DB 的连接关闭 (返还给连接池)

```
public void close() throws Exception
```

5) DB 的插入数据 (带重复主键判断)

```
public int insert(String sql, Object entity) throws Exception
```

6) DB 的更新数据 (不带重复主键判断)

```
public int update(String sql, Object entity) throws Exception
```

7) DB 的多次插入或更新数据 (带重复主键判断)

```
public int updates(String[] sqls, Integer[] effects, Object[] entities)
throws Exception
```

8) DB 中检索 1 条记录

```
public Object selectSingle(String sql, String entityName, Object entity)
throws Exception
```

9) DB 中检索多条记录

```
public Object selectMultiple(String sql, String entityName, Object entity)
throws Exception
```

10) DB 中进行分页检索

```
public void selectPage(String sql, IntSelPageselPage, Object entity) throws
Exception
```

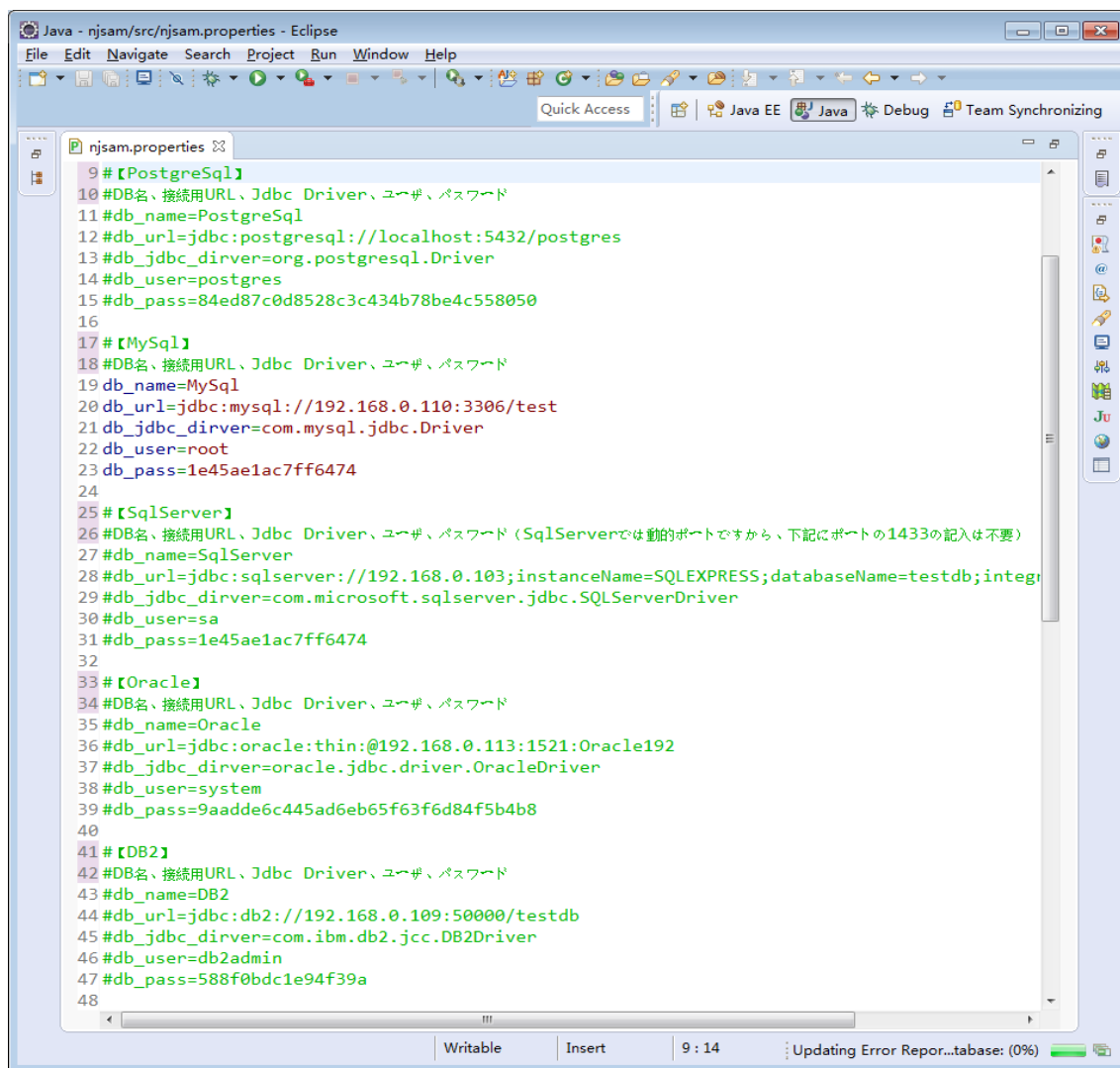
※详细请参照以下的 njcomm-1.00 API 式样。

解压文件 njweb_chinese_1.00.zip 中的 njcomm-api (用 index.html 启动)。

1 2 . 5 支持五种类型的数据库

宁静框架软件支持 PostgreSQL、MySQL、SqlServer、Oracle、DB2 等五种数据库。只要将下图中的设定变更后，所有的机能就全部适用。

(图 1 2 — 3 支持五种类型的数据库)



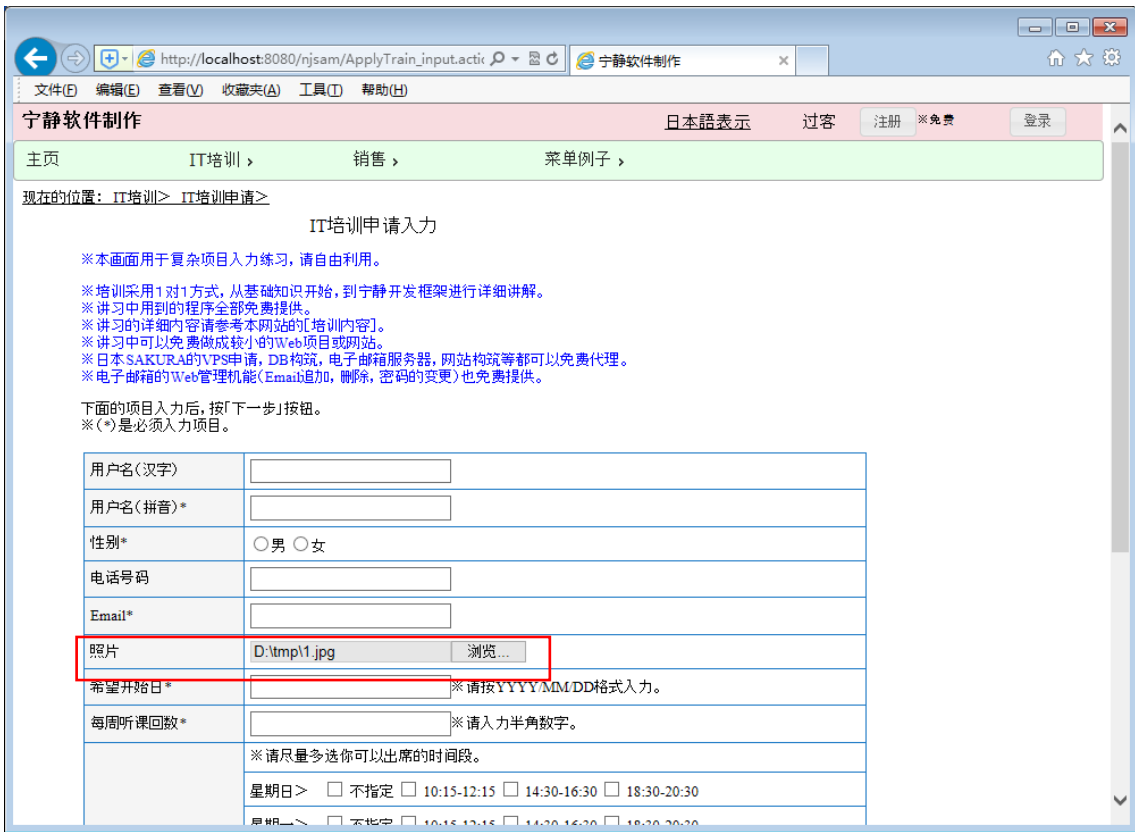
1 3 . 文件上传

宁静框架软件支持 Post 方式和 Ajax 方式的文件上传机能。

1 3 . 1 单个文件上传（Post 方式）

以 Post 方式，会同其它输入项目一起提交时，上传的文件通过文件的类型及文件大小的检查后，将会保存到 Java 实体类的 byte[] 类型的属性中。以下的例子是「照片」项目的文件上传。

（图 1 3 － 1 Post 方式的文件上传）



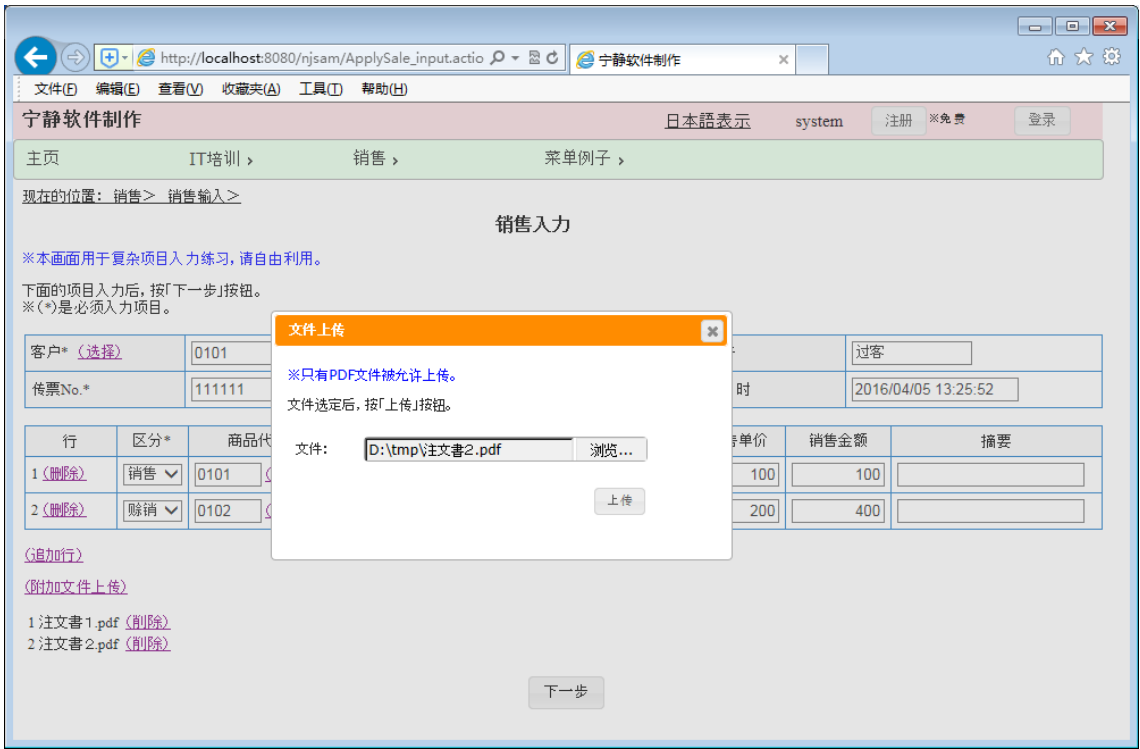
※关于单个文件上传（Post 方式）的作成，可以参照以下源代码。

```
/njsam/src/nj/action/ApplyTrain.java
/njsam/src/nj/action/inter/InterUpload.java
/njsam/WebContent/jsp/train/train-input.jsp
/njsam/WebContent/jsp2/train/train-input.jsp
```

1 3 . 2 多个文件上传 (Ajax 方式)

以 Ajax 方式的弹出画面，一个一个文件上传后，通过文件的类型及文件大小的检查后，将会保存到服务器的临时目录下，在最后由主画面提交时一起处理。

(图 1 3 - 2 Ajax 方式文件上传)



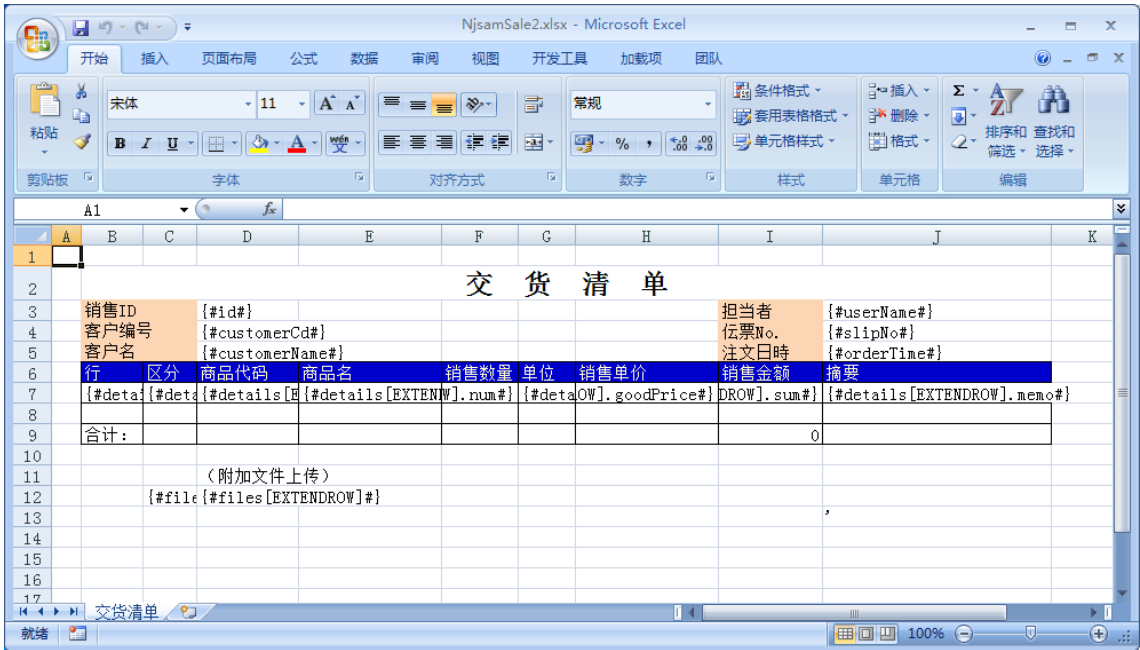
※关于多个文件上传 (Ajax 方式) 的作成，可以参照以下源代码。

```
/njsam/src/nj/action/ajax/Upload. java
/njsam/src/nj/action/inter/InterUpload. java
/njsam/WebContent/jsp/include/upload. jsp
/njsam/WebContent/jsp2/include/upload. jsp
```

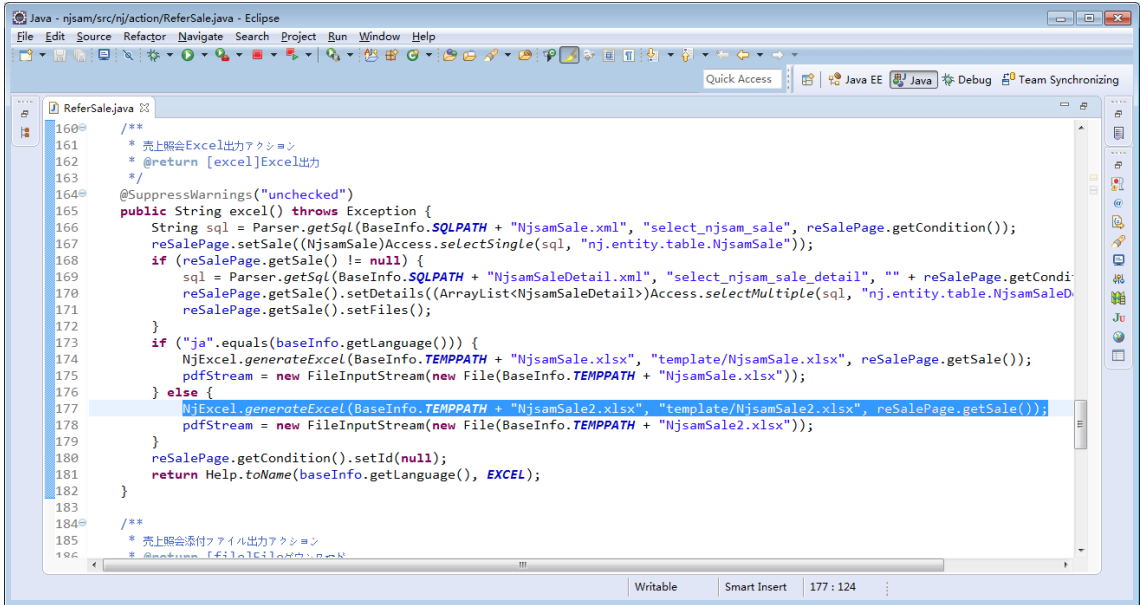
1 4 . Excel 文件输出

宁静框架软件利用 Poi，可以简单地进行 Excel 文件的作成和输出，通过把 Excel 模板文件和 Java 实体类的对接，包括实体类中的子类、数组、和列表等可以直接输出到 Excel 模板文件中。另外数组和列表可以根据设定，自动在 Excel 文件中进行横展开或竖展开，对于二重数组或列表可以同时进行横・竖展开。下面的例子是根据模板文件的设定，对商品明细按行进行展开。

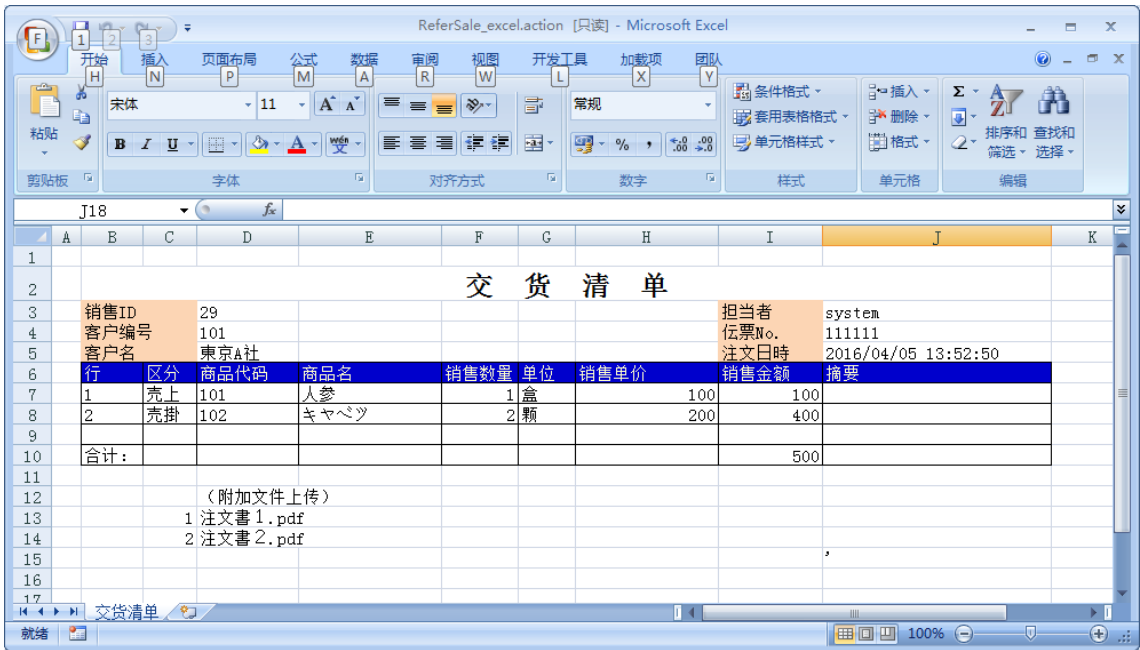
(图 1 4 - 1 Excel 模板)



(图 1 4 - 2 Excel 作成编码)



(图 1 4 - 3 Excel 文件输出)

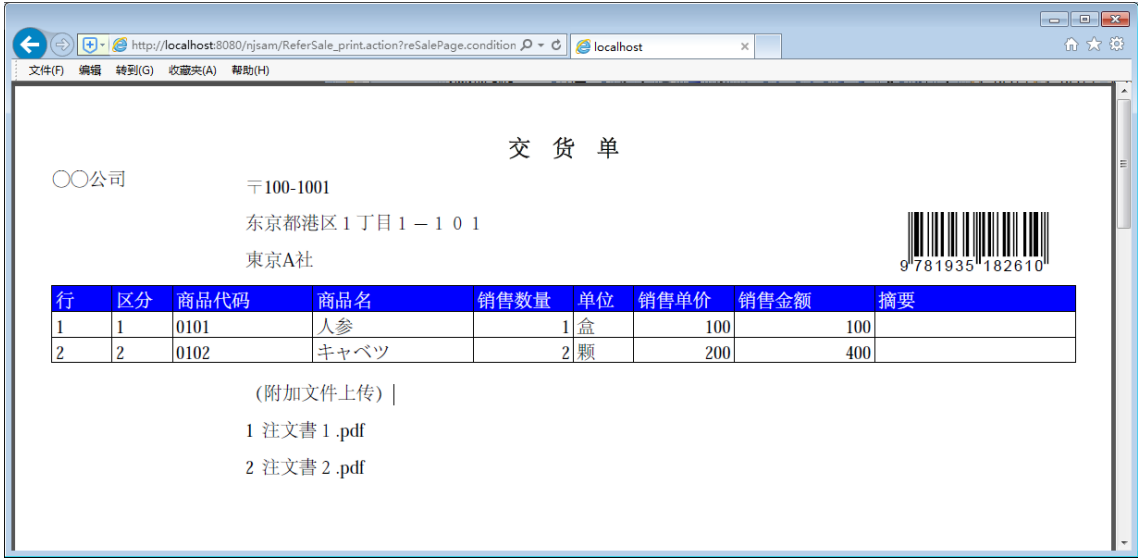


※关于 Excel 文件输出的作成，可以参照以下源代码。

/njsam/src/nj/action/ReferSale.java
/njsam/src/template/NjsamSale.xlsx
/njsam/src/template/NjsamSale2.xlsx
/njsam/WebContent/jsp/sale/sale-refer-result.jsp
/njsam/WebContent/jsp2/sale/sale-refer-result.jsp

1 5 . PDF 文件及条形码的输出

宁静框架软件利用 iText，可以输出 PDF 文件的表格、条形码等，能够细致地输出各种文字、表格和图形。（图 1 5－1 PDF 文件输出）



※关于 PDF 文件及条形码输出的作成，可以参照以下源代码。

```
/njsam/src/nj/action/ext/SalePdf. java  
/njsam/src/nj/action/ext/SalePdf2. java  
/njsam/WebContent/jsp/sale/sale-refer-result. jsp  
/njsam/WebContent/jsp2/sale/sale-refer-result. jsp
```

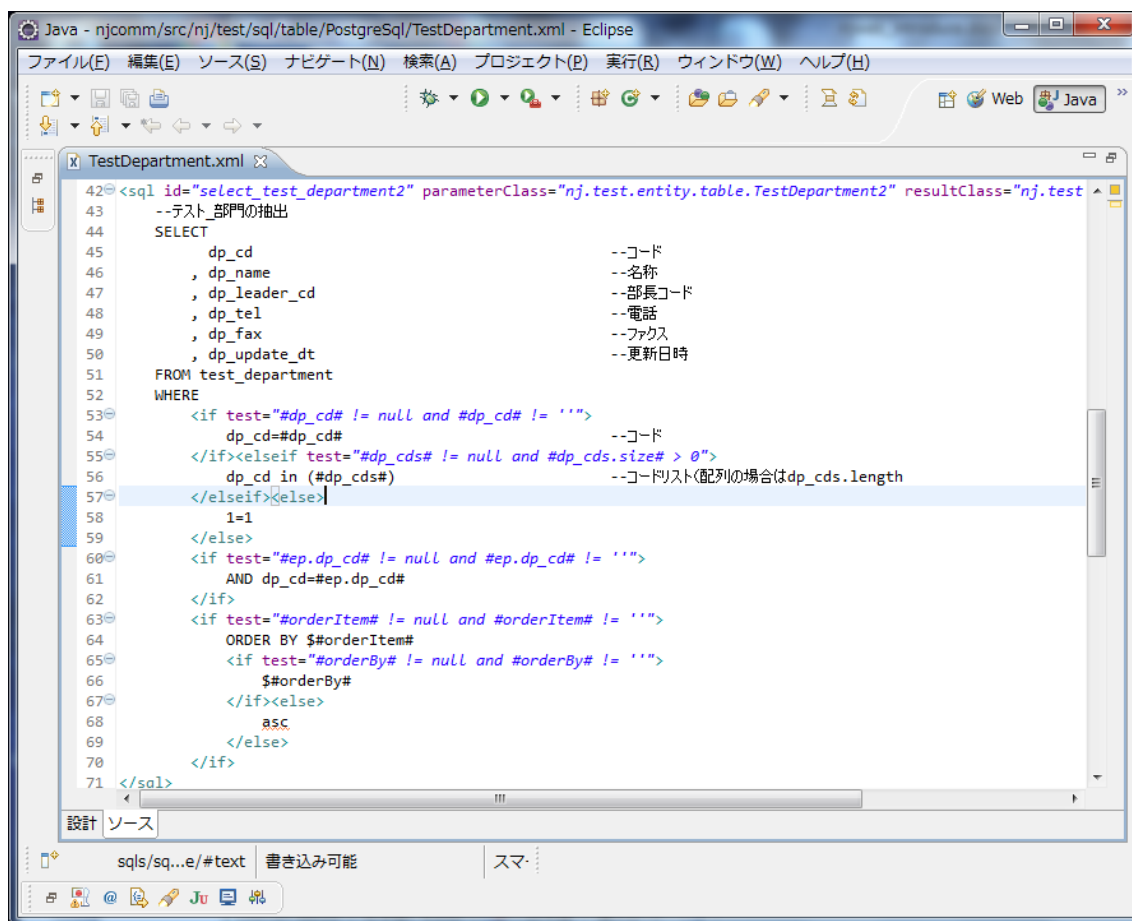
1 6 . 检索和分页

宁静框架软件具有分页检索机能，只需要对普通的检索 SQL 文，指定检索页和每页的件数，就可以把该当页的记录检索出来，并且如果指定的检索页超出范围の場合，也能自动地将最后一页检索出来。

用户实际开发时，只需要专注于业务逻辑，编写普通的检索 SQL 文就可以了，分页 SQL 文的编写及其 Java 逻辑的实装，系统将自动完成,对画面的 jsp 文件中，也提供了分页的部品，可以简单的完成分页检索的画面编码。

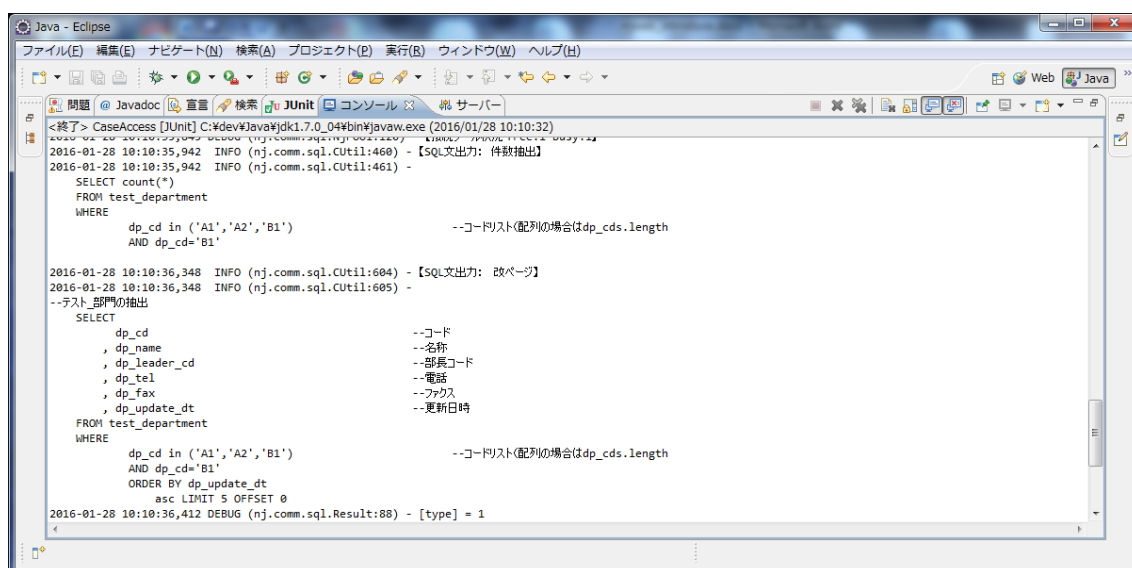
分页机能同样支持五种数据库类型，只要指定所用数据库名，系统就可以从普通的 SQL 文，自动生成相应数据库的全件检索 SQL 文和该当页的检索 SQL 文。

1) 下面的例子是从 PostgreSQL 的普通的 SQL 文, 自动生成件数检索 SQL 文和分页检索 SQL 文。(图 1 6 - 3 PostgreSQL の分页 SQL)



```

42 <sql id="select_test_department2" parameterClass="nj.test.entity.table.TestDepartment2" resultClass="nj.test
43 --テスト_部門の抽出
44 SELECT
45     dp_cd                                --コード
46     , dp_name                            --名称
47     , dp_leader_cd                       --部長コード
48     , dp_tel                             --電話
49     , dp_fax                             --ファクス
50     , dp_update_dt                       --更新日時
51 FROM test_department
52 WHERE
53     <if test="#dp_cd# != null and #dp_cd# != ''">
54         dp_cd=#dp_cd#                    --コード
55     </if><elseif test="#dp_cds# != null and #dp_cds.size# > 0">
56         dp_cd in (#dp_cds#)              --コードリスト(配列の場合はdp_cds.length
57     </elseif><else>
58         1=1
59     </else>
60     <if test="#ep.dp_cd# != null and #ep.dp_cd# != ''">
61         AND dp_cd=#ep.dp_cd#
62     </if>
63     <if test="#orderItem# != null and #orderItem# != ''">
64         ORDER BY $#orderItem#
65     <elseif test="#orderBy# != null and #orderBy# != ''">
66         $#orderBy#
67     </elseif>
68     asc
69     </else>
70     </if>
71 </sql>
    
```



```

<終了> CaseAccess [JUnit] C:\dev\Java\jdk1.7.0_04\bin\javaw.exe (2016/01/28 10:10:32)
2016-01-28 10:10:35,942 INFO (nj.comm.sql.CUtil:460) - 【SQL文出力: 件数抽出】
2016-01-28 10:10:35,942 INFO (nj.comm.sql.CUtil:461) -
SELECT count(*)
FROM test_department
WHERE
    dp_cd in ('A1','A2','B1')              --コードリスト(配列の場合はdp_cds.length
    AND dp_cd='B1'
2016-01-28 10:10:36,348 INFO (nj.comm.sql.CUtil:604) - 【SQL文出力: 改ページ】
2016-01-28 10:10:36,348 INFO (nj.comm.sql.CUtil:605) -
--テスト_部門の抽出
SELECT
    dp_cd                                --コード
    , dp_name                            --名称
    , dp_leader_cd                       --部長コード
    , dp_tel                             --電話
    , dp_fax                             --ファクス
    , dp_update_dt                       --更新日時
FROM test_department
WHERE
    dp_cd in ('A1','A2','B1')              --コードリスト(配列の場合はdp_cds.length
    AND dp_cd='B1'
    ORDER BY dp_update_dt
    asc LIMIT 5 OFFSET 0
2016-01-28 10:10:36,412 DEBUG (nj.comm.sql.Result:88) - [type] = 1
    
```

2) 下面的例子是从 DB2 的普通的 SQL 文，自动生成件数检索 SQL 文和分页检索 SQL 文。

(图 1 6 - 4 DB2 的分页 SQL)

```

41
42 <sql id="select_test_department2" parameterClass="nj.test.entity.table.TestDepartment2" resultClass="nj.test.entity.table.TestDepartment">
43   --テスト_部門の抽出
44   SELECT
45     dp_cd          AS "dp_cd"          --コード
46     , dp_name      AS "dp_name"        --名称
47     , dp_leader_cd AS "dp_leader_cd"    --部長コード
48     , dp_tel       AS "dp_tel"         --電話
49     , dp_fax       AS "dp_fax"         --ファクス
50     , dp_update_dt AS "dp_update_dt"    --更新日時
51   FROM test_department
52   WHERE
53     <if test="#dp_cd# != null and #dp_cd# != ''">
54       dp_cd=#dp_cd# --コード
55     </if><elseif test="#dp_cds# != null and #dp_cds.size# > 0">
56       dp_cd in (#dp_cds#) --コードリスト(配列の場合はdp_cds.length
57     </elseif><else>
58       1=1
59     </else>
60     <if test="#ep.dp_cd# != null and #ep.dp_cd# != ''">
61       AND dp_cd=#ep.dp_cd#
62     </if>
63     <if test="#orderItem# != null and #orderItem# != ''">
64       ORDER BY "#orderItem#"
65     </if><elseif test="#orderBy# != null and #orderBy# != ''">
66       $orderBy#
67     </if><else>
68       asc
69     </else>
70   </if>
71 </sql>

```

```

<終了> CaseAccess [JUnit] C:\dev\Java\jdk1.7.0_04\bin\javaw.exe (2016/01/28 10:53:18)
2016-01-28 10:53:21,009 DEBUG (nj.commm.sql.NjPool:120) - 【接続プール状況 free:1 busy:1】
2016-01-28 10:53:21,040 INFO (nj.commm.sql.CUtil:460) - 【SQL文出力: 件数抽出】
2016-01-28 10:53:21,040 INFO (nj.commm.sql.CUtil:461) -
SELECT count(*)
FROM test_department
WHERE
  dp_cd in ('A1','A2','B1') --コードリスト(配列の場合はdp_cds.length
  AND dp_cd='B1'

2016-01-28 10:53:21,119 INFO (nj.commm.sql.CUtil:604) - 【SQL文出力: 改ページ】
2016-01-28 10:53:21,119 INFO (nj.commm.sql.CUtil:605) -
SELECT *
FROM (
  --テスト_部門の抽出
  SELECT
    row_number() over (
      ORDER BY dp_update_dt
      asc ) as rowno,
    dp_cd          AS "dp_cd"          --コード
    , dp_name      AS "dp_name"        --名称
    , dp_leader_cd AS "dp_leader_cd"    --部長コード
    , dp_tel       AS "dp_tel"         --電話
    , dp_fax       AS "dp_fax"         --ファクス
    , dp_update_dt AS "dp_update_dt"    --更新日時
  FROM test_department
  WHERE
    dp_cd in ('A1','A2','B1') --コードリスト(配列の場合はdp_cds.length
    AND dp_cd='B1'
) tbl
WHERE rowno > 0 AND rowno <= 5
ORDER BY "dp_update_dt"
asc

2016-01-28 10:53:21,135 DEBUG (nj.commm.sql.Result:88) - [type] = -5

```


※关于检索和分页的作成，可以参照以下源代码。

```
/njsam/src/nj/action/ReferSale.java
/njsam/src/nj/entity/ReSalePage.java
/njsam/WebContent/jsp/sale/sale-refer.jsp
/njsam/WebContent/jsp2/sale/sale-refer.jsp
/njsam/WebContent/jsp/include/pager.jsp
/njsam/WebContent/jsp2/include/pager.jsp
```

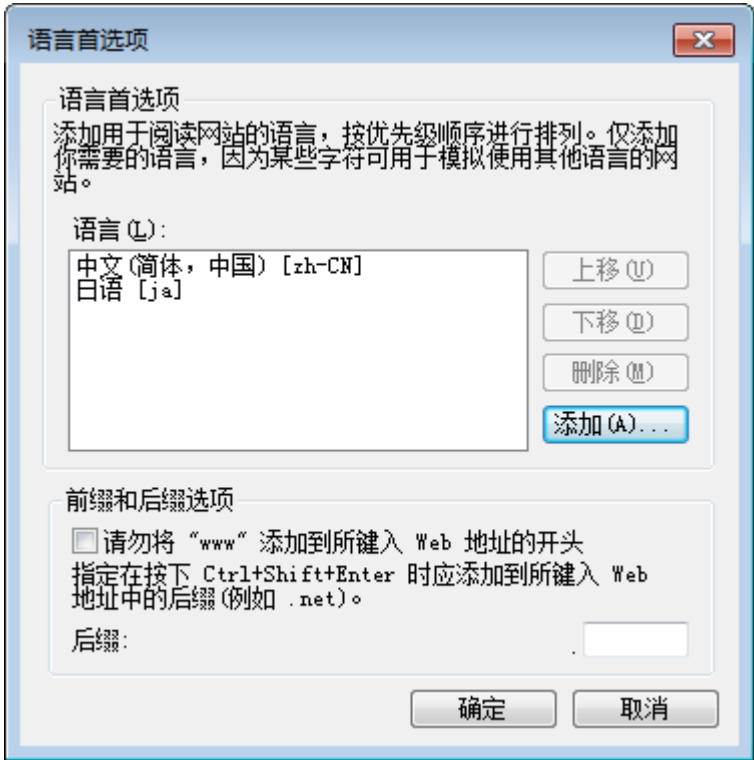
※详细请参照以下的 njcomm-1.00 API 式样。

解压文件 njweb_chinese_1.00.zip 中的 njcomm-api (用 index.html 启动)。

1 7 . 多国语言

宁静框架软件支持多国语言。当然，只使用一种语言也没有问题。初始画面显示时，根据浏览器中设定的语言首选项进行显示。在这以后用户可以通过画面上的语言切换，选择需要显示的语言。

(图 1 7 - 1 多国语言设定)



(图 1 7 - 2 多国语言选择)



※关于多国言語的作成，可以参照以下源代码。

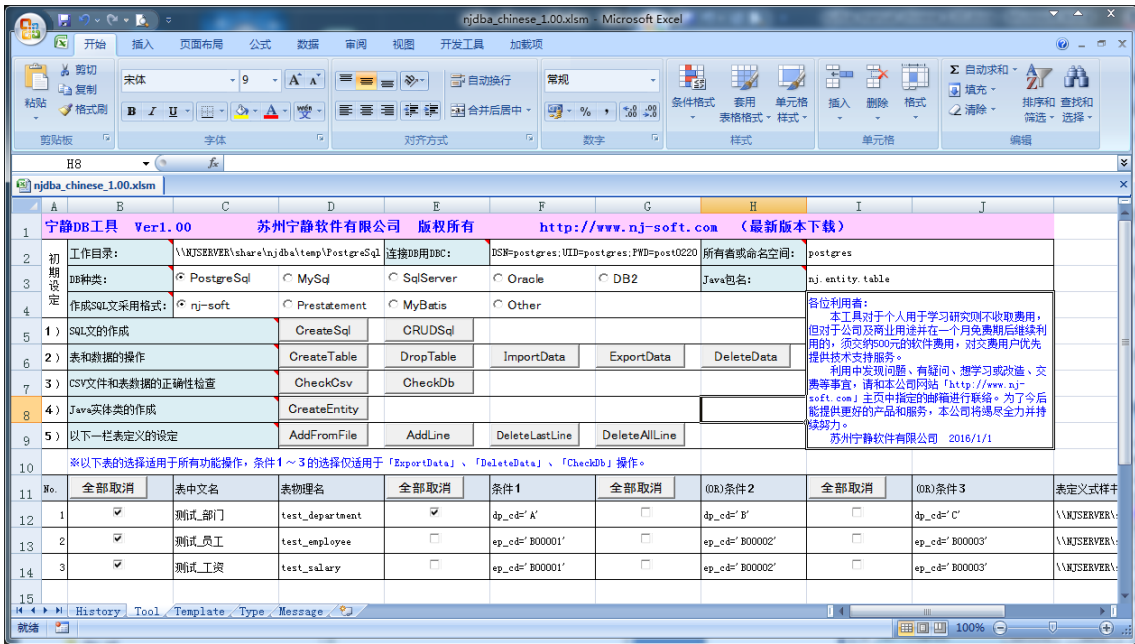
```
/njsam/src/nj/action/ApplyTrain.java
/njsam/src/nj/action/chk/Check.java
/njsam/src/nj/action/chk/Message.java
/njsam/WebContent/jsp/train/train-input.jsp
/njsam/WebContent/jsp2/train/train=input.jsp
/njsam/WebContent/jsp/include/messenger.jsp
/njsam/WebContent/jsp2/include/messenger.jsp
```

1 8 . 和宁静 DB 工具配合

宁静框架软件和宁静 DB 工具可以配合使用。宁静 DB 工具生成的动态 SQL 文，Java 实体类等可以在宁静框架软件中直接使用，能大幅度提高开发的效率和品质。

宁静 DB 工具采用 Excel 的 VBA 作成，基于表定义式样，能够完成对 DB 的各种操作和对数据的检查，以及自动生成 SQL 文和 Java 实体类。宁静 DB 工具支持 PostgreSql、MySql、SqlServer、Oracle、DB2 等五种数据库。

(图 1 8 - 1 和宁静 DB 工具配合)



※详细请参照宁静 DB 工具式样。

<http://www.nj-soft.com/njweb/Download.action?baseInfo.menuId=3030&baseInfo.curPage=0>

(全文结束)

苏州宁静软件有限公司 版权所有

<http://www.nj-soft.com>

2 0 1 6 年 5 月 1 日 劳动节