
ECサイト初級

商品購入機能の作成

5 時間目

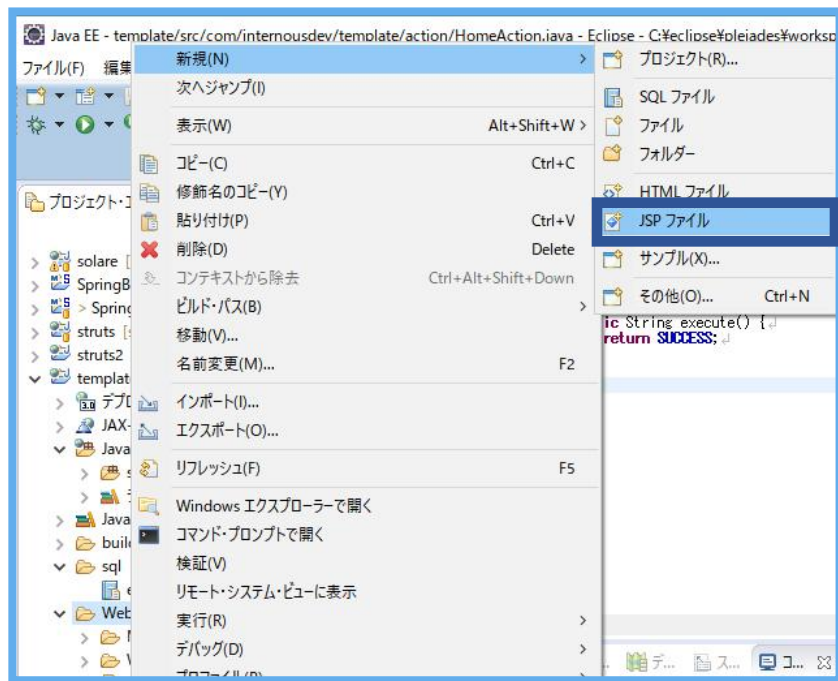
作業目次

作成機能一覧

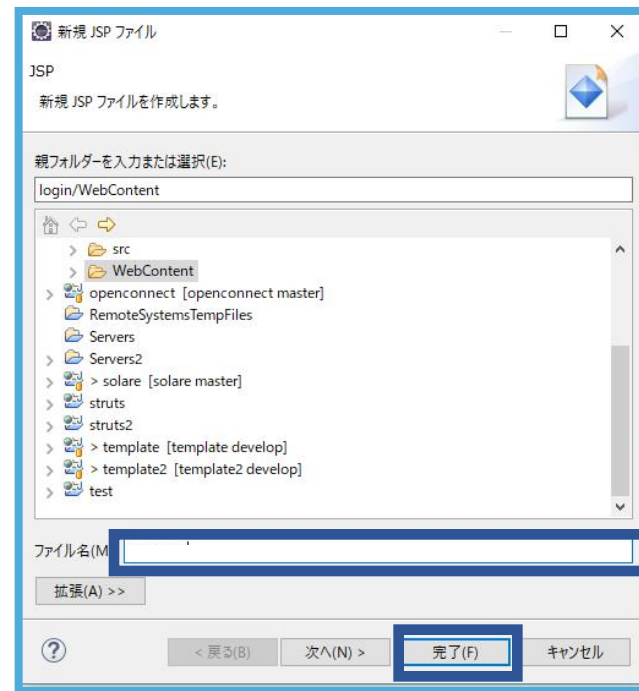
1. ログイン認証機能
2. ユーザー登録機能
- 3. 商品購入機能**
4. 購入履歴機能

byItem.jspの作成

1 byItem.jsp



① 「プロジェクト」「WebContent」を選択し、「右クリック」「新規」「JSPファイル」を選択します。



② 「名前(M):」欄に「buyItem.jsp」を入力し、完了ボタンをクリックします。

buyItem.jspの作成

③ 以下の内容を写経します。

buyItem.jsp(jspファイル)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="s" uri="/struts-tags"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta http-equiv="Content-Style-Type" content="text/css" />
    <meta http-equiv="Content-Script-Type" content="text/javascript" />
    <meta http-equiv="imagetoolbar" content="no" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <title>buyItem画面</title>

    <style type="text/css">
/* =====TAG LAYOUT===== */
```

次へ続きます。

前の続きです。

```
body {  
    margin:0;  
    padding:0;  
    line-height:1.6;  
    letter-spacing:1px;  
    font-family:Verdana, Helvetica, sans-serif;  
    font-size:12px;  
    color:#333;  
    background:#fff;  
}  
  
table {  
    text-align:center;  
    margin:0 auto;  
}  
/* =====ID LAYOUT===== */  
#top {  
    width:780px;  
    margin:30px auto;
```

次へ続きます。

前の続きです。

```
border:1px solid #333;
}

#header {
width: 100%;
height: 80px;
background-color: black;
}

#main {
width: 100%;
height: 500px;
text-align: center;
}

#footer {
width: 100%;
height: 80px;
background-color: black;
clear:both;
}
```

次へ続きます。

前の続きです。

```
</style>
</head>
<body>
  <div id="header">
    <div id="pr">
    </div>
  </div>
  <div id="main">
    <div id="top">
      <p>BuyItem</p>
    </div>
    <div>
      <s:form action="BuyItemAction">
        <table>
          <tr>
            <td>
              <span>商品名</span>
            </td>
```

次へ続きます。

前の続きです。

```
<td>
    <s:property value="session.buyItem_name" />
</td>
</tr>
<tr>
    <td>
        <span>値段</span>
    </td>
    <td>
        <s:property value="session.buyItem_price" />
        <span>円</span>
    </td>
</tr>
<tr>
    <td>
        <span>在庫</span>
    </td>
    <td>
        <select name="stock">
```

次へ続きます。

前の続きです。

```

                                <option value="1" selected="selected">1</option>
                                <option value="2">2</option>
                                <option value="3">3</option>
                                <option value="4">4</option>
                                <option value="5">5</option>
                                </select>
                                </td>
</tr>
<tr>
    <td>
        <span>支払い方法</span>
    </td>
    <td>
        <input type="radio" name="pay" value="1"
checked="checked">現金払い
        <input type="radio" name="pay" value="2">クレ
ジットカード
    </td>
</tr>
<tr>
```

次へ続きます。

前の続きです。

```
<td>
    <s:submit value="購入" />
</td>
</tr>
</table>
</s:form>
<div>
    <span>前画面に戻る場合は</span>
    <a href='<s:url action="HomeAction" />'>こちら</a>
</div>
</div>
</div>
<div id="footer">
    <div id="pr">
    </div>
</div>
</body>
</html>
```

byItem.jspの作成

byItem.jspの画面



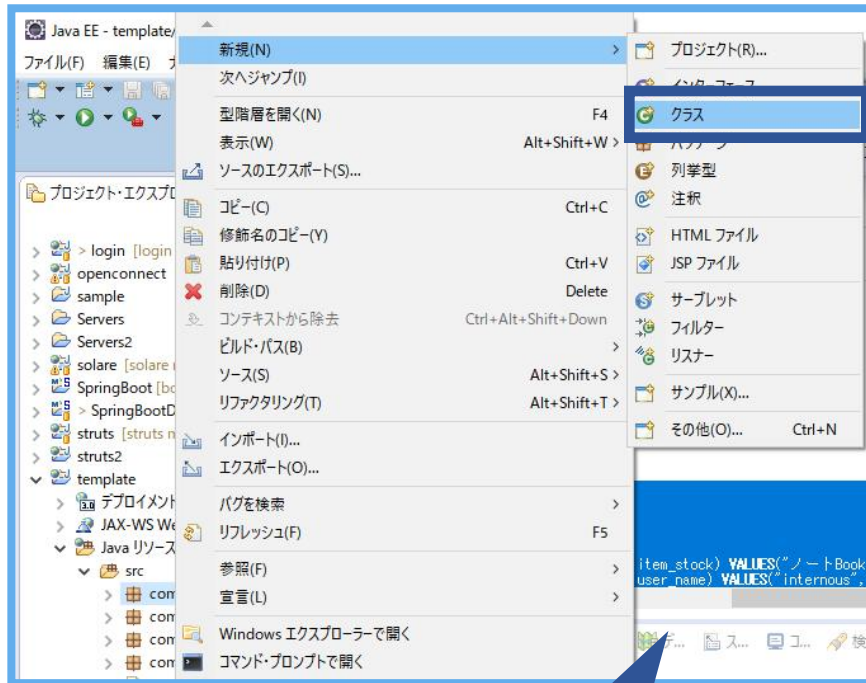
The screenshot displays a web page titled "BuyItem" within a black header and footer. The page content is centered and includes the following elements:

- A title bar labeled "BuyItem".
- Product details: "商品名" (Product Name) as "NoteBook", "値段" (Price) as "100円", and "在庫" (Inventory) as a dropdown menu showing "1".
- Payment method: "支払い方法" (Payment Method) with radio buttons for "現金払い" (Cash Payment, selected) and "クレジットカード" (Credit Card).
- A "購入" (Purchase) button.
- A link: "前画面に戻る場合は[こちら](#)" (To return to the previous screen, click [here](#)).

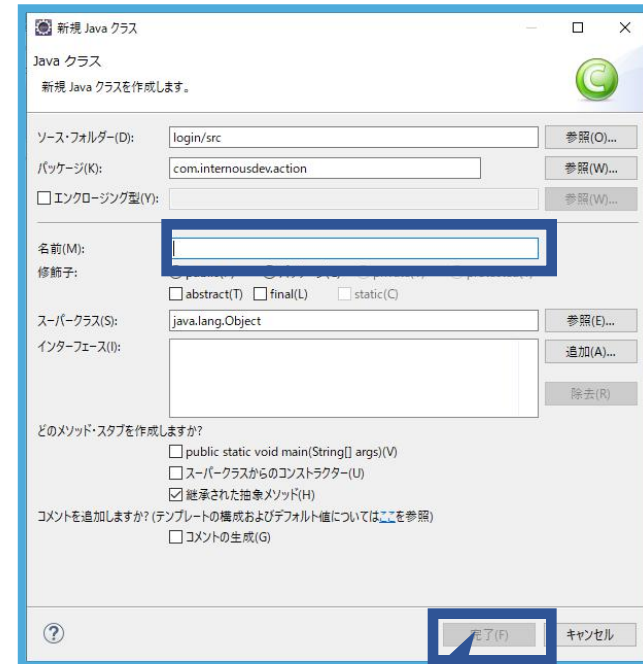
上記の画面が表示できれば成功です。

BuyItemAction.javaの作成

2 BuyItemAction.java



① 「com.internousdev.tempalte.action」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「BuyItemAction」を入力し、完了ボタンをクリックします。

BuyItemAction.javaの作成

③ 以下の内容を写経します。

BuyItemAction.java
(javaファイル)

```
package com.internousdev.template.action;
import java.util.Map;
import org.apache.struts2.interceptor.SessionAware;
import com.opensymphony.xwork2.ActionSupport;

public class BuyItemAction extends ActionSupport implements SessionAware {

    private int stock;
    private String pay;
    public Map<String, Object> session;
    private String result;

    public String execute() {
        result = SUCCESS;

        session.put("stock", stock);
        int intStock = Integer.parseInt(session.get("stock").toString());
        int intPrice = Integer.parseInt(session.get("buyItem_price").toString());
```

次へ続きます。

BuyItemAction.javaの作成

前の続きです。

```
session.put("buyItem_price", intStock * intPrice);  
String payment;  
  
if(pay.equals("1")) {  
    payment = "現金払い";  
    session.put("pay", payment);  
} else {  
    payment = "クレジットカード";  
    session.put("pay", payment);  
}  
return result;  
}
```

次へ続きます。

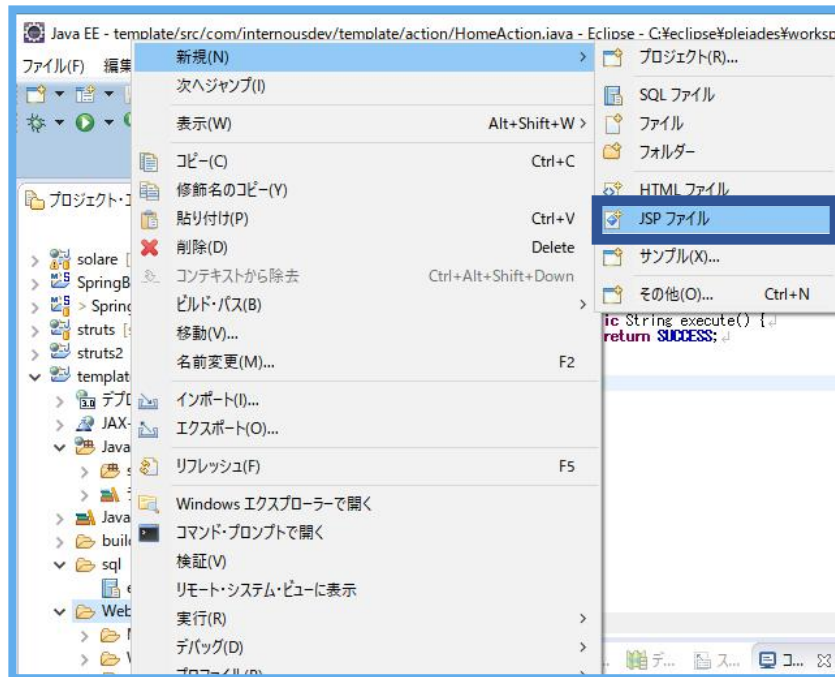
BuyItemAction.javaの作成

前の続きです。

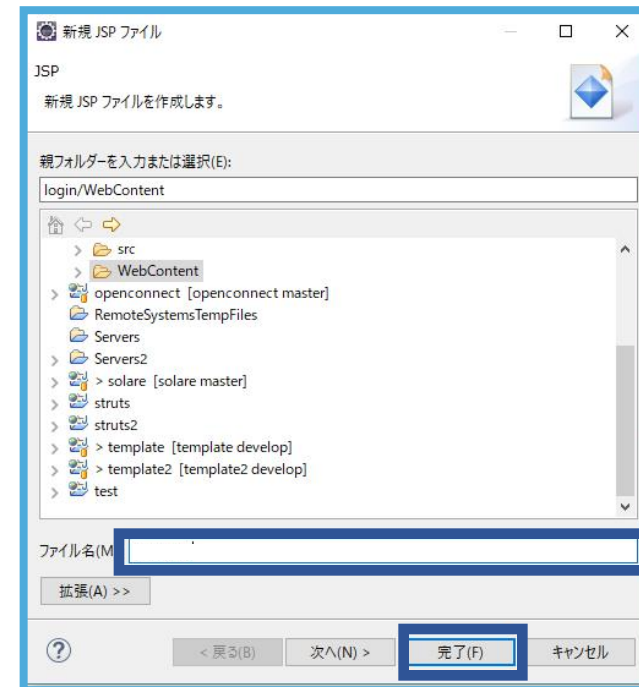
```
public int getStock() {  
    return stock;  
}  
public void setStock(int stock) {  
    this.stock = stock;  
}  
public String getPay() {  
    return pay;  
}  
public void setPay(String pay) {  
    this.pay = pay;  
}  
@Override  
public void setSession(Map<String, Object> session) {  
    this.session = session;  
}  
}
```

buyItemConfirm.jspの作成

4 buyItemConfirm.jsp



① 「プロジェクト」「WebContent」を選択し、「右クリック」「新規」「JSPファイル」を選択します。



② 「名前(M):」欄に「buyItemConfirm.jsp」を入力し、完了ボタンをクリックします。

buyItemConfirm.jspの作成

③ 以下の内容を写経します。

buyItemConfirm.jsp
(jspファイル)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="s" uri="/struts-tags"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta http-equiv="Content-Style-Type" content="text/css" />
    <meta http-equiv="Content-Script-Type" content="text/javascript" />
    <meta http-equiv="imagetoolbar" content="no" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <title>buyItemConfirm画面</title>

    <style type="text/css">
/* =====TAG LAYOUT===== */
```

次へ続きます。

buyItemConfirm.jspの作成

前の続きです。

```
body {  
    margin:0;  
    padding:0;  
    line-height:1.6;  
    letter-spacing:1px;  
    font-family:Verdana, Helvetica, sans-serif;  
    font-size:12px;  
    color:#333;  
    background:#fff;  
}  
  
table {  
    text-align:center;  
    margin:0 auto;  
}  
/* =====ID LAYOUT===== */  
#top {  
    width:780px;  
    margin:30px auto;
```

次へ続きます。

前の続きです。

```
border:1px solid #333;
}

#header {
width: 100%;
height: 80px;
background-color: black;
}

#main {
width: 100%;
height: 500px;
text-align: center;
}

#footer {
width: 100%;
height: 80px;
background-color: black;
clear:both;
}
```

次へ続きます。

buyItemConfirm.jspの作成

前の続きです。

```
</style>
</head>
<body>
  <div id="header">
    <div id="pr">
    </div>
  </div>
  <div id="main">
    <div id="top">
      <p>BuyItemConfirm</p>
    </div>
    <div>
      <s:form action="BuyItemConfirmAction">
        <table>
          <tr>
            <td>商品名</td>
            <td><s:property value="session.buyItem_name" /></td>
          </tr>
        </table>
      </s:form>
    </div>
  </div>
</body>
```

次へ続きます。

buyItemConfirm.jspの作成

前の続きです。

```
</tr>
<tr>
  <td>値段</td>
  <td>
    <s:property value="session.buyItem_price" />
    <span>円</span>
  </td>
</tr>
<tr>
  <td>購入回数</td>
  <td>
    <s:property value="session.stock" />
    <span>個</span>
  </td>
</tr>
<tr>
  <td>支払い方法</td>
  <td><s:property value="session.pay" /></td>
</tr>
```

次へ続きます。

buyItemConfirm.jspの作成

前の続きです。

```

                                <tr>
                                <td><s:submit value="完了" /></td>
                                </tr>
                                </table>
                                </s:form>
                                </div>
</div>
<div id="footer">
    <div id="pr">
    </div>
</div>
</body>
</html>
```

buyItemConfirm.jspの作成

buyItemConfirm.jspの画面

BuyItemConfirm

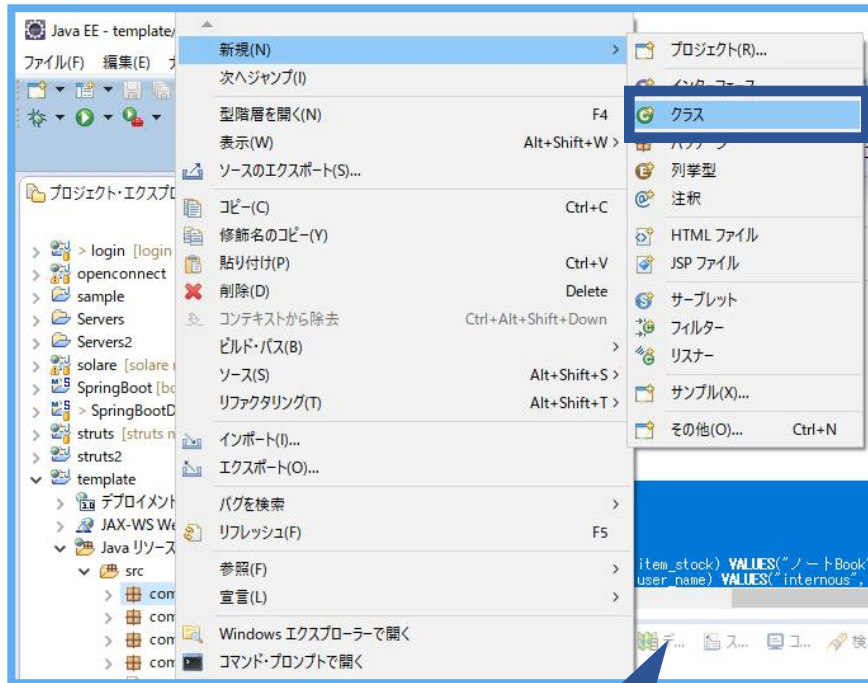
商品名	NoteBook
値段	100円
購入個数	1個
支払い方法	現金払い

完了

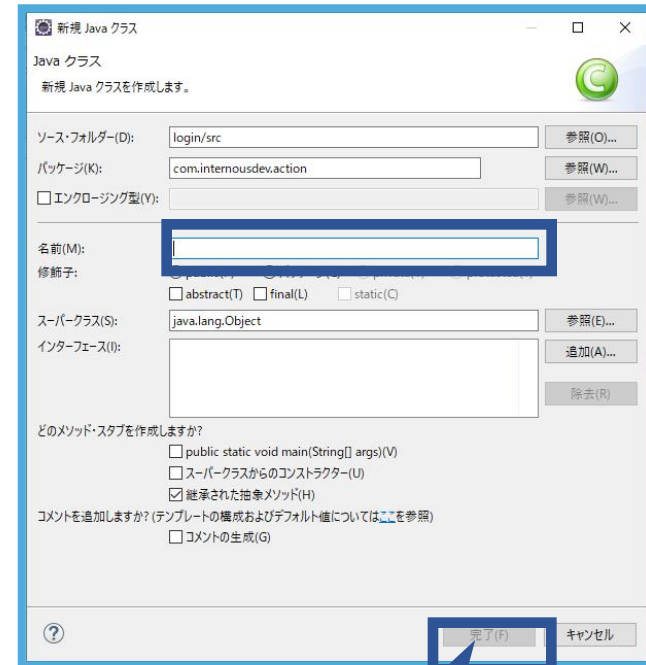
上記の画面が表示できれば成功です。

BuyItemConfirmAction.javaの作成

5 BuyItemConfirmAction.java



① 「com.internousdev.tempalte.action」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「BuyItemConfirmAction」を入力し、完了ボタンをクリックします。

BuyItemConfirmAction.javaの作成

③ 以下の内容を写経します。

BuyItemConfirmAction.java
(javaファイル)

```
package com.internousdev.template.action;
import java.sql.SQLException;
import java.util.Map;
import org.apache.struts2.interceptor.SessionAware;
import com.internousdev.template.dao.BuyItemCompleteDAO;
import com.opensymphony.xwork2.ActionSupport;

public class BuyItemConfirmAction extends ActionSupport implements SessionAware{
    public Map<String,Object> session;
    private String result;
    private BuyItemCompleteDAO buyItemCompleteDAO = new BuyItemCompleteDAO();
    public String execute() throws SQLException {

        buyItemCompleteDAO.buyItemInfo(
            session.get("id").toString(),
```

次へ続きます。

BuyItemConfirmAction.javaの作成

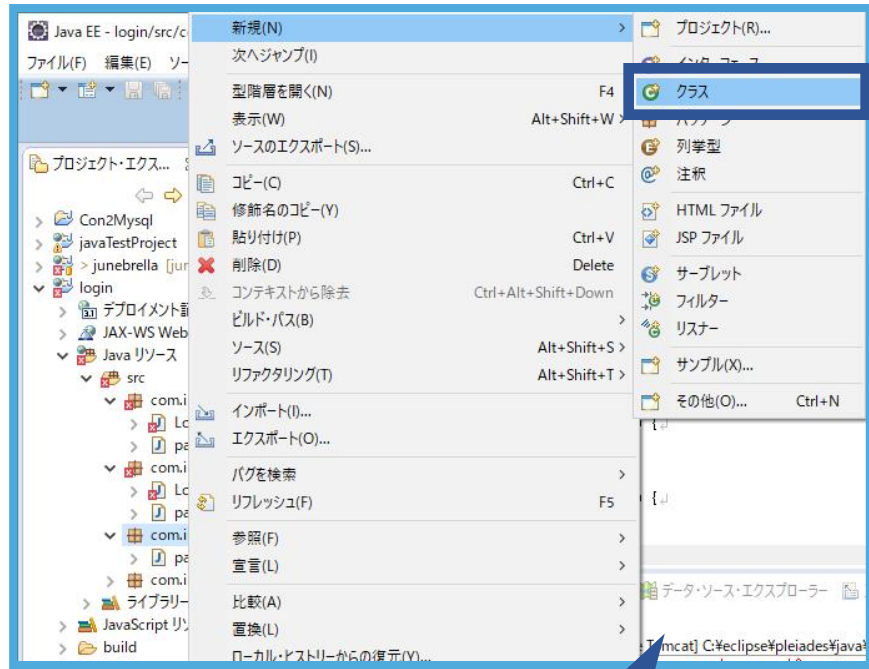
前の続きです。

```
        session.get("login_user_id").toString(),  
        session.get("buyItem_price").toString(),  
        session.get("stock").toString(),  
        session.get("pay").toString());  
  
        result = SUCCESS;  
        return result;  
    }  
  
    @Override  
    public void setSession(Map<String, Object> session) {  
        this.session = session;  
    }  
}
```

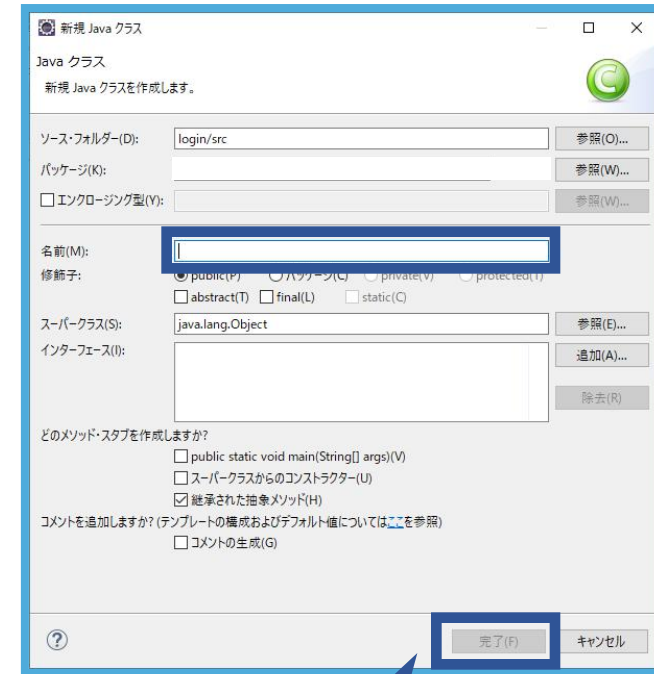
次へ続きます。

BuyItemCompleteDAO.javaの作成

6 BuyItemCompleteDAO.java



① 「com.internousdev.template.dao」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「BuyItemCompleteDAO」を入力し、完了ボタンをクリックします。

BuyItemCompleteDAO.javaの作成

BuyItemCompleteDAO.java
(javaファイル)

③ 以下の内容を写経します。

```
package com.internousdev.template.dao;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import com.internousdev.template.util.DBConnector;
import com.internousdev.template.util.DateUtil;

public class BuyItemCompleteDAO {

    private DBConnector dbConnector = new DBConnector();
    private Connection connection = dbConnector.getConnection();
    private DateUtil dateUtil = new DateUtil();

    private String sql = "INSERT INTO user_buy_item_transaction (item_transaction_id,
total_price, total_count, user_master_id, pay, insert_date) VALUES(?, ?, ?, ?, ?, ?)";

    public void buyItemInfo(String item_transaction_id, String user_master_id, String
total_price, String total_count, String pay) throws SQLException {
```

次へ続きます。

BuyItemCompleteDAO.javaの作成

前の続きです。

```
try {
    PreparedStatement preparedStatement = connection.prepareStatement(sql);
    preparedStatement.setString(1, item_transaction_id);
    preparedStatement.setString(2, total_price);
    preparedStatement.setString(3, total_count);
    preparedStatement.setString(4, user_master_id);
    preparedStatement.setString(5, pay);
    preparedStatement.setString(6, dateUtil.getDate());

    preparedStatement.execute();

} catch (Exception e) {
    e.printStackTrace();
} finally {
    connection.close();
}
}
```

BuyItemCompleteDAO.javaの作成

セキュリティ対策を考慮してJavaではPreparedStatementを利用します。

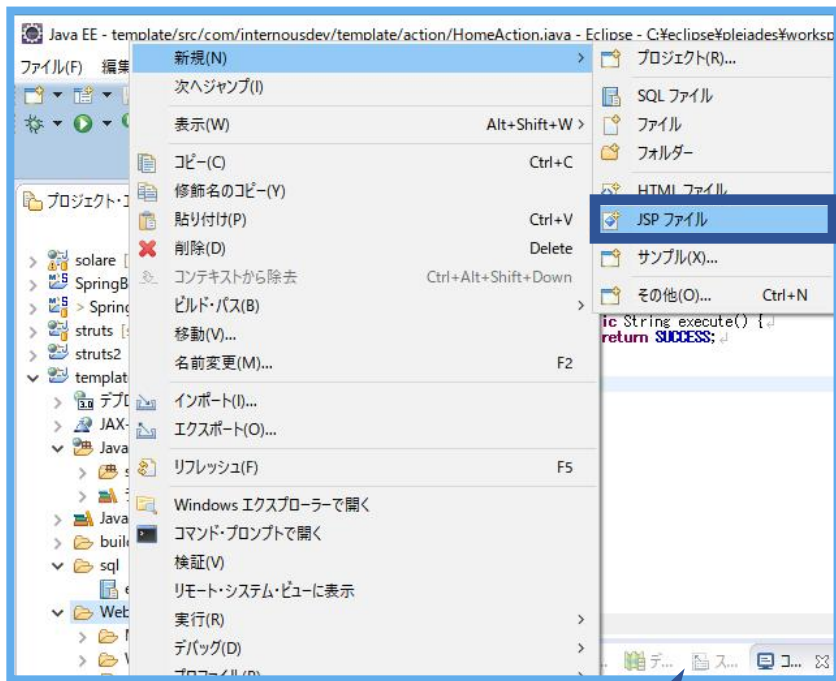
```
PreparedStatement preparedStatement = connection.prepareStatement(sql);  
preparedStatement.setString(1, item_transaction_id);  
preparedStatement.setString(2, total_price);  
preparedStatement.setString(3, total_count);  
preparedStatement.setString(4, user_master_id);  
preparedStatement.setString(5, pay);  
preparedStatement.setString(6, dateUtil.getDate());
```

SQL文の「？」パラメータに指定した値を挿入することができます。

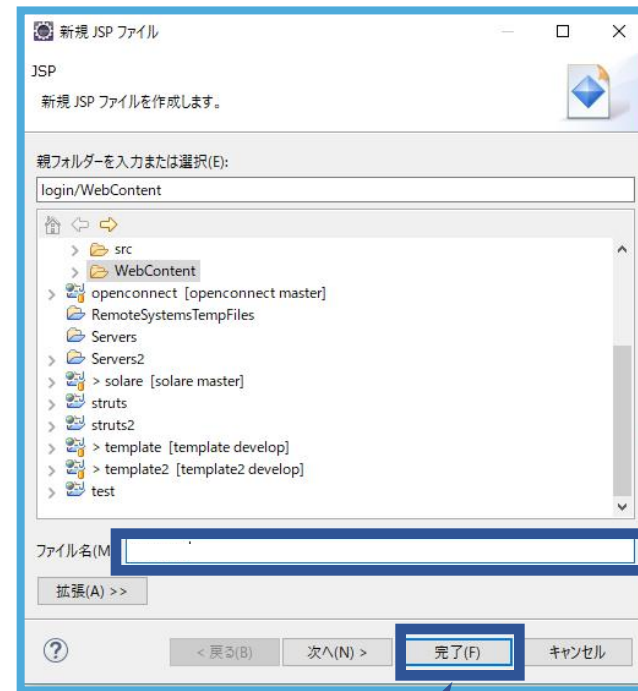
```
private String sql = "INSERT INTO user_buy_item_transaction  
(item_transaction_id, total_price, total_count, user_master_id, pay, insert_date)  
VALUES(?, ?, ?, ?, ?, ?)";
```

buyItemComplete.jspの作成

7 buyItemComplate.jsp



① 「プロジェクト」「WebContent」を選択し、「右クリック」「新規」「JSPファイル」を選択します。



② 「名前(M):」欄に「buyItemComplete.jsp」を入力し、完了ボタンをクリックします。

buyItemComplete.jspの作成

③ 以下の内容を写経します。

buyItemComplete.jsp
(jspファイル)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="s" uri="/struts-tags"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta http-equiv="Content-Style-Type" content="text/css" />
    <meta http-equiv="Content-Script-Type" content="text/javascript" />
    <meta http-equiv="imagetoolbar" content="no" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <title>buyItemComplete画面</title>

    <style type="text/css">
/* =====TAG LAYOUT===== */
```

次へ続きます。

buyItemComplete.jspの作成

前の続きです。

```
body {  
    margin:0;  
    padding:0;  
    line-height:1.6;  
    letter-spacing:1px;  
    font-family:Verdana, Helvetica, sans-serif;  
    font-size:12px;  
    color:#333;  
    background:#fff;  
}  
  
table {  
    text-align:center;  
    margin:0 auto;  
}  
/* =====ID LAYOUT===== */  
#top {  
    width:780px;  
    margin:30px auto;
```

次へ続きます。

buyItemComplete.jspの作成

前の続きです。

```
border:1px solid #333;
}

#header {
width: 100%;
height: 80px;
background-color: black;
}

#main {
width: 100%;
height: 500px;
text-align: center;
}

#footer {
width: 100%;
height: 80px;
background-color: black;
clear:both;
}
```

次へ続きます。

buyItemComplete.jspの作成

前の続きです。

```
</style>
</head>
<body>
  <div id="header">
    <div id="pr">
    </div>
  </div>
  <div id="main">
    <div id="top">
      <p>BuyItemComplete</p>
    </div>
    <div>
      <p>購入手続きが完了致しました。 </p>
      <div>
        <a href='<s:url action="MyPageAction" />'>
          マイページ</a><span>から購入履歴の確認が可能です。 </span>

```

次へ続きます。

buyItemComplete.jspの作成

前の続きです。

```
        </div>
    </div>
</div>
<div id="footer">
    <div id="pr">
    </div>
</div>
</body>
</html>
```

次へ続きます。

buyItemComplete.jspの作成

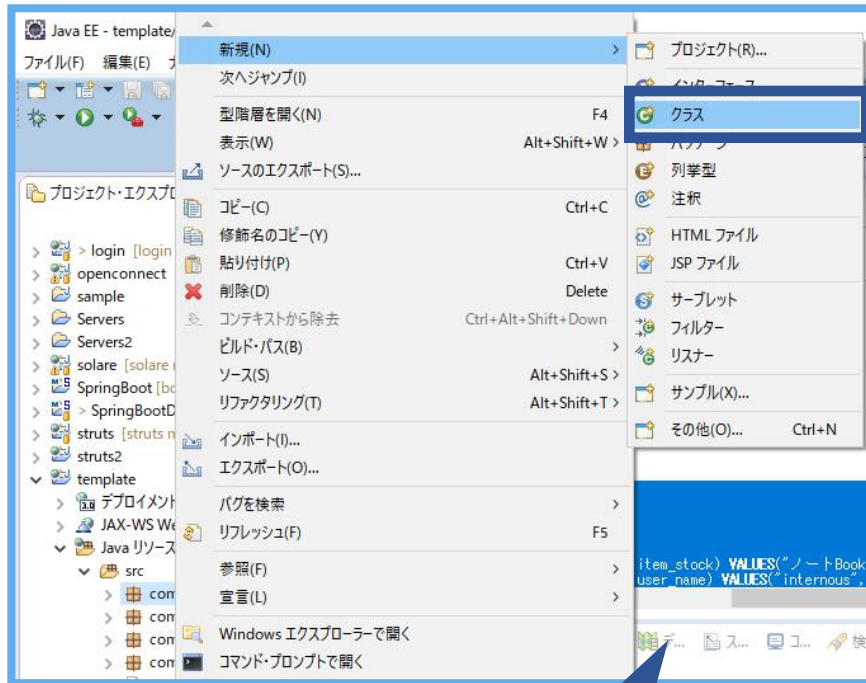
buyItemComplete.jspの画面



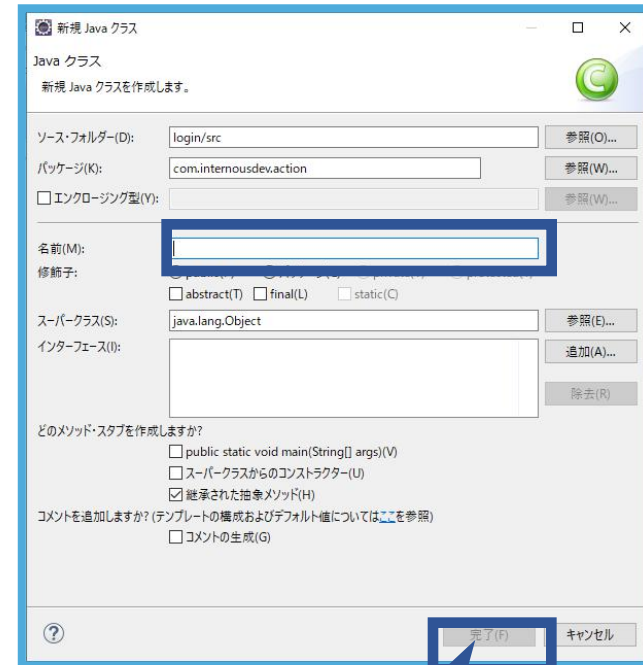
上記の画面が表示できれば成功です。

BuyItemCompleteAction.javaの作成

5 BuyItemConfirmAction.java



① 「com.internousdev.tempalte.action」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「BuyItemCompleteAction」を入力し、完了ボタンをクリックします。

BuyItemCompleteAction.javaの作成

③ 以下の内容を写経します。

BuyItemCompleteAction.java
(javaファイル)

```
package com.internousdev.template.action;
import java.util.Map;
import org.apache.struts2.interceptor.SessionAware;
import com.opensymphony.xwork2.ActionSupport;
public class BuyItemCompleteAction extends ActionSupport implements SessionAware{

    public Map<String,Object> session;
    private String result;

    public String execute() {

        result = SUCCESS;
        return result;
    }
}
```

次へ続きます。

BuyItemConfirmAction.javaの作成

前の続きです。

```
@Override
public void setSession(Map<String, Object> session) {
    this.session = session;
}
}
```