From Forms to APEX
Summit Case Study

Using PITSS.CON APEX Assistant
to re-engineer an Oracle Forms application
to Oracle Application Express

PITSS.CON 12.3.1

Case Study, May 2014

Introduction

PITSS.CON Forms2APEX Assistant

This assistant helps to convert your Oracle Forms application to Oracle Application Express. This How-To will go through the conversion process for the Oracle Forms demo application, Summit, which can be downloaded from OTN.

1. Audience

This guide is for Oracle Forms developers with basic knowledge of APEX programming.

2. Prerequisites

2.1 Demo schema installation

For this guide it is not necessary to run the Summit Demo Application. It is sufficient to create the database user and import the database dump. Download the Forms Summit Application from OTN and follow the enclosed installation instructions.

To use the TIF images included in the Summit Demo Application they have to be converted to GIFs, then these images have to be copied to the image folder of the web server. But not everyone has access to a web server, so this document shows how to do it another way, which everyone can use (Application Builder -> Shared Components -> Files -> Images -> Upload as Application Image or as Workspace Image).

2.2 APEX installation

As described in the Oracle Application Express Installation Guide—How to install Oracle Application Express. To access Oracle APEX there are 4 ways to configure APEX on your database:

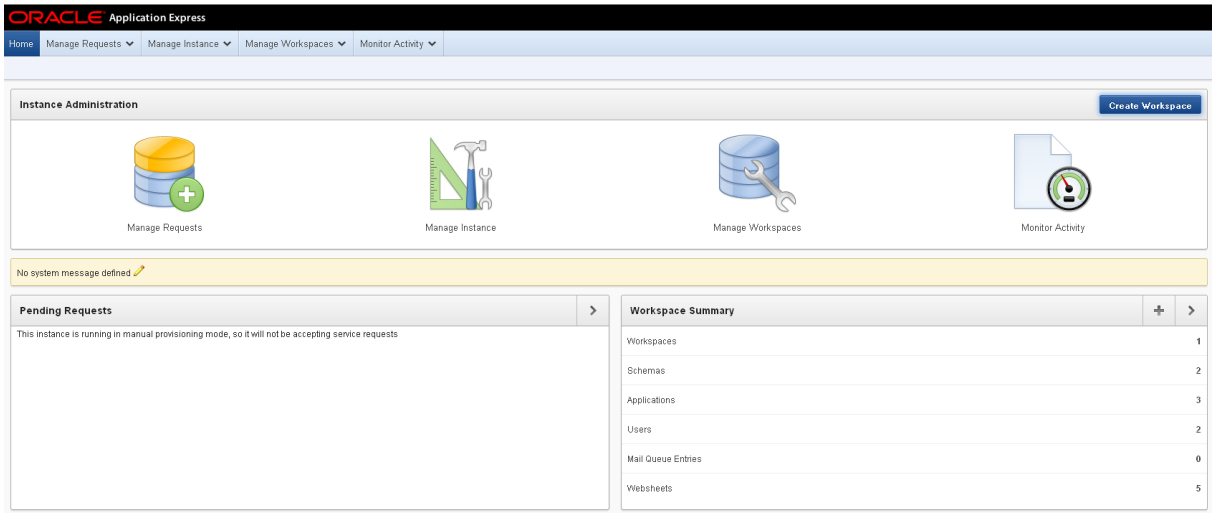
- Oracle Rest Data Services (formerly known as Oracle APEX Listener) in standalone mode
- Oracle Rest Data Services deployed on Application servers (Weblogic, Glassfish, Tomcat)
- Oracle HTTP Server
- Embedded PL/SQL Gateway

Please choose the appropriate approach for your environment.

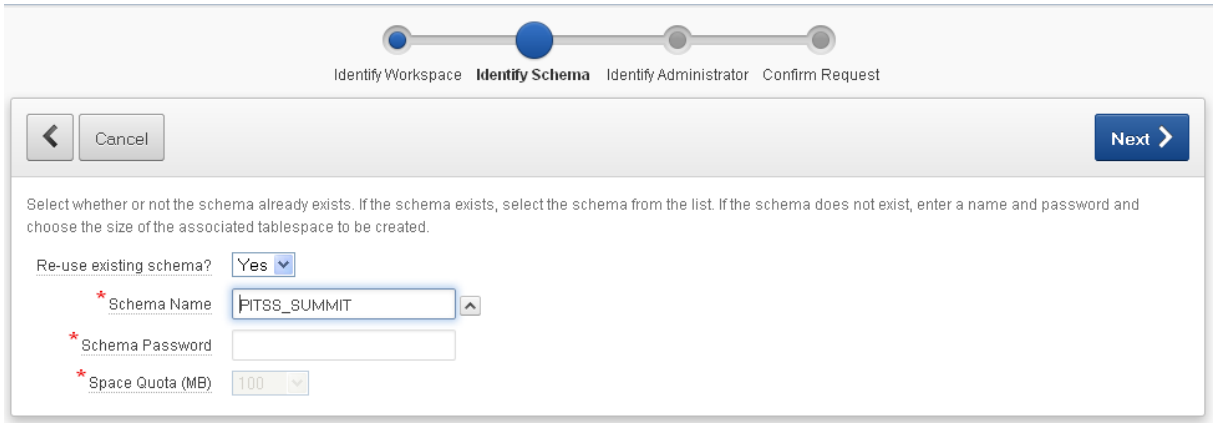
2.3 APEX environment preparation

After the installation please set up a workspace for the demo schema and create a developer user so that the generated APEX application can be imported and edited.

To create a new workspace, login as Administrator and click on the button “Create Workspace”.

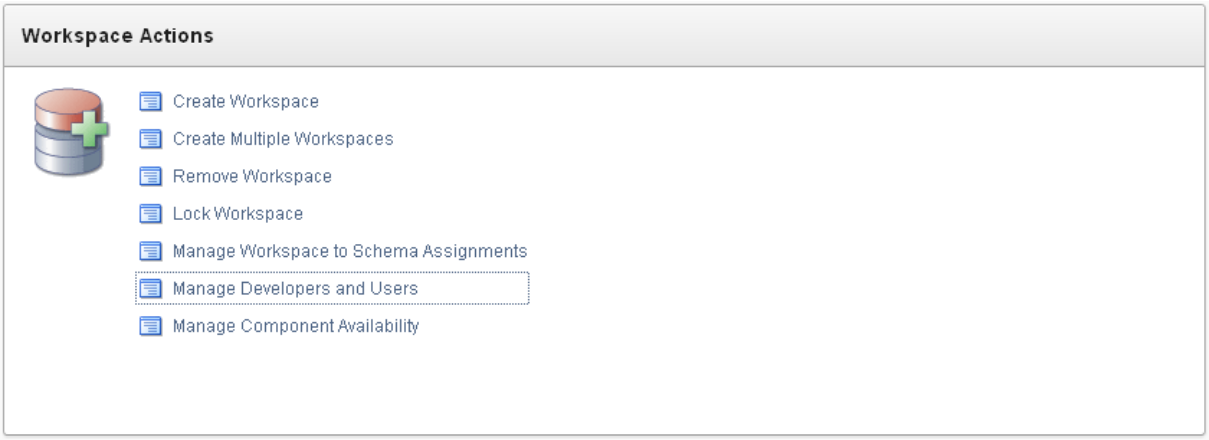


Set a name for the workspace in this dialog and select to reuse an existing schema—choose SUMMIT as the schema.

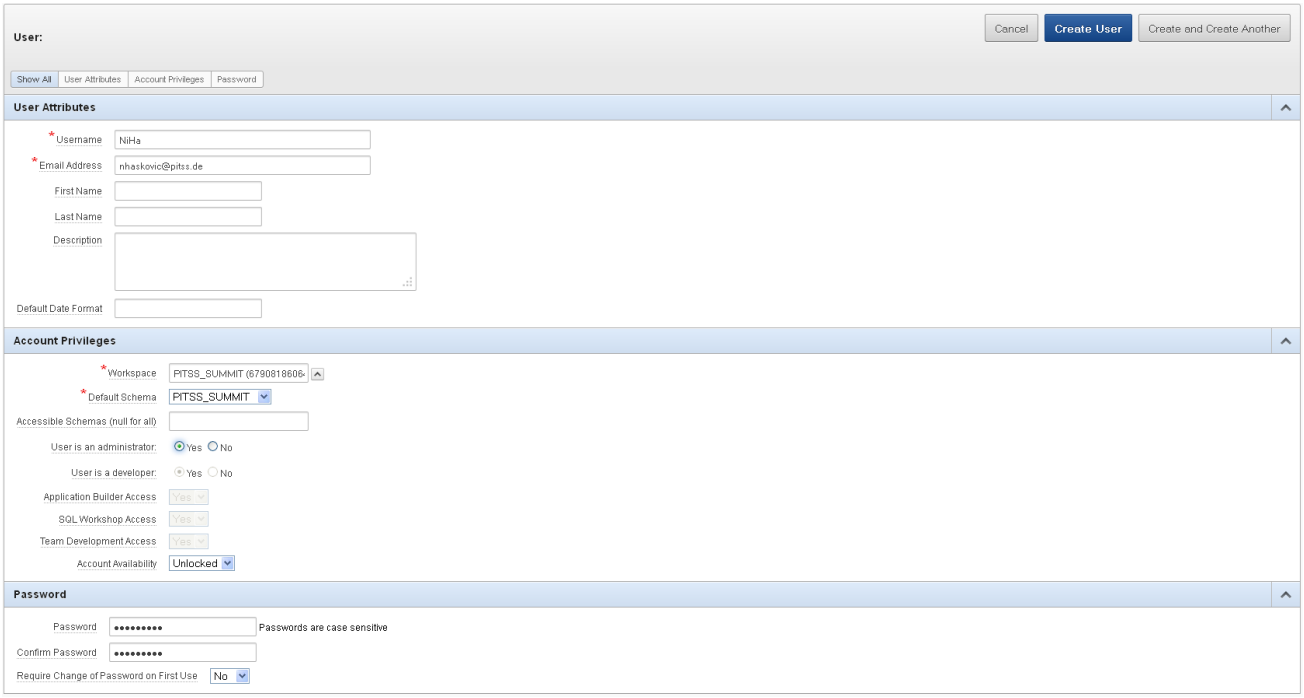


Choose an administrator name and a password to finish creating the new workspace.

To use the workspace please create users. To do so click the link “Manage Developers and Users” in the Workspace Actions module.



Fill in the requested information and set SUMMIT as the default schema.



Further information can be found in the Oracle Application Express Administration Guide.

3. Conversion

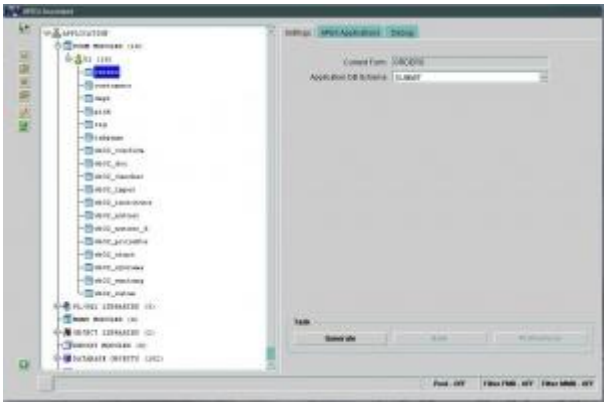
3.1 Import application to the PITSS.CON repository

- Copy all FMBs files to the fmb folder of the PITSS.CON user.
- Copy all libraries (OLB, PLL) to the olb folder of the PITSS.CON user.
- Load all following objects of the Summit Demo application with the PITSS.CON Maintenance Module in the repository:
  a. database schema
  b. Forms libraries
  c. Forms

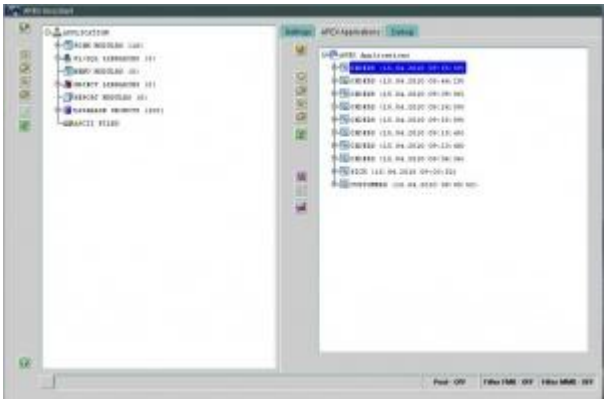
3.2 Forms2APEX Assistant

1. Open the Forms2APEX Assistant.

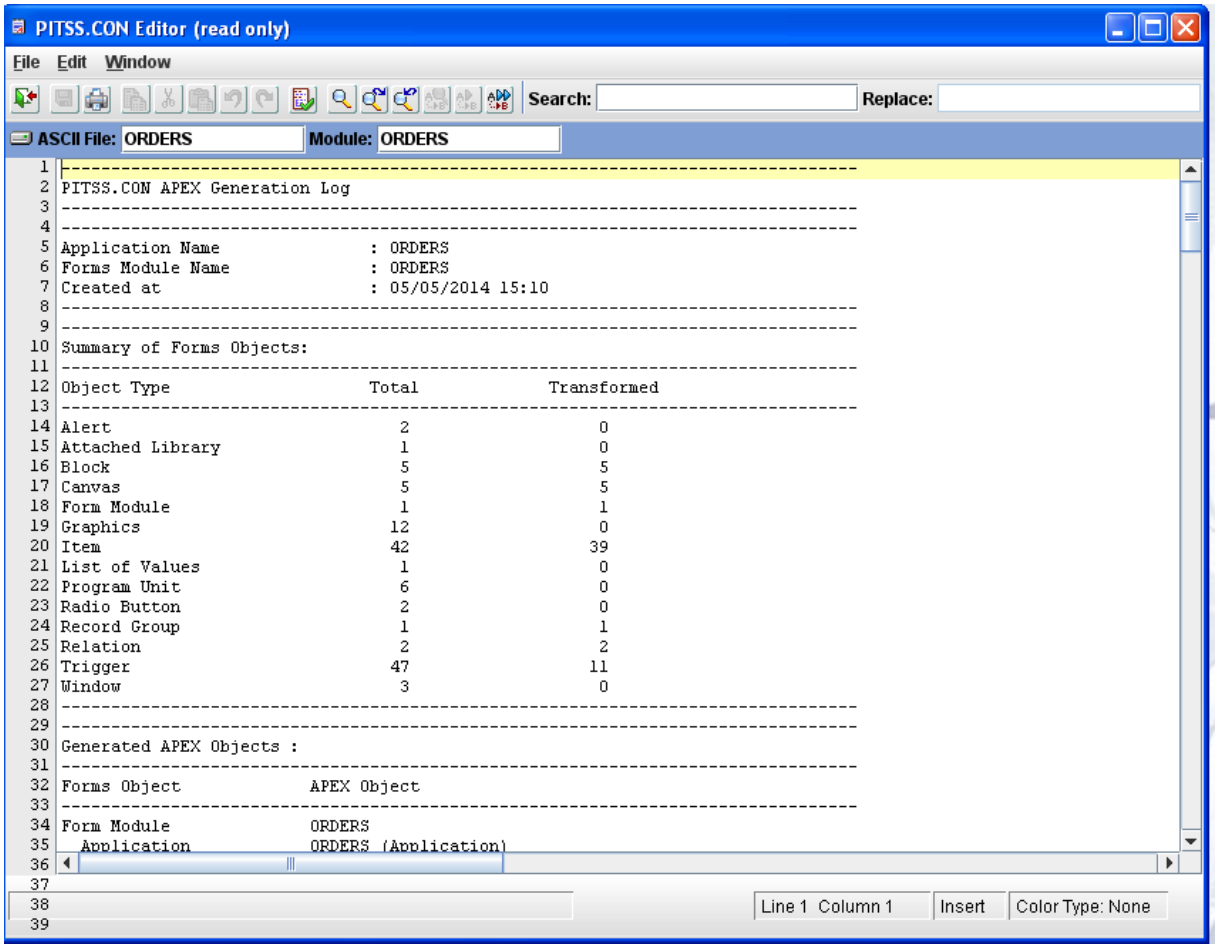
2. Choose the ORDERS Form from the Form Modules branch.



3. Choose the SUMMIT schema on the **Settings** tab page.
4. Press the **Generate** button.
5. Change to the **APEX Applications** tab page.

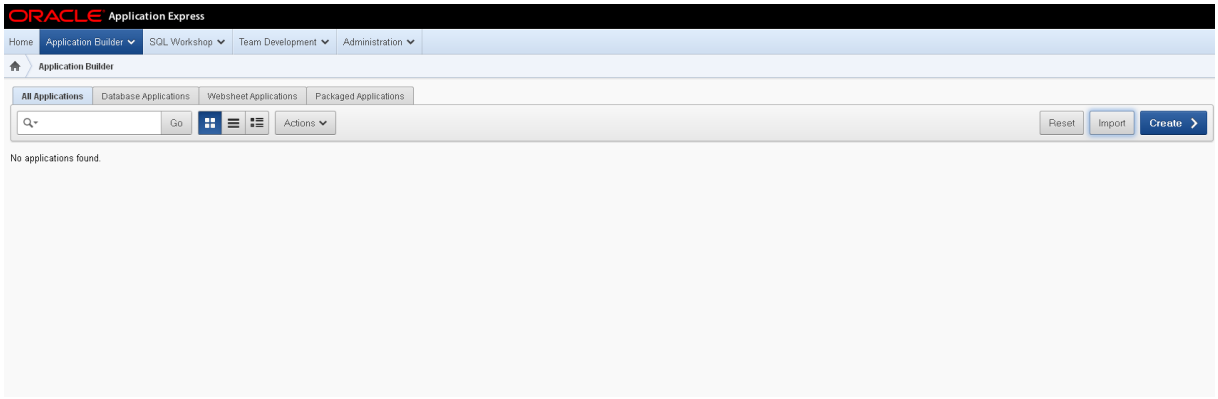


6. Click on the converted ORDERS Form in the tree.
7. Press the **Save** button to write the APEX PL/SQL script to the file system.
8. Right-click the on root of the converted ORDERS Form application and choose **View Log**.
9. The PITSS.CON Editor opens and the log file can be saved or printed.

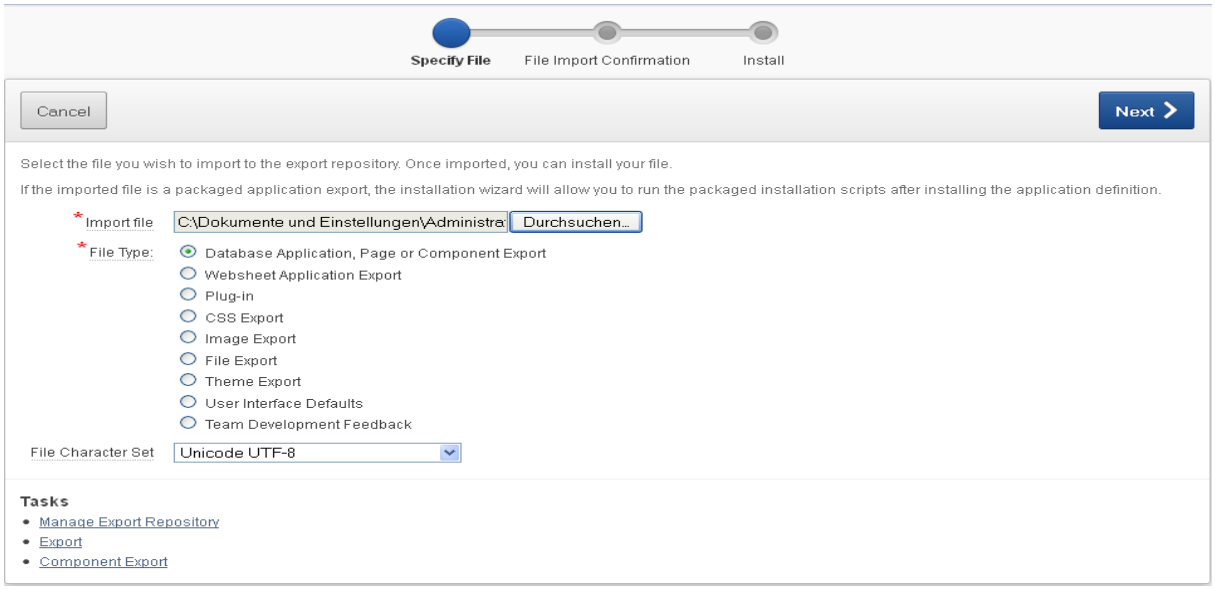


3.3 Import the conversion output

1. Log into Apex as developer or administrator.
2. Open the Application Builder.

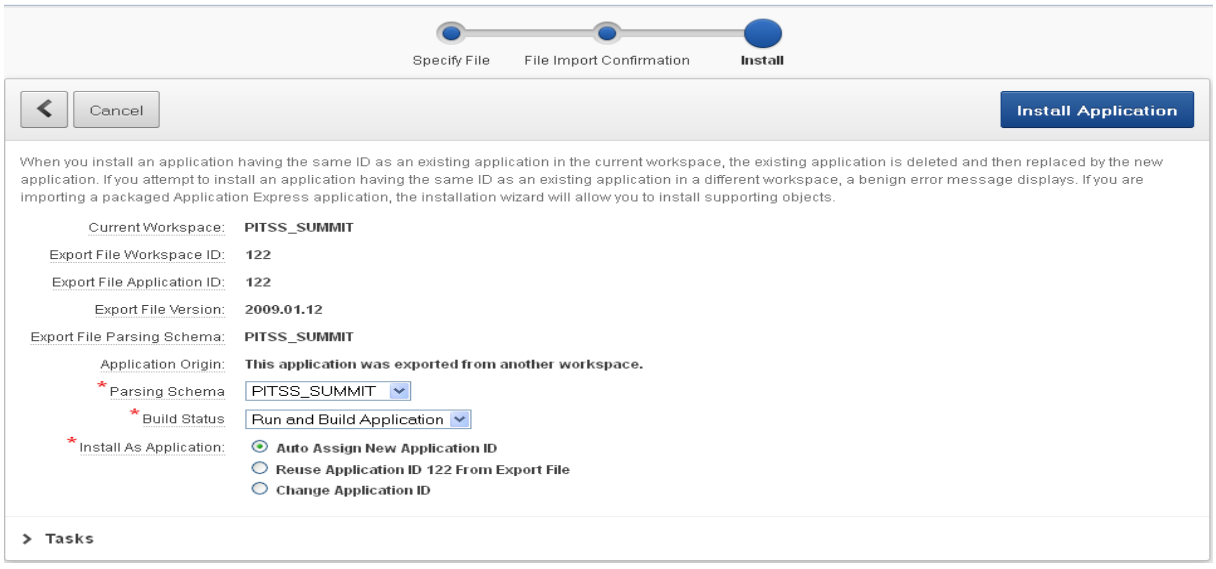


3. Click the **Import** button.
4. For the Import file select the APEX PL/SQL script which has been saved before.
   - **File Type**: Application, Page or Component Export
   - **File Character Set:** Unicode UTF-8

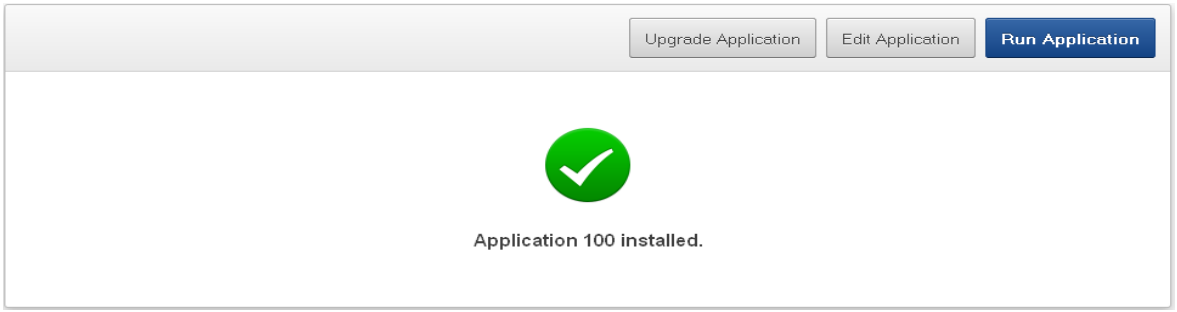


5. After uploading the file choose SUMMIT as parsing schema.

   - **Build Status:** Run and Build Application
   - **Install As Application:** Auto Assign New Application ID



6. Install and edit the application.

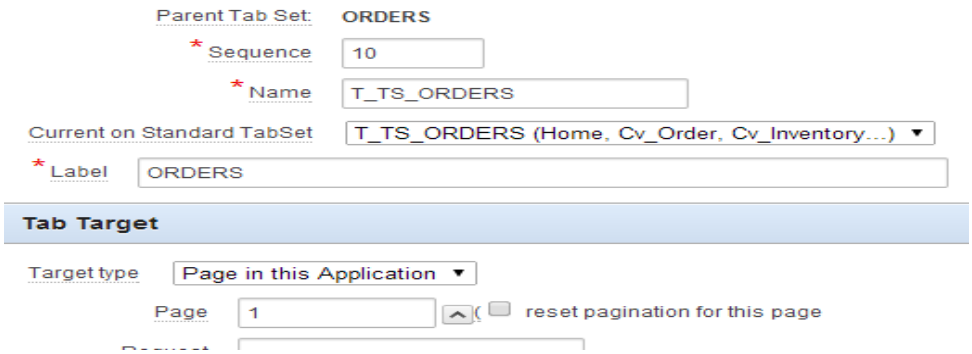


4. Manual adjustments

4.1 Edit the order page

Go to Shared Components -> Navigation -> Tabs -> Manage tabs (Tab Set: T_TS_ORDERS) and click on "Cv_Order". Change the label to "Orders".

At "Manage Tabs" click on the parent tab called "Orders" and change the attributes as seen here:



Go to the edit page and change the title to "Orders".

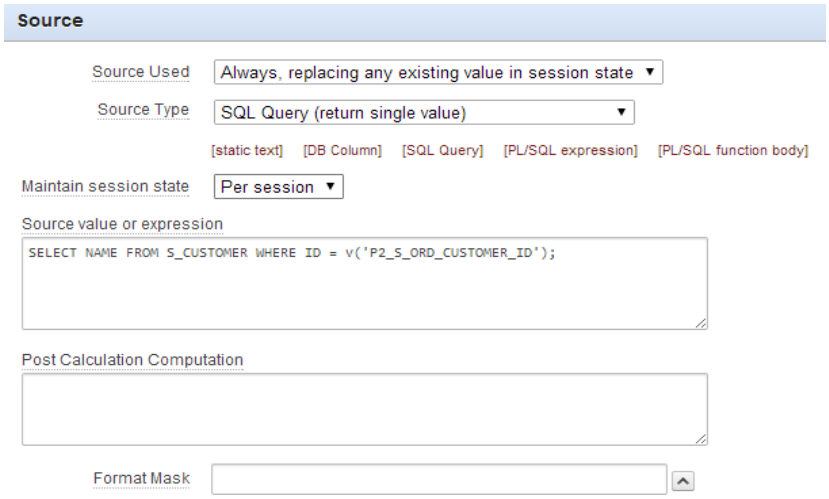
Editing the region S_ORD

- Go to Page 2.
- Change the title to "Order Information" and change User Interface -> Template -> to "Reports Region".
- Apply the changes.
- Go to the "Delete" button in "Order Information", set the condition to "Not Exists" and enter the following code:

```
SELECT 1
FROM S_ITEM
WHERE ORD_ID = :P2_S_ORD_ID
```

➔ This is needed so that only the order data that has no more references can be deleted.

Set the customer name to be displayed-> Go to P2_S_ORD_CUSTOMER_NAME -> Source:



Editing the region S_ITEM

Some columns have to be added to this report region. Columns are missing for the name of the image of an item as well as one column for displaying the image itself. Before the function called “get_product_image” can be used it has to be compiled in the SUMMIT schema using SQL Developer (or SQL PLUS, TOAD etc.):

```
create or replace FUNCTION "GET_PRODUCT_IMAGE" (product_number IN
NUMBER) RETURN VARCHAR2 IS
v_filename VARCHAR2(20);--
BEGIN
       SELECT s_image.filename INTO v_filename
       FROM   s_image, s_product
       WHERE  s_image.id = s_product.image_id
              AND
              s_product.id = product_number;
       if v_filename is null then
          v_filename := 'No file';
       end if;
       RETURN v_filename;
EXCEPTION
       WHEN no_data_found THEN return('No file');
END;
```

- Change the region title to “Items” and change User Interface -> Template -> to “Reports Region”
- Adapt the region source SQL query :
  - Add a column for the Item Picture

    ,replace(get_product_image(PRODUCT_ID), '.tif','.gif') image_name

The new SQL Query should like this:

```
SELECT
 "ROWID"
,"ITEM_ID"
,"ORD_ID"
,"PRICE"
,"PRODUCT_ID"
,"QUANTITY"
,"QUANTITY_SHIPPED"
,(QUANTITY_SHIPPED*PRICE) Item_Total
,"PRODUCT_ID" PRODUCT_ID_DISPLAY1
,replace(get_product_image(PRODUCT_ID), '.tif','.gif') image_name
from S_ITEM
where ORD_ID=:ORDERS_ID
```

➔ Apply your changes

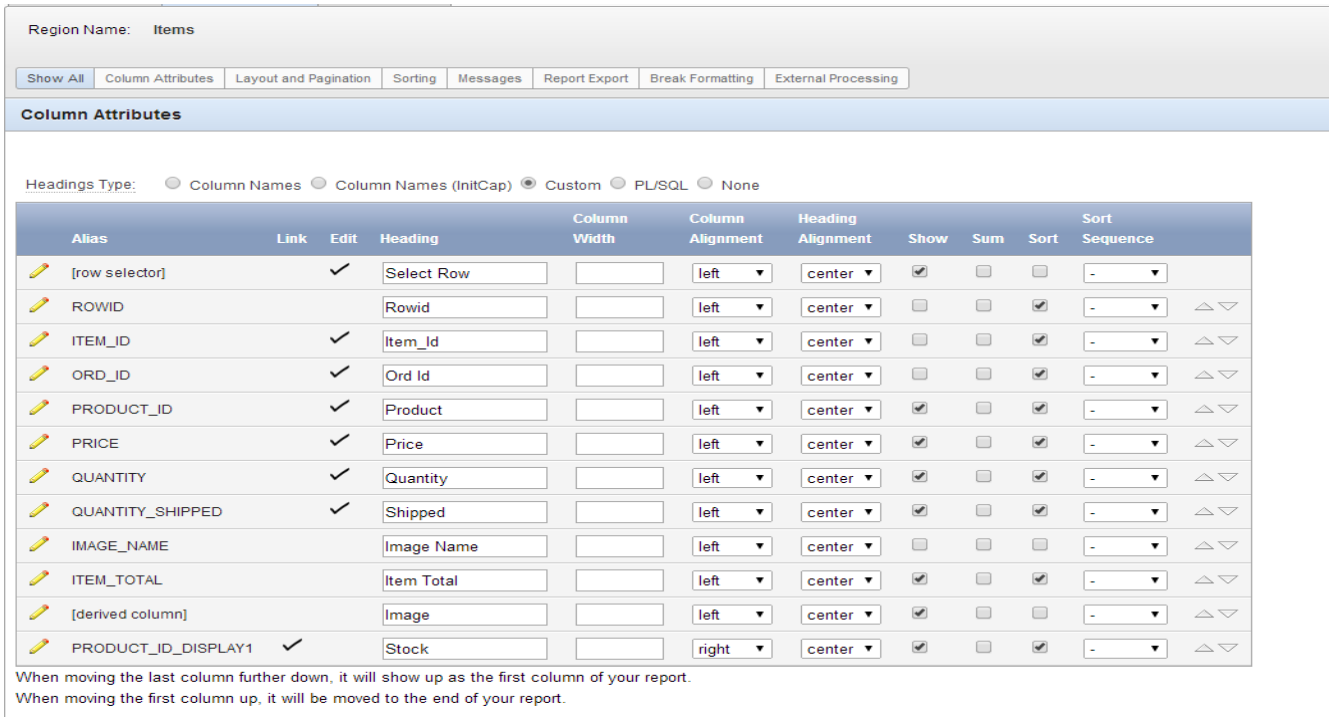
- Change to the “Report Attributes” tab.
  - Rearrange the columns (see screen shot).
  - Edit the headings (see screen shot).
  - In Column Image (Derived Column) set the display as Standard report column.
  - Enter the following text in Column Image “Column Formatting” -> HTML Expression:

```
<img style="border: 4px solid #CCC; -moz-border-radius:
4px; -webkit-border-radius: 4px;"
src="#WORKSPACE_IMAGES##IMAGE_NAME#" height="75"
width="75" alt="Product Image" title="Product Image">
```

  - Upload the images into APEX Application Builder -> Application 100 -> Shared Components -> Images (for being displayed in the report).

➔ If the images are not located in the environment of the server, or if a server (e.g. standalone) is not used this way, the best option is to upload the images in the so-called Shared Components section of your APEX Application. Then there is the choice of how to reference them, between Application Images and Workspace Images. If Workspace Images is chosen they are not related only to one specific application, so the images can be used (by referencing them) in every application of your workspace. In this Conversion of SUMMIT the Reference of Workspace Images is used.

- Go to Product_ID -> Set Display as “Select List (named LOV)” and click Apply Changes.



Adding the insert functionality for "Items" Region

First compile the following Sequence in SQL Developer (or SQL Plus, TOAD etc.):

```
CREATE SEQUENCE „SUMMIT"."S_ITEM_ID" MINVALUE 1 MAXVALUE 9999999
INCREMENT BY 1 START WITH 1 NOCACHE NOORDER NOCYCLE;
```

The next Step is to create this Trigger:

```
create or replace TRIGGER BI_S_ITEM
before insert on S_ITEM
for each row
begin
if :NEW.ITEM_ID is null then
select S_ITEM_ID.nextval into :NEW.ITEM_ID from dual;
end if;
end;
```

Add order sum

In the "Orders" region, add an item to this region to display the value of the order:

- Item: Display only
- Item Type: Display as Text (does not save state)
- Item Name: P2_ORDER_TOTAL
- Sequence: 115

- Region: Items
- Label: Order Total
- Label alignment: Left
- Label Template: No Template
- Item Source: SQL Query:

```
Select
to_char(sum(price*quantity_shipped),'999G999G999G999G990D00')
from s_item where ord_id=:P2_S_ORD_ID
```

Edit Control region

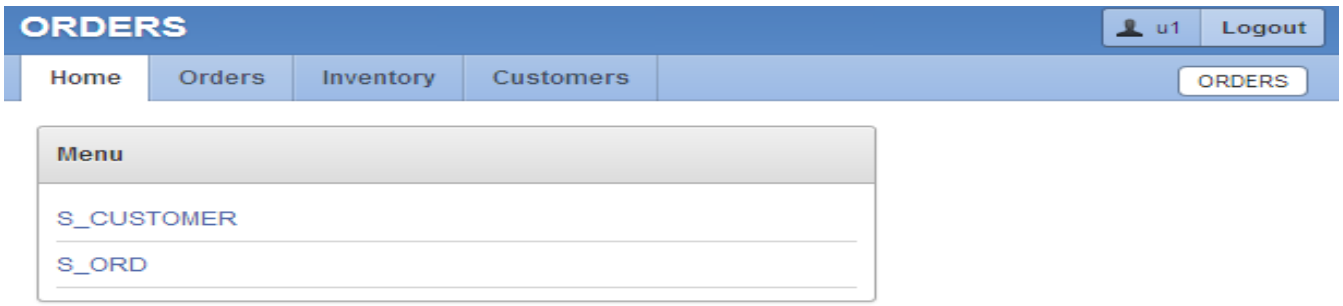
- Set the condition of the “Control” region to “Never”.

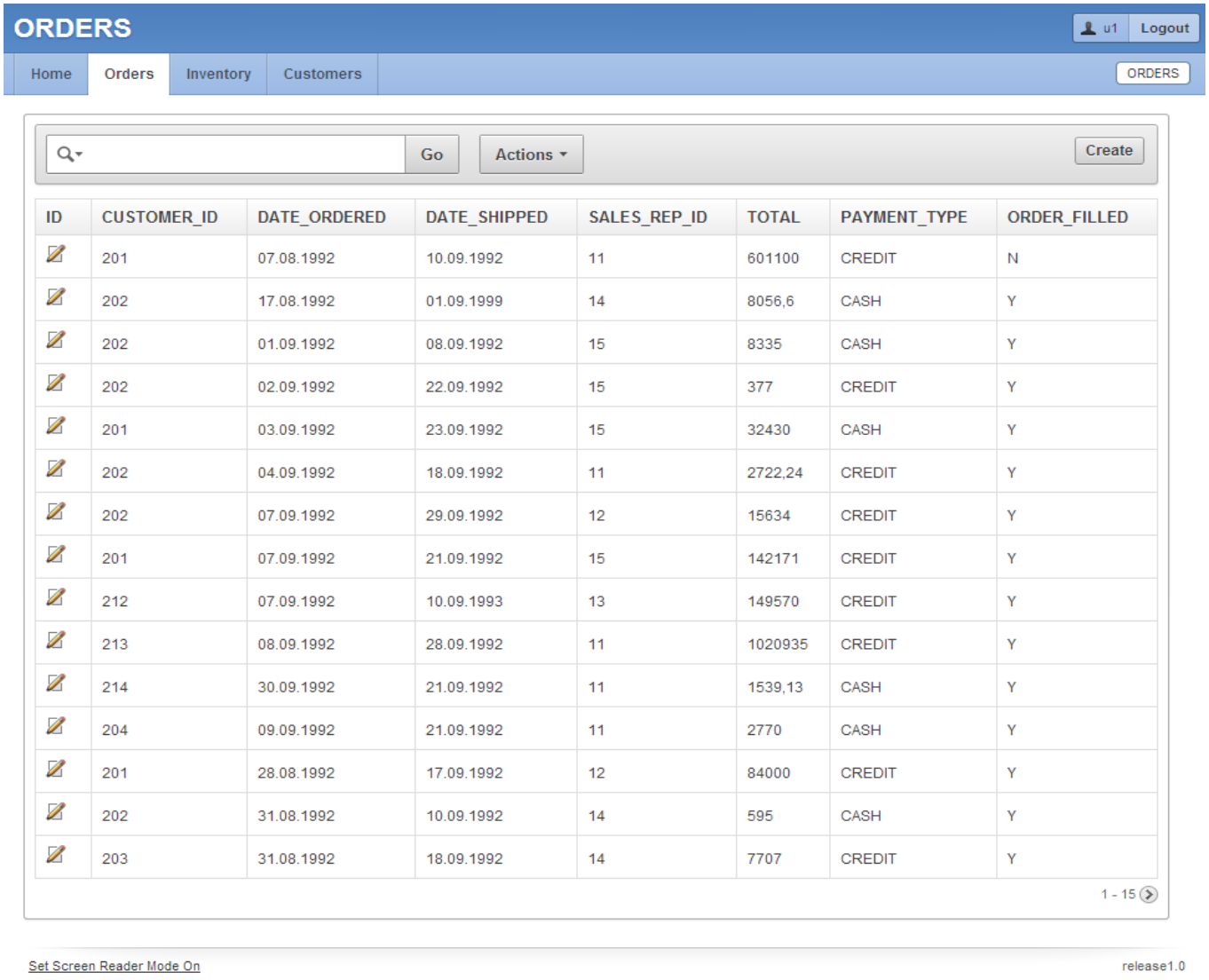
4.2 Edit Page Zero

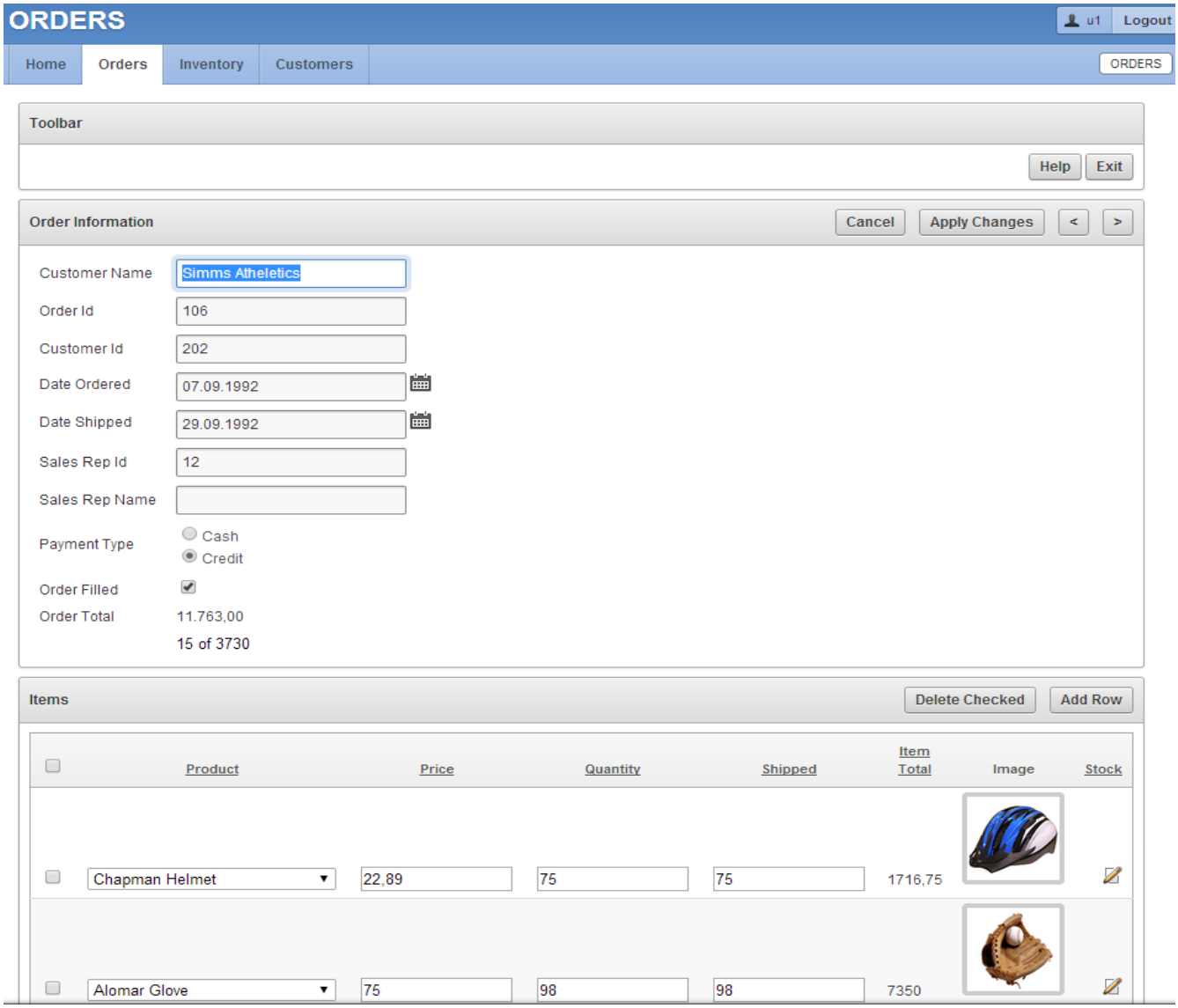
- Go to Page 0 (Page Zero/Global Page)
- Set the title to “Toolbar” and column: 1, sequence: 1, and change User Interface -> Template -> to “Reports Region”.
- Delete items P0_CONTROL_IMMEDIATE, P0_CONTROL_AUTO_QUERY.
- Set the conditions of IMAGE_BUTTON and STOCK_BUTTON to “Never”.
- Edit the EXIT button:
    a. Redirect to URL:

```
f?p=&APP_ID.:101
```

    b. Apply Changes.




ORDERS
u1
Logout
Home
Orders
Inventory
Customers
ORDERS
Go
Actions
Create
ID
CUSTOMER_ID
DATE_ORDERED
DATE_SHIPPED
SALES_REP_ID
TOTAL
PAYMENT_TYPE
ORDER_FILLED
201 07.08.1992 10.09.1992 11 601100 CREDIT N
202 17.08.1992 01.09.1999 14 8056,6 CASH Y
202 01.09.1992 08.09.1992 15 8335 CASH Y
202 02.09.1992 22.09.1992 15 377 CREDIT Y
201 03.09.1992 23.09.1992 15 32430 CASH Y
202 04.09.1992 18.09.1992 11 2722,24 CREDIT Y
202 07.09.1992 29.09.1992 12 15634 CREDIT Y
201 07.09.1992 21.09.1992 15 142171 CREDIT Y
212 07.09.1992 10.09.1993 13 149570 CREDIT Y
213 08.09.1992 28.09.1992 11 1020935 CREDIT Y
214 30.09.1992 21.09.1992 11 1539,13 CASH Y
204 09.09.1992 21.09.1992 11 2770 CASH Y
201 28.08.1992 17.09.1992 12 84000 CREDIT Y
202 31.08.1992 10.09.1992 14 595 CASH Y
203 31.08.1992 18.09.1992 14 7707 CREDIT Y
1 - 15
Set Screen Reader Mode On
release1.0



4.3 Edit the stock page

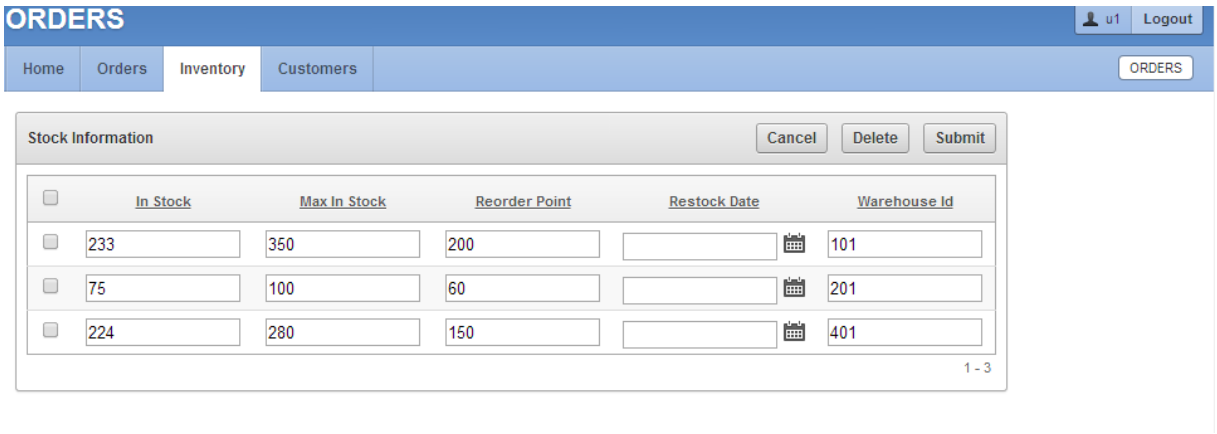
Go to the parent tabs and open the tab “Cv_Inventory”. Change the label to “Inventory”.

Go to the edit page and change the title to “Inventory”.

Change the title of region C_INVENTORY to “Stock Information” and change User Interface -> Template -> to “Reports Region”.

Editing the report region S_INVENTORY

Set the Add_Row button to “Never” (-> Conditions) because it won’t be used. The report region is displayed for updating and deleting but not for creating a new entry.



4.4 Edit the customers page

Go to the parent tabs and open the tab “Cv_customer”. Change the label to “Customers”.

Go to the edit page and change the title to “Customers”.

Change the title of region S_CUSTOMER to “Customer Information” and change User Interface -> Template -> to “Reports Region”.

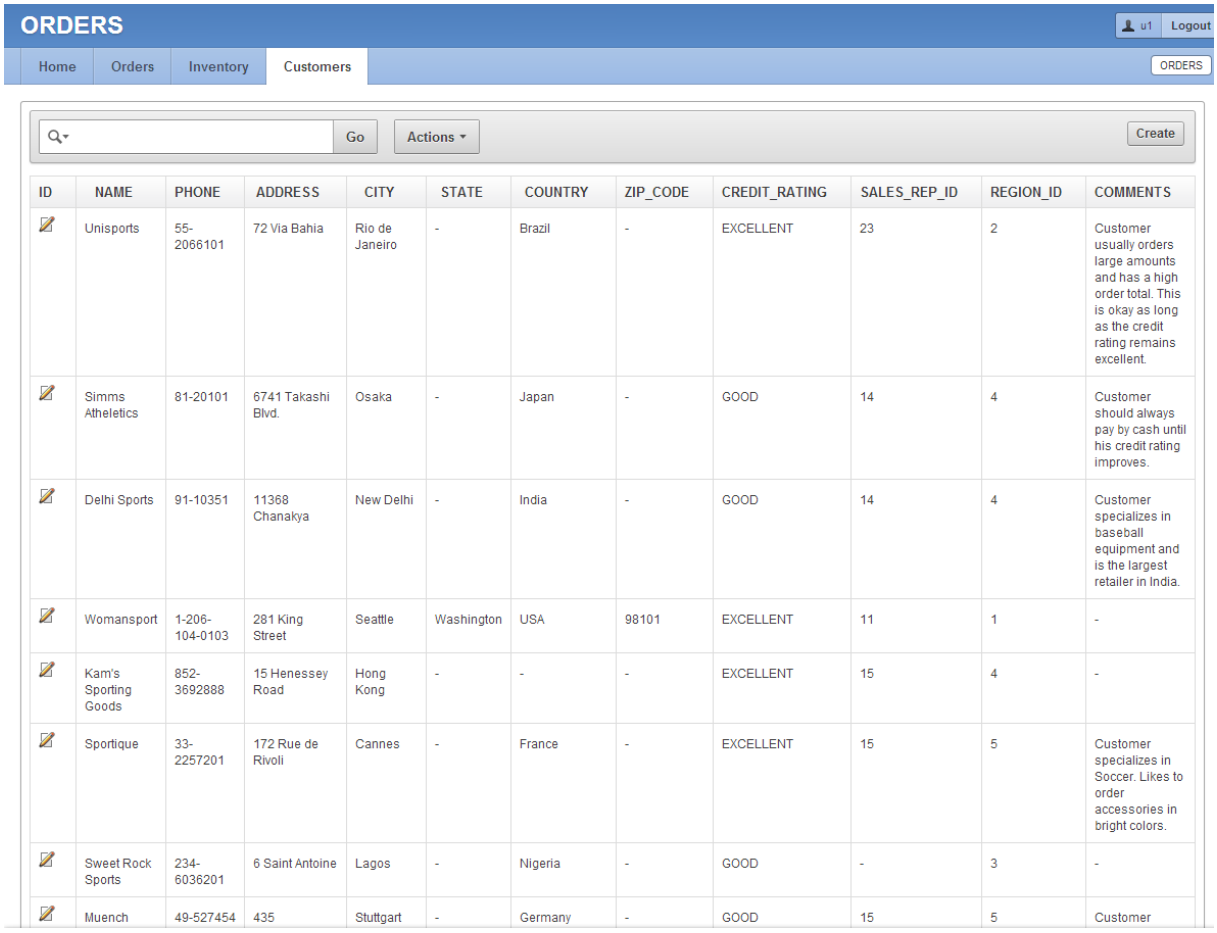
Add a process to the page:

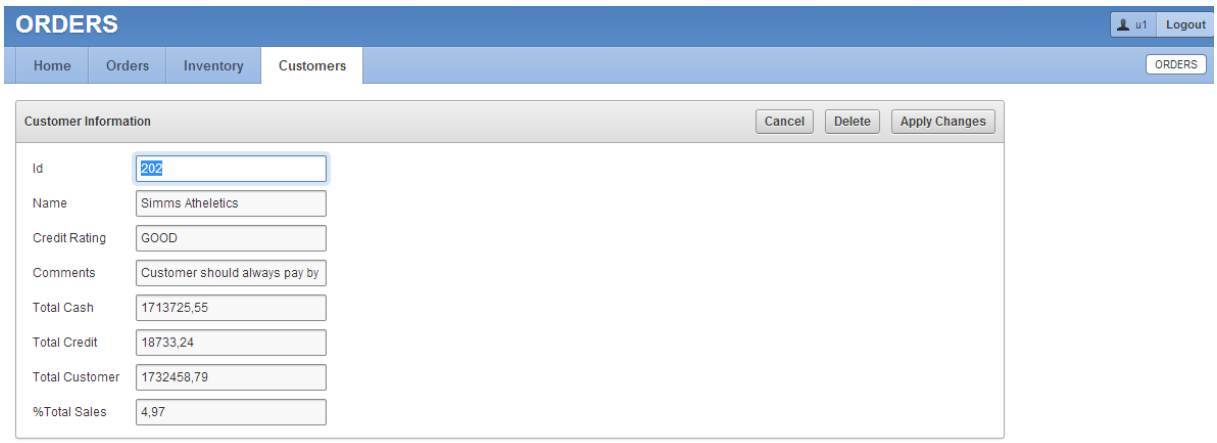
- Type: PL/SQL
- Name: set customer totals
- Point: On Load-After Header
- PL/SQL Process:

```
DECLARE
v_sum number;
BEGIN
select nvl(sum(total),0)
INTO :P6_S_CUSTOMER_TOTAL_CASH
from s_ord
where customer_id = :P6_S_CUSTOMER_ID
  and  payment_type = 'CASH';
select nvl(sum(total),0)
INTO :P6_S_CUSTOMER_TOTAL_CREDIT
from s_ord
where customer_id = :P6_S_CUSTOMER_ID
  and  payment_type = 'CREDIT';
:P6_S_CUSTOMER_TOTAL:= round(:P6_S_CUSTOMER_TOTAL_CREDIT +
:P6_S_CUSTOMER_TOTAL_CASH,2);
select sum(total)
INTO v_sum
from s_ord;
:P6_S_CUSTOMER_OF_TOTAL :=
round((:P6_S_CUSTOMER_TOTAL/v_sum)*100,2);
```

```
END;
```

Condition EXISTS: `select distinct 1 from s_ord where customer_id = :P6_S_CUSTOMER_ID`





5. Summary

This case study initially defines the audience for which this document was made for and which prerequisites have to be fulfilled to create a successful conversion of the "Summit" application from Oracle Forms to Oracle Apex. The installations of the Demo Schema and of Apex have been described as well as the necessary prerequisites for the Apex environment. After that the approach for the conversion is shown: Starting with the import of the FMBs-files, to the right handling of the PITSS.CON Apex Assistant through to the final export of the SQL-File, which then can be imported in Apex without problems. In a last step the case study shows how the few manual adjustments for converting the "Summit" Application completely and successfully are covered. Looking to the future it can be said that the PITSS.CON Apex Assistant becomes more efficient, the degree of automation for migrations further increases and one is well prepared for the release of Apex 5.0.

From Forms to APEX - Summit Case Study
May 2014
Authors: Markus Salm, Nihad Haskovic
Reviewer: Madi Serban

PITSS in Europe
Germany
+49-711-728.752.00
sales@pitss.com
www.pitss.com

PITSS in Americas
USA
+ 1 248.740.0935
pitssamerica@pitss.com
www.pitss.com