
C3. Creating APEX reports

In order to display data, APEX uses **REPORTS**. A report is a grid view of the fields needed to be displayed, each field using its own column. The fields can be selected from one table or from multiple, joined tables. A column can have **sorting** enabled, in which case the user can click on its header in order to change the order of the displayed fields. Any SQL SELECT statement can be used as a data source for a report, so the columns can also display computed data, such as **COUNT** or **MAX**, not only field values.

There are several forms of reports, but the most used ones are:

- **Interactive Report** – it provides many already implemented actions, such as searching, filtering, column breaks and it is quite flexible for many tasks, but it is harder to customize beyond what it already offers
- **Classic Report** – it is a lower level report than the “Interactive Report” and it requires more time to implement what an “Interactive Report” already provides, but it can be customized more, in order to suit more demanding reports

A report is created from the Application builder in the same way as any other page, using the **Create Page** button and selecting the **Report** group.



ACTIVITY 1: create an Interactive Report from the Departments table, showing the departments name and for

each department the number of persons assigned to it, following the steps below:

Select the Interactive Report and give it the name “Departments Report”

The screenshot shows the 'Page and Report Attributes' tab in the Oracle APEX interface. At the top, there are four tabs: 'Page and Report Attributes' (selected), 'Tab Options', 'SQL Query', and 'Buttons'. Below the tabs, there is a 'Cancel' button on the left and a 'Next >' button on the right. The main area contains a text input for 'Identify a page number and name. If this page number already exists, the wizard creates the page for you.' Below this, there are several fields: 'Application' is '11000 - Persons', 'Page Number' is 'd', 'Page Title' is 'Departments Report', 'Page Template' is 'Region with Headers and Tabs', 'Page Items' is 'Report 1', and 'Buttons' is 'do not use breadcrumbs on page'.

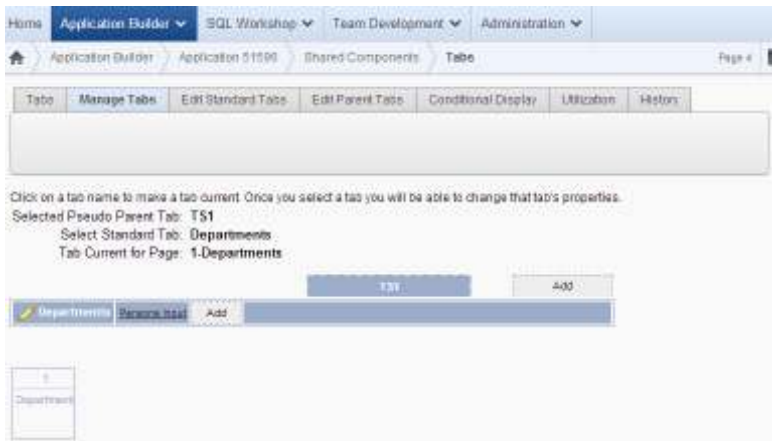
On the Tab Options page, leave the default settings.

On the SQL Query page, type the following SQL SELECT:

```
select DEPARTMENTS.NAME as NAME,  
       COUNT(PERSONS.SEX) AS NPERSONS  
from DEPARTMENTS, PERSONS  
where DEPARTMENTS.ID=PERSONS.DEP_ID  
group by DEPARTMENTS.NAME
```

To create simple queries, the **Query Builder** tool can be used. The Query Builder button opens a new page in which tables and columns together with their associated conditions can be selected. As our query contains computed fields and the **group by** clause, it is easier to be created by hand.

On the next pages use the default settings and finally generate the report page. As a final step, add it to the main navigation menu, using the Shared Components tab from the application main window and using the Navigation=>Tabs option and Manage Tabs tab.



When the Add button is clicked, wizard lets us chose the navigation title (Departments Report) and after that the page to be linked to this navigation button (Tab Current for Page). The other steps of the wizard can be left with their default values. If all the things went well, the resulted page should look like in the following image.



Here the Actions button is pressed and it can be seen that there are many predefined actions, already implemented for our report, alongside searching columns sorting.



ACTIVITY 2: create a Classic Report, showing for each department the persons assigned to it.

Select the Classic Report and give it the name “Persons Report”. Tab Options page remain the same. In the Report Source page we can use the Query Builder to generate our query.



The final query should look like in the following image.

Report Source SQL Query ▼

* Enter a SQL SELECT statement or a PL/SQL function returning a SQL SELECT statement

```
select DEPARTMENTS.NAME as DEPARTMENT,
PERSONS.NAME as NAME,
PERSONS.EMAIL as EMAIL,
PERSONS.NASTERE as NASTERE,
PERSONS.SEX as SEX
from PERSONS PERSONS,
DEPARTMENTS DEPARTMENTS
where PERSONS.DEP_ID =DEPARTMENTS.ID
```

The next pages of the wizard keep their default values. After the report is generated, it should be added to the main

navigation menu in the same way as the last report. Its name in the navigation menu will be “Persons Report”. If all went well, the application should look like in the following image.

Persons


Supermarkets Personnel Departments Report **Persons Report**

Report 1

DEPARTMENT	NAME	EMAIL	SALARY	SEX
Production	Joe	joe@gmail.com	12,000-13	M
Production	Serg	serg@gmail.com	17,000-18	M
Production	Alex	alex@gmail.com	16,000-14	F
Quality Control	Maria	maria@gmail.com	10,000-12	M

Summary

1 - 4

 **ACTIVITY 3:** Modify the Persons Report in order to remove the duplicate names of the departments.

It can be seen that a department can have multiple persons assigned to it.

If the report is ordered by the department name, it is redundant to write the department name for each person of the same department. It would be better if the department name is shown only when it is changed from the last person. In order to accomplish this, we open the report settings page.

In this page there are displayed all the page general options and its regions. The report is only one region on this page, so in order to access its properties we click on the “Report” link of the “Report 1” from the Regions tab. This opens the page with the report specific settings. The specific formatting we want to apply to our report is synonym with a **group by DEPARTMENT.NAME** clause. This splits the report on many sub-reports, in accord with the “group by” field. In the report options page, this formatting is named **Break Formatting** and it can be found on the Report Attributes tab. In this tab we set the first column as the break column. It can be seen that the break columns must be positioned at the beginning of the report.

The screenshot shows the 'Report Attributes' tab in a software application. The 'Report Name' is 'Report 1'. The 'Break Column' is set to 'First Column'. The 'Break Column Date' is set to 'Default Break Formatting'. The 'For report heading breaks use this format' field contains 'COLUMN_NAME'. The 'Display this field as report header using this format' field contains 'COLUMN_NAME'. The 'When displaying a break line, display this text before break column' field is empty. The 'When displaying a break line use this format' field contains 'COLUMN_NAME'. The 'When displaying a break line, display this text after all columns' field is empty. The 'Report Attributes' tab is selected, and the 'Apply Changes' button is visible.

The final report should look like in the following figure.

Persons Website Output

[Superfriends](#) [Persons Report](#) [Superfriends Report 1](#) **Persons Report**

Report 1

EMAIL (click)	NAME	EMAIL	DOB	AGE
Procter	Ken	ken@myk.com	12-04-75	M
Gogo	Gogo	gogo@mail.com	07-Jul-75	M
Alva	Alva	alva@mail.com	14-04-74	F
Quake Control	Marion	marion@mail.com	18-08-72	M

Download

1 / 4