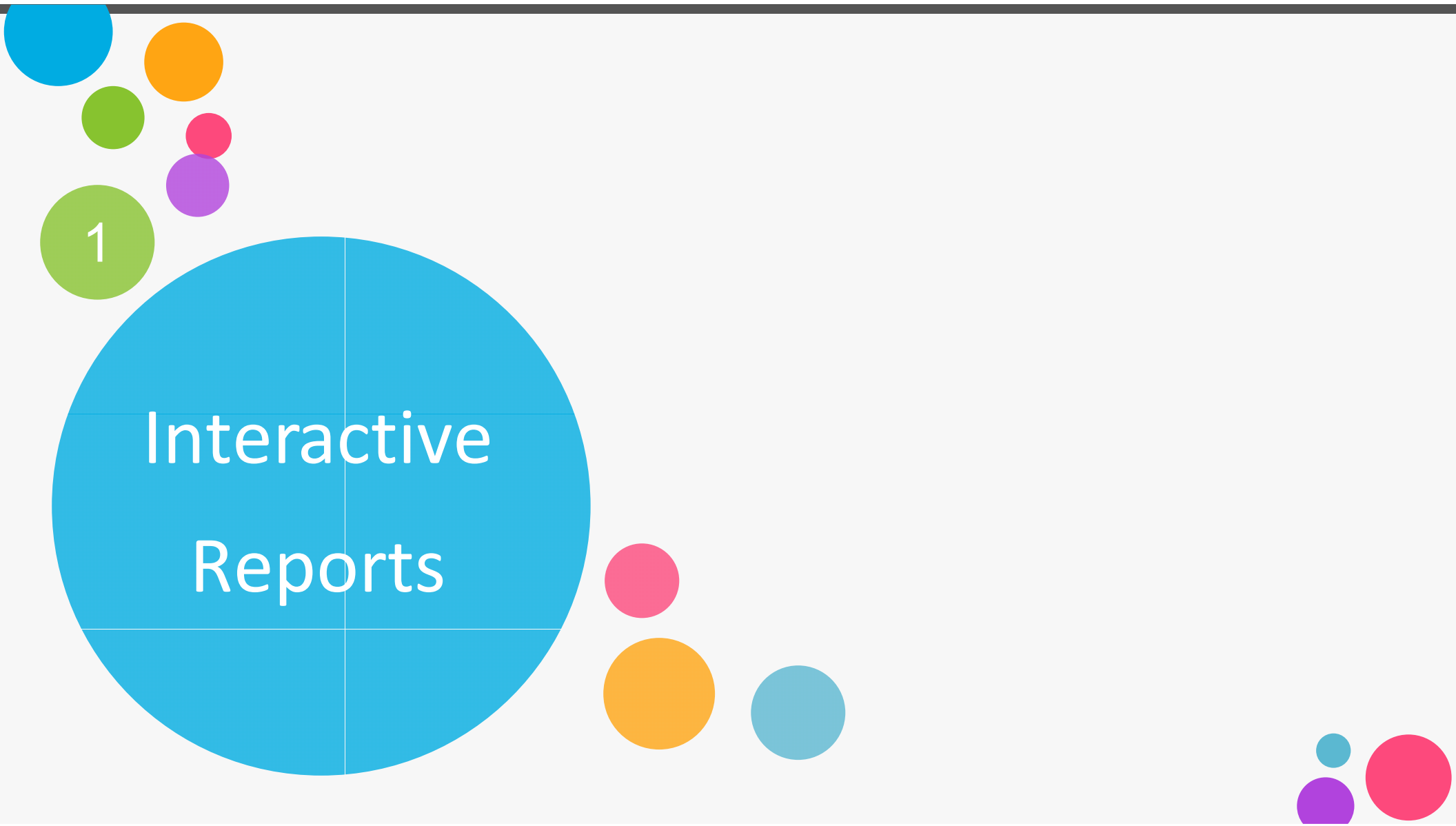
Creating readable outputs using sql plus


1
Interactive
Reports

Interactive Reports

Here we can create reports that prompt the user to supply their own values different times to a single same query.
To create interactive reports, we can Use ***substitution variables*** in a command file or in a single SQL statement.

2

Substitution variable

Substitution Variable

Substitution Variables

single-ampersand (&) substitution variables are use to temporarily store values. These variables can be used to make the query dynamic means single query can be use multiple times after substituting different *substitution variables*

Notation	Description
&user_variable	Indicates a variable in a SQL statement; if the variable does not exist, SQL*Plus prompts the user for a value (SQL*Plus discards a new variable once it is used)

Substitution Variable

SET VERIFY Command

To confirm the changes in the SQL statement we use the SQL*Plus ***SET VERIFY*** command.

Setting SET VERIFY ON forces SQL*Plus to display the text of a command before and after it replaces substitution variables with values.

By default it is **On** so to experience this we will ***Set Verify OFF***

Substitution Variable

Character & Date Values with Substitution Variables

In a WHERE clause, date and character values must be enclosed within single quotation marks. The same rule applies to the substitution variables.

We can also Use character functions with Substitution variable like UPPER('&job_title').

3

Customizing the SQL*Plus Environment

Customizing SQL* Plus Environment

SQL*Plus Environment can be controlled by using the SET commands which we have also discussed in earlier labs

SYNTAX

SET system_variable value

To show all Systems variable at any instant you can type ***SHOW ALL***
By default Echo is off *then now will* ***set Echo on*** then again we will Type ***SHOW ALL &*** see that among all Set varaibles Echo changed

Set Echo

Set Echo off : will execute the command of file without being displaying(echo) on the screen

Set Echo on : will execute the command of file with displaying(echo) on the screen

Customizing SQL* Plus Environment

Lets check some Command variables & their descriptions

Notation	Description

The value n above represents a numeric value.

The underlined value indicates default values. If we enter no values with the variable, SQL*Plus assumes the default value.

4

Saving Customization in SQL File

Saving Customizations in login.sql File

1. ***login.sql*** file contains standard ***SET*** and ***other SQL*Plus commands*** that may be required for every session.
2. SQL reads that file and implements the respective commands at logon.
3. When the session is logged out, all customized settings are lost.
4. We need to see how permanent changes are added to the login.sql file.
5. Default path of this file is at drive where you installed Oracle with a current user as **E:\app\ALI\product\11.2.0\dbhome_1\sqlplus\admin**

5

SQL Plus Format Commands

SQL* PLUS Format Commands

We can control the report features by using the following commands:-

S.NO	Command	Description

Lets study **The COLUMN Command** which controls display of a column.

Option	Description

SQL* PLUS Format Commands

To better learn formatting we will compare Original data with the formatted data

Original DATA:

```
SQL> select ename,sal,mgr from emp;

ENAME             SAL        MGR
---------- ---------- ----------
SMITH             800       7902
ALLEN            1600       7698
WARD             1250       7698
JONES            2975       7839
MARTIN           1250       7698
BLAKE            2850       7839
CLARK            2450       7839
SCOTT            3000       7566
KING             5000
TURNER           1500       7698
ADAMS            1100       7788

ENAME             SAL        MGR
---------- ---------- ----------
JAMES             950       7698
FORD             3000       7566
MILLER           1300       7782

14 rows selected.
```

Formatted Data:

```
SQL> COLUMN ENAME HEADING 'Employee|Name'Format A15
SQL> COLUMN SAL JUSTIFY LEFT FORMAT $99,990.00
SQL> COLUMN MGR FORMAT 9999999999 NULL 'No Manager'
SQL> select ename,sal,mgr from emp;

Employee
Name            SAL                  MGR
--------------- ---------- ----------
SMITH              $800.00         7902
ALLEN            $1,600.00         7698
WARD             $1,250.00         7698
JONES            $2,975.00         7839
MARTIN           $1,250.00         7698
BLAKE            $2,850.00         7839
CLARK            $2,450.00         7839
SCOTT            $3,000.00         7566
KING             $5,000.00 No Manager
TURNER           $1,500.00         7698

Employee
Name            SAL                  MGR
--------------- ---------- ----------
ADAMS            $1,100.00         7788
JAMES              $950.00         7698
FORD             $3,000.00         7566
MILLER           $1,300.00         7782

14 rows selected.
```

This is My Formatting

A15 means At a time 15 Characters in a Single line and rest will go in second line

Comma applied after 3 digits from Right

SQL* PLUS Format Commands

Display the current setting for the a column. We write:

COLUMN COLUMN_NAME

Display the current setting for Employee name column:

COLUMN ENAME

```
SQL> COLUMN ENAME
COLUMN   ENAME ON
HEADING  'Employee|Name' headsep '|'
FORMAT   A15
```

To clear settings for the ENAME column

COLUMN ENAME CLEAR

So if we will again type **COLUMN** ENAME then :

```
SQL> COLUMN ENAME
SP2-0046: COLUMN 'ENAME' not defined
```

SQL* PLUS Format Commands

NOTE: If the command is lengthy, we can continue on the next line by ending the current line with a hyphen (-).

```
SELECT * from -
Emp  where rownum <=4;
```

Row num brings Top Mentioned record from Result set

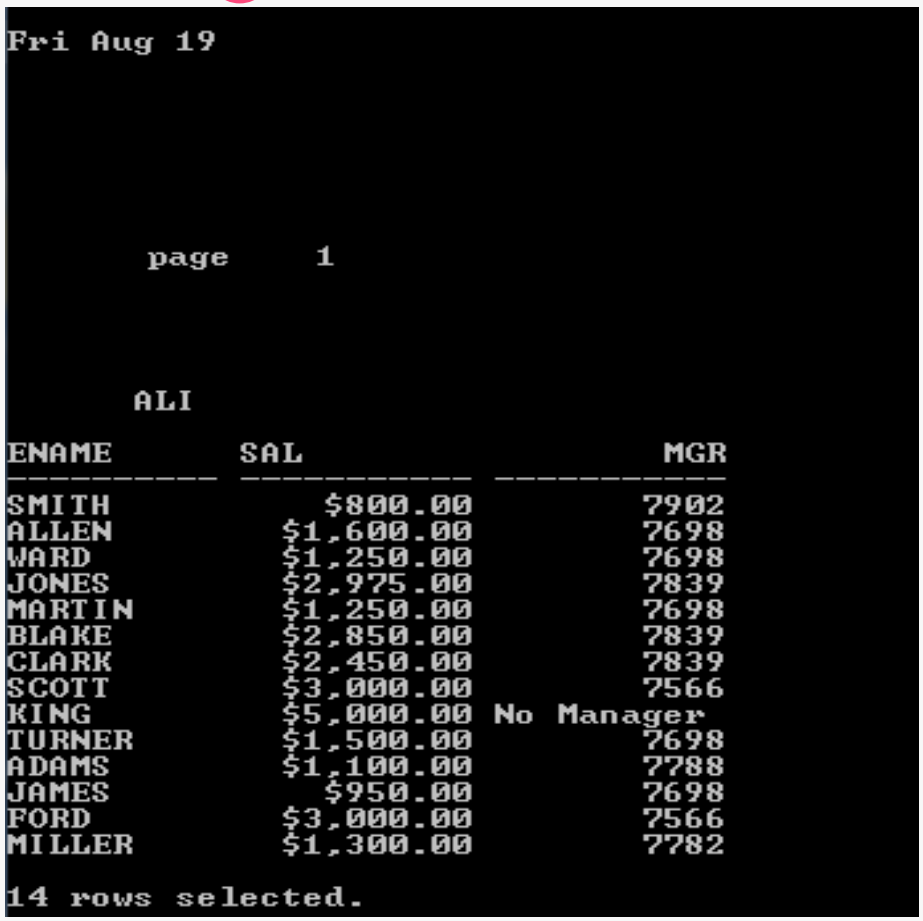
```
SQL> select * from -
> emp where rownum <=4;

     EMPNO ENAME      JOB              MGR HIREDATE  SAL          COMM     DEPTNO
---------- ---------- --------- ---------- --------- ---------- ---------- ----------
      7369 SMITH      CLERK           7902 17-DEC-80     $800.00                   20
      7499 ALLEN      SALESMAN        7698 20-FEB-81   $1,600.00        300       30
      7521 WARD       SALESMAN        7698 22-FEB-81   $1,250.00        500       30
      7566 JONES      MANAGER         7839 02-APR-81   $2,975.00                  20
```

Lets study **TTITLE Command** which format the page header

TITLE automatically puts date and page number on the report. If I write following cmd

TTITLE ALI

SQL* PLUS Format Commands



```
Fri Aug 19

                      page    1

                     ALI
ENAME      SAL          MGR
---------- ---------- ----------
SMITH         $800.00       7902
ALLEN       $1,600.00       7698
WARD        $1,250.00       7698
JONES       $2,975.00       7839
MARTIN      $1,250.00       7698
BLAKE       $2,850.00       7839
CLARK       $2,450.00       7839
SCOTT       $3,000.00       7566
KING        $5,000.00 No Manager
TURNER      $1,500.00       7698
ADAMS       $1,100.00       7788
JAMES         $950.00       7698
FORD        $3,000.00       7566
MILLER      $1,300.00       7782

14 rows selected.
```

SQL* PLUS Format Commands

Lets study **BTITLE Command** which create a footer text. Footers appear at the bottom of the page according to the PAGESIZE value. If I write :

BTITLE Confidential

```
Fri Aug 19

        page     1

              .

       ALI
ENAME       SAL                        MGR
---------- ---------- ----------
SMITH            $800.00                7902
ALLEN          $1,600.00                7698
WARD           $1,250.00                7698
JONES          $2,975.00                7839
MARTIN         $1,250.00                7698
BLAKE          $2,850.00                7839
CLARK          $2,450.00                7839
SCOTT          $3,000.00                7566
KING           $5,000.00 No Manager
TURNER         $1,500.00                7698
ADAMS          $1,100.00                7788
JAMES            $950.00                7698
FORD           $3,000.00                7566
MILLER         $1,300.00                7782

 confidential
14 rows selected.
```

This is page footer

SQL* PLUS Format Commands

Lets study **The Break Command** which section out rows and suppress duplicate values.

Section out rows means make a bunch(Merge) on a particular mentioned column. This bunch will easy to understand when we will apply order by clause in the data

To understand break concept just take a following simple query

Select job,deptno from emp

JOB	DEPTNO
CLERK	20
SALESMAN	30
SALESMAN	30
MANAGER	20
SALESMAN	30
MANAGER	30
MANAGER	10
ANALYST	20
PRESIDENT	10
SALESMAN	30
CLERK	20

JOB	DEPTNO
CLERK	30
ANALYST	20
CLERK	10

SQL* PLUS Format Commands

Now if I break on Job via following command

Break On Job SKIP 2 Skip 2 means ***skip 2 lines*** on job change

Impact of the above command would be:

Select job,deptno from emp

Lets **order by** the result for better merging

Select job,deptno from emp order by job

```
SQL> break on job skip 2
SQL> select job,deptno from emp;

JOB          DEPTNO
--------- ----------
CLERK            20

SALESMAN         30
                 30

MANAGER          20

SALESMAN         30
JOB          DEPTNO
--------- ----------

MANAGER          30
                 10

ANALYST          20

PRESIDENT        10

JOB          DEPTNO
--------- ----------
SALESMAN         30

CLERK            20
                 30

ANALYST          20

JOB          DEPTNO
--------- ----------
CLERK            10


14 rows selected.
```

```
SQL> select job,deptno from emp order by job;

JOB          DEPTNO
--------- ----------
ANALYST          20
                 20

CLERK            10
                 30
                 20
                 20

MANAGER          30
JOB          DEPTNO
--------- ----------
MANAGER          20
                 10

PRESIDENT        10

SALESMAN         30
                 30
                 30
                 30

14 rows selected.
```

SQL* PLUS Format Commands

We can clear all BREAK settings by using:

CLEAR BREAK

You will get a response from SQL that "***breaks Cleared***"

```
SQL> clear break
breaks cleared
```

6

Create script file to Run a report

Create script file to Run a report

1. Create the SQL SELECT statement at the SQL prompt.
2. Save the SELECT statement to a script file.
3. Edit the script file to enter the SQL*Plus commands.
4. Add the required formatting commands before the SELECT statement.
5. Verify that the SELECT statement is followed by a run character, either a semicolon (;) or a slash (/).
6. Clear formatting commands after the SELECT statement.
7. Save the script file.
8. Enter “START filename” to run the script.