Oracle Basic

- Install Oracle 11.2 and higher version only
  - Oracle is providing some default user i.e
    1. username : scott | password : tiger
    2. username : system | password : manager

C:\Users\class>sqlplus

SQL*Plus: Release 11.2.0.1.0 Production on Mon May 24 16:36:00 2021

Copyright (c) 1982, 2010, Oracle. All rights reserved.

Enter user-name: scott

Enter password: tiger

Connected to:

Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production

With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL>

SQL command must be end with ;

SQL * Plus command are optional to end with ;

SQL> show user;

USER is "SCOTT"

SQL>Cl SCR;

- to clear SQL Screen

Create Command :

- Create a table to store the data

Syn:

SQL >Create <table> <tablename>
(<column_name> <datatype>(size),.....,
 <column_n> <datatype>(size));

Eg:
SQL>Create table stu_pyt
(sno number(3),sname varchar(20),scity varchar(10));

SQL> select * from tab;
- It will display all list of table | objects are existed in the current user.

```
TNAME                          TABTYPE
------------------------------ -------
BONUS                          TABLE
DEPT                           TABLE
EMP                            TABLE
EMPLOYEE                       TABLE
SALGRADE                       TABLE
SSINFO                         TABLE
STUDENT                        TABLE
STU_PYT                        TABLE

8 rows selected.
```

DESC[RIBE]

Syn: SQL>DESC[RIBE] <tablename>;
SQL> desc Stu_pyt;

```
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------
 SNO                                                NUMBER(3)
 SNAME                                              VARCHAR2(10)
 SCITY                                              VARCHAR2(10)
```

ALTER

- It is used to make the changes in the table structure.
- Adding the new column(s)
- Change the name of the column
- We Delete column(s)
- We change the datatype of the column.

Syn:

SQL>ALTER <table> <Tablename>

ADD|MODIFY|DROP ----> [Clauses]

(<column> <datatype>(size),.....)

Adding New Column To the Table:

==========

```
SQL> ALTER TABLE STU_PYT
  2  ADD
  3  (FNAME VARCHAR(10),PIN NUMBER(6));

Table altered.

SQL> DESC STU_PYT;
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------
```

SNO	NUMBER(3)
SNAME	VARCHAR2(10)
SCITY	VARCHAR2(10)
FNAME	VARCHAR2(10)
PIN	NUMBER(6)

To Change the Name of the column:

==========

Syn:

SQL>ALTER <table> <tablename>

RENAME COLUMN

Old_col_Name to New_Col_Name;

Eg:

```
SQL> ALTER TABLE STU_PYT
  2  RENAME COLUMN
  3  SNAME TO STU_NAME;
```

```
SQL> DESC STU_PYT;
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------
 SNO                                                NUMBER(3)
 STU_NAME                                           VARCHAR2(10)
 SCITY                                              VARCHAR2(10)
 FNAME                                              VARCHAR2(10)
 PIN                                                NUMBER(6)
```

To Delete a column:

==========

Syn:

SQL>ALTER <TABLE> <tablename>
DROP <column> (<Columnname>);

```
SQL> alter table stu_pyt
  2  drop column  sno;

Table altered.

SQL> desc stu_pyt;
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------

 STU_NAME                                           VARCHAR2(10)
 SCITY                                              VARCHAR2(10)
 FNAME                                              VARCHAR2(10)
 PIN                                                NUMBER(6)
```

Deleting More than one column:

==========

SQL>ALTER <TABLE> <TABLENAME>
DROP
(<column1>,<column2>,.....);

```
SQL> ALTER TABLE STU_PYT
  2  DROP
  3  (SCITY,PIN);

Table altered.

SQL> DESC STU_PYT;
```

Name

STU_NAME

FNAME

3.DROP command:

==========

- Used to delete unwanted table or database Object.

Syn:

SQL>DROP <table> <Tablename>;

SQL>DROP table stu_pyt;

SQL> drop table stu_pyt;

Table dropped.

SQL> SELECT * FROM TAB;

TNAME	TABTYPE
BIN$0WByLhuARxCia3ZiCosUkw==$0	TABLE
BONUS	TABLE
DEPT	TABLE
EMP	TABLE
EMPLOYEE	TABLE
SALGRADE	TABLE
SSINFO	TABLE
STUDENT	TABLE

CREATE | ALTER | DROP Command

4.INSERT :

- Used to insert the data into the particular column(s)

Inserting Data into particular Column(s):

==========

SYN:

```
SQL>INSERT INTO <TABLENAME>
        (<COLUMN1>,<Column2>,.....,<Column n>)
          VALUES
          (<value1>,<value2>,.....,<value n>);
```

* Note: While inserting the data into char | varchar data type and date data type then that values must be Enclosed in ' '

```
SQL> INSERT INTO STU_PYT
  2  (SNO,SCITY)
  3  VALUES
  4  (101,'KMM');
1 row created.
```

Inserting Data into All Column(s):

==========

```
SQL> INSERT INTO STU_PYT
  2  VALUES
  3  (102,'JAMES','KMM');

1 row created.
```

Note: While inserting data into all the columns then

doesn't required to specify the column names rather
values order should be same as order of the columns existed
in the table.

CREATE
INSERT
ALTER
DROP

SELECT Command :
==========

- Used to read the data from the table

Syn:

```
SQL>SELECT <column(s)>/[*]
            [DISTINCT   <COLUMNAME>]
            FROM        <tablename>
                [WHERE <condition>]
                [ORDER By <column> [DESC] ];
```

```
SQL> SELECT SNO,SNAME
  2  FROM STU_PYT;

       SNO SNAME
---------- ----------
       101
       102 JAMES
       103 RAMESH
```

SQL> SELECT SNAME,SNO,SCITY

```
  2  FROM STU_PYT;

SNAME                SNO SCITY
---------- ---------- ------
                    101 KMM
JAMES               102 KMM
RAMESH               103 HYD
```

Reading Data From All the column(s)

==========

```
SQL> SELECT *
  2  FROM STU_PYT;

       SNO SNAME      SCITY
---------- ---------- ------
       101            KMM
       102 JAMES      KMM
       103 RAMESH     HYD
```

Note:

Scott is predefined and it is one of default user.

Scott is providing Some predefined Tables in it.

Such As : Emp ,

Dept , Bonus , Salgrade.

Note: The default width for numerical colums in oracle database 10 digit

```
SQL>Show numwidth;
SQL>set numwidth 5;

SQL> SELECT * FROM EMP;

EMPNO ENAME      JOB              MGR HIREDATE       SAL    COMM
----- ---------- --------- ----- --------- ----- -----
 7369 SMITH      CLERK          7902 17-DEC-80    4133
 7499 ALLEN      SALESMAN       7698 20-FEB-81    4933     300
 7521 WARD       SALESMAN       7698 22-FEB-81    4583     500
 7654 MARTIN     SALESMAN       7698 28-SEP-81    4583    1400
 7788 SCOTT      ANALYST        7566 19-APR-87    6000
 7839 KING       PRESIDENT           17-NOV-81    5000
 7844 TURNER     SALESMAN       7698 08-SEP-81    4833       0
 7876 ADAMS      CLERK          7788 23-MAY-87    4433
 7900 JAMES      CLERK          7698 03-DEC-81    4283
 7902 FORD       ANALYST        7566 03-DEC-81    6000
 7934 MILLER     CLERK          7782 23-JAN-82    4633


SQL> SELECT EMPNO,ENAME,JOB,
  2  HIREDATE
  3  FROM EMP;

EMPNO ENAME      JOB          HIREDATE
----- ---------- --------- ---------
 7369 SMITH      CLERK        17-DEC-80
 7499 ALLEN      SALESMAN     20-FEB-81
 7521 WARD       SALESMAN     22-FEB-81
 7654 MARTIN     SALESMAN     28-SEP-81
```

7788 SCOTT ANALYST 19-APR-87
7839 KING PRESIDENT 17-NOV-81
7844 TURNER SALESMAN 08-SEP-81
7876 ADAMS CLERK 23-MAY-87
7900 JAMES CLERK 03-DEC-81
7902 FORD ANALYST 03-DEC-81
7934 MILLER CLERK 23-JAN-82

SQL> SELECT * FROM DEPT;

DEPTNO DNAME LOC
------ -------------- ---------
10 ACCOUNTING NEW YORK
20 RESEARCH DALLAS
30 SALES CHICAGO

DISTINCT CLAUSE :

- It is the first clause of the select statement
- used to eliminate duplicate values from the selected column

Syn:

SQL>SELECT [DISTNICT] <columnname>
FROM <tablename>;

SQL> select JOB FROM EMP;

JOB

CLERK

SALESMAN
SALESMAN
SALESMAN
ANALYST
PRESIDENT
SALESMAN
CLERK
CLERK
ANALYST
CLERK

11 rows selected.

SQL> SELECT DISTINCT JOB FROM EMP;

JOB

CLERK
SALESMAN
PRESIDENT
ANALYST

SQL> SELECT DISTINCT DEPTNO FROM EMP;
DEPTNO

30
20
10

SORTING THE RECORDS WHILE READING DATA FROM TABLE:

==========

ORDER BY CLAUSE

Syn:

```
SQL>SELECT [DISTINCT] <COLUMNNAME>
            FROM <TABLENAME>
                ORDER BY <COLUMNNAME> [DESC];
```

```
SQL> SELECT ENAME
  2  FROM EMP
  3  ORDER BY ENAME;

ENAME
----------
ADAMS
ALLEN
FORD
JAMES
KING
MARTIN
MILLER
SCOTT
SMITH
TURNER
WARD

11 rows selected.
```

```
SQL> SELECT ENAME FROM EMP
  2  ORDER BY ENAME DESC;
```

ENAME

WARD

TURNER

SMITH

SCOTT

MILLER

MARTIN

KING

JAMES

FORD

ALLEN

ADAMS

Example:

```
SQL> select ENAME,JOB,SAL
  2  FROM EMP
  3  ORDER BY SAL DESC;
```

ENAME	JOB	SAL
FORD	ANALYST	6000
SCOTT	ANALYST	6000
KING	PRESIDENT	5000
ALLEN	SALESMAN	4933
TURNER	SALESMAN	4833
MILLER	CLERK	4633
MARTIN	SALESMAN	4583
WARD	SALESMAN	4583
ADAMS	CLERK	4433

JAMES CLERK 4283
SMITH CLERK 4133

Example:

```
SQL> SELECT * FROM EMP
  2  ORDER BY HIREDATE;

EMPNO ENAME      JOB              MGR HIREDATE
----- ---------- --------- ----- ---------
 7369 SMITH      CLERK          7902 17-DEC-80
 7499 ALLEN      SALESMAN       7698 20-FEB-81
 7521 WARD       SALESMAN       7698 22-FEB-81
 7844 TURNER     SALESMAN       7698 08-SEP-81
 7654 MARTIN     SALESMAN       7698 28-SEP-81
 7839 KING       PRESIDENT           17-NOV-81
 7900 JAMES      CLERK          7698 03-DEC-81
 7902 FORD       ANALYST        7566 03-DEC-81
 7934 MILLER     CLERK          7782 23-JAN-82
 7788 SCOTT      ANALYST        7566 19-APR-87
 7876 ADAMS      CLERK          7788 23-MAY-87
```

WHERE CLAUSE:

==========

- If u want read the data from the table based some condition then we have to use where clause

SYN:

```
SQL>SELECT [DISTINCT] <Columnname>
                FROM <TABLENAME>
                      [WHERE <CONDITION>]
```

[ORDER BY <COLUMN> [DESC]];

```
SQL> SELECT EMPNO,ENAME,JOB
  2  FROM EMP
  3  WHERE ENAME='SMITH';

EMPNO ENAME      JOB
----- ---------- ---------
 7369 SMITH      CLERK
```

Example:

```
SQL> SELECT * FROM EMP
  2  WHERE ENAME='SMITH';

EMPNO ENAME      JOB       MGR HIREDATE       SAL    COMM DEPTNO
----- ---------- --------- ---- ------------ ----- ----- ------
 7369 SMITH      CLERK     7902 17-DEC-80    4133                20
```

Example:

```
SQL> SELECT EMPNO,ENAME,JOB,
  2  SAL,HIREDATE
  3  FROM EMP
  4  WHERE JOB='SALESMAN';

EMPNO ENAME      JOB              SAL HIREDATE
----- ---------- --------- ----- ---------
 7499 ALLEN      SALESMAN     4933 20-FEB-81
 7521 WARD       SALESMAN     4583 22-FEB-81
 7654 MARTIN     SALESMAN     4583 28-SEP-81
```

7844 TURNER SALESMAN 4833 08-SEP-81

Example:

```
SQL> SELECT * FROM EMP
  2  WHERE HIREDATE<='20-FEB-81'
  3  ORDER BY HIREDATE;

EMPNO ENAME      JOB              MGR HIREDATE
----- ---------- --------- ----- ---------
 7369 SMITH      CLERK         7902 17-DEC-80
 7499 ALLEN      SALESMAN      7698 20-FEB-81
```

Example :

```
SQL> SELECT * FROM EMP
  2  WHERE SAL>=4500;

EMPNO ENAME      JOB              MGR HIREDATE       SAL
----- ---------- --------- ----- --------- -----
 7499 ALLEN      SALESMAN      7698 20-FEB-81    4933
 7521 WARD       SALESMAN      7698 22-FEB-81    4583
 7654 MARTIN     SALESMAN      7698 28-SEP-81    4583
 7788 SCOTT      ANALYST       7566 19-APR-87    6000
 7839 KING       PRESIDENT          17-NOV-81    5000
 7844 TURNER     SALESMAN      7698 08-SEP-81    4833
 7902 FORD       ANALYST       7566 03-DEC-81    6000
 7934 MILLER     CLERK         7782 23-JAN-82    4633
```

More than One condition using logical Operators :

and | or

==========

AND

```
SQL> SELECT *
  2  FROM EMP
  3  WHERE JOB='SALESMAN' AND
  4  SAL>=4600;

EMPNO ENAME      JOB              MGR HIREDATE       SAL
----- ---------- --------- ----- --------- -----
 7499 ALLEN      SALESMAN     7698 20-FEB-81    4933
 7844 TURNER     SALESMAN     7698 08-SEP-81    4833
```

or

```
SQL> SELECT * FROM EMP
  2  WHERE JOB='CLERK' OR
  3        SAL<=4800;

EMPNO ENAME      JOB              MGR HIREDATE
----- ---------- --------- ----- ---------
 7369 SMITH      CLERK         7902 17-DEC-80
 7521 WARD       SALESMAN      7698 22-FEB-81
 7654 MARTIN     SALESMAN      7698 28-SEP-81
 7876 ADAMS      CLERK         7788 23-MAY-87
 7900 JAMES      CLERK         7698 03-DEC-81
 7934 MILLER     CLERK         7782 23-JAN-82
```

UPDATE:

==========

- used to make the changes in the data which is already

existed in the table.

Syn:

```
SQL>UPDATE <TABLENAME>
SET <COLUMN>=<VALUE>,
        <COLUMN>=<VALUE>
            [WHERE <CONDITION>];
```

```
SQL> UPDATE EMP
  2  SET COMM=5000;

11 rows updated.

SQL> SELECT * FROM EMP;

EMPNO ENAME      JOB              MGR HIREDATE       SAL    COMM
----- ---------- --------- ---------- ----------- ----- -----
 7369 SMITH      CLERK           7902 17-DEC-80    4133    5000
 7499 ALLEN      SALESMAN        7698 20-FEB-81    4933    5000
 7521 WARD       SALESMAN        7698 22-FEB-81    4583    5000
 7654 MARTIN     SALESMAN        7698 28-SEP-81    4583    5000
 7788 SCOTT      ANALYST         7566 19-APR-87    6000    5000
 7839 KING       PRESIDENT            17-NOV-81    5000    5000
 7844 TURNER     SALESMAN        7698 08-SEP-81    4833    5000
 7876 ADAMS      CLERK           7788 23-MAY-87    4433    5000
 7900 JAMES      CLERK           7698 03-DEC-81    4283    5000
 7902 FORD       ANALYST         7566 03-DEC-81    6000    5000
 7934 MILLER     CLERK           7782 23-JAN-82    4633    5000
```

```
SQL> UPDATE EMP
  2  SET SAL=SAL+(SAL*5)/100,
  3        COMM=COMM+(COMM*3)/100;

SQL> UPDATE EMP
  2  SET SAL=SAL+(SAL*10)/100
  3  WHERE JOB='SALESMAN';
```

ROLLBACK:

==========

- It is used to cancel the last transaction

  Syn: SQL> Rollback;

- Rollback will work on [Delete | Insert | update]

```
SQL>rollback;
```

DELETE:

=======

- IT Is Used to delete a record | group of records | all records from the table.

  SYN:

  SQL>DELETE <FROM> <TABLENAME>
  [WHERE <CONDITION>];

```
SQL>DELETE FROM EMP
WHERE ENAME='SMITH';
```

```
SQL>DELETE FROM EMP
WHERE JOB='SALESMAN';

SQL>DELETE FROM EMP;

SQL> ROLLBACK;
```

COMMIT:

=======

- IT is used to make the transaction to save.
- Once transaction is commited rollback doesn't works on it.

```
SQL>Commit;
```