LEARNING OUTCOME

By the end of the lesson the student will be able to learn about :

The basic SQL-DML commands.
Insert rows into tables.
Update rows in tables.
Delete rows from tables.

DEMO

1. Download the sql script **DreamHome Script Oracle Apex** file or click **HERE**. **Save** the downloaded document as**DreamHomeScript.txt** on your computer.

2. Go to Application Express - Sign In (oracle.com) or click HERE . APEX sign-in page will be displayed.

3. After successful login, click SQL Workshop  .

4. Click SQL Scripts.

5. Click the Upload button as shown in Figure 1.

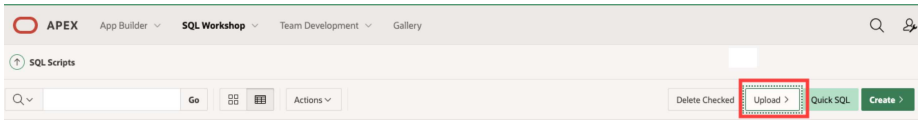


Figure 1.

6. Click **Choose File** and find a script at step 1 above as shown in **Figure 2** below.

Figure 2

7. Click the Upload button.

8. Click Run Button for the script DreamHome Oracle Apex.txt as shown in **Figure 3** below.

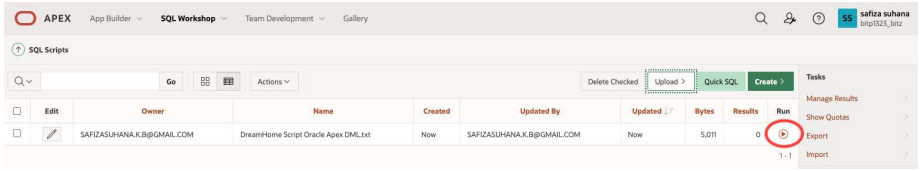


Figure 3

9. Figure 4 will be displayed. Click the Run Now button.

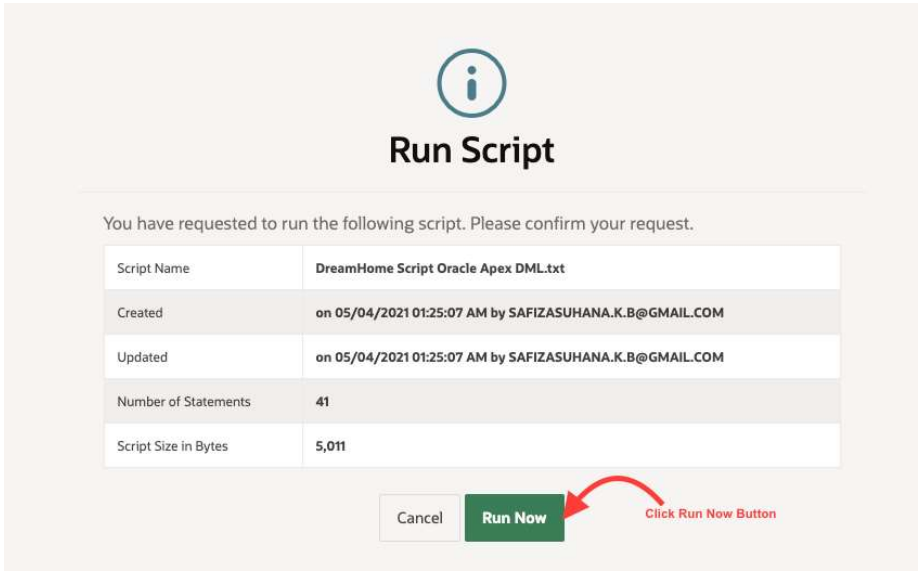


Figure 4

10. Figure 5 will be displayed to show that there are seven (7) tables have been created. Which are BRANCH, STAFF, PRIVATEOWNER, PROPERTYFORRENT, CLIENT, VIEWING, and REGISTRATION tables (red line) (**Figure 5**) as well as several records inserted for each table (green line) **(Figure 6).**

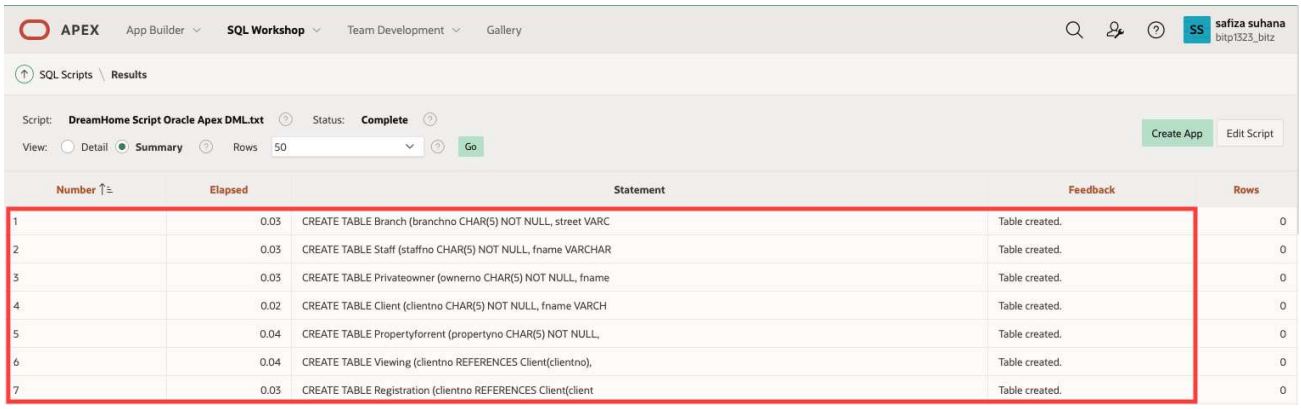


Figure 5

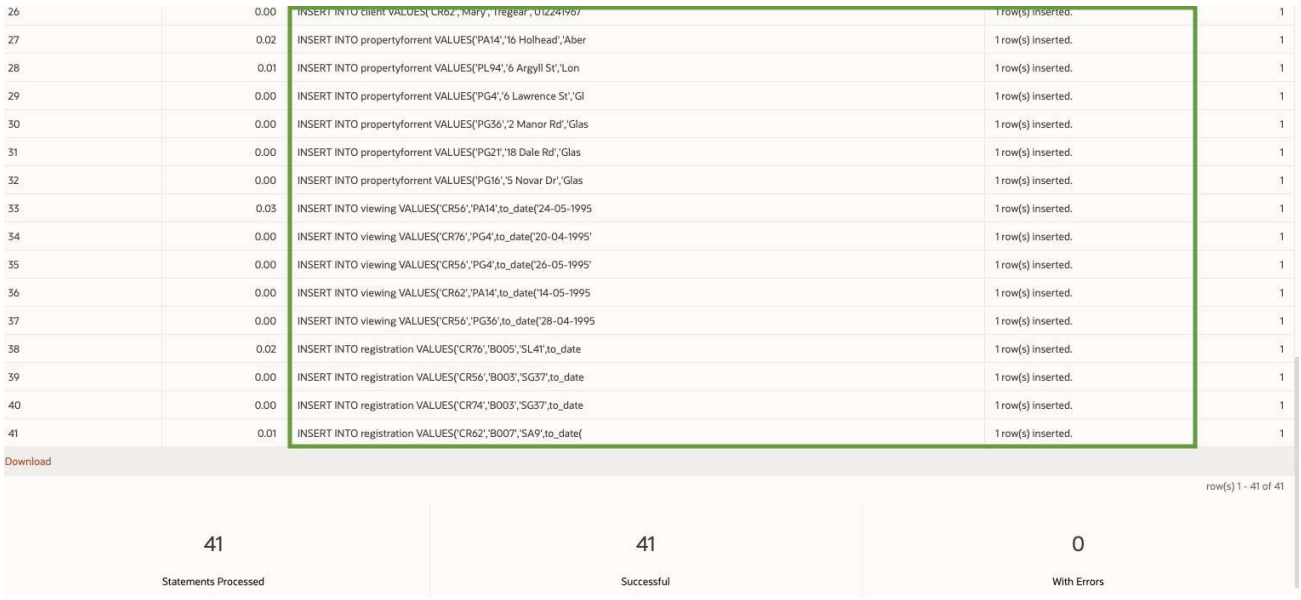
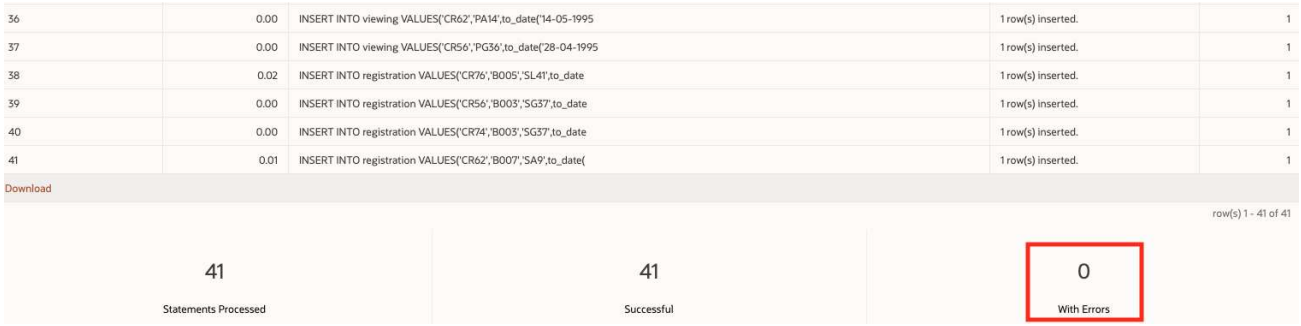


Figure 6

11. Check if there are errors from your code as shown in **Figure 7.**



36	0.00	INSERT INTO viewing VALUES('CR62','PA14',to_date('14-05-1995	1 row(s) inserted.	1
37	0.00	INSERT INTO viewing VALUES('CR56','PG36',to_date('28-04-1995	1 row(s) inserted.	1
38	0.02	INSERT INTO registration VALUES('CR76','B005','SL41',to_date	1 row(s) inserted.	1
39	0.00	INSERT INTO registration VALUES('CR56','B003','SG37',to_date	1 row(s) inserted.	1
40	0.00	INSERT INTO registration VALUES('CR74','B003','SG37',to_date	1 row(s) inserted.	1
41	0.01	INSERT INTO registration VALUES('CR62','B007','SA9',to_date(	1 row(s) inserted.	1

Figure 7

12. If you got 0 errors, means you have created your tables successfully and insert related data into the tables. Congratulations!! Well Done!

VIEW DATA

1. There is the syntax of queries if we want to access the data inserted to the tables which are:

SYNTAX

SELECT * FROM <tablename>

2. Now, confirm your data insertion to the tables by issuing the following queries one by one by clicking the SQL Workshop button as shown below :

Figure 8

3. Click the SQL Commands button. Write the SQL statement below.

SELECT * FROM BRANCH

4. Click Run button.

What did you get from this queries? (Hint: The data of BRANCH table will be displayed).

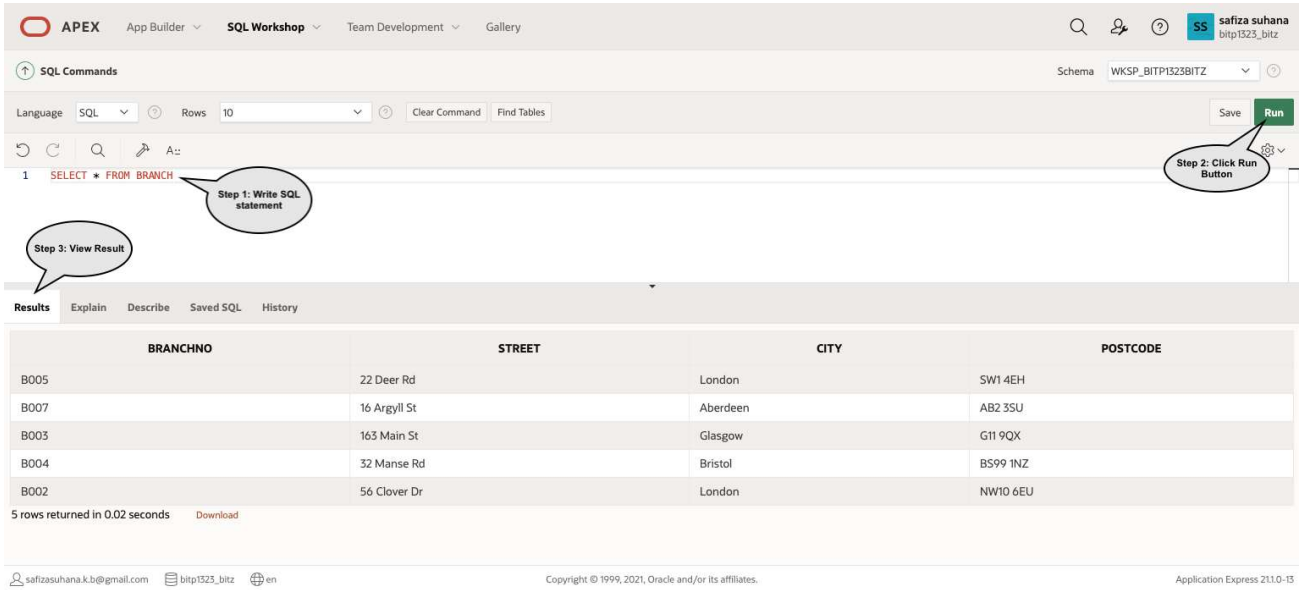


Figure 9

5. Continue to view the data from the rest of the tables: STAFF, PRIVATEOWNER, PROPERTYFORRENT, CLIENT, VIEWING, REGISTRATION;

6. Congratulations! You have 7 tables with their own data.

INSERT DATA

This section will show you how to insert new data to the table. Suppose that you want to insert the data as specified below:

TABLE: BRANCH

BRANCHNO	B006
STREET	Anna Beijerstraat
CITY	Glasgow
POSTCODE	G129QW

1. We can insert new data above using SQL statement by following below syntax. There are 2 scenarios that you can considered.

Scenario 1: Insert all data into tables

INSERT INTO <table> VALUES (<data1, data2, ...))

This example shows how to insert all data into BRANCH table:

1. In **SQL Commands** write SQL statement below:

INSERT INTO Branch VALUES ('B006', 'Anna Beijerstraat', 'Glasgow', 'G129QW');

2. Click **Run** button.
3. The message *1 row(s) inserted* will be displayed once data has been inserted to the table successfully.

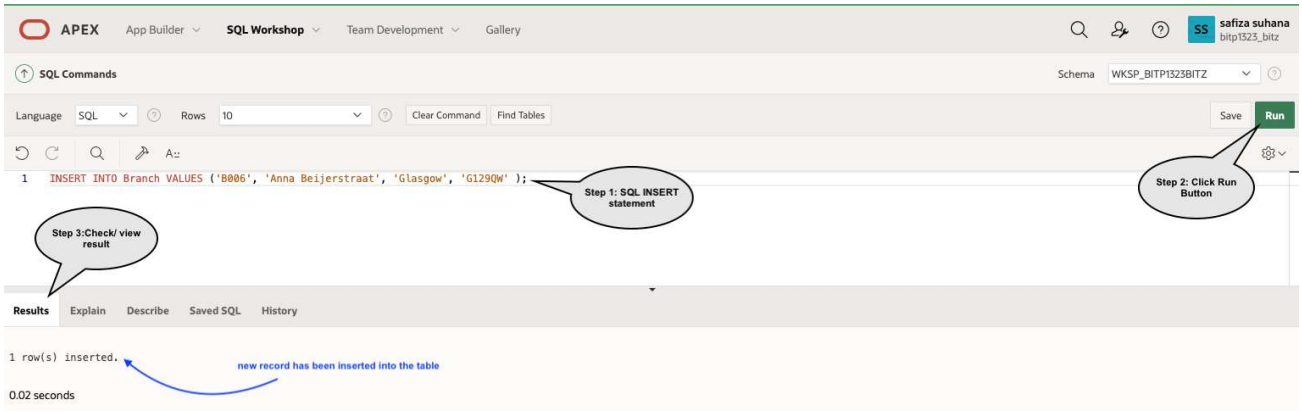


Figure 10

4. You can check the new data by using SQL SELECT statement as shown in section VIEW DATA above (Figure 9). The new data is successfully inserted is shown in Figure 11.

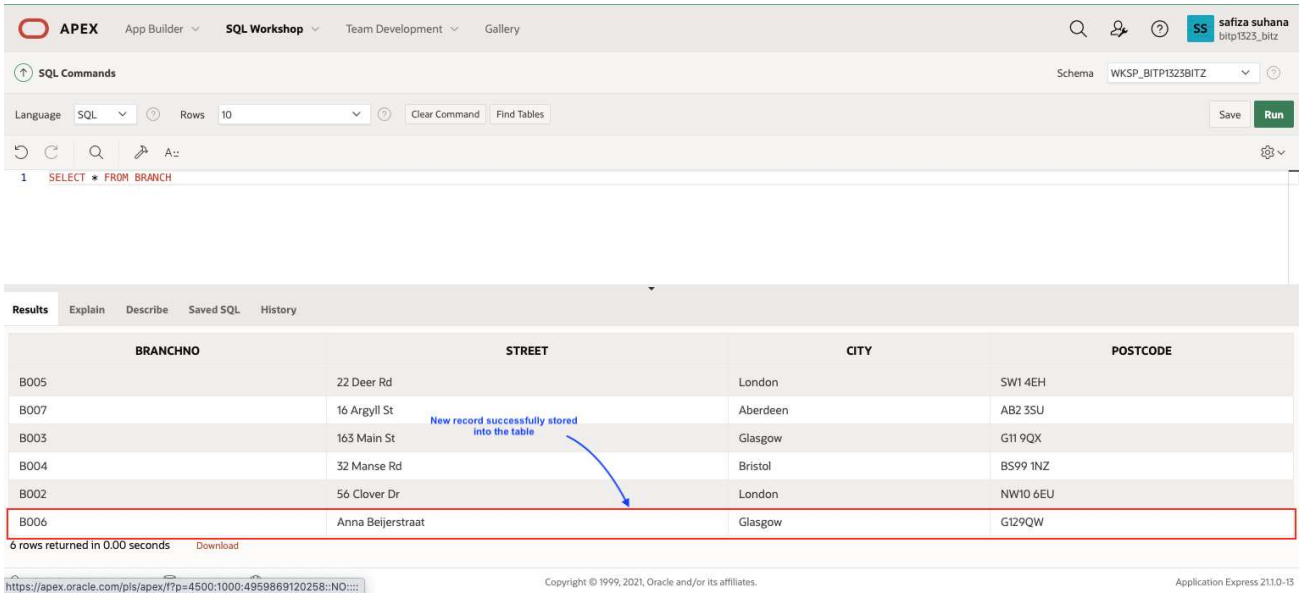


Figure 11

Scenario 2: Insert data into selected column

INSERT INTO <table> (<fieldname1, fieldname2, ... >) VALUES (<data1, data2, ...))

EXAMPLE:

INSERT INTO Branch (BRANCHNO, CITY, POSTCODE) VALUES ('B008', 'Bradford', 'BD1 2DX,');

In this scenario, we not insert the **STREET** data yet into **BRANCH** table. However, it depends on the data specified in Data Dictionary, if the data is not required (or in other words, it can be null), then this statement is allowed.

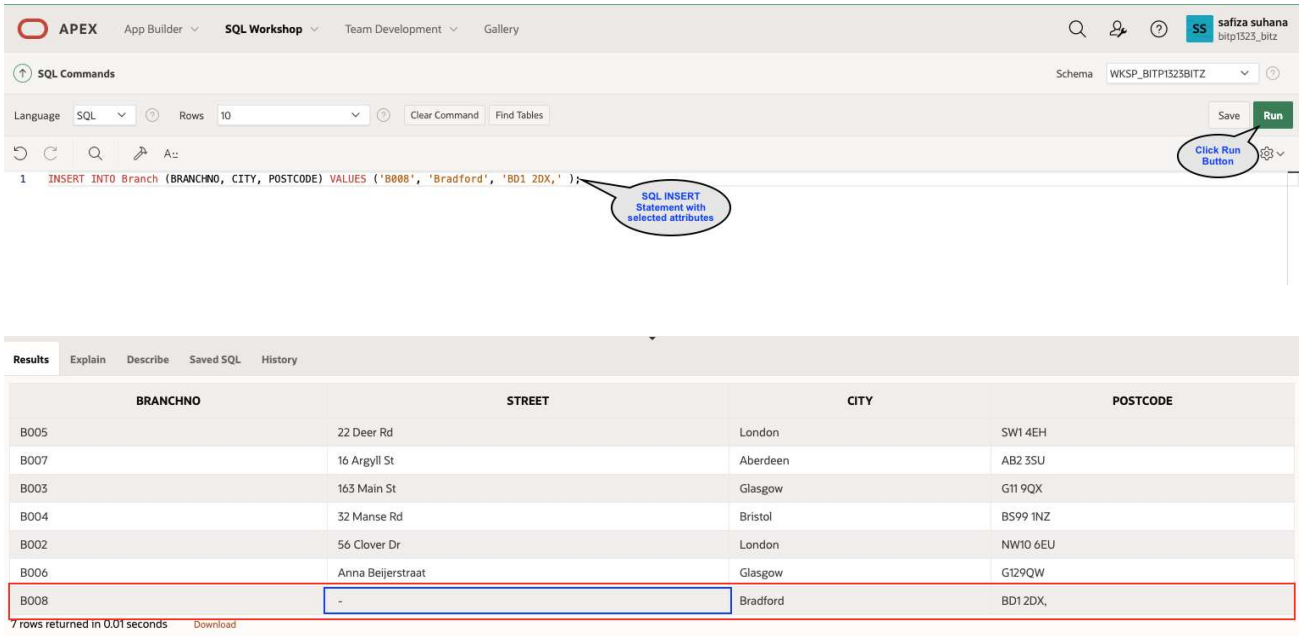


Figure 12

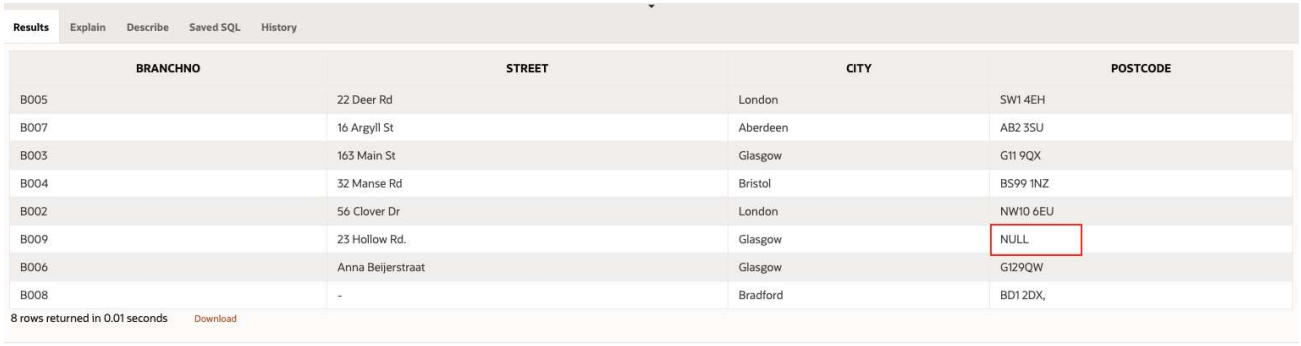
Figure 12 shows that the new data for BRANCH table that is B008 is successfully inserted into the table (red line). However the STREET data for is not inserted yet (null) (blue line).

Scenario 3: Insert all data into table which has null value.

Figure 13 (SQL statement) and 14 (Result) below showed the SQL statement to insert all data into the **BRANCH** table which involved '**NULL**' value for the **POSTCODE** data.

INSERT INTO Branch VALUES ('B009', '23 Hollow Rd.', 'Glasgow', 'NULL');

Figure 13



Results | Explain | Describe | Saved SQL | History

BRANCHNO	STREET	CITY	POSTCODE
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU
B009	23 Hollow Rd.	Glasgow	NULL
B006	Anna Beijerstraat	Glasgow	G129QW
B008	-	Bradford	BD1 2DX,

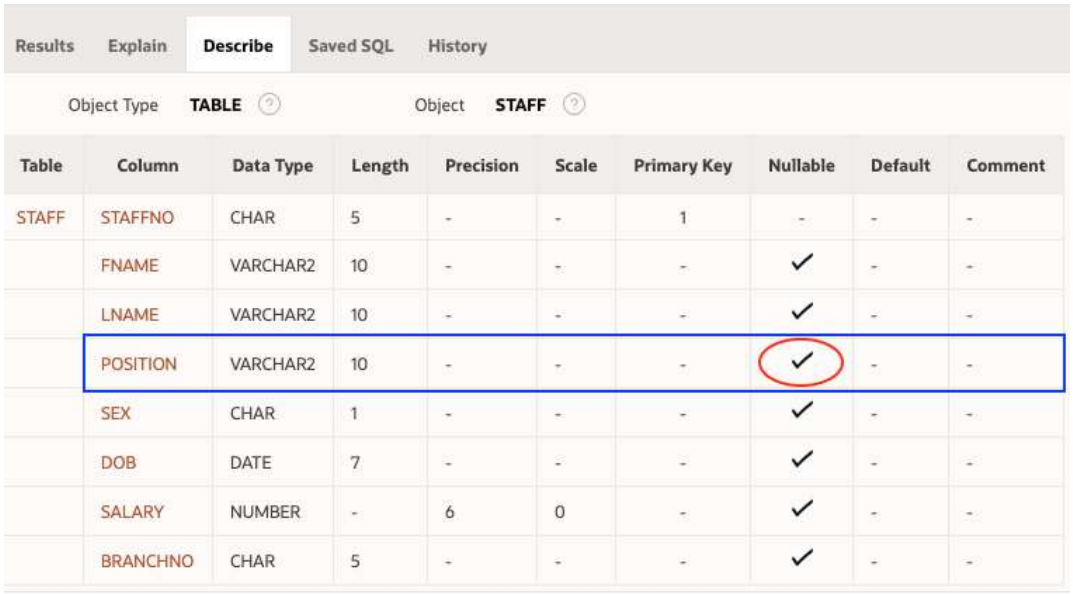
8 rows returned in 0.01 seconds Download

Figure 14

Activity - Do It By Yourself - DDL

1. ADD CONSTRAINT TO STAFF TABLE

a. Assume that we need to change the table structure of STAFF to make sure that the data POSITION is REQUIRED DATA. How do you do that?



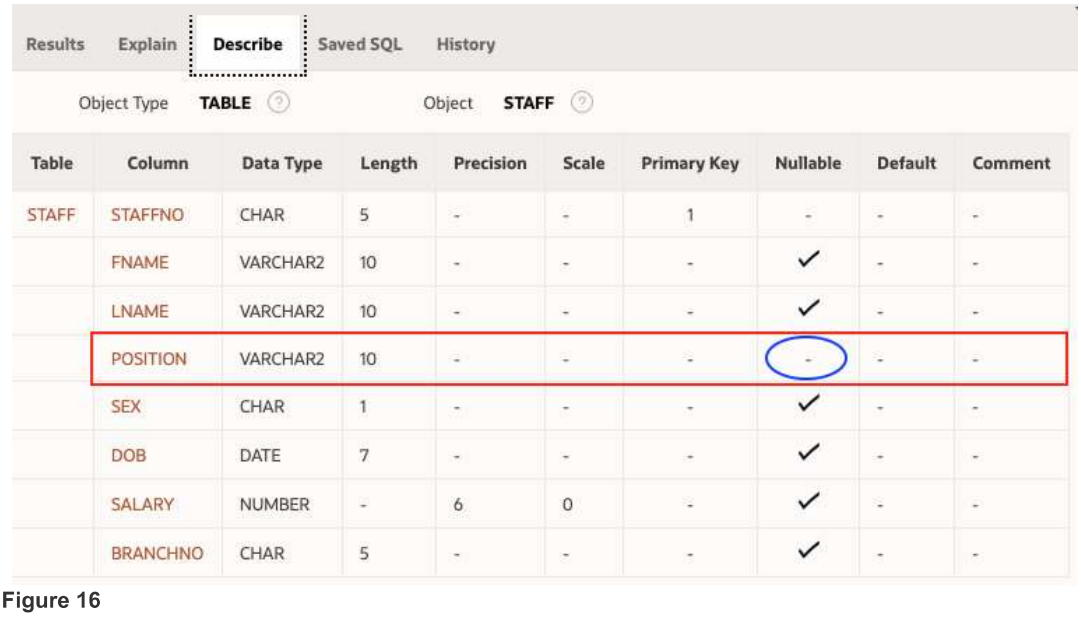
Results | Explain | Describe | Saved SQL | History

Object Type TABLE Object STAFF

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STAFF	STAFFNO	CHAR	5	-	-	1	-	-	-
	FNAME	VARCHAR2	10	-	-	-	✓	-	-
	LNAME	VARCHAR2	10	-	-	-	✓	-	-
	POSITION	VARCHAR2	10	-	-	-	✓	-	-
	SEX	CHAR	1	-	-	-	✓	-	-
	DOB	DATE	7	-	-	-	✓	-	-
	SALARY	NUMBER	-	6	0	-	✓	-	-
	BRANCHNO	CHAR	5	-	-	-	✓	-	-

Figure 15

Figure 15 shows that all fields allowed the NULL data, how we want to change its constraints to NOT ACCEPT Null value as shown in Figure 16?



Results | Explain | Describe | Saved SQL | History

Object Type TABLE Object STAFF

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STAFF	STAFFNO	CHAR	5	-	-	1	-	-	-
	FNAME	VARCHAR2	10	-	-	-	✓	-	-
	LNAME	VARCHAR2	10	-	-	-	✓	-	-
	POSITION	VARCHAR2	10	-	-	-	-	-	-
	SEX	CHAR	1	-	-	-	✓	-	-
	DOB	DATE	7	-	-	-	✓	-	-
	SALARY	NUMBER	-	6	0	-	✓	-	-
	BRANCHNO	CHAR	5	-	-	-	✓	-	-

Figure 16

b. In SQL Commands, write these SQL statements
ALTER TABLE STAFF MODIFY (POSITION VARCHAR(10) CONSTRAINT NN_Position NOT NULL)

c. Write SQL statement below
DESC STAFF;
Was the structure of the table changed?

Activity - Do It By Yourself - DML

ACTIVITY WEEK 9 - DO-IT BY YOURSELF - DML
Please insert the rest of the data to specified tables as shown below:
a. Write SQL Statement to insert the data below.

TABLE: STAFF

STAFFNO	SG72
FNAME	Joseph
LNAME	Martin
POSITION	Supervisor
SEX	M
DOB	15 July 1975
SALARY	15000
BRANCHNO	B006

b. What happens if you write below SQL statement? Why did you get that output?
INSERT INTO STAFF VALUES ('SL09', 'Monica', 'Fareast',", 'F', NULL, 16800, 'B009');

Yes, you will get the error below.

```
ORA-01400: cannot insert NULL into ("MY_A712_SQL_T01"."STAFF"."POSITION")
```

c. Continue to insert all the data below.

TABLE: PRIVATEOWNER

OWNERNO	CO88
FNAME	Suzanne
LNAME	Wilson
ADDRESS	St. Andrew's Drive, Paisley PA3 2TJ, UK
TELNO	01224-786212
EMAIL	Suzanne.wil@emailme.com
PASSWORD	********

TABLE: CLIENT

CLIENTNO	CR58
FNAME	Fiona
LNAME	Lindsay

TELNO	01876745632
PREFFTYPE	Apartment
MAXRENT	800
EMAIL	fionalind@gmail.com

PROPERTYFORRENT

PROPERTYNO	PB23
STREET	1st Floor, 156 Grattan Road, UK
CITY	Bradford,
POSTCODE	BD1 2JL
TYPE	Apartment
ROOMS	3
RENT	750
OWNERNO	CO88
STAFFNO	SG72
BRANCHNO	B005

TABLE: VIEWING

CLIENTNO	CR56
PROPERTYNO	PB23
VIEWDATE	20 March 2020
COMMENTS	Perfect!

TABLE: REGISTRATION

CLIENTNO	CR58
BRANCHNO	B003
STAFFNO	SG72
DATEJOINED	20 Jan 2019

UPDATE DATA

This section will show you how to update data in the table (table row). Suppose that you want to update the data as specified below:

TABLE:BRANCH

BRANCHNO	B006	
STREET	Anna Beijerstraat	12, Anna Beijerstraat
CITY	Glasgow	
POSTCODE	G129QW	

We can Update the data above using SQL statements by following the below syntax.

UPDATE <tablename> SET <columnname> = <expression> , <columnname = <expression>

WHERE <condition list>

This example shows how to **Update** STREET data from old data ('Anna Beijerstraat') to new data ('12, Anna Beijerstraat') for row **BRANCHNO B006** in a **BRANCH** table:

1. In **SQL Commands** write the SQL statement below:

***UPDATE BRANCH SET STREET* = '12, Anna Beijerstraat' WHERE BRANCHNO = 'B006'**

(If more than one attribute is to be updated in a row, separated the correction with commas.)

UPDATE branch SET city='Glassgow', postcode = 'AB6 3SU' WHERE branchno = 'B005'

2. Click the **Run** button.

3. The message *1 row(s) updated* will be displayed once data has been updated to the table successfully. You can check the updated data by using SQL SELECT statement as shown in section VIEW DATA above (Figure 9). The new data is successfully updated is shown in Figure 19.

BRANCHNO	STREET	CITY	POSTCODE
B005	22 Deer Rd	Glassgow	AB6 3SU
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU
B009	23 Hollow Rd.	Glasgow	NULL
B006	12, Anna Beijerstraat	Glasgow	G129QW
B008	-	Bradford	BD1 2DX,

Figure 19

DELETE DATA

This section will show you how to delete data in the table (table row). Suppose that you want to delete the data as specified below:

TABLE: BRANCH

BRANCHNO	B006
STREET	12 Anna Beijerstraat
CITY	Glasgow
POSTCODE	G129QW

We can Delete the data above using the SQL statement by following the below syntax.

DELETE FROM <tablename> WHERE <condition list>

This example shows how to **Delete** BRANCHNO B002 from the table BRANCH

1. In **SQL Commands** write the SQL statement below:

***DELETE FROM BRANCH* WHERE BRANCHNO = 'B002'**

2. Click the **Run** button.

3. The message *1 row(s) deleted* will be displayed once data has been deleted from the table successfully. You can check the deleted data by using SQL SELECT statement as shown in section VIEW DATA above (Figure 9). The new data is successfully deleted is shown in Figure 20.

BRANCHNO	STREET	CITY	POSTCODE
B005	22 Deer Rd	Glassgow	AB6 3SU
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004 Latest record from BRANCH table	32 Manse Rd	Bristol	BS99 1NZ
B009	23 Hollow Rd.	Glasgow	NULL
B006	12, Anna Beijerstraat	Glasgow	G129QW
B008	-	Bradford	BD1 2DX,

Figure 20.

There is **no record *B002*** anymore in the **Branch table**.

RESTORE DATA (NO NEED TO BE IMPLEMENTED AS IT'S NOT SUPPORTED BY APEX)

If we not yet used COMMIT commands to store the changes permanently in the database, we can restore the database to its previous condition with the ROLLBACK command.

ROLLBACK command undoes any changes since the last COMMIT command and brings the data back to the values that existed before the changes were made.

1. To restore the data to their "pre-change" condition, write SQL Command

ROLLBACK

2. If you confirm with the data, write SQL Command COMMIT

Exercise

Write and execute DML statements for the following tasks:

1. Change the first name of staff with staff number SL21 to Johnny. Verify the changes made to the table.
2. Change the salary to 40000 for staff who works as a manager. Verify the changes made to the table.
3. Change the rent to 375 for a Flat with 3 rooms. Verify the changes made to the table.
4. Delete a registration made by the client with clientno CR74. Verify the changes you made to the table.
5. Delete a branch that is located in Bristol from the branch table. Verify the changes made to the table.
6. Delete a branch which is located in Aberdeen from the branch table. Explain the result you get after processing this query.

CHALLENGE YOUR SELF

1. Create a new table named PrivateownerGlasgow based on the selected columns (ownerno, fname, telno, email) and rows of an existing table (Private owner who lives in Glasgow). Confirm the creation of PrivateownerGlasgow table and your addition to the table.