



SQL

Rajsaranya

DDL COMMANDS:

- ddl stands for data definition language
- ddl is a set of sql commands used to create, modify, and delete database structures but not data.

List of DDL commands:

- **CREATE**: This command is used to create the database or its objects (like table, index, function, views, store procedure, and triggers).
DROP: This command is used to delete objects from the database.
ALTER: This is used to alter the structure of the database.
TRUNCATE: This is used to remove all records from a table, including all spaces allocated for the records are removed.
COMMENT: This is used to add comments to the data dictionary.
RENAME: This is used to rename an object existing in the database.

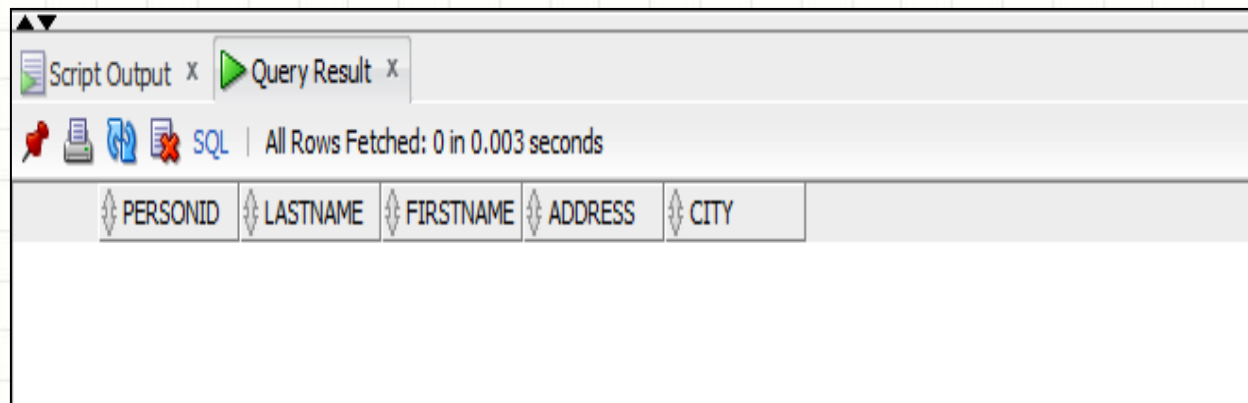


CREATE

TO CREATE A TABLE SYNTAX

CREATE TABLE *table_name* (column's datatypes);

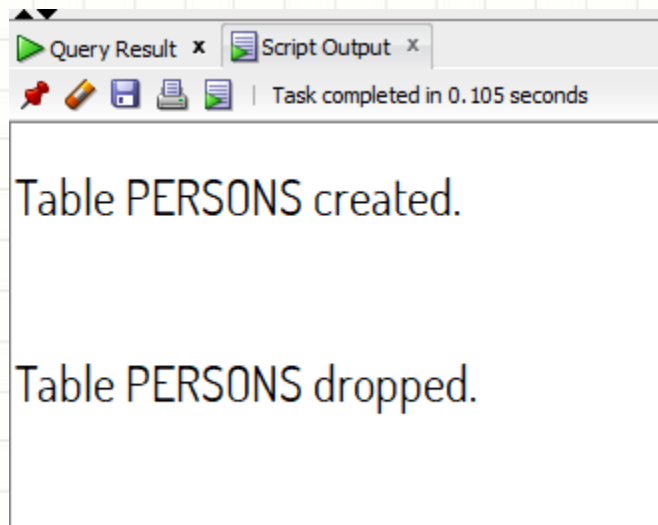
```
CREATE TABLE Persons (  
    PersonID int,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255);
```





DROP

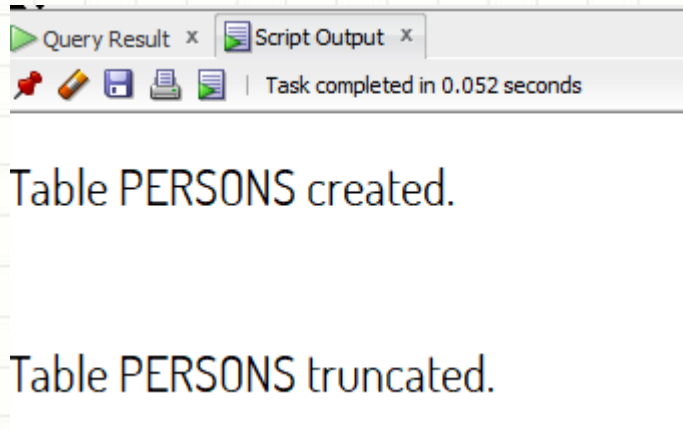
- **DROP THE TABLE**
- **SYNTAX**
- DROP TABLE table_name;
- **DROP TABLE Persons;**





TRUNCATE

- **TRUNCATE THE TABLE**
- **SYNTAX**
- *TRUNCATE TABLE table_name;*
- *TRUNCATE TABLE Persons;*





RENAME


- **Rename THE TABLE**
- **SYNTAX**
- **RENAME TABLE Old_Table_Name TO New_Table_Name;**
-



ALTER

- TO ADD COLOUMNS IN A TABLE
- SYNTAX
- ALTER TABLE *table_name* (*column's datatypes*);
- VARCHAR 2000BYTES
- VARCHAR2 4000 BYTES IT HAS ALL THE SPECIAL CHARACTERS ALSO
- ALTER TABLE Persons
ADD Email varchar(255);

Script Output x Query Result x

 | All Rows Fetched: 5 in 0.002 seconds

	PERSONID	LASTNAME	FIRSTNAME	ADDRESS	CITY	EMAIL	SALARY	SINCREMENT	BGRP	DOB
1	1A	Raisaranva	Adambakkam	chennai	rai@gmail...	23000	10	0+	2309	
2	1V	Vengat	karumandapam	trichv	venkv@g...	23000	10	B+	1204	
3	2V	Vengat	karumandapam	trichv	venkv@g...	23000	10	B+	1204	
4	3S	Ashok	Adambakkam	chennai	ashok@g...	23000	10	0+	1602	
5	4S	Gaia	Adambakkam	chennai	gaia@gm...	23000	10	0+	2509	

Alter the table using modify

- ALTER TABLE Persons MODIFY (personid varchar(20));

- **INSERT VALES IN TABLE**

- **SYNTAX**

- **INSERT INTO *table_name* (*column's1*)**

VALUES (*value's*);

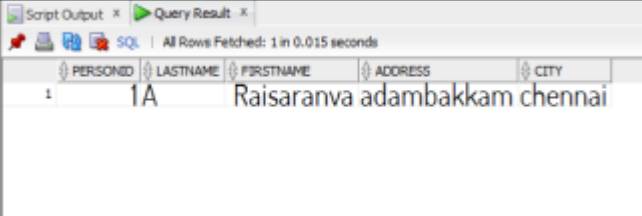
- **INSERT INTO Persons (PersonID,LastName,
FirstName,Address,City)**

VALUES ('0001','A','Rajsaranya','adambakkam','chennai');

1 row inserted.

```
SQL> INSERT INTO Persons ( PersonID,LastName, FirstName,Address,City)  
2 VALUES ('0001','A','Rajsaranya','adambakkam','chennai');
```

1 row inserted.



The screenshot shows a 'Query Result' window with a table containing one row of data. The table has five columns: PERSONID, LASTNAME, FIRSTNAME, ADDRESS, and CITY. The row contains the values 1, A, Rajsaranya, adambakkam, and chennai. The status bar at the top indicates 'All Rows Fetched: 1 in 0.015 seconds'.

PERSONID	LASTNAME	FIRSTNAME	ADDRESS	CITY
1	A	Rajsaranya	adambakkam	chennai

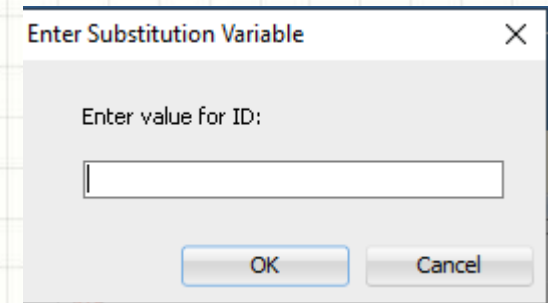
- **INSERT VALES IN TABLE MANUALLY BY USER**

- **SYNTAX**

- **INSERT INTO *table_name* (*column's1* ,...) VALUES ('&ALIASNAME',...);**


- **INSERT INTO Persons (PersonID,LastName, FirstName,Address,City)**

- **VALUES ('&ID','&LN','&FN','&ADD','&CITY');**



A dialog box titled "Enter Substitution Variable" with a close button (X) in the top right corner. Inside the dialog, there is a label "Enter value for ID:" followed by a text input field. At the bottom of the dialog, there are two buttons: "OK" and "Cancel".

Script Output x Query Result x

 | All Rows Fetched: 4 in 0.003 seconds

	PERSONID	LASTNAME	FIRSTNAME	ADDRESS	CITY
1	1A	Raisaranva	adambakkam	chennai	
2	2V	VENGAT	KARUMANDAPAM	TRICHY	
3	3S	ASHOK	KILIYUR	KUMBAKONAM	

- TO ADD COLOUMNS IN THE TABLE
- SYNTAX
- ALTER TABLE *table_name*
ADD *column_name datatype(value);*
- ALTER TABLE Persons
ADD Persons_Salary varchar(data);
- ALTER TABLE Persons
ADD Persons_Salary varchar2(250);

Script Output x

Query Result x

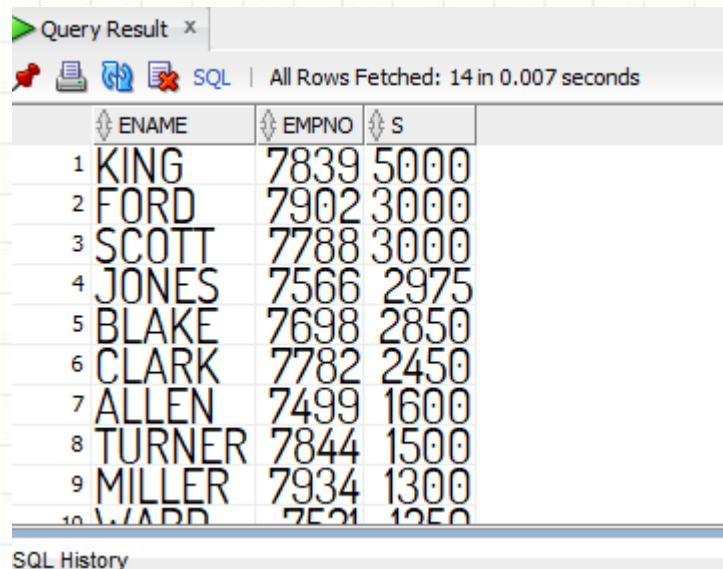
Query Result 1 x

SQL | All Rows Fetched: 5 in 0.004 seconds

	PERSONID	LASTNAME	FIRSTNAME	ADDRESS	CITY	PERSONS_SALARY
1	1A	Raisaranva	adambakkam	chennai	(null)	
2	2V	Vengat	karumandapam	trichv	(null)	
3	3S	Gaia	haridwarmagalam	kumakonam	(null)	
4	4S	Ashok	kattur	taniavur	(null)	
5	5L	abi	chengalbattu	chennai	(null)	

Sorting – NAME WISE

- This query is used to sort the values based on name
- `SELECT * FROM TABLENAME ORDER BY COLUMNNAME DESC;`
- `SELECT * FROM TABLENAME ORDER BY COLUMNNAME ASC;`
- `SELECT ENAME,EMPNO,SAL S FROM EMP ORDER BY S DESC ;`



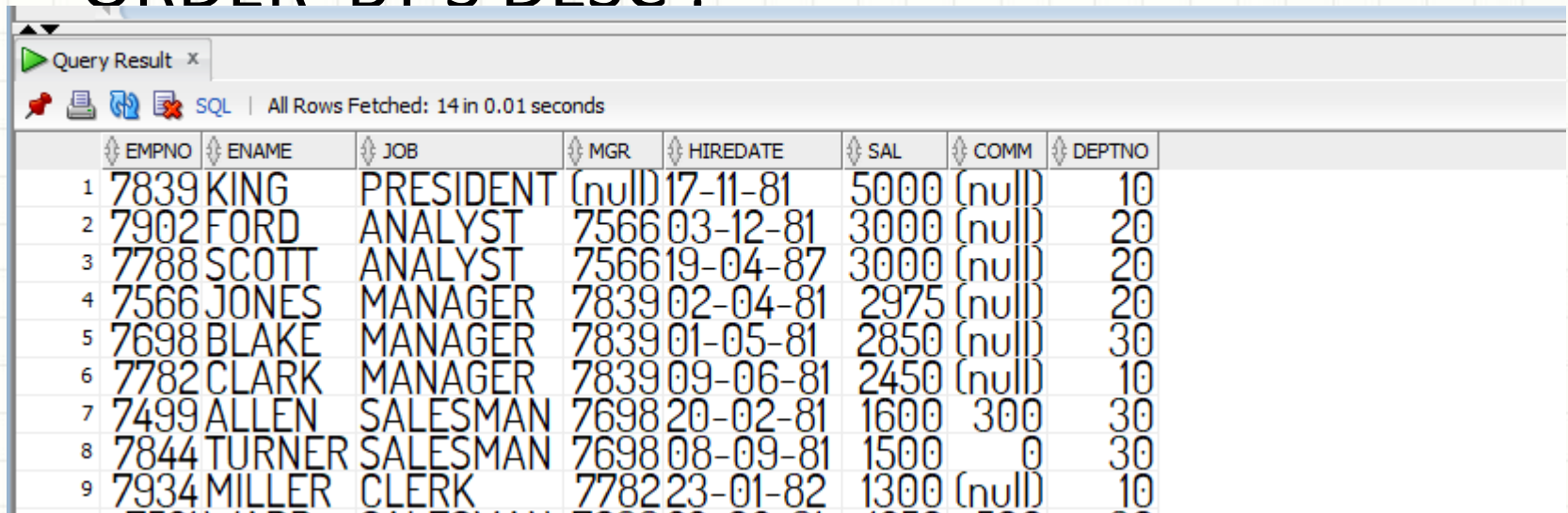
The screenshot shows a 'Query Result' window with a toolbar containing icons for a red pin, a printer, a refresh, a document with a red X, and the text 'SQL'. To the right of the toolbar, it says 'All Rows Fetched: 14 in 0.007 seconds'. Below the toolbar is a table with three columns: 'ENAME', 'EMPNO', and 'S'. The table contains 10 rows of data, sorted by salary in descending order. The first row is KING with salary 5000, and the last row is WARD with salary 1250. Below the table is a section labeled 'SQL History'.

	ENAME	EMPNO	S
1	KING	7839	5000
2	FORD	7902	3000
3	SCOTT	7788	3000
4	JONES	7566	2975
5	BLAKE	7698	2850
6	CLARK	7782	2450
7	ALLEN	7499	1600
8	TURNER	7844	1500
9	MILLER	7934	1300
10	WARD	7521	1250

SQL History

Sorting – ALIAS wise

- This query is used to sort the values based on the column name and provide alias name .
- **SELECT ENAME,EMPNO,SAL S FROM EMP ORDER BY S DESC :**

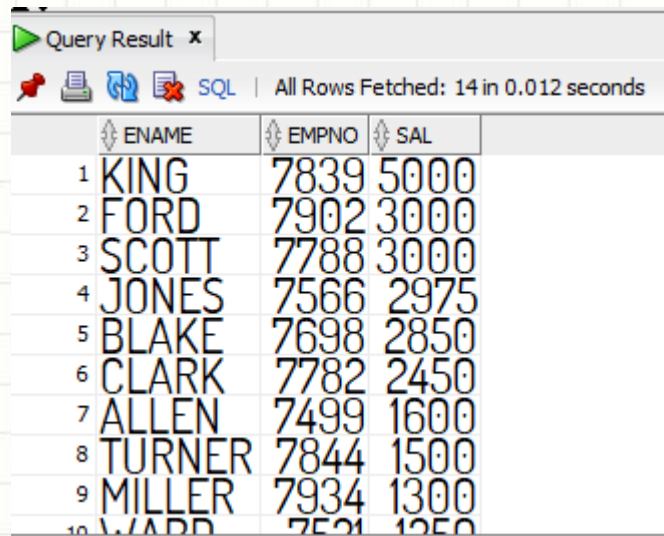


The screenshot shows a 'Query Result' window with a table of employee data. The table has 9 rows and 8 columns: EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, and DEPTNO. The data is sorted by the SAL column in descending order. The first row is King (5000) and the last row is Miller (1300). The COMM column contains null values for most employees, except for Allen (300) and Turner (0).

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7839	KING	PRESIDENT	(null)	17-11-81	5000	(null)	10
2	7902	FORD	ANALYST	7566	03-12-81	3000	(null)	20
3	7788	SCOTT	ANALYST	7566	19-04-87	3000	(null)	20
4	7566	JONES	MANAGER	7839	02-04-81	2975	(null)	20
5	7698	BLAKE	MANAGER	7839	01-05-81	2850	(null)	30
6	7782	CLARK	MANAGER	7839	09-06-81	2450	(null)	10
7	7499	ALLEN	SALESMAN	7698	20-02-81	1600	300	30
8	7844	TURNER	SALESMAN	7698	08-09-81	1500	0	30
9	7934	MILLER	CLERK	7782	23-01-82	1300	(null)	10

Sorting – POSITION wise

- This query is used to sort the values based on the column name by its positions.
- SELECT COLOUMN NAMES1,COLOUMN NAME 2,... FROM TABLENAME ORDER BY NO DESC;
- SELECT COLOUMN NAMES1,COLOUMN NAME 2,... FROM TABLENAME ORDER BY NO ASC;
- SELECT ENAME,EMPNO,SAL FROM EMP ORDER BY 3 DESC;



The screenshot shows a 'Query Result' window with a toolbar containing icons for save, print, SQL, and error. The status bar indicates 'All Rows Fetched: 14 in 0.012 seconds'. The table below displays the results of a query sorting employees by salary (SAL) in descending order.

	ENAME	EMPNO	SAL
1	KING	7839	5000
2	FORD	7902	3000
3	SCOTT	7788	3000
4	JONES	7566	2975
5	BLAKE	7698	2850
6	CLARK	7782	2450
7	ALLEN	7499	1600
8	TURNER	7844	1500
9	MILLER	7934	1300
10	WARD	7521	1250



COMMENT

BACKUP TABLE

- CREATE TABLE NEW TABLENAME AS (SELECT * FROM OLD TABLE NAME);
- CREATE TABLE PERSON_INFO AS (SELECT * FROM PERSONS);

Table PERSON_INFO created.