SQL STATEMENTS

KRISHNA

SQL SYNTAX

DATATYPE :

DATATYPE	SYNTAX
1. CHAR	CHAR (SIZE)
2. VARCHAR	VARCHAR(SIZE)
3. VARCHAR 2	VARCHAR(SIZE) VARCHAR 2 (SIZE)
4. DATE	DATE
5. LARGE OBJECT CHARACTER LARGE OBJECT	CLOB
BINARY LARGE OBJECT	BLOB

DATA QUERY LANGUAGE :

PROJECTION :

SYNTAX:

SELECT */[DISTINCT] COLUMN_NAME /EXPRESSION [ALIAS]

FROM TABLE_NAME ;

SELECTION :

SYNTAX

SELECT */[DISTINCT] COLUMN_NAME /EXPRESSION [ALIAS]

FROM TABLE_NAME

WHERE < FILTER CONDITION> ;

SPECIAL OPERATORS :

1.IN:

---

column_name/expression IN (v1,v2,...Vn)

2.NOT IN :

---

column_name/expression NOT IN (v1,v2,...Vn)

3.BETWEEN:

---

column_name/expression BETWEEN lower_range AND higher_range

4.NOT BETWEEN :

---

column_name/expression NOT BETWEEN lower_range AND higher_range

5.IS :

---

column_name/expression IS NULL

6.IS NOT :

---

column_name/expression IS NOT NULL

7.LIKE:

---

column_name/expression LIKE 'pattern_to_match'

8.NOT LIKE:

---

column_name/expression NOT LIKE 'pattern_to_match' ;

ESCAPE CHARACTER

---

column_name/expression LIKE/NOT LIKE 'pattern_to_match' ESCAPE 'char'

GROUP BY CLAUSE :

SYNTAX:

SELECT group_by_expression / group_function

FROM table_name

[WHERE <filter_condition>]

GROUP BY column_name/expression ;

HAVING CLAUSE :

SYNTAX:

SELECT group_by_expression / group_function

FROM table_name

[WHERE <filter_condition>]

GROUP BY column_name/expression

HAVING <group_filter_condition> ;

ORDER BY CLAUSE :

SYNTAX:

SELECT group_by_expression / group_function

FROM table_name

[WHERE <filter_condition>]

[GROUP BY column_name/expression]

[HAVING <group_filter_condition>]

ORDER BY Col_name/expression [ASC]/DESC ;

JOINS :

1.CARTESIAN JOIN/CROSS JOIN

SYNTAX:ANSI

SELECT col_name

FROM table_name1 CROSS JOIN table_name2;

SYNTAX:ORACLE

SELECT col_name

FROM table_name1, table_name2;

2.INNER JOIN/EQUI JOIN

SYNTAX:ANSI

```
SELECT col_name
FROM table_name1 INNER JOIN table_name2
ON table_name1.col_name=table_name2.col_name;
```

SYNTAX:ORACLE

```
SELECT col_name
FROM table_name1, table_name2
WHERE table_name1.col_name=table_name2.col_name;
```

3.NATURAL JOIN

SYNTAX:ANSI

```
SELECT col_name
FROM table_name1 NATURAL JOIN table_name2 ;
```

NOTE: NO ORACLE SYNTAX FOR NATURAL JOIN

4.OUTER JOIN

I. LEFT OUTER JOIN

SYNTAX:ANSI

```
SELECT col_name
FROM table_name1 LEFT [OUTER] JOIN table_name2
ON table_name1.col_name=table_name2.col_name;
```

SYNTAX:ORACLE

```
SELECT col_name
FROM table_name1, table_name2
WHERE table_name1.col_name=table_name2.col_name(+);
```

II. RIGHT OUTER JOIN

SYNTAX:ANSI

```
SELECT col_name
FROM table_name1 RIGHT [OUTER] JOIN table_name2
ON table_name1.col_name=table_name2.col_name;
```

SYNTAX:ORACLE

```
SELECT col_name
FROM table_name1, table_name2
WHERE table_name1.col_name(+)=table_name2.col_name;
```

III. FULL OUTER JOIN

SYNTAX:ANSI

```
SELECT col_name
FROM table_name1 FULL [OUTER] JOIN table_name2
ON table_name1.col_name=table_name2.col_name;
```

[NOTE: NO ORACLE SYNTAX FOR FULL OUTER JOIN]

5.SELF JOIN

SYNTAX:ANSI

```
SELECT col_name
FROM table_name1 T1 JOIN table_name1 T2
ON T1.col_name=T2.col_name;
```

SYNTAX:ORACLE

```
SELECT col_name
FROM table_name1 T1, table_name1 T2
WHERE T1.col_name=T2.col_name;
```

DATA DEFINITION LANGUAGE:

1.CREATE:

CREATE TABLE Table_Name

(Column_Name1 datatype constraint_type ,

Column_Name2 datatype constraint_type ,

Column_Name3 datatype constraint_type ,

.

.

Column_NameN datatype constraint_type) ;

2.RENAME

SYNTAX: RENAME current_table_name TO New_table_name;

3.ALTER

1.TO ADD A COL :

ALTER TABLE table_name

ADD COLUMN_NAME DATATYPE CONSTRAINT;

2.TO RENAME THE COLUMN:

ALTER TABLE table_name

RENAME COLUMN current_name TO new_name;

3.TO DROP A COL :

ALTER TABLE table_name

DROP COLUMN COLUMN_NAME ;

4.TO CHANGE THE DATATYPE:

ALTER TABLE table_name

MODIFY COLUMN_NAME new_datatype;

5.TO CHANGE THE CONSTRAINT[NULL/ NOT NULL]:

ALTER TABLE table_name

MODIFY COLUMN_NAME existing_datatype NULL/NOTNULL;

4. TRUNCATE:

SYNTAX: TRUNCATE TABLE table_name;

5.DROP:

SYNTAX: DROP TABLE table_name;

TO RECOVER THE TABLE:(only in oracle)

SYNTAX: FLASHBACK TABLE table_name

TO BEFORE DROP ;

TO DROP THE TABLE FROM RECYCLE BIN

SYNTAX: PURGE TABLE table_name;

DATA MANIPULATION LANGUAGE

1.INSERT:

SYNTAX 1: INSERT INTO table_name VALUES (V1,V2,....,Vn);

(OR)

SYNTAX 2 : INSERT INTO TABLE_NAME VALUES (&COL1,COL2,…COL N) ;

2.UPDATE:

SYNTAX: UPDATE table_name

SET COL1=V1,COL2=V2,.....,COLn=Vn

[WHERE <filter_condition>];

3.DELETE:

SYNTAX : DELETE

FROM table_name

[WHERE <filter_condition>];

TRANSACTION CONTROL LANGUAGE

1.COMMIT:

SYNTAX: COMMIT;

2.SAVEPOINT:

SYNTAX: SAVEPOINT savepoint_name**;**

3.ROLLBACK:

SYNTAX: ROLLBACK;

ROLLBACK TO SAVEPOINT

SYNTAX: ROLLBACK TO savepoint_name;

DATA CONTROL LANGUAGE:

1.GRANT:

SYNTAX: GRANT sql_statement ON table_name

TO user_name;

2.REVOKE:

SYNTAX: REVOKE sql_statement ON table_name

FROM user_name;

CONTACT

EMAIL_ID : krishnalalvasu1@gmail.com

INSTA_ID : mr.____krishna___