

DAY 20 / 16

Thursday, July 9, 2020 8:45 AM

STATEMENTS ARE CLASSIFIED INTO 5 DIFFERENT TYPES

- DATA DEFINITION LANGUAGE (DDL)
- DATA MANIPULATION LANGUAGE (DML)
- TRANSACTION CONTROL LANGUAGE (TCL)
- DATA CONTROL LANGUAGE (DCL)
- DATA QUERY LANGUAGE (DQL)

1. DATA DEFINITION LANGUAGE (DDL):

" DDL is used to construct an object in the database and deals with the Structure of the Object "

It has 5 statements :

1. CREATE
2. RENAME
3. ALTER
4. TRUNCATE
5. DROP

1. CREATE : " IT IS USED TO BUILD / CONSTRUCT AN OBJECT "

Object / Entity can be a Table or a View (Virtual Table) .

How to Create a Table :

- Name of the table
 - ▶ Tables cannot have same names .
- Number of Columns .
- Names of the columns .
- Assign datatypes for the Columns.
- Assign Constraints [**NOT MANDATORY**] .

Example 1:

Table_Name : **CUSTOMER**

Number of Columns : 4

Customer

Column_Name	CID	CNAME	CNO	ADDRESS
Datatypes	Number(2)	Varchar(10)	Number (10)	Varchar(15)
Null / Not Null	Not Null	Not Null	Not Null	Null
Unique	Unique		Unique	
Check			Check (length(CNO) = 10)	
Primary Key	Primary Key			
Foreign Key				

Primary Key	Primary Key			
Foreign Key				

Not Mandatory

Syntax to create a table :

```
CREATE TABLE Table_Name
(
    Column_Name1 datatype constraint_type ,
    Column_Name2 datatype constraint_type ,
    Column_Name3 datatype constraint_type ,
    .
    .
    Column_NameN datatype constraint_type
);
```

Example :

```
CREATE TABLE CUSTOMER
(
    CID Number(2) primary key ,
    CNAME Varchar(10) ,
    CNO Number(10) not null check( length( CNO ) = 10 ) ,
    ADDRESS Varchar(15)
);
```

NOTE :

To Describe the table:

```
Syntax:  DESC Table_Name ;
```

Example 2:

Table_Name : **PRODUCT**

Number of Columns : 4

Product

Column_Name	PID	PNAME	PRICE	CID
Datatypes	Number(2)	Varchar(10)	Number (7,2)	Number(2)
Null / Not Null	Not Null	Not Null	Not Null	Null
Unique	Unique			
Check			Check (Price > 0)	
Primary Key	Primary Key			
Foreign Key				Foreign Key

Syntax to create a table :

```
CREATE TABLE Table_Name
(
    Column_Name1 datatype constraint_type ,
    Column_Name2 datatype constraint_type ,
    Column_Name3 datatype constraint_type ,
    .
    .
    Column_NameN datatype ,
    Constraint Foreign key references Parent_Table_Name(Column_Name)
);
```

Example :

```
CREATE TABLE PRODUCT
(
    PID Number(2) primary key ,
    PNAME Varchar(10) ,
    PRICE Number(7,2) check( Price > 0) ,
    CID Number(2) ,
    Constraint CID_FK Foreign Key(CID) references CUSTOMER( CID )
);
```

2.RENAME : "IT IS USED TO CHANGE THE NAME OF THE OBJECT "

Syntax: RENAME Table_Name TO New_Name ;

Example :

```
RENAME Customer TO Cust ;
```

3. ALTER : " IT IS USED TO MODIFY THE STRUCTURE OF THE TABLE "

➤ TO ADD A COLUMN :

Syntax: ALTER TABLE Table_Name
ADD Column_Name Datatype Constraint_type ;

Example : ALTER TABLE Cust
ADD MAIL_ID Varchar(15) ;

➤ TO DROP A COLUMN :

Syntax: ALTER TABLE Table_Name
DROP COLUMN Column_Name ;

Example : ALTER TABLE Cust
DROP COLUMN MAIL_ID ;

➤ **TO RENAME A COLUMN :**

Syntax: ALTER TABLE Table_Name
RENAME COLUMN Column_Name TO new_Column_Name ;

Example : ALTER TABLE Cust
RENAME COLUMN CNO TO PHONE_NO ;

➤ **TO MODIFY THE DATATYPE :**

Syntax: ALTER TABLE Table_Name
MODIFY COLUMN_NAME New_Datatype;

Example : ALTER TABLE Cust
MODIFY CNAME CHAR(10) ;

➤ **TO MODIFY NOT NULL CONSTRAINTS :**

Syntax: ALTER TABLE Table_Name
MODIFY COLUMN_NAME Existing_datatype [NULL]/NOT NULL;

Example : ALTER TABLE Cust
MODIFY ADDRESS Varchar(15) Not Null ;

4. **TRUNCATE :** " IT IS USED TO REMOVE ALL THE RECORDS FROM THE TABLE PERMANENTLY "

Syntax: TRUNCATE TABLE Table_Name ;

Cust

<u>Cid</u>	<u>Cname</u>	<u>Phone_no</u>	<u>Address</u>
1	A	1234567890	BANGALORE
2	B	1234567899	MYSORE
3	C	1234567880	MANGALORE

Example : TRUNCATE TABLE Cust ;

Cust

<u>Cid</u>	<u>Cname</u>	<u>Phone_no</u>	<u>Address</u>

5. **DROP :** " IT IS USED TO REMOVE THE TABLE FROM THE DATABASE "

Syntax: DROP TABLE Table_Name ;

Example :

DATABASE

.. . .

Example :

DATABASE

Cust			
<u>Cid</u>	<u>Cname</u>	<u>Phone_no</u>	<u>Address</u>
1	A	1234567890	BANGALORE
2	B	1234567899	MYSORE
3	C	1234567880	MANGALORE

DROP

Address

Bin Folder



TO RECOVER THE TABLE :

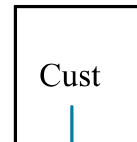
Syntax: FLASHBACK TABLE Table_Name
TO BEFORE DROP ;

Example :

DATABASE

Address : BIN\$123ABCXYZ

Bin Folder



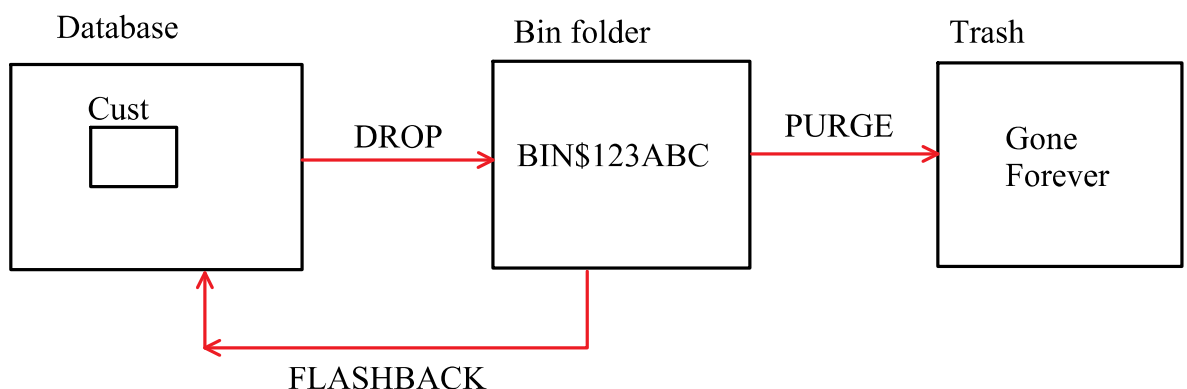
FLASHBACK

FLASHBACK TABLE Cust
TO BEFORE DROP ;

TO DELETE THE TABLE FROM BIN FOLDER :

Syntax: PURGE TABLE Table_Name ;

Example : PURGE TABLE Cust ;



NOTE : DDL STATEMENTS ARE AUTO-COMMIT STATEMENTS

DAY 21 / DAY 17

Friday, July 10, 2020 9:36 AM

DATA MANIPULATION LANGUAGE (DML)

It is used to Manipulate the Object by performing insertion , updating and deletion .

1. INSERT
2. UPDATE
3. DELETE

1. **INSERT** : It is used to insert / create records in the table .

Syntax: INSERT INTO Table_Name VALUES(v1 , v2 , v3) ;

CID	CNAME	CNO	ADDRESS
NUMBER(2)	VARCHAR(10)	NUMBER(10)	VARCHAR(20)
1	ABHI	1234567890	BANGALORE
2	ABDUL	1234567891	MANGALORE

Ex: INSERT INTO CUSTOMER VALUES(1 , 'ABHI' , 1234567890 , 'BANGALORE');
INSERT INTO CUSTOMER VALUES(2 , 'ABDUL' , 1234567891 , 'MANGALORE');

PID	PNAME	PRICE	CID
NUMBER(2)	VARCHAR(10)	NUMBER(6,2)	NUMBER(3)
10	IPHONE	10000	2
20	ZENPHONE	4000	1
30	ONEPLUS	NULL	NULL

Ex: INSERT INTO PRODUCT VALUES(10 , 'IPHONE' , 10000 , 2);
INSERT INTO PRODUCT VALUES(20 , 'ZENPHONE' , 4000 , 1);
INSERT INTO PRODUCT VALUES(20 , 5000 , 'EARPHONES', NULL); // ERROR
INSERT INRO PRODUCT VALUES(30 , 'ONEPLUS8', NULL , NULL);

2.UPDATE :It is used to modify an existing value .

Syntax: UPDATE Table_Name
SET Col_Name = Value , Col_Name = Value ,,,,
[WHERE stmt] ;

CID	CNAME	CNO	ADDRESS
NUMBER(2)	VARCHAR(10)	NUMBER(10)	VARCHAR(20)
1	ABHI	1234567890	BANGALORE
2	ABDUL	1234567891	MANGALORE

Change cno of abdul to 9876543210

EX:

*UPDATE CUSTOMER
SET CNO = 9876543210
WHERE CNAME ='ABDUL ' ;*

CID	CNAME	CNO	ADDRESS
NUMBER(2)	VARCHAR(10)	NUMBER(10)	VARCHAR(20)
1	ABHI	1234567890	BANGALORE
2	ABDUL	9876543210	MANGALORE

3.DELETE : It is used to remove a particular record from the table .

Syntax: DELETE FROM Table_Name
[WHERE stmt];

CID	CNAME	CNO	ADDRESS
NUMBER(2)	VARCHAR(10)	NUMBER(10)	VARCHAR(20)
1	ABHI	1234567890	BANGALORE
2	ABDUL	1234567891	MANGALORE

I have to remove a customer ABHI

*Ex:
DELETE FROM CUSTOMER
WHERE CNAME ='ABHI' ;*

CID	CNAME	CNO	ADDRESS
NUMBER(2)	VARCHAR(10)	NUMBER(10)	VARCHAR(20)
2	ABDUL	1234567891	MANGALORE

1. WAQT update the salary of employee to double their salary if He is working as a manager .

**UPDATE EMP
SET SAL = SAL*2
WHERE JOB ='MANAGER' ;**

2. WAQT change the name of SMITH to SMITH .

**UPDATE EMP
SET ENAME ='SMITH'
WHERE ENAME ='SMITH' ;**

3. WAQT modify the job of KING to 'PRESIDENT' .

**UPDATE EMP
SET JOB ='PRESIDENT'
WHERE ENAME ='KING' ;**

4. WAQT to change name of ALLEN to ALLEN MORGAN .

```
UPDATE EMP  
SET ENAME ='ALLEN MORGAN'  
WHERE ENAME ='ALLEN' ;
```

5. WAQT hike the salary of the employee to 10% . If employees earn less than 2000 as a salesman .

```
UPDATE EMP  
SET SAL = SAL+SAL*10/100  
WHERE SAL < 2000 AND JOB ='SALESMAN' ;
```

6. WAQ TO delete the employees who don't earn commission .

```
DELETE FROM CUSTOMER  
WHERE COMM IS NULL ;
```

7. WAQ to remove all the employees hired before 1987 in dept 20

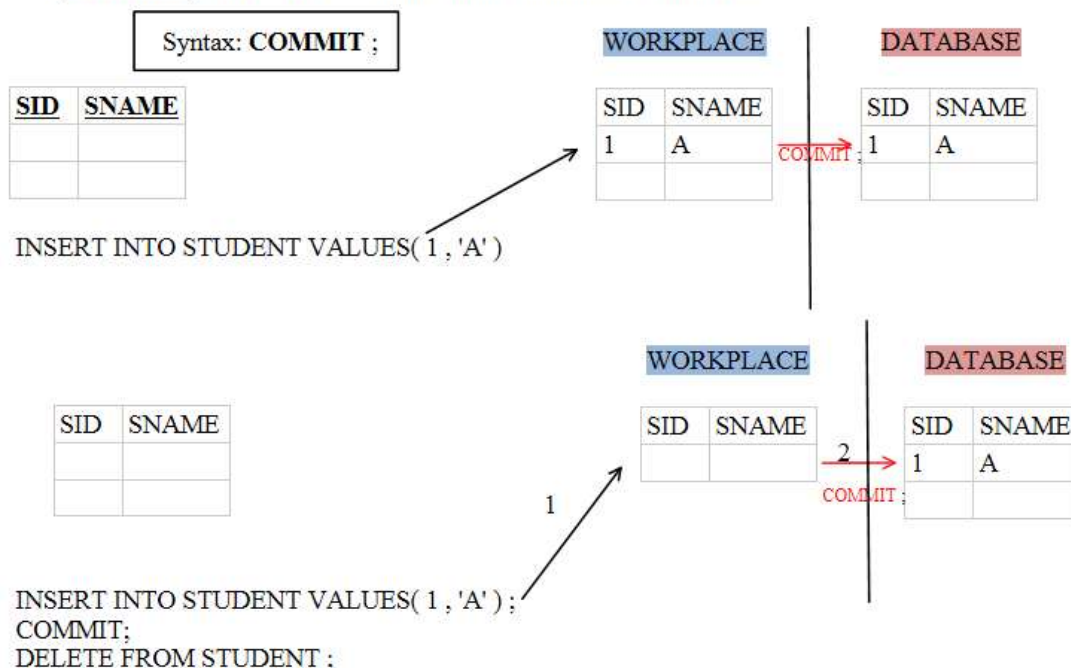
```
DELETE FROM EMP  
WHERE HIREDATE < '01-JAN-1987' AND DEPTNO = 20 ;
```

8. Differentiate between TRUNCATE and DELETE statements .

<u>TRUNCATE</u>	<u>DELETE</u>
Belongs to DDL	Belongs to DML
Removes all the records from the Table permanently .	Removes a particular record from the Table .
Auto COMMIT	Not auto COMMIT .

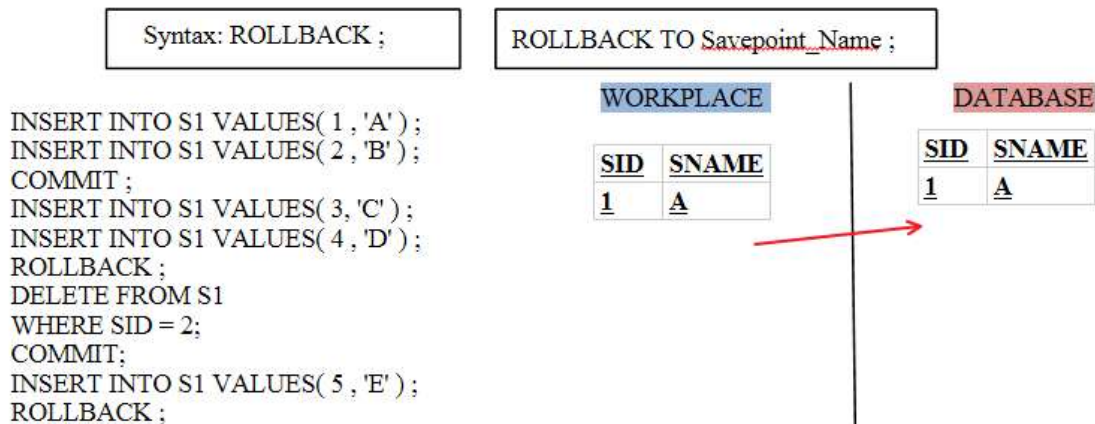
TRANSACTION CONTRL LANGUAGE (TCL)

1. **COMMIT** : It is used to save all the transactions into the database.



NOTE : Transactions : The operations performed by using DML commands on the table is known as Transaction .

2. **ROLLBACK** : It is used to get back the *latest saved transactions* .



3. **SAVEPOINT** : It is used to Mark the transaction and it is used with rollback

Syntax: **SAVEPOINT** Savepoint_Name ;

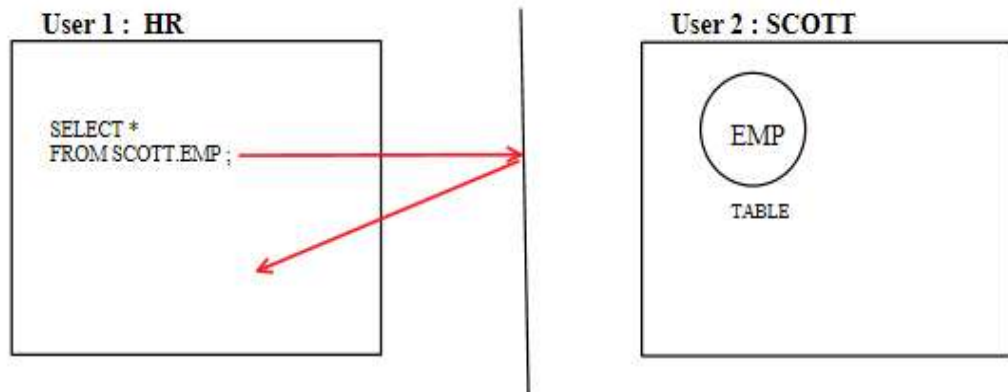
DATA CONTROL LANGUAGE

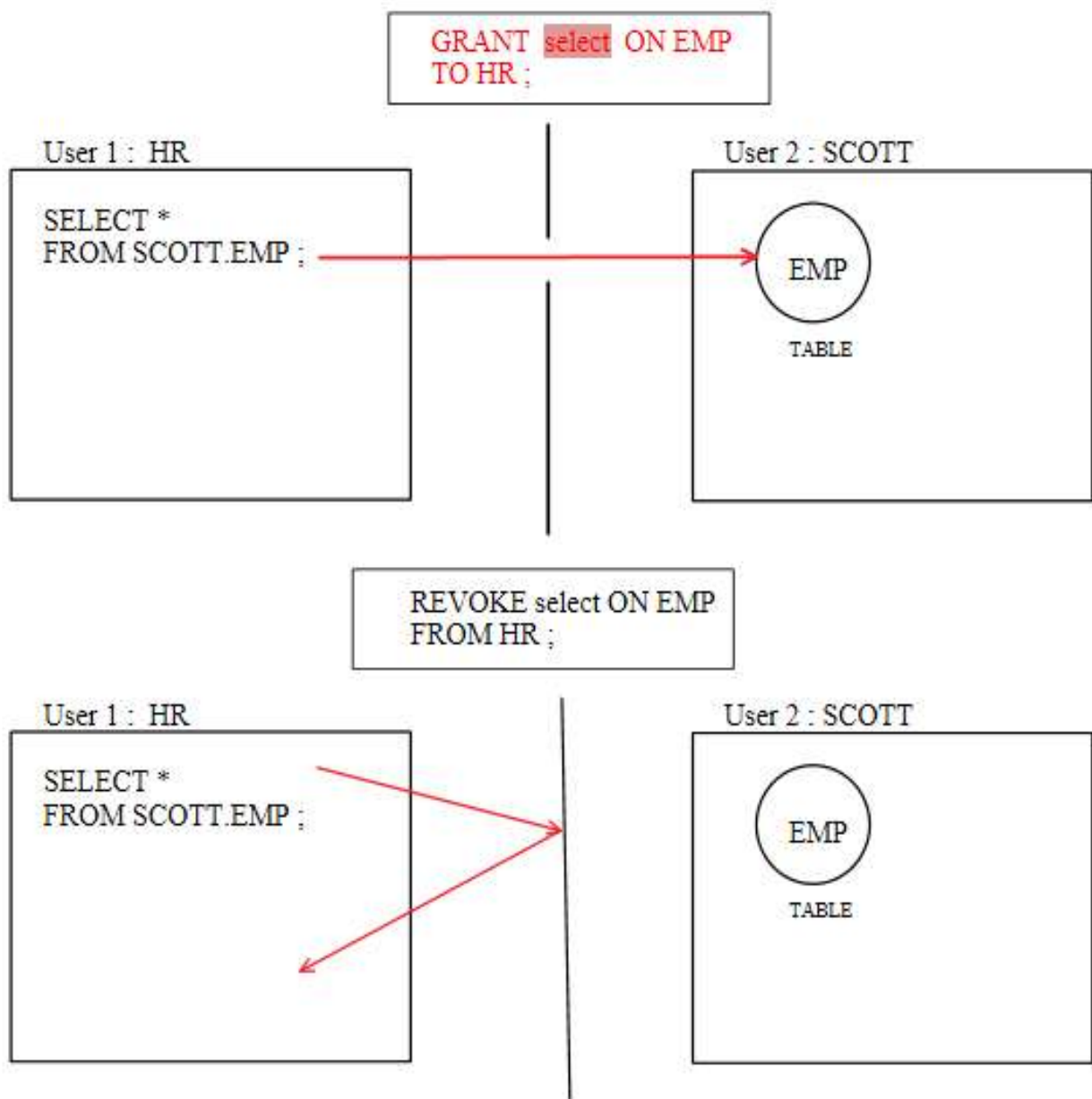
IS used to control the data flow between the users .

1. GRANT
2. REVOKE

1.GRANT : IT is used to give permission to a user .

Syntax: GRANT sql_stmt ON Table_Name
TO User_Name ;





2.REVOKE : IT is used to take back the permission from a user .

Syntax: REVOKE sql_stmt ON Table_Name
FROM User_Name ;

*EX: REVOKE select ON EMP
FROM HR ;*

EXAMPLE :

```
SQL> CONNECT
Enter user-name: SCOTT
Enter password: *****
Connected.
SQL> GRANT DELETE ON EMP
2 TO HR ;

Grant succeeded.

SQL> CONNECT
Enter user-name: HR
Enter password: *****
Connected.
SQL> SHOW USER
USER is "HR"
SQL> DELETE FROM SCOTT.EMP
2 WHERE ENAME LIKE 'A%';

2 rows deleted.

SQL> SELECT *
2 FROM SCOTT.EMP ;
FROM SCOTT.EMP
*
ERROR at line 2:
ORA-01031: insufficient privileges
```