

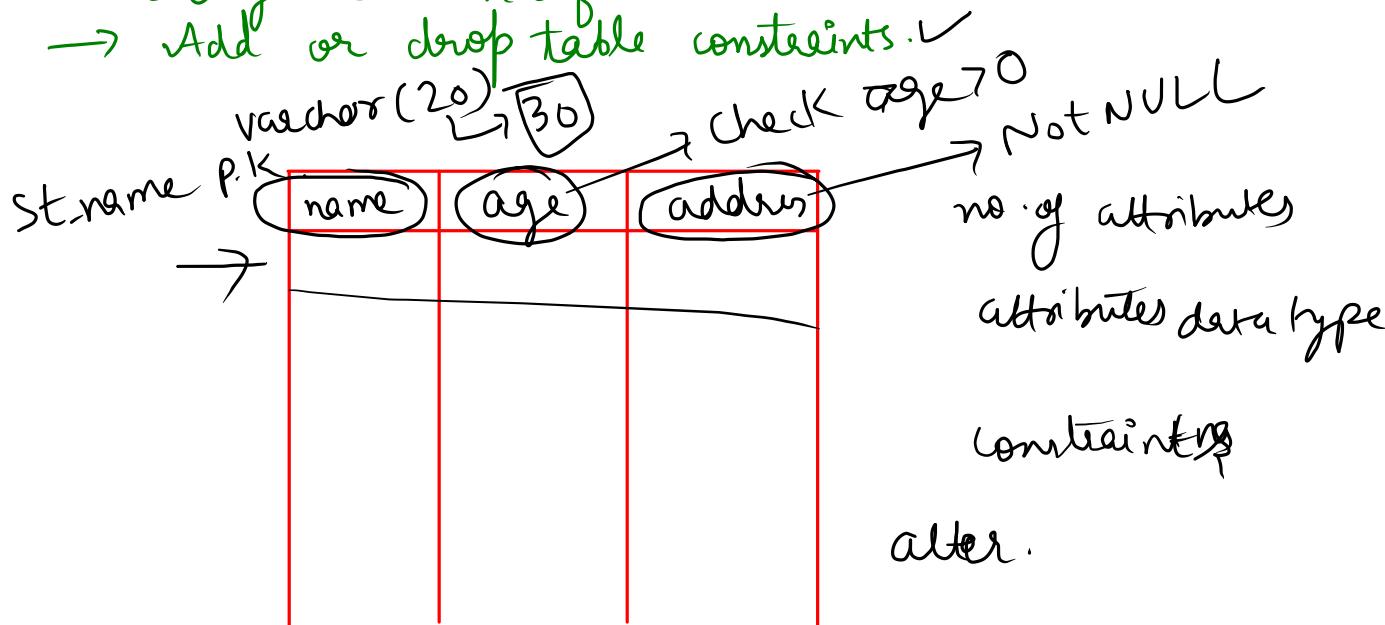
DDL Command 2

Alter Table Command

→ To change the definition of a base table.

features

- Add a column. ✓
- remove a column. ✓
- change column definition. ✓
- Add or drop table constraints. ✓



DDL commands -

~~Create~~
~~Alter~~
~~Drop~~
~~Truncate~~

DML

~~Insert~~
~~update~~
~~delete~~

Table schema = Table

Add a column

Alter table table-name add column-name datatype;

drop a column

Alter table table-name drop column column-name;

change name of column

Alter table table-name rename old-col-name to new-col-name;

change datatype of column

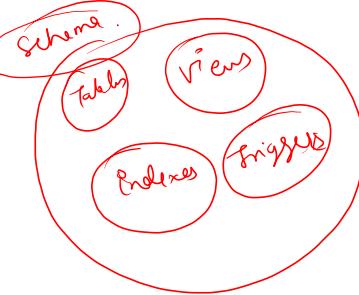
Alter table table-name alter column column-name type new-data-type;

Add a constraint

Alter table table-name add constraint constraint-name;

drop

Alter table table-name drop constraint constraint-name;



Drop table table-name

restrict(default)

Cascade

restrict

data present ✓

for F.K. Constraints X

all other constraints

data ✓

F.K. constraint ✓

all other constraints

drop

* elements → components used in definition of schema & NOT the data inside tables.

** default → If you have not mentioned explicitly Restrict/Cascade, then by default mode will be Restrict.

D.DL Command 3

Drop command ↗

→ If you try to drop a table, all the values present inside the table will be deleted, constraints present on that table will be deleted & then table too will be deleted.

Drop [schema | Table | constraint] <name> [Cascade | Restrict];

→ Cascade option will create a chain reaction and delete all the elements associated with element.

→ In restrict option, an element is deleted if it is not referenced in any constraints or views. otherwise, drop will not be executed & result in an error.

Truncate

Truncate table student;

Truncate command will delete all the data inside the table.

DDL Command 4

Truncate table table-name;

- Deletes all the values inside the table without altering or deleting schema of the table.

Qn:

Why is Truncate a DDL command & not DML?

Student

name	id	age
A	1	10
B	2	15
BCD	3	20

DML

age = 10
Delete From Student where age % 10 == 0 = truncate

A	1	10
BCD	3	20

Delete From Student =



truncate can't be rolled back
delete can be rolled back

DML Commands



- Insert ✓
- Update ✓
- Delete ✓
- Select ✓

Insert Command

student

name	id	age
abc	2	30

i) Insert into table-name values (v1, v2, v3, ...vn);

* here values should be provided in the order columns inside table are created while table creation.

ii) Insert into table-name (att1, att2, att3 ... - attn)
values (v1, v2, v3, ...vn);

* v1 value will be inserted against att1
v2 " " " " " att2
and so on.

Insert operation will result in error if

- i) datatype mismatch ✓
- ii) constraint violation ✓

* If values for certain columns are not provided while performing insert operation, then database will allocate default value corresponding to that column.

BLOB →



Insert into student values ('abc', 2, 30);

Insert into student (name, age, id)
values ('abc', 30, 2);

// Create table (

) With data (

select

)

Delete Command

→ It is used to remove tuples from a relation.

Syntax :-

Delete FROM table-name
Where column-name condition value ;

Update Command

- It is used to alter already present ^{Imp.} **VALUES** inside a table.

Update table-name
 Set attr1=val1, attr2=val2, attrn=valn
 where attrK condition valueK.

* Can throw an error if we have data type or constraint violation.

Update Student
 Set age = ~~10~~ id age
 Where id > 2 = 0

	name	id	age				
→	A	1	10	10	10	10	10
→	B	2	20	20	20	20	20
→	C	3	30	30	30	30	30
→	D	4	15	15	15	15	15
→	E	5	40	40	40	40	40

Employee

Ename	ESSN	EDNO
A	1234	H
B	6789	M
-	-	-
-	-	-

Department

DNUM	DName	DLOC
H	HR	Both
-	-	-
-	-	-
-	-	-

- i) Set null
- ii) Set default
- iii) cascade

update cascade
 cascade

foreign key (Attr1) referenced table2 (attr2) on delete

- i) NULL
- ii) set default
- iii) cascade

Select Command

```
SELECT attr1, attr2, attr3, ...  
FROM tablename  
WHERE attr condition attr2 ... ;
```

attr1	attr2	attr3	attrn

from

Where

Select

1. 'Select' and 'From' are mandatory, however 'Where' is an optional clause.

2. Select * → This will output all the columns present in table.

3. Select *
From student, marks;
table1, table2 ...
→ Cartesian product.

4. If 2 tables in join operation have same attribute names, then we use below

a) Select student.name, student.age, marks.name, marks.age
from student, name
where student.name = marks.name and student.age = marks.age;

b) Select S.name, S.age, M.name, M.age
FROM Student as S, Marks as M
where S.name = M.name and S.age = M.age;

c) Select S.name, S.age, M.name, M.age
FROM Student S, Marks M
where S.name = M.name and S.age = M.age;

Aliasing is used to resolve ambiguity.

* Aliasing can be used in Select clause as well.

```
Select S.name as name_of_student, M.marks as marks_of_student  
From Student S, Marks M  
where S.name = M.name and S.age = M.age;
```

SQL treats table not as a set but as a multiset.

→ Duplicate entries are allowed in tables, views, query results.

⇒ An SQL table with a key is restricted to being a set.

Select DISTINCT keyword is used to eliminate duplicate tuples from the result set.

Select ALL keyword is used to select all the tuples from the result set.

Select
From All Salary
Employee;

Select Distinct Salary
from Employee;