

HW.

1. Create database ,delet
2. Create table ,delet
3. Create columns ,delet

-
1. If we wanna create database write below command:
CREATE DATABASE (name of our data;)
DROP DATABASE (name of our data ;)
 2. If we wanna create table and columns write below command:
CREATE TABLE (name of table)(name of column + data type + constraints if there is)
Or if we wanna delete write
DROP TABLE (name of table;)
DROP COLUMN (name of column ;)

```
postgres=#  
postgres=# CREATE DATABASE university;  
ERROR: database "university" already exists  
postgres=# CREATE TABLE students(  
postgres(# id BIGSERIAL NOT NULL PRIMARY KEY,  
postgres(# first_name VARCHAR(50) NOT NULL,  
postgres(# last_name VARCHAR(50) NOT NULL,  
postgres(# gender VARCHAR(8) NOT NULL,  
postgres(# check_registration VARCHAR(50) NOT NULL,  
postgres(# check_graduation VARCHAR(50) NOT NULL,  
postgres(# identity_card VARCHAR(50) NOT NULL,  
postgres(# year INT NOT NULL,  
postgres(# date_of_birth DATE NOT NULL);  
CREATE TABLE  
postgres=# \d students
```

Column	Type	Collation	Nullable	Default
id	bigint		not null	nextval('students_id_seq')
first_name	character varying(50)		not null	
last_name	character varying(50)		not null	
gender	character varying(8)		not null	
check_registration	character varying(50)		not null	
check_graduation	character varying(50)		not null	
identity_card	character varying(50)		not null	
year	integer		not null	
date_of_birth	date		not null	

```
Indexes:  
"students_pkey" PRIMARY KEY, btree (id)
```

3. INSERT INTO table name like students (first_name,last_name,check_registration ,check_graduation,gender,date_of_birth)
4. VALUES ('suha','Mohammed','FEMALE', '2017' , '2020' , date '1998-06-16')

```

"students_pkey" PRIMARY KEY, btree (id)

postgres=# INSERT INTO students(first_name,last_name,gender,check_registration,check_graduation,identity_card,year,date_of_birth)
postgres=# VALUES ('Suha','Mohammed','FEMALE','twenty siventen','twenty twenty','passport','2021',date '1998-16-06')
postgres=#
postgres=# VALUES ('Suha','Mohammed','FEMALE','twenty siventen','twenty twenty','passport','2021',date '1998-16-06');
ERROR: syntax error at or near "VALUES"
LINE 3: VALUES ('Suha','Mohammed','FEMALE','twenty siventen','twenty...
               ^
postgres=# INSERT INTO students (first_name,last_name,gender,check_registration,check_graduation,identity_card,year,date_of_birth)
postgres=# VALUES ('Suha','Mohammed','FEMALE','twenty siventen','twenty twenty','passport','2021',date '1998-16-06');
ERROR: date/time field value out of range: "1998-16-06"
LINE 2: ... siventen','twenty twenty','passport','2021',date '1998-16-0...
               ^
HINT: Perhaps you need a different "datestyle" setting.
postgres=#
postgres=# \d students

```

Column	Type	Table "public.students"	collation	Nullable	Default
id	integer			not null	nextval('students_id_seq'::regclass)

5. I can insert several data via write one after one and isolated their by comma (,) but the last one should write ; in the end

```

CREATE TABLE
postgres=# INSERT INTO students(full_name,gender,year)
postgres=# VALUES('saja mohammed jasim','female','2005'),
postgres=# ('retag mohanad mohammed','female','2009'),
postgres=# ('Aldus Mohanad Mohammed','female','2014');
INSERT 0 3
postgres=# \d students;
postgres=# SELECT * FROM students;

```

id	full_name	gender	year
1	saja mohammed jasim	female	2005
2	retag mohanad mohammed	female	2009
3	Aldus Mohanad Mohammed	female	2014

(3 rows)

- 6.To add column to my table I should write commands below:

#ALTER TABLE students * (I didn't write semicolon because I am not complete there are a bunch of commands)

#ADD COLUMN like date_of_birth TIMESTAMP; *(data type is TIMESTAMP here)

```

postgres=# ALTER TABLE students
postgres=# ADD COLUMN phone_no INT NOT NULL;
ERROR: column "phone_no" of relation "students" contains null values
postgres=# ALTER TABLE students
postgres=# ADD COLUMN phone_no INT;
ALTER TABLE
postgres=# VALUES('0780246853');
column1
-----
0780246853
(1 row)

postgres=#
postgres=# SELECT * FROM students;

```

id	full_name	gender	year	date of birth	phone
----	-----------	--------	------	---------------	-------

7. We can edit our data type through :

```
UPDATE 0
postgres=# \d students;
postgres=# ALTER TABLE students ALTER COLUMN date_of_birth TYPE TIMESTAMP;
ALTER TABLE
postgres=# \D students
invalid command \D
Try \? for help.
postgres=# \d students
postgres=# INSERT INTO students(first_name,last_name,gender)
postgres=# VALUES('saja','mohammed','female');
ERROR: null value in column "last_name" violates not-null constraint
```

```
postgres=# INSERT INTO students(full_name,gender,year,date_of_birth)
postgres=# VALUES('Noor AL-khatib','female','2021',DATE '1998-11-13');
INSERT 0 1
postgres=# SELECT * FROM students;
 id |      full_name      | gender | year |      date_of_birth
-----+-----+-----+-----+-----
  1 | saja mohammed jasim | female | 2005 | 
  2 | retag mohanad mohammed | female | 2009 | 
  3 | Aldus Mohanad Mohammed | female | 2014 | 
  4 | Noor AL-khatib      | female | 2021 | 1998-11-13 00:00:00
(4 rows)
```

8. we can use SELECT to editing any table or column we have or we can search about specific data we want it to check :

```
postgres=#
postgres=# INSERT INTO teacher(f_name,l_name,level)
VALUES('Reyam','Enad','AI Expert')
VALUES('Reyam','Enad','AI Expert');
ERROR: syntax error at or near "VALUES"
LINE 3: VALUES('Reyam','Enad','AI Expert');
      ^
postgres=# INSERT INTO teacher(f_name,l_name,level)
postgres=# VALUES('Reyam','Enad','AI Expert');
INSERT 0 1
postgres=# INSERT INTO teacher(f_name,l_name,level)
postgres=# VALUES('Rasha','Mohammed','offcer Eng');
INSERT 0 1
postgres=# \d teacher
postgres=# SELECT * FROM teacher;
 id | f_name | l_name | level
-----+-----+-----+-----
  1 | suha   | mohammed | ph dr
  2 | Reyam  | Enad    | AI Expert
  3 | Rasha  | Mohammed | offcer Eng
(3 rows)
```

9-connect to my database UNI:

\c name of database

```
es Terminal
suhamohammed@userhp:~$ sudo su - postgres
[sudo] password for suhamohammed:
postgres@userhp:~$ psql
psql (13.4 (Ubuntu 13.4-1.pgdg20.04+1))
Type "help" for help.

postgres=# \c uni
You are now connected to database "uni" as user "postgres".
uni=#
```

10- Create table with FK write :

- Name of table_id datatype references name of table(id)

```
uni=# create table students(
uni(# id bigserial primary key not null,
uni(# full_name varchar(60) not null,
uni(# tell_no bigint not null,
uni(# reg_date bigint not null,
uni(# grad_date bigint not null,
uni(# identity varchar(70) not null,
uni(# year int not null,
uni(# gender varchar(8) not null,
uni(# naterial_id bigint references naterial (id),
uni(# teacher_id bigint references teacher (id),
uni(# study_id bigint references study (id));
CREATE TABLE
uni=# create table finance(
uni(# id bigserial primary key not null,
uni(# type varchar(70) not null,
uni(# date_time timestamp not null);
CREATE TABLE
uni=# create table classroom (
uni(# id bigserial primary key not null,
uni(# room_type varchar(50) not null,
uni(# study_id bigint references study (id),
uni(# naterial_id bigint references naterial (id),
uni(# teacher_id bigint references teacher (id),
uni(# study_id bigint references study (id));
ERROR: column "study_id" specified more than once
```

11- add relationship in completed table write :

- Alter table name of table
- Add name of table_id datatype references name of table(id)

```
es Terminal
suhamohammed@userhp:~$ sudo su - postgres
postgres@userhp:~$ psql
psql (13.4 (Ubuntu 13.4-1.pgdg20.04+1))
Type "help" for help.

postgres=# \c uni
You are now connected to database "uni" as user "postgres".
uni=# alter table material
uni=# add study_id bigint references study (id),
uni=# add course_id bigint references course (id),
uni=# add department_id bigint references department (id);
ALTER TABLE
uni=#
```

* After adding FK :

```
es Terminal
suhamohammed@userhp:~$ sudo su - postgres
postgres@userhp:~$ psql
psql (13.4 (Ubuntu 13.4-1.pgdg20.04+1))
Type "help" for help.

postgres=# \c uni
You are now connected to database "uni" as user "postgres".
uni=# alter table material
uni=# add study_id bigint references study (id),
uni=# add course_id bigint references course (id),
uni=# add department_id bigint references department (id);
ALTER TABLE
uni=# select * from material;
 id | name | date_time | study_id | course_id | department_id
-----+-----+-----+-----+-----+-----
(0 rows)

uni=#
```


12- this is my tables inside database uni:

```
suhamohammed@userhp:~$ sudo su - postgres
[sudo] password for suhamohammed:
Sorry, try again.
[sudo] password for suhamohammed:
postgres@userhp:~$ psql
psql (13.4 (Ubuntu 13.4-1.pgdg20.04+1))
Type "help" for help.

postgres=# \c uni
You are now connected to database "uni" as user "postgres".
uni=# \dt
          List of relations
Schema |      Name      | Type  | Owner
-----+-----+-----+-----
public | build           | table | postgres
public | classes         | table | postgres
public | classroom       | table | postgres
public | course          | table | postgres
public | department      | table | postgres
public | finance         | table | postgres
public | image           | table | postgres
public | naterial        | table | postgres
public | registration     | table | postgres
public | students        | table | postgres
public | study           | table | postgres
public | teacher         | table | postgres
public | university      | table | postgres
(13 rows)

uni=#
```

