/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. create placementdb
2. create 4 schemas under that:
3. loader\_schema
4. user\_schema
5. job\_schema
6. role\_function\_schema

3) execute the the scripts below

4) under loader\_schema:

loader

5) under user\_schema:

userdetails, personal\_profile, professional\_profile, qualification

professional\_profile\_qualification, professional\_profile\_job

6) under job\_schema:

criteria, company, job, company\_job

7) under role\_function\_schema

role\_master, function\_master, role\_function\_map

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

CREATE TABLE loader\_schema.loader

(

roll\_no text NOT NULL,

name text,

branch text,

gender text,

dob text,

mobile text,

email text,

ssc\_per text,

hsc\_or\_dip text,

hsc\_or\_dip\_per text,

s1\_mark text,

s1\_max text,

s1\_per text,

s1\_res text,

s1\_kt text,

s2\_mark text,

s2\_max text,

s2\_per text,

s2\_res text,

s2\_kt text,

s3\_mark text,

s3\_max text,

s3\_per text,

s3\_res text,

s3\_kt text,

s4\_mark text,

s4\_max text,

s4\_per text,

s4\_res text,

s4\_kt text,

s5\_mark text,

s5\_max text,

s5\_per text,

s5\_res text,

s5\_kt text,

s6\_mark text,

s6\_max text,

s6\_per text,

s6\_res text,

s6\_kt text,

s7\_mark text,

s7\_max text,

s7\_per text,

s7\_res text,

s7\_kt text,

s8\_mark text,

s8\_max text,

s8\_per text,

s8\_res text,

s8\_kt text,

deg\_per text,

deg\_kt text,

drops text,

CONSTRAINT loader\_pkey PRIMARY KEY (roll\_no)

)

WITH (

OIDS=FALSE

);

ALTER TABLE loader\_schema.loader

OWNER TO postgres;

CREATE TABLE user\_schema.userdetails

(

username text NOT NULL,

user\_password text,

account\_active text,

current\_state text,

last\_login timestamp without time zone,

created\_date date,

created\_by text,

modified\_by text,

modified\_date date,

role\_id text,

CONSTRAINT "UserDetails\_pkey" PRIMARY KEY (username),

CONSTRAINT userdetails\_role\_id\_fkey FOREIGN KEY (role\_id)

REFERENCES role\_function\_schema.role\_master (role\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE user\_schema.userdetails

OWNER TO postgres;

CREATE TABLE user\_schema.personal\_profile

(

username text NOT NULL,

name text,

gender text,

dob text,

mobile\_no text,

email\_id text,

category text,

created\_date date,

created\_by text,

modified\_date date,

modified\_by text,

correspondence\_address text,

permanent\_address text,

family\_income text,

CONSTRAINT personal\_profile\_pkey PRIMARY KEY (username),

CONSTRAINT personal\_profile\_username\_fkey FOREIGN KEY (username)

REFERENCES user\_schema.userdetails (username) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE user\_schema.personal\_profile

OWNER TO postgres;

CREATE TABLE user\_schema.professional\_profile

(

username text NOT NULL,

branch text,

status text,

field\_of\_interest text,

achievements text,

projects text,

internships text,

co\_curricular\_activities text,

extra\_curricular\_activities text,

created\_date date,

created\_by text,

modified\_date date,

modified\_by text,

year text,

work\_experience text,

degree\_percentage text,

degree\_kts text,

degree\_drops text,

CONSTRAINT professional\_profile\_pkey PRIMARY KEY (username),

CONSTRAINT professional\_profile\_username\_fkey FOREIGN KEY (username)

REFERENCES user\_schema.userdetails (username) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE user\_schema.professional\_profile

OWNER TO postgres;

CREATE TABLE user\_schema.qualification

(

username text NOT NULL,

created\_date date,

created\_by text,

modified\_date date,

modified\_by text,

result text,

no\_of\_kts text,

qualification\_id text NOT NULL,

percentage text,

gpa text,

total\_marks text,

marks\_obtained text,

CONSTRAINT qualification\_pkey PRIMARY KEY (qualification\_id, username),

CONSTRAINT qualification\_username\_fkey FOREIGN KEY (username)

REFERENCES user\_schema.userdetails (username) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT qualification\_qualification\_id\_key UNIQUE (qualification\_id)

)

WITH (

OIDS=FALSE

);

ALTER TABLE user\_schema.qualification

OWNER TO postgres;

CREATE TABLE user\_schema.professional\_profile\_qualification

(

username text NOT NULL,

qualification\_id text NOT NULL,

CONSTRAINT professional\_profile\_qualification\_pkey PRIMARY KEY (username, qualification\_id),

CONSTRAINT professional\_profile\_qualification\_qualification\_id\_fkey FOREIGN KEY (qualification\_id)

REFERENCES user\_schema.qualification (qualification\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT professional\_profile\_qualification\_username\_fkey FOREIGN KEY (username)

REFERENCES user\_schema.professional\_profile (username) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE user\_schema.professional\_profile\_qualification

OWNER TO postgres;

CREATE TABLE user\_schema.professional\_profile\_job

(

username text NOT NULL,

job\_id text NOT NULL,

CONSTRAINT professional\_profile\_job\_pkey PRIMARY KEY (username, job\_id),

CONSTRAINT professional\_profile\_job\_job\_id\_fkey FOREIGN KEY (job\_id)

REFERENCES job\_schema.job (job\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT professional\_profile\_job\_username\_fkey FOREIGN KEY (username)

REFERENCES user\_schema.professional\_profile (username) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE user\_schema.professional\_profile\_job

OWNER TO postgres;

CREATE TABLE role\_function\_schema.role\_master

(

role\_id text NOT NULL,

role\_name text,

created\_by text,

created\_date date,

modified\_by text,

modified\_date date,

CONSTRAINT role\_master\_pkey PRIMARY KEY (role\_id)

)

WITH (

OIDS=FALSE

);

ALTER TABLE role\_function\_schema.role\_master

OWNER TO postgres;

CREATE TABLE role\_function\_schema.function\_master

(

function\_id text NOT NULL,

function\_name text,

function\_url text,

created\_by text,

created\_date date,

modified\_by text,

modified\_date date,

CONSTRAINT function\_master\_pkey PRIMARY KEY (function\_id)

)

WITH (

OIDS=FALSE

);

ALTER TABLE role\_function\_schema.function\_master

OWNER TO postgres;

CREATE TABLE role\_function\_schema.role\_function\_map

(

role\_id text NOT NULL,

function\_id text NOT NULL,

CONSTRAINT role\_function\_map\_pkey PRIMARY KEY (role\_id, function\_id),

CONSTRAINT role\_function\_map\_function\_id\_fkey FOREIGN KEY (function\_id)

REFERENCES role\_function\_schema.function\_master (function\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT role\_function\_map\_role\_id\_fkey FOREIGN KEY (role\_id)

REFERENCES role\_function\_schema.role\_master (role\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE role\_function\_schema.role\_function\_map

OWNER TO postgres;

CREATE TABLE job\_schema.criteria

(

criteria\_id text NOT NULL,

company\_id text,

eligible\_branches text,

year\_of\_passing text,

placed\_students\_allowed text,

percentage text,

cgpa text,

no\_of\_live\_kts\_allowed text,

no\_of\_dead\_kts\_allowed text,

year\_gap\_allowed text,

ssc\_percentage text,

hsc\_or\_dip\_percentage text,

last\_date\_to\_apply text,

CONSTRAINT "Criteria\_pkey" PRIMARY KEY (criteria\_id),

CONSTRAINT criteria\_company\_id\_fkey FOREIGN KEY (company\_id)

REFERENCES job\_schema.company (company\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE job\_schema.criteria

OWNER TO postgres;

CREATE TABLE job\_schema.company

(

company\_id text NOT NULL,

company\_name text,

company\_address text,

created\_date date,

created\_by text,

modified\_date date,

modified\_by text,

CONSTRAINT company\_pkey PRIMARY KEY (company\_id)

)

WITH (

OIDS=FALSE

);

ALTER TABLE job\_schema.company

OWNER TO postgres;

CREATE TABLE job\_schema.job

(

job\_id text NOT NULL,

event\_id text,

job\_description text,

company\_id text,

job\_category text,

ctc text,

skills\_required text,

docs\_required text,

drive\_date date,

criteria\_id text,

created\_date date,

created\_by text,

modified\_date date,

modified\_by text,

CONSTRAINT job\_pkey PRIMARY KEY (job\_id),

CONSTRAINT job\_company\_id\_fkey FOREIGN KEY (company\_id)

REFERENCES job\_schema.company (company\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT job\_criteria\_id\_fkey FOREIGN KEY (criteria\_id)

REFERENCES job\_schema.criteria (criteria\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE job\_schema.job

OWNER TO postgres;

CREATE TABLE job\_schema.company\_job

(

company\_id text NOT NULL,

job\_id text NOT NULL,

CONSTRAINT company\_job\_pkey PRIMARY KEY (company\_id, job\_id),

CONSTRAINT company\_job\_company\_id\_fkey FOREIGN KEY (company\_id)

REFERENCES job\_schema.company (company\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT company\_job\_job\_id\_fkey FOREIGN KEY (job\_id)

REFERENCES job\_schema.job (job\_id) MATCH SIMPLE

ON UPDATE CASCADE ON DELETE CASCADE

)

WITH (

OIDS=FALSE

);

ALTER TABLE job\_schema.company\_job

OWNER TO postgres;