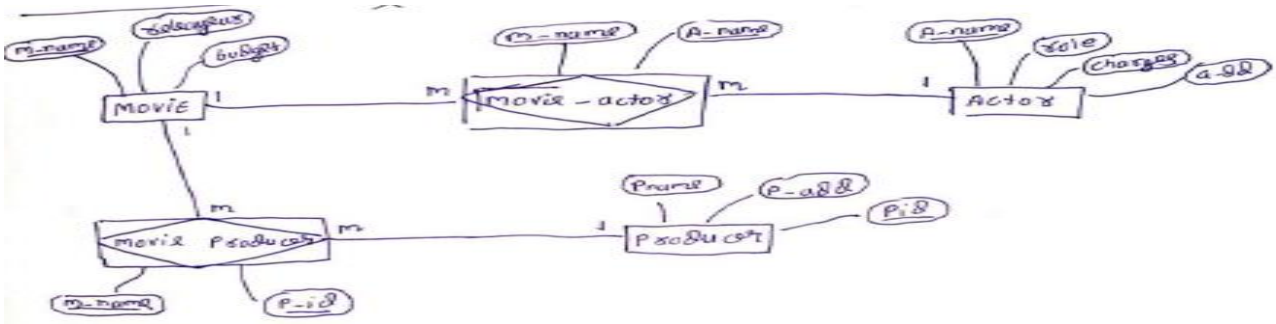
Q1) Practical Questions on PostgresSQL

Movie (M_Name,release_year,budget) Actor (A_name, role,charges,a_address) Producer (producer_id, Name, P_address)

a) **Draw the ER diagram for above relational schema and normalize it in 3NF.**



b) Create the above database in 3NF form in PostgresSQL using constraints.

CREATE TABLE Movie (M_Name VARCHAR(100) PRIMARY KEY, release_year INTEGER NOT NULL, budget NUMERIC NOT NULL);

CREATE TABLE Actor (A_name VARCHAR(100) PRIMARY KEY, role VARCHAR(100) NOT NULL, charges NUMERIC NOT NULL, a_address VARCHAR(100) NOT NULL);

CREATE TABLE Producer (producer_id SERIAL PRIMARY KEY, Name VARCHAR(100) NOT NULL, P_address VARCHAR(100) NOT NULL);

CREATE TABLE Movie_Actor (M_Name VARCHAR(100) REFERENCES Movie, A_name VARCHAR(100) REFERENCES Actor, role VARCHAR(100), PRIMARY KEY (M_Name, A_name, role));

CREATE TABLE Movie_Producer (M_Name VARCHAR(100) REFERENCES Movie, producer_id INTEGER REFERENCES Producer, PRIMARY KEY (M_Name, producer_id));

INSERT INTO Movie (M_Name, release_year, budget) VALUES ('Bahubali', 2015, 100000000), ('Dangal', 2016, 80000000), ('Tanhaji', 2020, 120000000);

INSERT INTO Actor (A_name, role, charges, a_address) VALUES ('Rajkumar Rao', 'Hero', 5000000, 'Mumbai'), ('Deepika Padukone', 'Heroine', 8000000, 'Bengaluru'), ('Ranveer Singh', 'Hero', 6000000, 'Mumbai');

INSERT INTO Producer (Name, P_address) VALUES ('Karan Johar', 'Mumbai'), ('Sajid Nadiadwala', 'Mumbai'), ('Ajay Devgn', 'Pune');

INSERT INTO Movie_Actor (M_Name, A_name, role) VALUES ('Bahubali', 'Rajkumar Rao', 'Hero'), ('Dangal', 'Deepika Padukone', 'Heroine'), ('Tanhaji', 'Ranveer Singh', 'Hero');

INSERT INTO Movie_Producer (M_Name, producer_id) VALUES ('Bahubali', 1), ('Dangal', 2), ('Tanhaji', 3);

Q2) Using above database, solve the following queries:

a) List the names of movies with the highest budget.

```
SELECT M_Name FROM Movie WHERE budget = (SELECT MAX(budget) FROM Movie);
```

b) List the names of producers who produce the same movie as “Tanhaji”.

```
SELECT p.Name FROM Producer p JOIN Movie_Producer mp ON p.producer_id = mp.producer_id
WHERE mp.M_Name = 'Tanhaji';
```

c) List the names of actors who do not live in Pune or Mumbai city.

```
SELECT DISTINCT A_name FROM Actor WHERE a_address NOT IN ('Pune', 'Mumbai');
```

d) Delete the information of Producers who are living in Pune city.

```
DELETE FROM Movie_Producer WHERE producer_id = 3;
DELETE FROM Producer WHERE P_address = 'Pune';
```