P1 - As part of this assignment, we need to list down all the entities, their attributes and the table structures for the scenario mentioned in the previous slide. You also need to write the SQL queries required to create these tables along with few sample entries. Ensure the tables follow 1NF, 2NF, 3NF and BCNF rules.

The actual Movie Database I want to build: But this DB will have duplicate values and is not normalized:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| movie\_id | movie\_title | genre | view\_type | city | theater | date | show\_time |
| 1 | Jawan | Action | 2D | HYD | Cinepolis Mall A | 12-11-2023 | 3:00 PM |
| 1 | Jawan | Action | 2D | HYD | Cinepolis Mall B | 13-11-2023 | 6:00 PM |
| 1 | Jawan | Action | 2D | HYD | Cinepolis Towli Chowki | 14-11-2023 | 10:00 PM |
| 1 | Jawan | Action | 2D | HYD | INOX | 15-11-2023 | 2:00 PM |
| 1 | Jawan | Action | 2D | BLR | PVR | 15-11-2023 | 4:00 PM |
| 1 | Jawan | Action | 3D | MUM | PVR | 15-11-2023 | 4:00 PM |
| 2 | OPPENHEIMER | Drama | 2D | MUM | IMAX | 12-11-2023 | 3:00 PM |
| 2 | OPPENHEIMER | Drama | 2D | HYD | Cinepolis | 12-11-2023 | 6:00 PM |
| 2 | OPPENHEIMER | Drama | 2D | HYD | INOX | 13-11-2023 | 10:00 PM |
| 2 | OPPENHEIMER | Drama | 2D | BLR | INOX | 13-11-2023 | 2:00 PM |
| 3 | EVIL DEAD | Horror | 2D | BLR | PVR | 14-11-2023 | 4:00 PM |
| 3 | EVIL DEAD | Horror | 3D | KOL | PVR | 15-11-2023 | 4:00 PM |

Normalize the Tables:

We need to follow some normalizing techniques to separate the information in 3 separate Tables.

1. Movie Table
2. Theater Table
3. Show Table

**Movie Table:**

|  |  |  |
| --- | --- | --- |
| movie\_id | movie\_title | genre |
| 1 | Jawan | Action |
| 2 | OPPENHEIMER | Drama |
| 3 | Evil Dead | Horror |
|  |  |  |

Queries:

CREATE TABLE movies(

movie\_id varchar UNIQUE PRIMARY KEY,

movie\_title VARCHAR(50) NOT NULL, genre VARCHAR(50) NOT NULL)

**SELECT \* FROM movies**

**A screenshot of a computer

Description automatically generated**

**Theater Table:**

|  |  |  |
| --- | --- | --- |
| theater\_id | theater\_name | city |
| 1 | Cinepolis Mall A | HYD |
| 2 | Cinepolis Mall B | HYD |
| 3 | Cinepolis Towli Chowki | HYD |
| 4 | INOX | HYD |
| 5 | PVR | BLR |
| 6 | PVR | MUM |
| 7 | IMAX | MUM |
|  |  |  |
|  |  |  |

Queries:

CREATE TABLE theaters (

theather\_id varchar UNIQUE PRIMARY KEY,

theather\_name VARCHAR(50) NOT NULL,

city VARCHAR(50) NOT NULL );

SELECT \* FROM theaters;

**A screenshot of a computer

Description automatically generated**

**Shows Table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| show\_id | theater\_id | movie\_id | date | show\_time | view\_type |
| 1 | 1 | 1 | 12-11-2023 | 3:00 PM | 2D |
| 2 | 2 | 1 | 13-11-2023 | 6:00 PM | 2D |
| 3 | 3 | 1 | 14-11-2023 | 10:00 PM | 2D |
| 4 | 4 | 1 | 15-11-2023 | 2:00 PM | 2D |
| 5 | 5 | 1 | 15-11-2023 | 4:00 PM | 2D |
| 6 | 6 | 1 | 15-11-2023 | 4:00 PM | 2D |
| 7 | 7 | 1 | 15-11-2023 | 4:00 PM | 3D |
| 8 | 8 | 2 | 12-11-2023 | 3:00 PM | 2D |
| 9 | 9 | 2 | 13-11-2023 | 6:00 PM | 2D |
| 10 | 9 | 2 | 13-11-2023 | 10:00 PM | 2D |
| 11 | 10 | 3 | 14-11-2023 | 4:00 PM | 2D |

Queries:

CREATE TABLE shows (

show\_id varchar UNIQUE PRIMARY KEY,

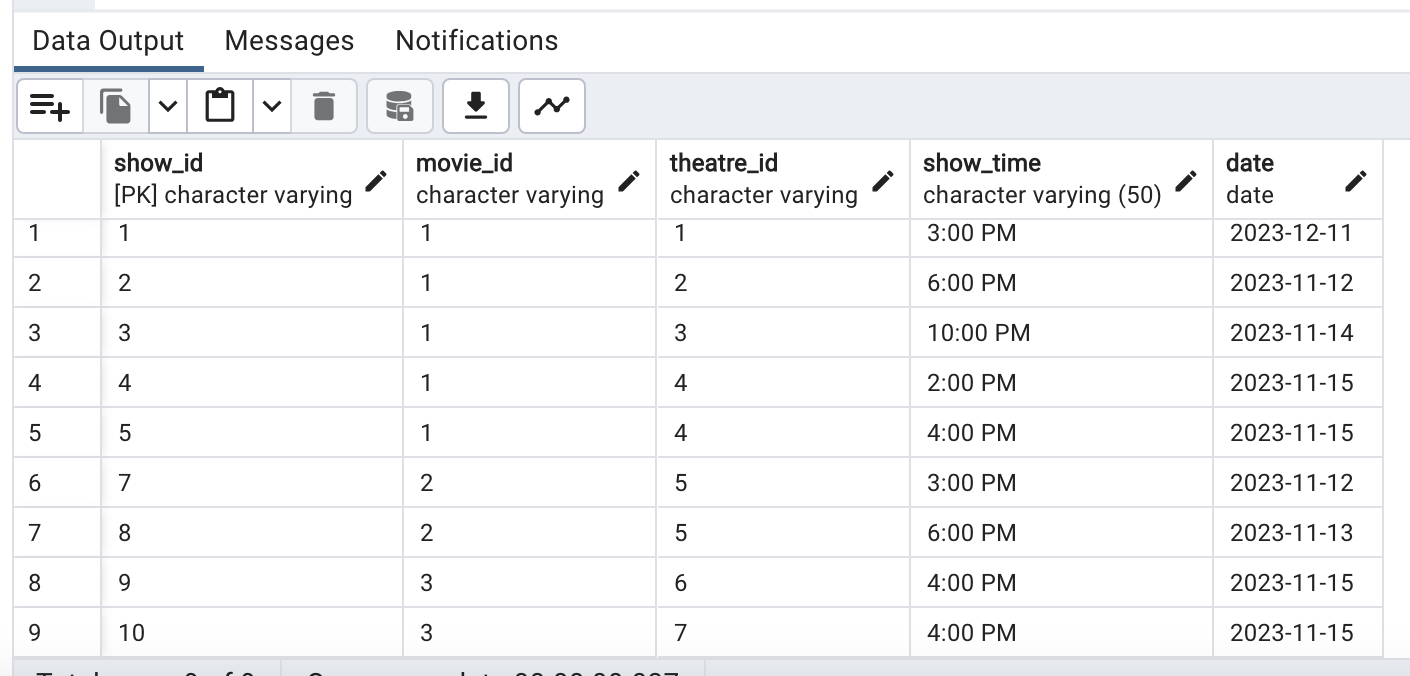
movie\_id varchar REFERENCES movies(movie\_id),

theatre\_id varchar REFERENCES theaters(theather\_id),

show\_time VARCHAR(50) NOT NULL,

date DATE NOT NULL);

SELECT \* FROM shows;



P2 - Write a query to list down all the shows on a given date at a given theatre along with their respective show timings.

Queries:

SELECT movies.movie\_title, shows.show\_time, theaters.theather\_name

FROM shows

JOIN movies ON shows.movie\_id = movies.movie\_id

JOIN theaters ON shows.theatre\_id = theaters.theather\_id

WHERE shows.date = '2023-11-12' and theaters.theather\_id = '2';

A screenshot of a phone

Description automatically generated