QUESTION 1

CREATE DATABASE QUESTION1;

USE QUESTION1;

-- Create Tables

CREATE TABLE Artists (

ArtistId INT PRIMARY KEY,

ArtistName VARCHAR(255)

);

SELECT\* FROM Artists;



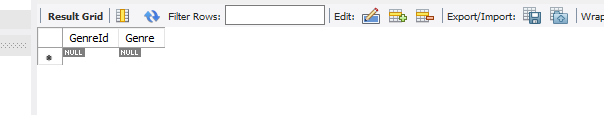
CREATE TABLE Genre (

GenreId INT PRIMARY KEY,

Genre VARCHAR(255)

);

select\* from Genre;



CREATE TABLE Albums (

AlbumId INT PRIMARY KEY,

AlbumName VARCHAR(255),

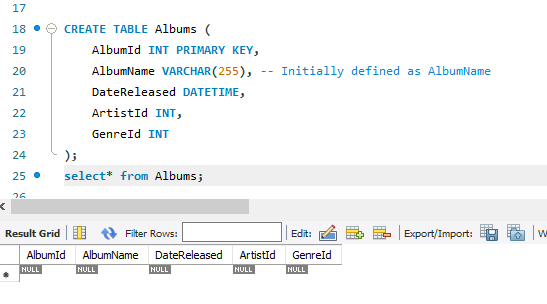
DateReleased DATETIME,

ArtistId INT,

GenreId INT

);

select\* from Albums;



-- Insert Data

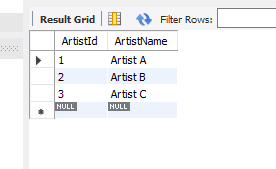
INSERT INTO Artists (ArtistId, ArtistName) VALUES

(1, 'Artist A'),

(2, 'Artist B'),

(3, 'Artist C');

select\* from Artists;



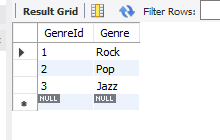
INSERT INTO Genre (GenreId, Genre) VALUES

(1, 'Rock'),

(2, 'Pop'),

(3, 'Jazz');

select\* from Genre;



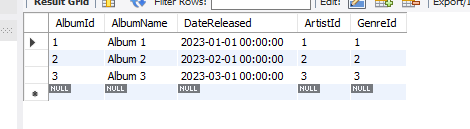
INSERT INTO Albums (AlbumId, AlbumName, DateReleased, ArtistId, GenreId) VALUES

(1, 'Album 1', '2023-01-01', 1, 1),

(2, 'Album 2', '2023-02-01', 2, 2),

(3, 'Album 3', '2023-03-01', 3, 3);

select\* from Albums;

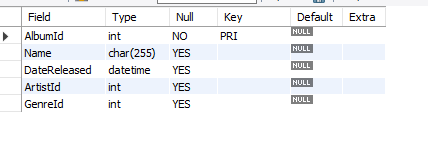


-- Modify Column Name and Datatype

ALTER TABLE Albums

CHANGE AlbumName Name CHAR(255);

show columns from Albums;



-- Add Foreign Keys with ON DELETE and ON UPDATE Actions

ALTER TABLE Albums

ADD CONSTRAINT FK\_Albums\_Artists

FOREIGN KEY (ArtistId) REFERENCES Artists(ArtistId)

ON DELETE SET NULL

ON UPDATE CASCADE,

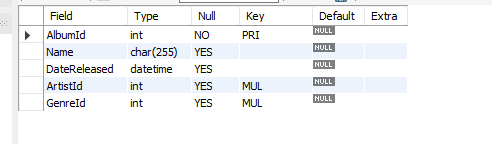
ADD CONSTRAINT FK\_Albums\_Genre

FOREIGN KEY (GenreId) REFERENCES Genre(GenreId)

ON DELETE SET NULL

ON UPDATE CASCADE;

show columns from Albums;



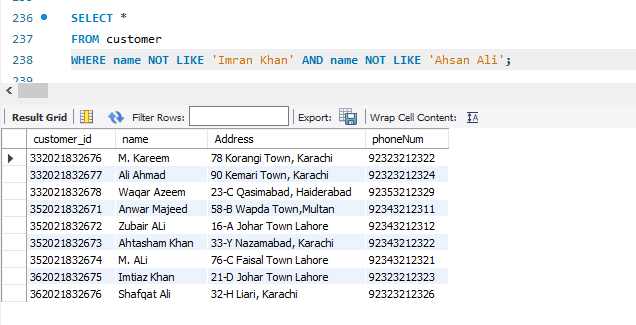
QUESTION 2:

Question 1

SELECT \*

FROM customer

WHERE name NOT LIKE 'Imran Khan' AND name NOT LIKE 'Ahsan Ali';

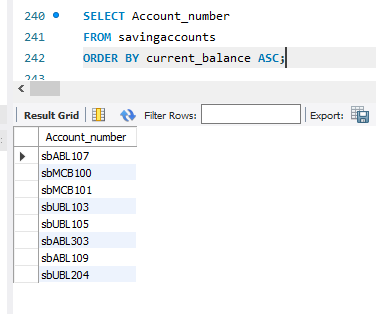


Question 2

SELECT Account\_number

FROM savingaccounts

ORDER BY current\_balance ASC;

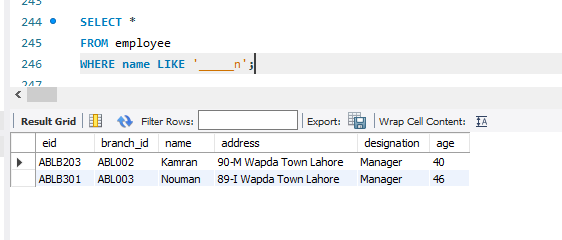


Question 3

SELECT \*

FROM employee

WHERE name LIKE '\_\_\_\_\_n';



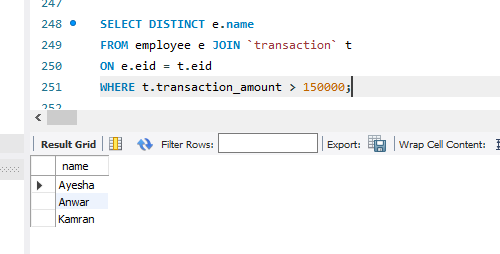
Question 4

SELECT DISTINCT e.name

FROM employee e JOIN `transaction` t

ON e.eid = t.eid

WHERE t.transaction\_amount > 150000;



QUESTION 3

