یک برنامه در زبان SQL که شامل کلمات کلیدی آن است:

-- Create a database called Books

CREATE DATABASE Books;

-- Use the Books database

USE Books;

-- Create a table called Authors with four columns: id, name, country, and birth\_year

CREATE TABLE Authors ( id INT PRIMARY KEY, name VARCHAR(50) NOT NULL, country VARCHAR(50), birth\_year INT );

-- Insert some data into the Authors table

INSERT INTO Authors (id, name, country, birth\_year)

VALUES (1, 'George Orwell', 'UK', 1903), (2, 'Maya Angelou', 'USA', 1928), (3, 'Yuval Noah Harari', 'Israel', 1976),

(4, 'J.K. Rowling', 'UK', 1965), (5, 'Rupi Kaur', 'Canada', 1992);

-- Create a table called Books with five columns: id, title, author\_id, price, and rating

CREATE TABLE Books ( id INT PRIMARY KEY, title VARCHAR(100) NOT NULL, author\_id INT NOT NULL, price DECIMAL(5,2), rating INT CHECK (rating BETWEEN 1 AND 5),

FOREIGN KEY (author\_id) REFERENCES Authors (id) );

-- Insert some data into the Books table

INSERT INTO Books (id, title, author\_id, price, rating)

VALUES (1, '1984', 1, 9.99, 5),

(2, 'Animal Farm', 1, 7.99, 4), (3, 'I Know Why the Caged Bird Sings', 2, 8.99, 5),

(4, 'Sapiens: A Brief History of Humankind', 3, 12.99, 5), (5, 'Harry Potter and the Philosopher''s Stone', 4, 6.99, 5),

(6, 'Milk and Honey', 5, 9.99, 4);

-- Alter the Authors table to add a column called genre

ALTER TABLE Authors ADD genre VARCHAR(50) CHECK (genre IN ('Fiction', 'Non-fiction', 'Poetry'));

-- Select the name and country of the authors who write fiction

SELECT name, country FROM Authors as au WHERE au.genre = 'fiction';

-- Select the title and price of the books that have a rating of 5

SELECT title, price FROM Books WHERE rating = 5;

-- Select the name and title of the authors and their books using a join

SELECT Authors.name, Books.title FROM Authors INNER JOIN Books ON Authors.id = Books.author\_id;

-- Update the price of the book '1984' to 10.99

UPDATE Books SET price = 10.99 WHERE title = '1984';

-- Delete the book 'Milk and Honey' from the Books table

DELETE FROM Books WHERE title = 'Milk and Honey';

-- Alter the Authors table to drop the column birth\_year

ALTER TABLE Authors DROP COLUMN birth\_year;

-- Drop the Books table

DROP TABLE Books;

-- Drop the Books database

DROP DATABASE Books;

-- Added code starts here

-- Create a database called Movies

CREATE DATABASE Movies;

-- Use the Movies database

USE Movies;

-- Create a table called Actors with three columns: id, name, and gender

CREATE TABLE Actors ( id INT PRIMARY KEY, name VARCHAR(50) NOT NULL, gender VARCHAR(10) CHECK (gender IN ('Male', 'Female', 'Other')) );

-- Insert some data into the Actors table

INSERT INTO Actors (id, name, gender) VALUES (1, 'Tom Hanks', 'Male'), (2, 'Meryl Streep', 'Female'),

(3, 'Will Smith', 'Male'), (4, 'Emma Watson', 'Female'), (5, 'Elliot Page', 'Other');

-- Create a table called Movies with four columns: id, title, year, and genre

CREATE TABLE Movies ( id INT PRIMARY KEY, title VARCHAR(100) NOT NULL, year INT, genre VARCHAR(50) );

-- Insert some data into the Movies table

INSERT INTO Movies (id, title, year, genre)

VALUES (1, 'Forrest Gump', 1994, 'Drama'), (2, 'The Devil Wears Prada', 2006, 'Comedy'),

(3, 'Men in Black', 1997, 'Sci-Fi'), (4, 'Beauty and the Beast', 2017, 'Fantasy'), (5, 'Inception', 2010, 'Thriller');

-- Create a table called Casts with three columns: movie\_id, actor\_id, and role

CREATE TABLE Casts ( movie\_id INT, actor\_id INT, role VARCHAR(50), PRIMARY KEY (movie\_id, actor\_id),

FOREIGN KEY (movie\_id) REFERENCES Movies (id), FOREIGN KEY (actor\_id) REFERENCES Actors (id) );

-- Insert some data into the Casts table

INSERT INTO Casts (movie\_id, actor\_id, role) VALUES (1, 1, 'Forrest Gump'), (2, 2, 'Miranda Priestly'),

(3, 3, 'Agent J'), (4, 4, 'Belle'), (5, 5, 'Ariadne');

-- Select the title and genre of the movies that were released after 2000

SELECT title, genre FROM Movies WHERE year > 2000;

-- Select the name and gender of the actors who played in 'Men in Black'

SELECT Actors.name, Actors.gender

FROM Actors INNER JOIN Casts ON Actors.id = Casts.actor\_id WHERE Casts.movie\_id = (SELECT id FROM Movies WHERE title = 'Men in Black');

-- Select the title and role of the movies that Tom Hanks played in

SELECT Movies.title, Casts.role FROM Movies

INNER JOIN Casts ON Movies.id = Casts.movie\_id WHERE Casts.actor\_id = (SELECT id FROM Actors WHERE name = 'Tom Hanks');

-- Select the distinct genres of the movies in the Movies table

SELECT DISTINCT genre FROM Movies;

-- Select the name and count of the movies that each actor played in, grouped by actor name

SELECT Actors.name, COUNT(Casts.movie\_id) AS movie\_count FROM Actors

LEFT JOIN Casts ON Actors.id = Casts.actor\_id GROUP BY Actors.name;

-- Select the name and average rating of the movies that each actor played in, grouped by actor name and ordered by rating in descending order

SELECT Actors.name, AVG(Movies.rating) AS avg\_rating FROM Actors

LEFT JOIN Casts ON Actors.id = Casts.actor\_id LEFT JOIN Movies ON Casts.movie\_id = Movies.id GROUP BY Actors.name ORDER BY avg\_rating DESC;

-- Select the name and gender of the actors who played in more than one movie, having a movie count greater than 1

SELECT Actors.name, Actors.gender FROM Actors LEFT JOIN Casts ON Actors.id = Casts.actor\_id

GROUP BY Actors.name HAVING COUNT(Casts.movie\_id) > 1;

-- Added code ends here

منبع:

* <https://join.skype.com/bot/cf0e6215-34fe-409b-9e4b-135d7f3aa13b>