Contents

[1. Create a database called LusuFlix and add a table called Movies, Customers and rentals 1](#_Toc138767941)

[2. Data Inserted into the tables: 3](#_Toc138767942)

[Create Login For Manager and Cashier: 3](#_Toc138767943)

[3. Grant Permissions to Manager and Cashier 4](#_Toc138767944)

[4. Top Movies: 5](#_Toc138767945)

[5. View Top Customers: 5](#_Toc138767946)

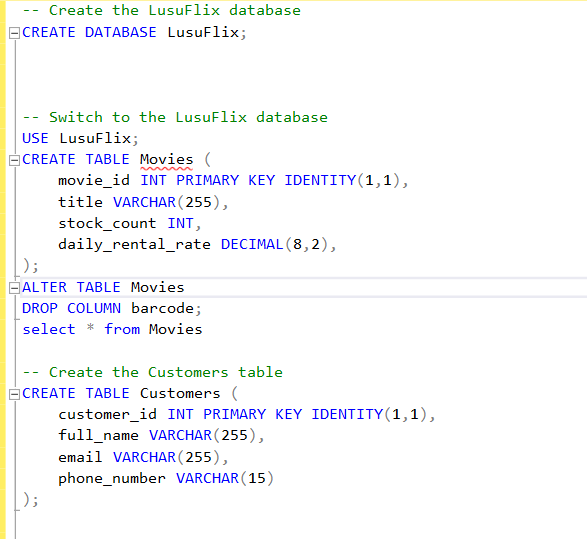
[6. View Top Revenue: 6](#_Toc138767947)

[7. View Monthly Revenue: 6](#_Toc138767948)

[8. View Yearly Revenue: 6](#_Toc138767949)

# 

# Create a database called LusuFlix and add a table called Movies, Customers and rentals



A picture containing text, screenshot, font

Description automatically generated

CREATE DATABASE LusuFlix;

USE LusuFlix;

CREATE TABLE Movies (

movie\_id INT PRIMARY KEY IDENTITY(1,1),

title VARCHAR(255),

stock\_count INT,

daily\_rental\_rate DECIMAL(8,2),

);

ALTER TABLE Movies

DROP COLUMN barcode;

select \* from Movies

-- Create the Customers table

CREATE TABLE Customers (

customer\_id INT PRIMARY KEY IDENTITY(1,1),

full\_name VARCHAR(255),

email VARCHAR(255),

phone\_number VARCHAR(15)

);

-- Create the Rentals table

CREATE TABLE Rentals (

rental\_id INT PRIMARY KEY IDENTITY(1,1),

customer\_id INT,

movie\_id INT,

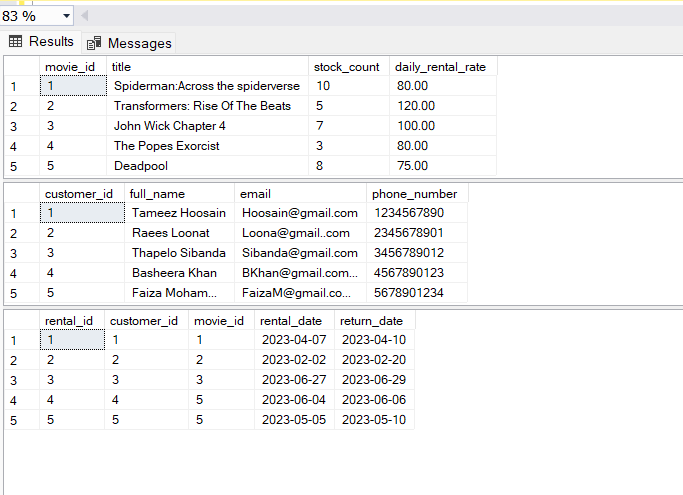
rental\_date DATE,

return\_date DATE,

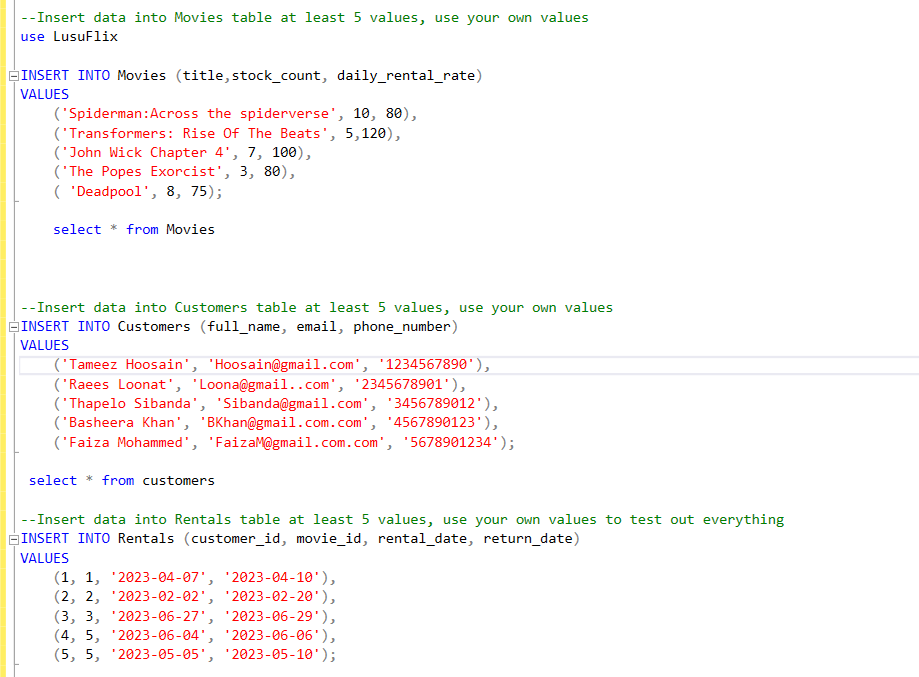
FOREIGN KEY (customer\_id) REFERENCES Customers (customer\_id),

FOREIGN KEY (movie\_id) REFERENCES Movies (movie\_id)

);



# Data Inserted into the tables:



--Insert data into Movies table at least 5 values, use your own values

use LusuFlix

INSERT INTO Movies (title,stock\_count, daily\_rental\_rate)

VALUES

('Spiderman:Across the spiderverse', 10, 80),

('Transformers: Rise Of The Beats', 5,120),

('John Wick Chapter 4', 7, 100),

('The Popes Exorcist', 3, 80),

( 'Deadpool', 8, 75);

select \* from Movies

--Insert data into Customers table at least 5 values, use your own values

INSERT INTO Customers (full\_name, email, phone\_number)

VALUES

('Tameez Hoosain', 'Hoosain@gmail.com', '1234567890'),

('Raees Loonat', 'Loona@gmail..com', '2345678901'),

('Thapelo Sibanda', 'Sibanda@gmail.com', '3456789012'),

('Basheera Khan', 'BKhan@gmail.com.com', '4567890123'),

('Faiza Mohammed', 'FaizaM@gmail.com.com', '5678901234');

select \* from customers

--Insert data into Rentals table at least 5 values, use your own values to test out everything

INSERT INTO Rentals (customer\_id, movie\_id, rental\_date, return\_date)

VALUES

(1, 1, '2023-04-07', '2023-04-10'),

(2, 2, '2023-02-02', '2023-02-20'),

(3, 3, '2023-06-27', '2023-06-29'),

(4, 5, '2023-06-04', '2023-06-06'),

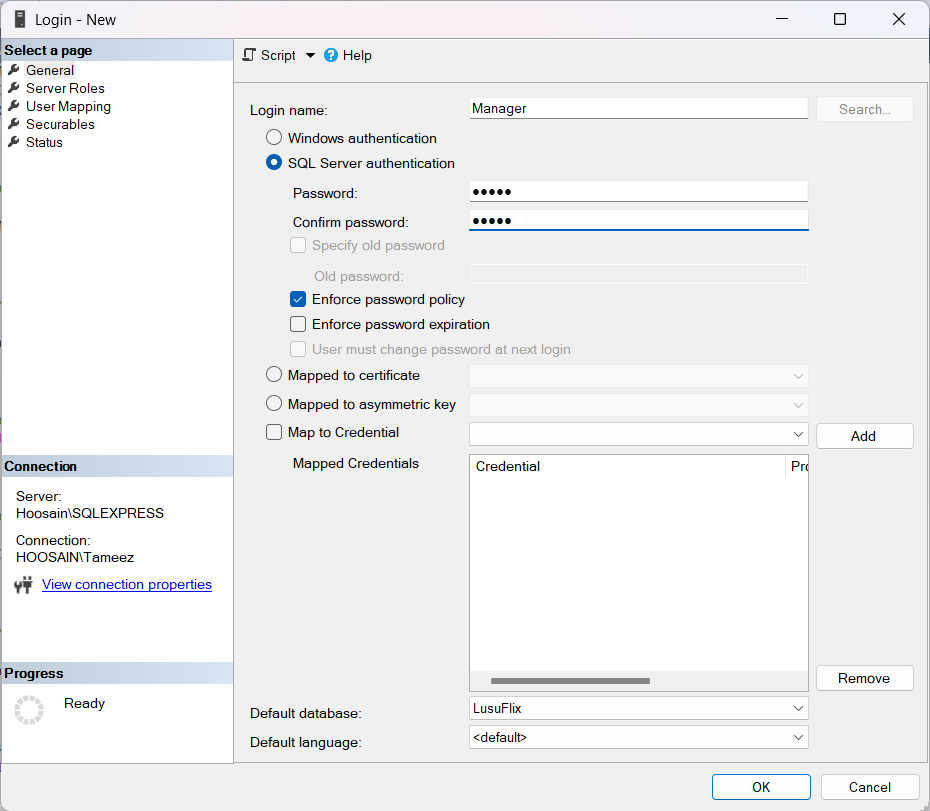
(5, 5, '2023-05-05', '2023-05-10');

Select \* From Movies;

Select \* From Customers;

Select \* From Rentals;

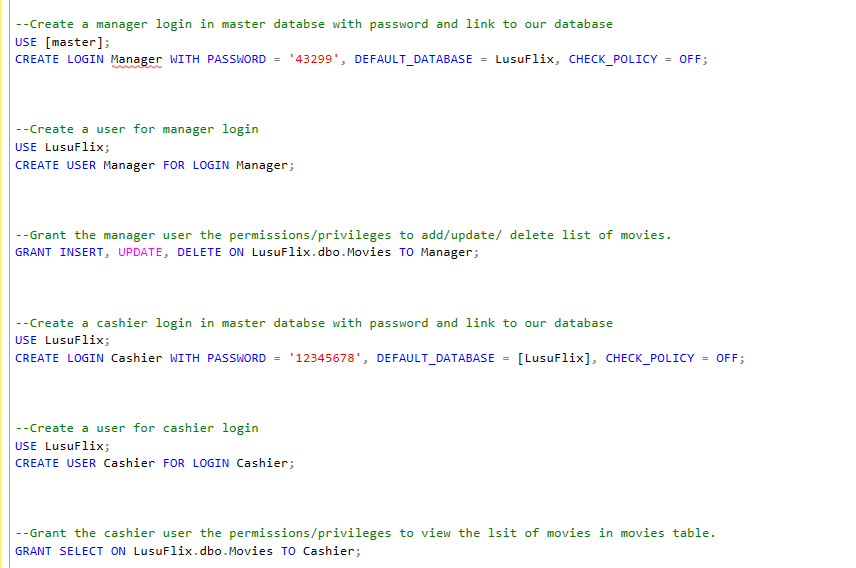
# Create Login For Manager and Cashier:



A screenshot of a computer

Description automatically generated

# Grant Permissions to Manager and Cashier



--Create a manager login in master databse with password and link to our database

USE [master];

CREATE LOGIN Manager WITH PASSWORD = '43299', DEFAULT\_DATABASE = LusuFlix, CHECK\_POLICY = OFF;

--Create a user for manager login

USE LusuFlix;

CREATE USER Manager FOR LOGIN Manager;

--Grant the manager user the permissions/privileges to add/update/ delete list of movies.

GRANT INSERT, UPDATE, DELETE ON LusuFlix.dbo.Movies TO Manager;

--Create a cashier login in master databse with password and link to our database

USE LusuFlix;

CREATE LOGIN Cashier WITH PASSWORD = '12345678', DEFAULT\_DATABASE = [LusuFlix], CHECK\_POLICY = OFF;

--Create a user for cashier login

USE LusuFlix;

CREATE USER Cashier FOR LOGIN Cashier;

--Grant the cashier user the permissions/privileges to view the lsit of movies in movies table.

GRANT SELECT ON LusuFlix.dbo.Movies TO Cashier;

# Top Movies:

A screenshot of a computer program

Description automatically generated with low confidence

--View Top Movies

SELECT TOP 10

m.title,

COUNT(r.movie\_id) AS rental\_count

FROM

Movies m

JOIN Rentals r ON m.movie\_id = r.movie\_id

GROUP BY

m.title

ORDER BY

rental\_count DESC;

# View Top Customers:

A screenshot of a computer program

Description automatically generated with medium confidence

--View Top Customers

SELECT TOP 10

c.full\_name,

COUNT(r.customer\_id) AS rental\_count

FROM

Customers c

JOIN Rentals r ON c.customer\_id = r.customer\_id

GROUP BY

c.full\_name

ORDER BY

rental\_count DESC;

# View Top Revenue:

A picture containing text, screenshot, font, display

Description automatically generated

--View Daily Revenue

SELECT

rental\_date,

SUM(m.daily\_rental\_rate) AS revenue

FROM

Rentals r

JOIN Movies m ON r.movie\_id = m.movie\_id

GROUP BY

rental\_date;

# View Monthly Revenue:

A screenshot of a computer code

Description automatically generated with low confidence

# View Yearly Revenue:

A picture containing text, screenshot, font, display

Description automatically generated

--View Revenue for the year

SELECT

YEAR(rental\_date) AS rental\_year,

SUM(m.daily\_rental\_rate) AS revenue

FROM

Rentals r

JOIN Movies m ON r.movie\_id = m.movie\_id

GROUP BY

YEAR(rental\_date);