

**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGIES**

**FALL 2023**

**CA EXAMINATION**

NAMES: NDE HURICH DILAN

MATRICULE:ICTU20223351

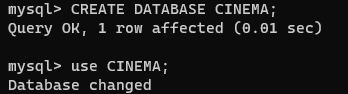
EMAIL: nde.dilan@ictuniversity.edu.cm

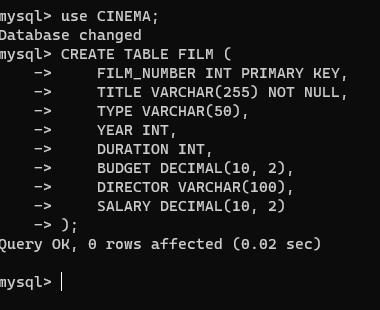
<TEL:+237694525931>

RELATIONAL DATABASE CA

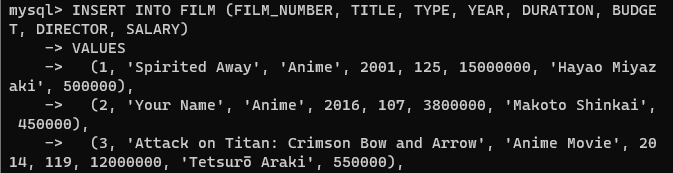
Exercice 1: CINEMA

First let’s create the database we will be working with, and fill it with some dummy data:

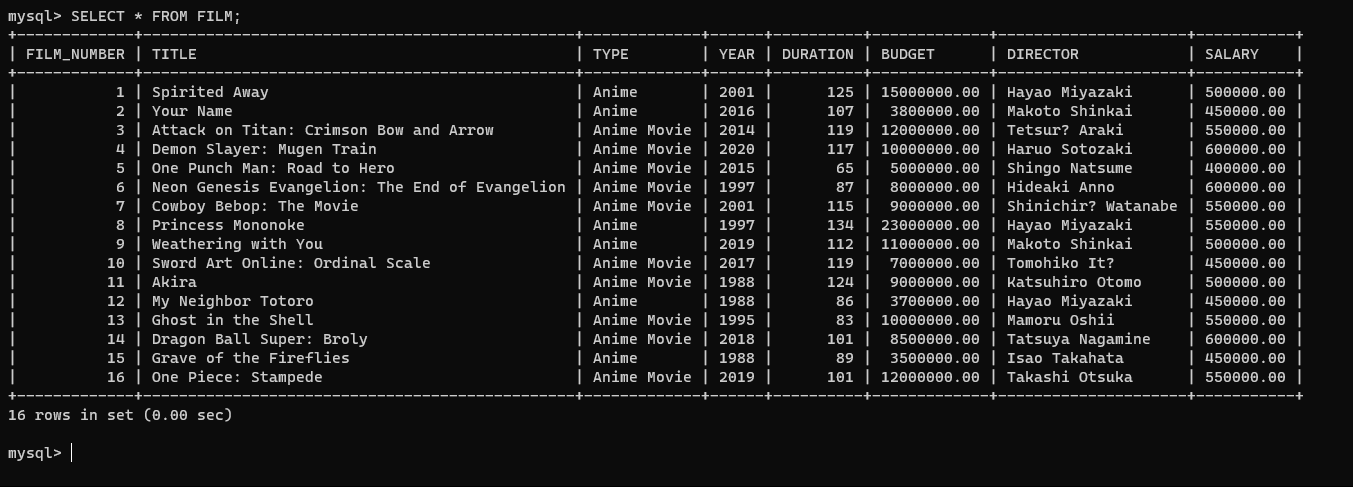




Here we are just creating the **FILM TABLE inside the CINEMA DB**

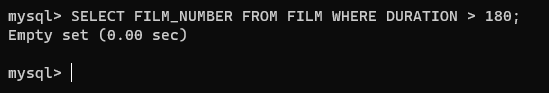


Now we insert some values in the film table(anime movie)

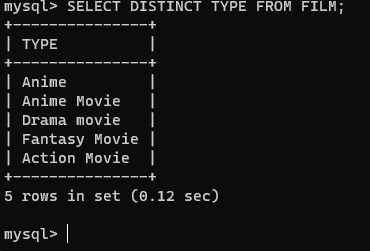
I) Find the list of all the films.

According to the previous table no film exceeds 180 mins

ii) Find the list of films whose length exceeds 180 minutes

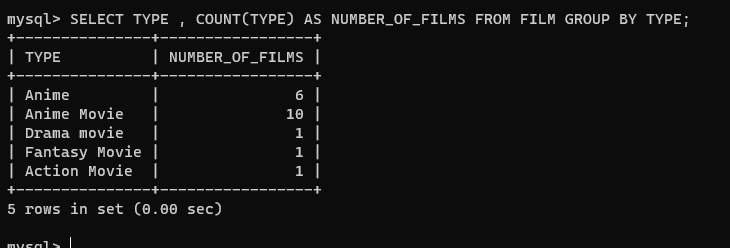


ii) Give the list of all the type of films

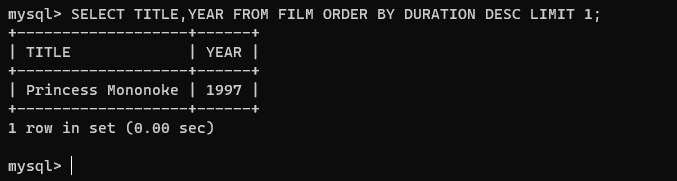


The different type of film present inside the FILM TABLE

iv) Give the number of films by type

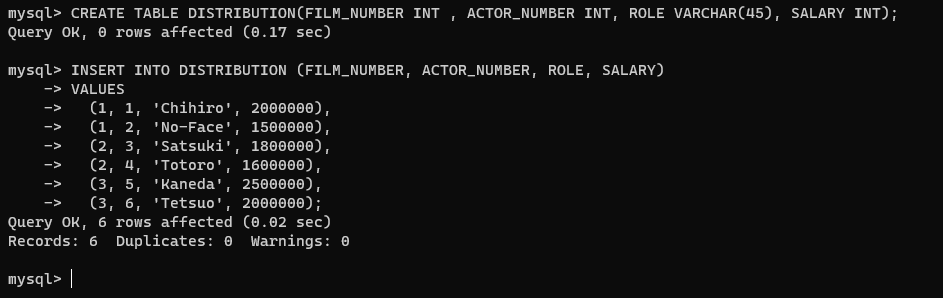


v) Find the title(s) and year(s) of the longest film(s).



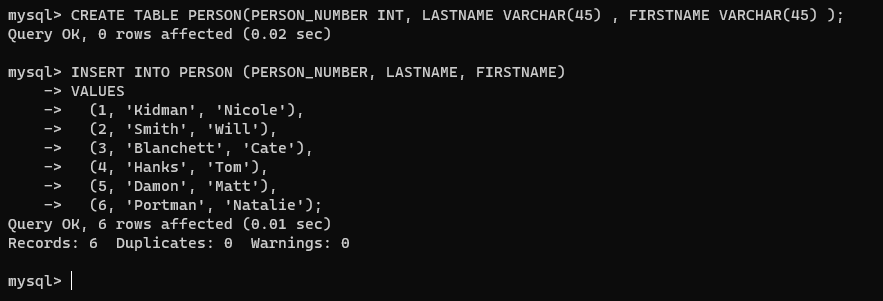
I have to say that here I was confused because of the ‘s’ at the end, but I choose to limit the result to **1.**

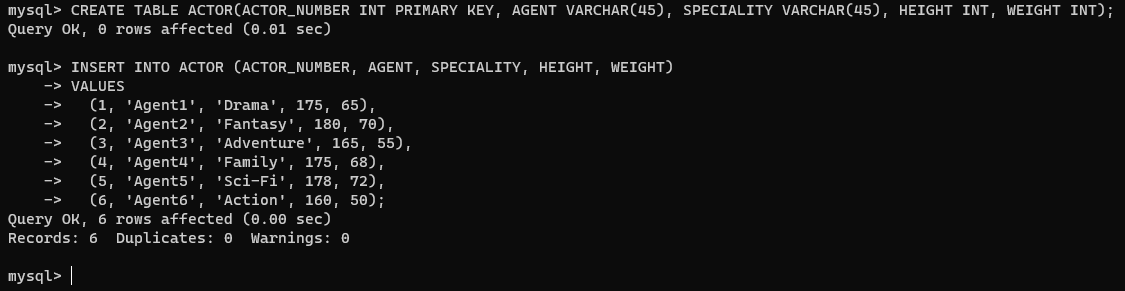
vi) Find all the "pairs of actors", i.e., the actors having played the "Lead" role in the same film.

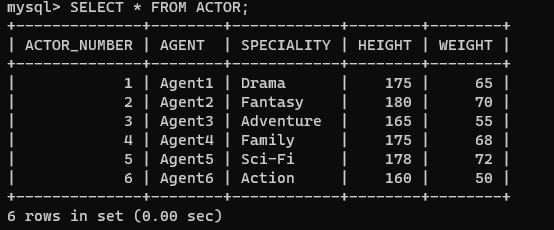


Let’s create and fill the **DISTRIBUTION** table first

Now we can perform operation on the **DISTRIBUTION table.**







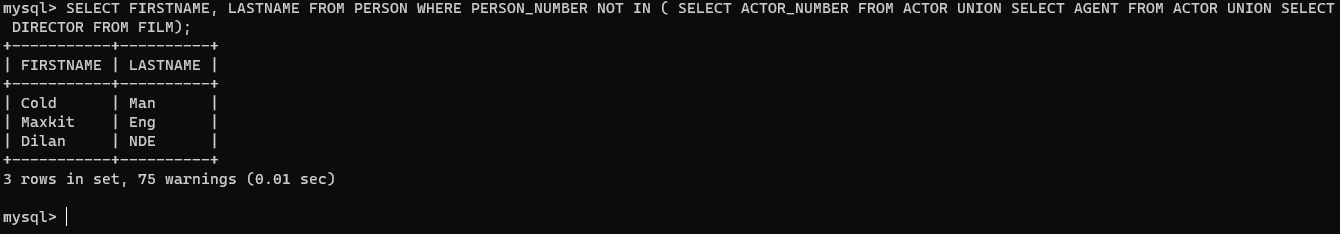


**vii) Find the names of people who are not agents, actors or directors.**

****

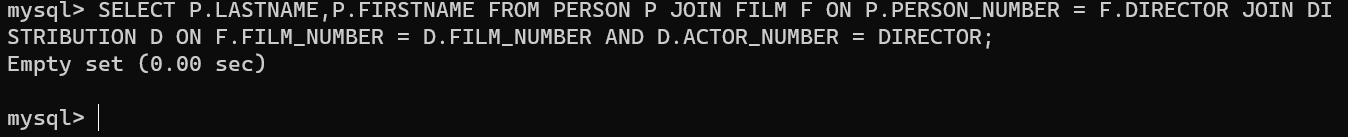
Here is the PERSON table, I’ve added some persons which are not actors(like myself) in the table.

Now we can perform operation on the **PERSON table.**

****

From the result we can see my name(the last😊)

**viii) Give the last name and first name of the directors who have acted in at least one of their own films**

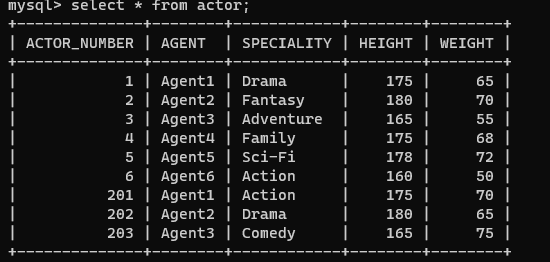
****

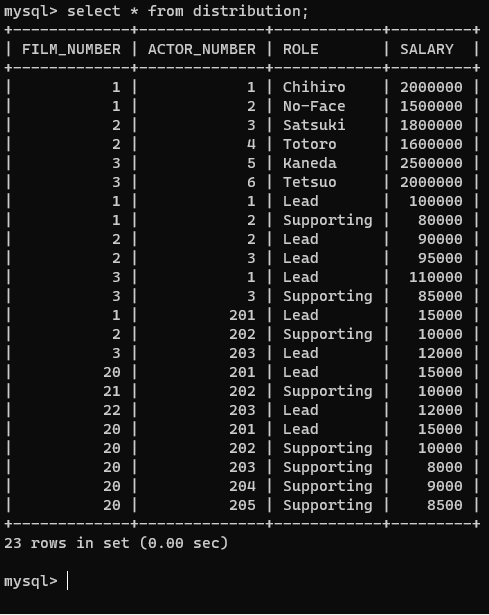
From our previous tables we can understand why it returns an empty set.

**After that I ran the command to feed my db with those dymy data.**

**ix) What is the total salary of the actors of the film "The White House"**

**for this query , I will create 5 actors of that particular film ,a distribution for each actor and the corresponding film which is "The White House";**

****

****

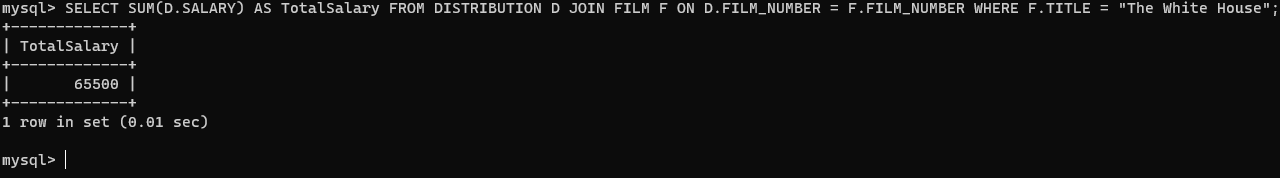
****

**Here is the query**

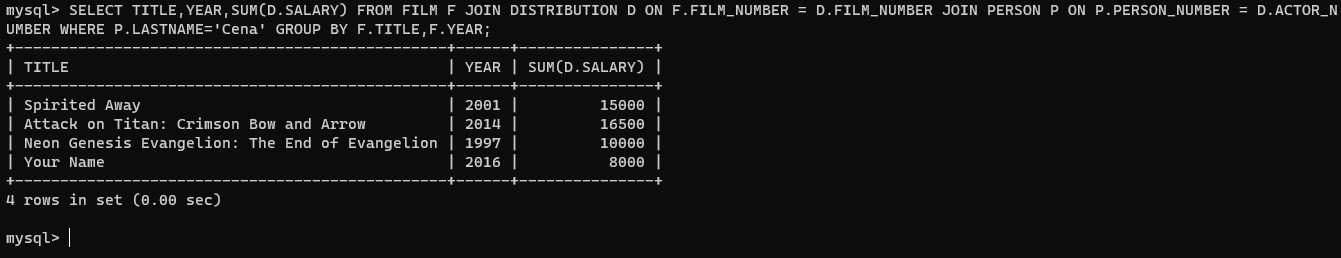
SELECT SUM(SALARY) AS TotalSalary

FROM DISTRIBUTION D

JOIN FILM F ON D.FILM\_NUMBER = F.FILM\_NUMBER WHERE F.TITLE = 'The White House';

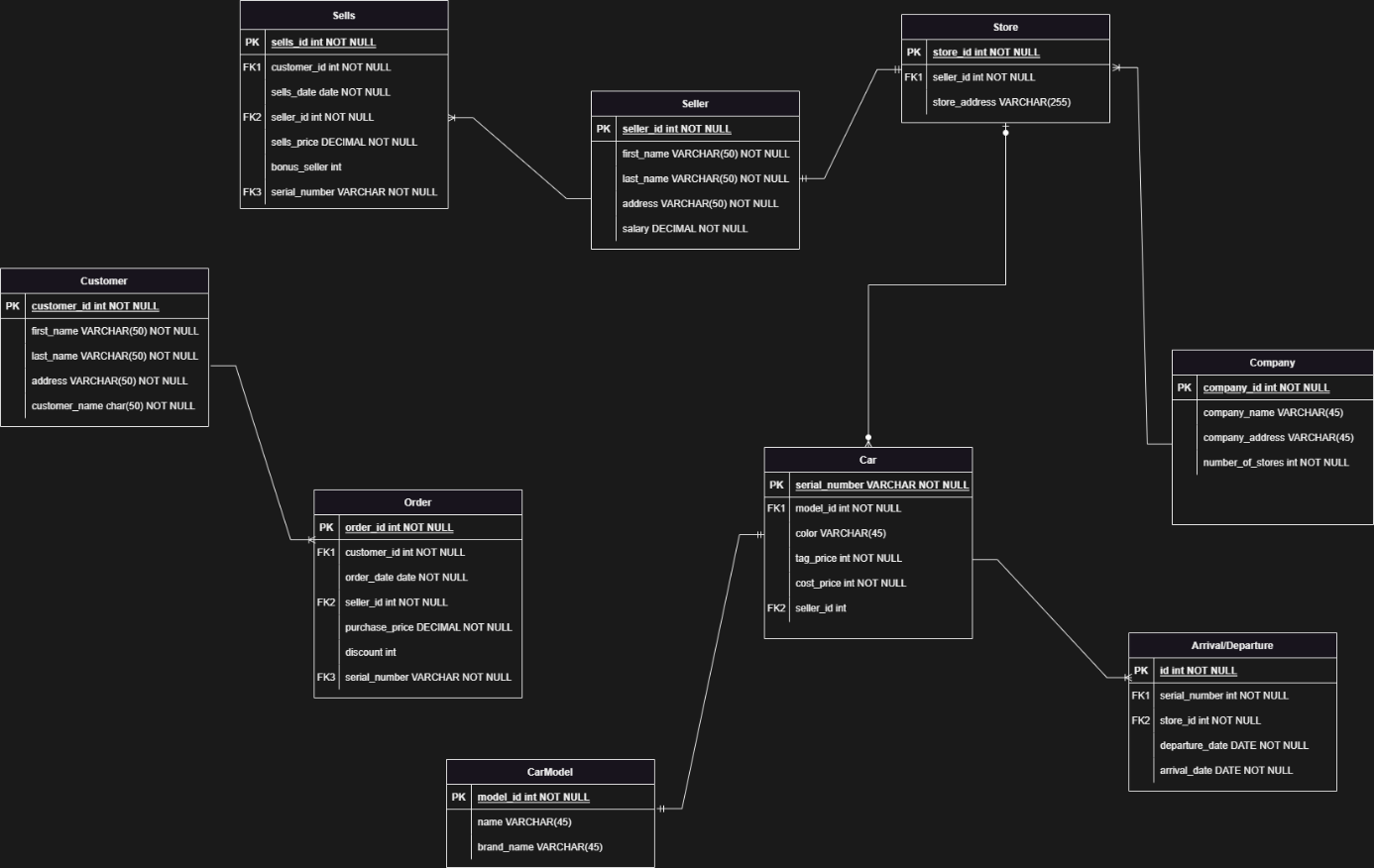
****

**x) For each Cena’s film (title and year), give the total actors’ salaries.**

****

Exercice 2

1. i) Give the conceptual data model of the above (E-R Diagram)



1. Convert this conceptual data model to **logical data model**. Hence, write the following SQL queries

• COMPANY (CompanyId (PK, INT), CompanyName (VARCHAR), SellerID (FK, VARCHAR), CompanyAddress (VARCHAR),NumberOfStore (INT, NULL))

• CAR (SerialNumber (PK, VARCHAR), Model (VARCHAR), Color (VARCHAR), TagPrice (DECIMAL), CostPrice (DECIMAL), SoldBy (FK, VARCHAR), StoredIn (FK, VARCHAR))

• CUSTOMER (CustomerID (PK, VARCHAR), LastName (VARCHAR), FirstName (VARCHAR), Address (VARCHAR))

• SELLER (SellerID(PK, VARCHAR), LastName(VARCHAR)

, FirstName (VARCHAR), Address (VARCHAR), Salary (DECIMAL))

• STORE (StoreID (PK, INT), StoreAddress (VARCHAR), SellerID (FK, VARCHAR))

• CAR\_MODEL (ModelId (PK, INT), Name (VARCHAR), BrandName ( VARCHAR))

• SELLS (SerialNumber (FK, VARCHAR), SellerID (FK, VARCHAR), SellsPrice (DECIMAL), SaleDate (DATE))

• CUSTOMER\_ORDER (CustomerID (FK, VARCHAR), SerialNumber (FK, VARCHAR), PurchaseDate (DATE))

• ARRIVAL\_OR\_DEPARTURE\_INFO (Id (PK, INT) , SerialNumber (FK, VARCHAR), StoreID (FK, VARCHAR), ArrivalDate (DATE),DepartureDate (DATE))

iii) Give the list of cars and serial numbers sold after the 1st of January 2023.

SELECT SerialNumber, ModelID

FROM Sale

JOIN Car ON Sale.SerialNumber = Car.SerialNumber

WHERE SaleDate > '2023-01-01';

iv) Give the name of the car with the highest sales

SELECT Brand, Name

FROM CarModel

WHERE ModelID = ( SELECT ModelID FROM Sale

GROUP BY ModelID

ORDER BY SUM(PurchasePrice - CostPrice) DESC

LIMIT 1

);

v) Give the seller with the lowest discount

SELECT FirstName, LastName

FROM Seller

WHERE SellerID = ( SELECT SellerID FROM Sale

ORDER BY Discount ASC

LIMIT 1

);

vi) Give the profit of each store for the month of December 2022

SELECT StoreID, SUM(PurchasePrice - CostPrice) AS Profit

FROM Sale JOIN Car ON Sale.SerialNumber = Car.SerialNumber

WHERE SaleDate >= '2022-12-01' AND SaleDate < '2023-01-01'

GROUP BY StoreID;

vii) Give the best customer (The customer who has paid in the most money to the company)

SELECT FirstName, LastName

FROM Customer

WHERE CustomerID = ( SELECT CustomerID FROM Sale GROUP BY CustomerID

ORDER BY SUM(PurchasePrice) DESC

LIMIT 1

);

viii) The brand of the car with the highest discount

SELECT Brand

FROM CarModel

WHERE ModelID = ( SELECT ModelID FROM Sale

ORDER BY Discount DESC

LIMIT 1

);

ix) The car with the highest tag price

SELECT Brand, Name

FROM CarModel

WHERE ModelID = ( SELECT ModelID FROM Car

ORDER BY TagPrice DESC

LIMIT 1

);