1. **Database** : Data base is the collections of data in systematically and electronically. It can store data like sales transactions, customer information, product information etc. The data will be stores in Row and Column format such like Table
2. **What is Normalization?**

* Normalization is the process of organizing the data in the database.
* Normalization is used to minimize the redundancy from a relation or set of relations. It is also used to eliminate undesirable characteristics like Insertion, Update, and Deletion Anomalies.
* Normalization divides the larger table into smaller and links them using relationships.
* The normal form is used to reduce redundancy from the database table.

1. **What is Difference between DBMS and RDBMS?**

**DBMS :** **Database Management System (DBMS)** is a software that is used to define, create and maintain a database and provides controlled access to the data.

DBMS stores data as file.

Normalization is not present.

It deals with small quantity of data.

It supports single user.

**RDBMS: Relational Database Management System (RDBMS)** is an advanced version of a DBMS.

RDBMS stores data in tabular form.

Normalization is present.

It deals with large amount of data.

It supports multiple users.

1. **What is MF Cod Rule of RDBMS Systems?**

**It was proposed by Dr. E.F.Codd.   
It uses the concept of relations to represent each and every file.**

**12 Rules are :**

1. Information Rule
2. Guaranteed Access Rule
3. Systematic Treatment of NULL Values

## Active Online Catalog

## Comprehensive Data Sub-Language Rule

## View Updating Rule

## High-Level Insert, Update, and Delete Rule

## Physical Data Independence

## Logical Data Independence

## Integrity Independence

## Distribution Independence

## Non-Subversion Rule

## What do you understand By Data Redundancy?

## ****Redundancy**** means having multiple copies of same data in the database. This problem arises when a database is not normalized

## What is DDL Interpreter?

## **DDL** stands for **Data Definition Language.**

**DDL** commands can be used to add, remove, or modify tables with in a database.

The DDL commands are: 

1. CREATE
2. ALTER
3. DROP
4. TRUNCATE
5. RENAME
6. **What is DML Compiler in SQL?**

DML Compiler: It processes the DML statements into low level instruction (machine language), so that they can be executed.

1. **What is SQL Key Constraints writing an Example of SQL Key Constraints**

Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

* NOT NULL - Ensures that a column cannot have a NULL value
* [UNIQUE](https://www.w3schools.com/sql/sql_unique.asp) - Ensures that all values in a column are different
* [PRIMARY KEY](https://www.w3schools.com/sql/sql_primarykey.asp) - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
* [FOREIGN KEY](https://www.w3schools.com/sql/sql_foreignkey.asp) - Prevents actions that would destroy links between tables
* [CHECK](https://www.w3schools.com/sql/sql_check.asp) - Ensures that the values in a column satisfies a specific condition
* [DEFAULT](https://www.w3schools.com/sql/sql_default.asp) - Sets a default value for a column if no value is specified
* [CREATE INDEX](https://www.w3schools.com/sql/sql_create_index.asp) - Used to create and retrieve data from the database very quickly

Ex: CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    PRIMARY KEY (ID)  
);

1. **What is save Point? How to create a save Point write a Query?**

* Savepoint is a command in SQL that is used with the rollback command.
* It is a command in Transaction Control Language that is used to mark the transaction in a table.
* Savepoint is helpful when we want to roll back only a small part of a table and not the whole table. In simple words, we can say savepoint is a bookmark in SQL.

Ex: mysql> SAVEPOINT ini;

1. **What is trigger and how to create a Trigger in SQL?**

 A trigger is a special type of stored procedure that automatically runs when an event occurs in the database server. DML triggers run when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view.

create trigger [trigger\_name]

[before | after]

{insert | update | delete}

on [table\_name]

[for each row]

[trigger\_body]

**Task – 1 : Database Name : School**

Table 1 : Exam table:

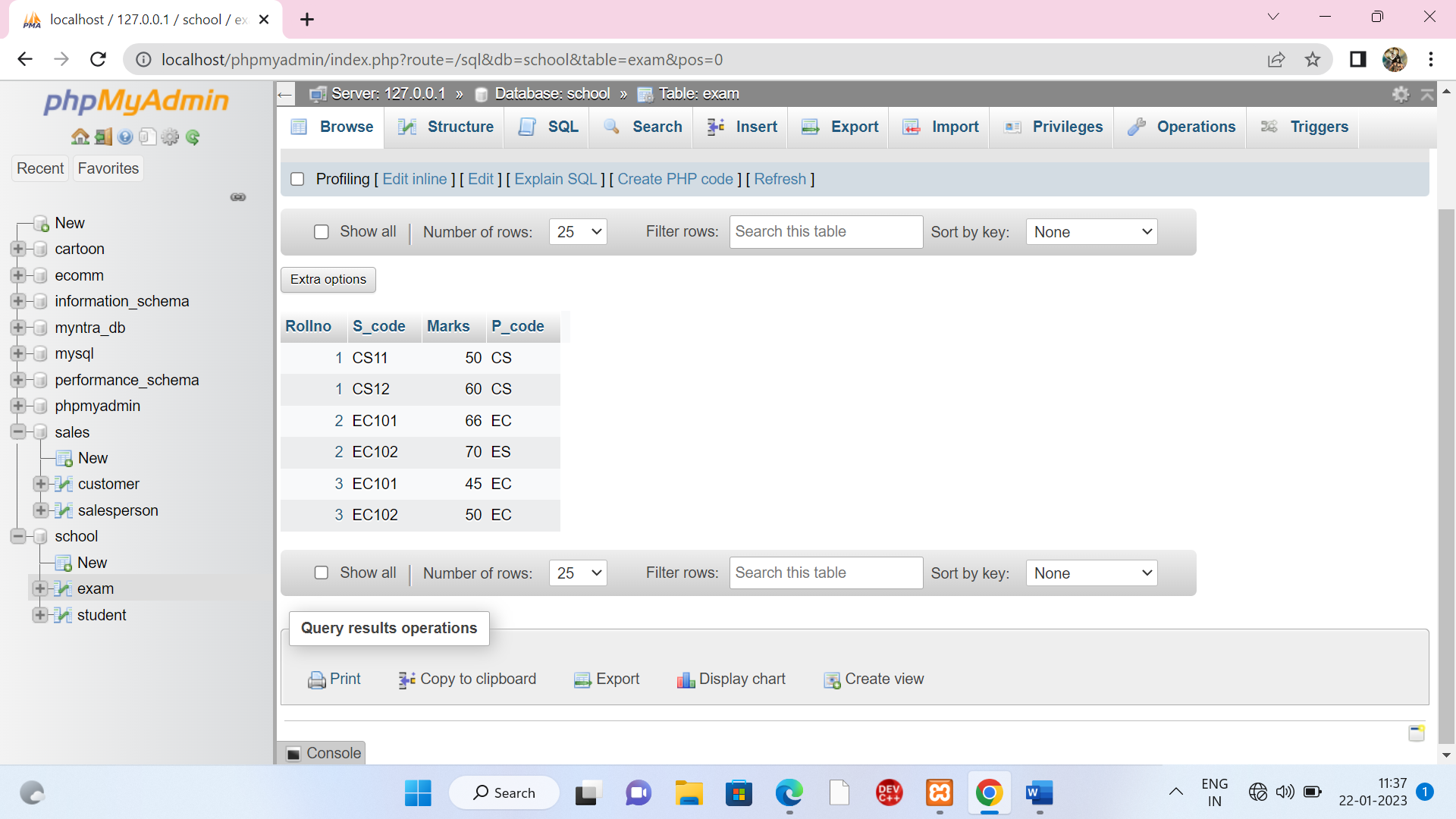
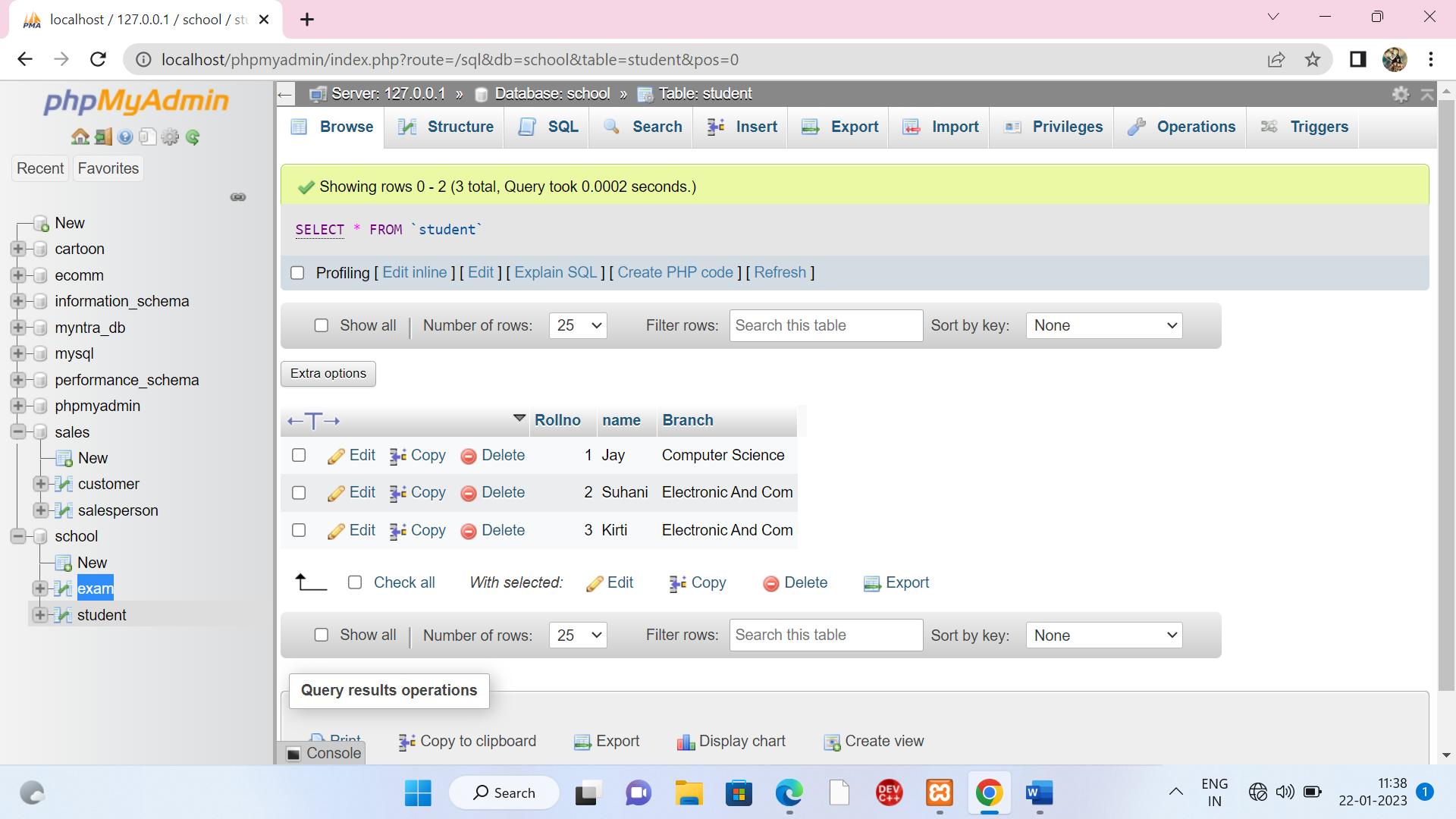
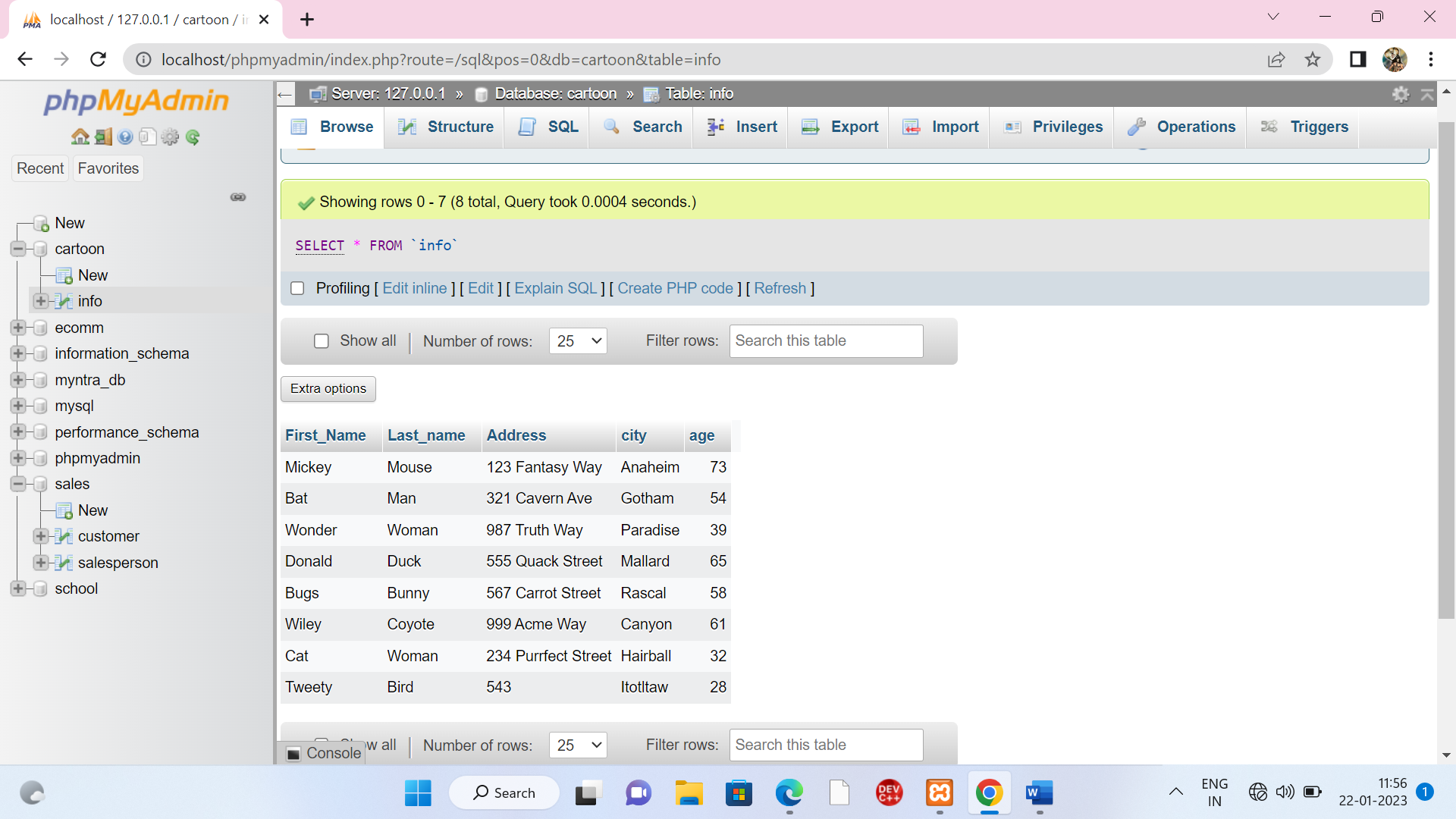


Table 2 : Student Table:



**Task -2 :- Database Name : Cartoon**

Table -1 : Info



**Task -3 : Databse Name : Bank**

Table -1: Employee

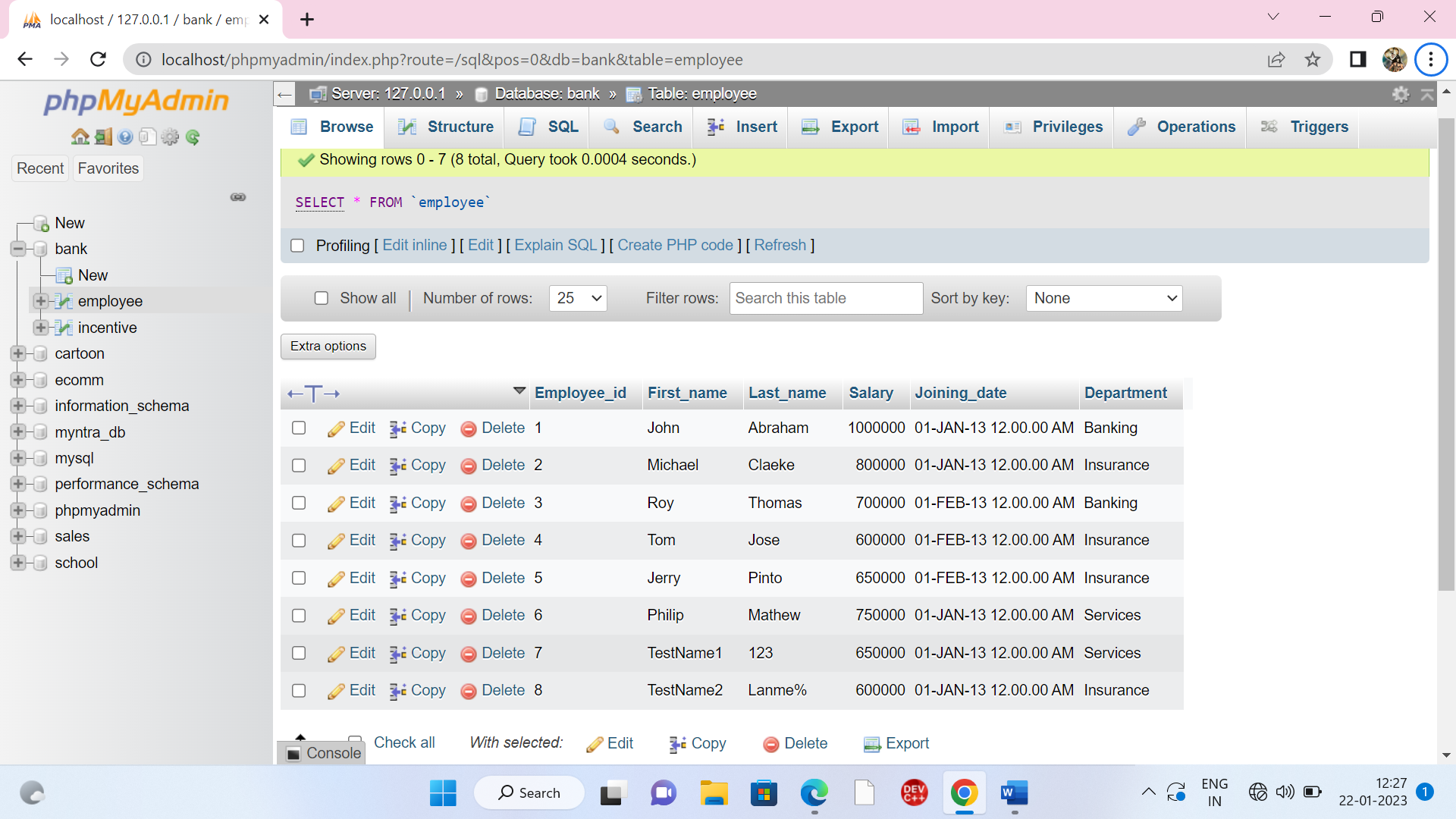
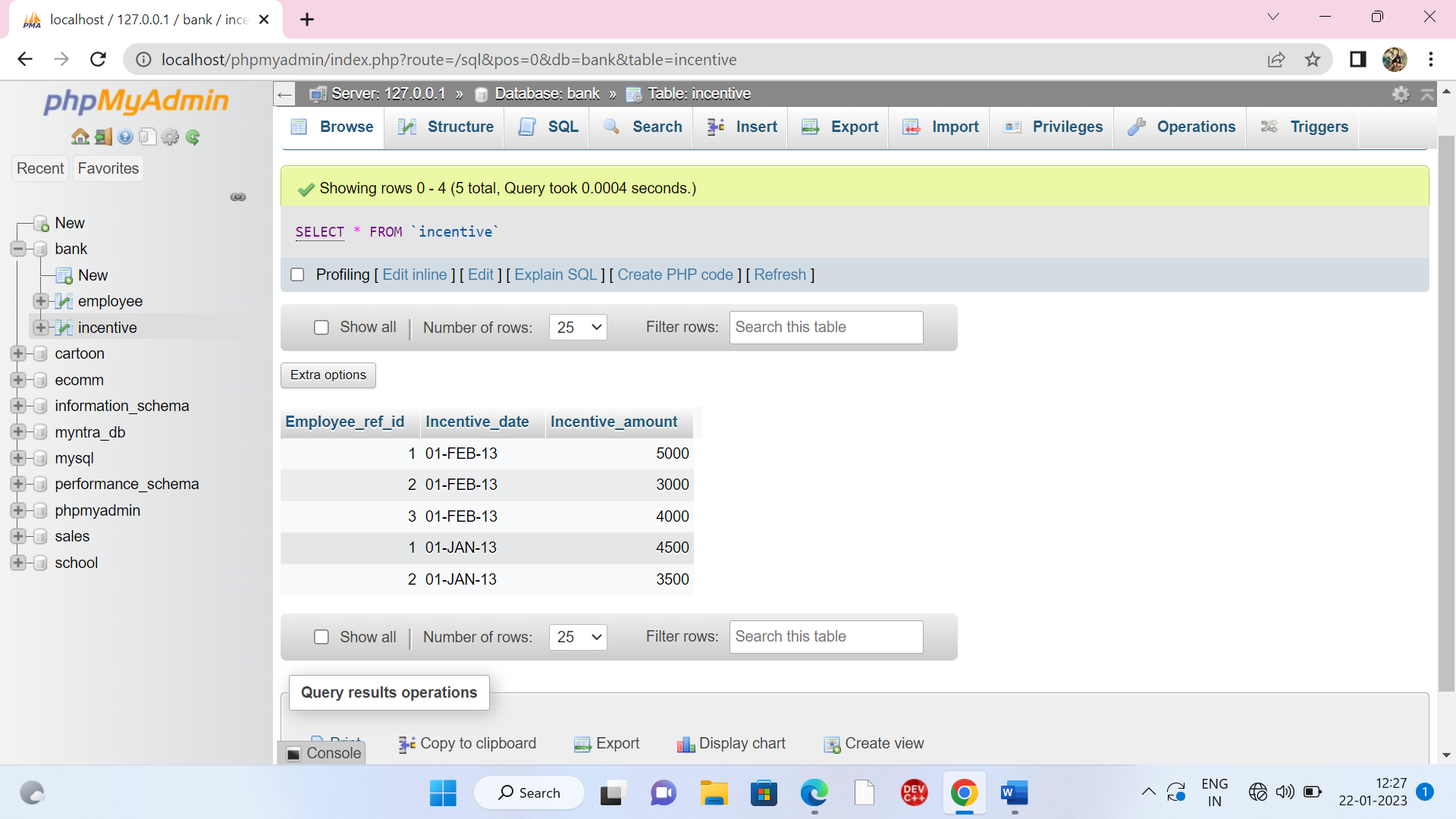


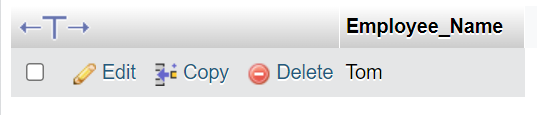
Table -2: Incentive



**Query**

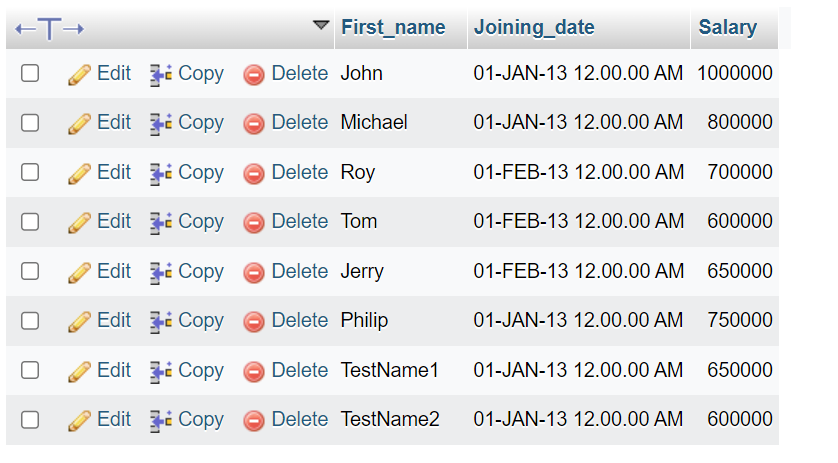
1. Get First\_Name from employee table using Tom name “Employee Name”.

* SELECT First\_name AS Employee\_Name from employee where First\_name='Tom';



1. Get FIRST\_NAME, Joining Date, and Salary from employee table.

🡺 SELECT First\_name, Joining\_date, Salary from employee;



1. Get all employee details from the employee table order by First\_Name Ascending and Salary descending?

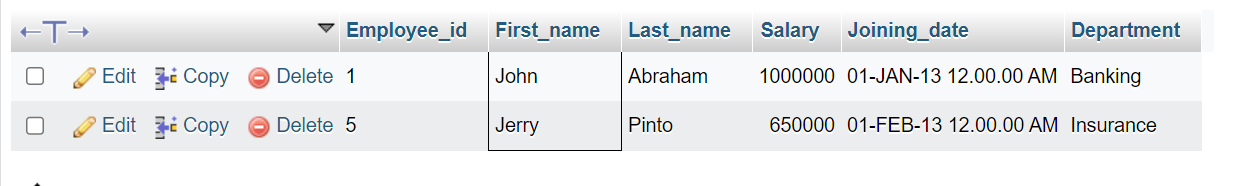
* SELECT \* from employee ORDER BY First\_name ASC;



* SELECT \* from employee ORDER BY Salary DESC;

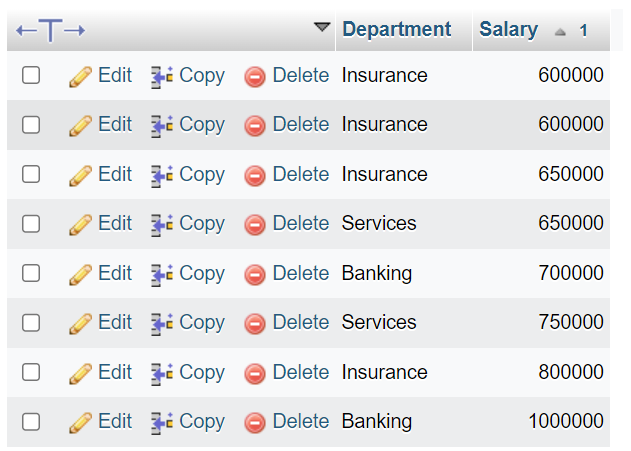
**Task 4 :- Database Name : Sales**

1. Get employee details from employee table whose first name contains ‘J’.

* SELECT \* from employee WHERE First\_name LIKE 'J%';

1. Get department wise maximum salary from employee table order by salary ascending?

* SELECT Department, Salary FROM `employee` ORDER BY Salary ASC;



1. Create After Insert trigger on Employee table which insert records in view table

DELIMITER $$

CREATE TRIGGER insertTrigger after INSERT on employee for Each ROW

BEGIN

INSERT into update\_trg(emp\_id,emp\_name,emp\_dpart,Date\_Time) VALUES(new.Employee\_id,new.First\_name,new.Department);

END

$$

**Task -4 : Database Name : Sales**

Table 1 : customer

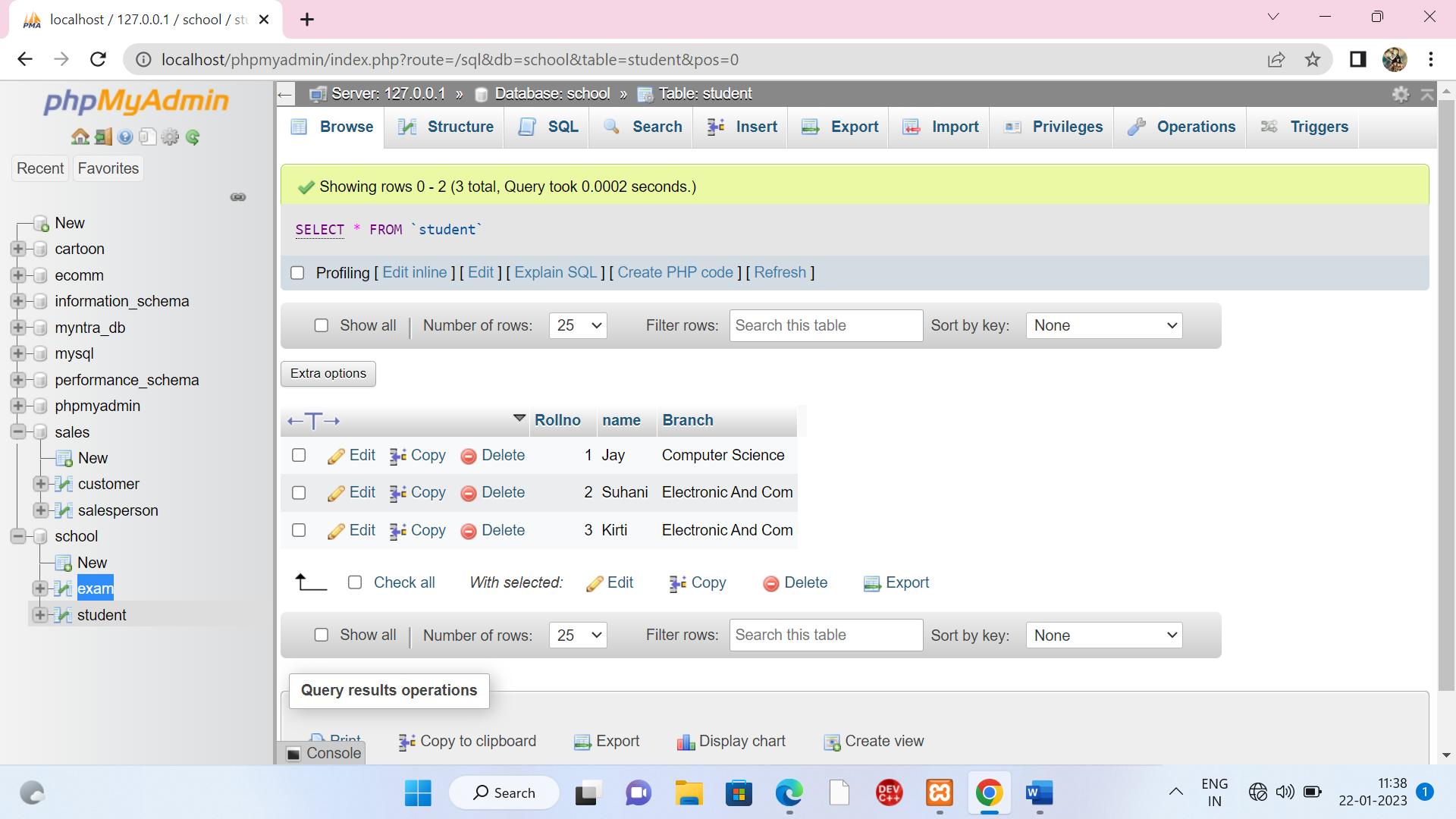
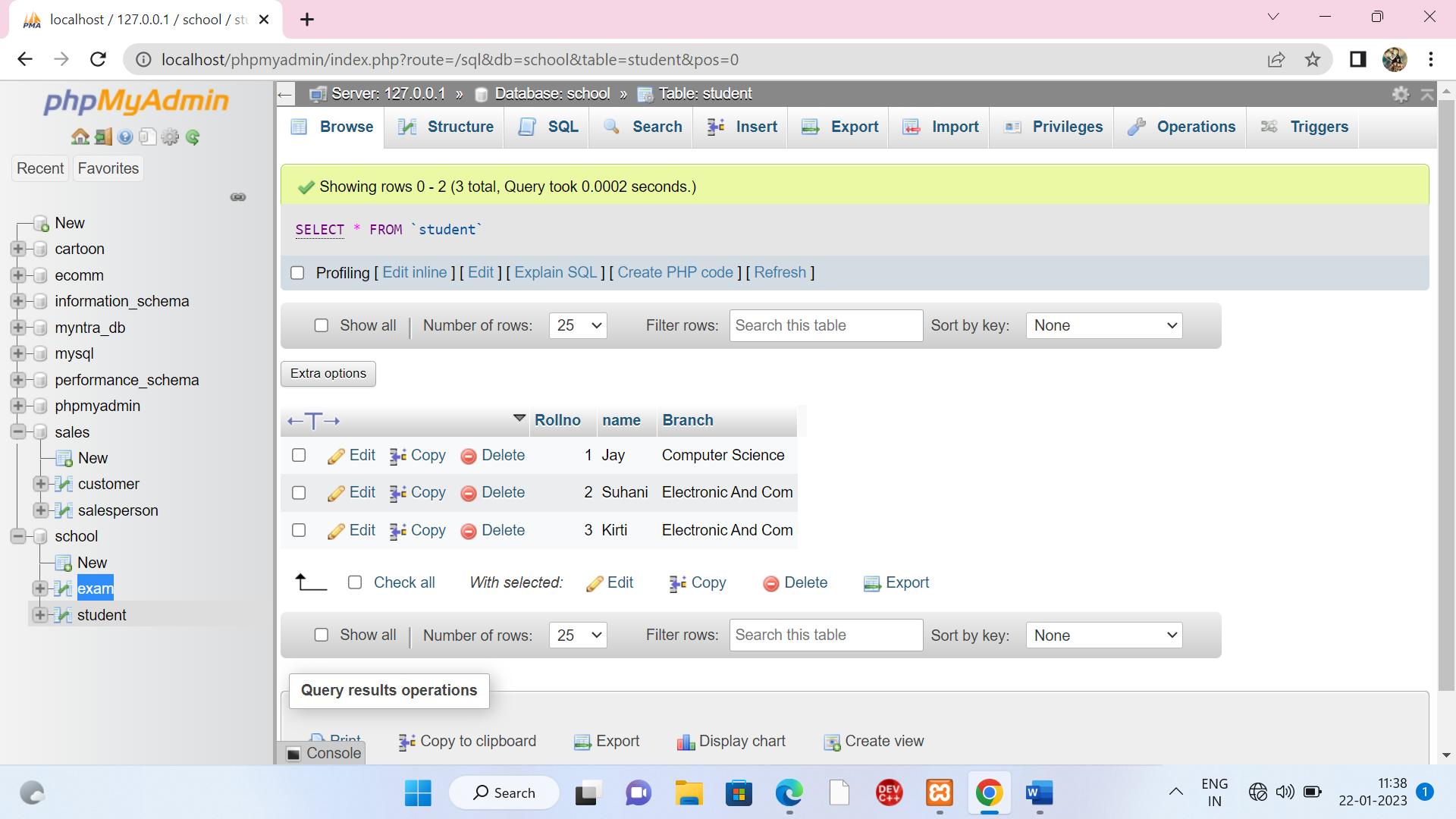


Table -2 : salesperson



**Query**

b) Names and cities of all salespeople in London with commission above 0.12

🡺 SELECT Sname,City FROM salesperson where City='London' And Comm >= '.12' ;

c) All salespeople either in Barcelona or in London

🡺 SELECT Sname from salesperson where City='Barcelona' or city='London'

d) All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded).

🡺 SELECT Sname from salesperson where Comm BETWEEN '.10' and '.12';

e) All customers excluding those with rating <= 100 unless they are located in Rome

🡺 SELECT \* from customer where Rating>=100 and City!='Roe'