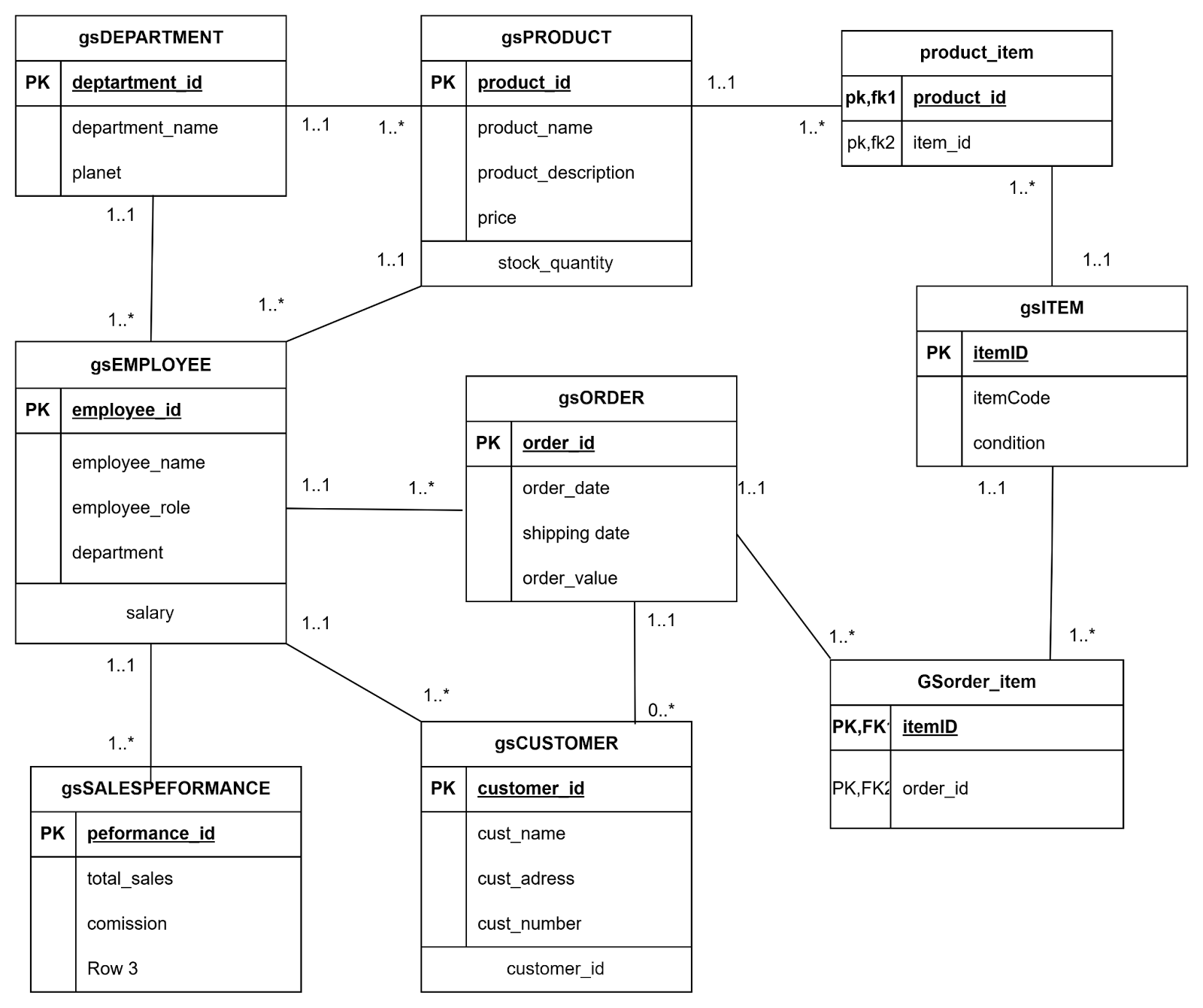
Rayhanur Rahman , StudentID = W2073057

Entities , keys and relationships:

|  |  |  |  |
| --- | --- | --- | --- |
| Entity | Primary Key | Foreign key | Rationale for PK |
| gsCUSTOMER | customer\_id | N/A | Customer id is unique for each customer |
| gsORDER | order\_id | customer\_id | Order id is unique for each order |
| customer\_order | Customer\_id , order\_id | Customer\_id , order\_id | Customer\_order is a link entity and a minimum PK is a  composite key made  up of parent PKs as  as well as being FKs. |
| gsITEM | itemID | product\_id | itemID is unique for each ID |
| GSorder\_item | Order\_id , itemID | Order\_id , itemID | Item\_order is a link entity and a minimum PK is a  composite key made  up of parent PKs as  as well as being FKs. |
| gsPRODUCT | product\_id | N/A | Product\_id is unique for each product |
| product\_item | product\_id , itemID | Product\_id, itemID | Product\_item is a link entity and a minimum PK is a  composite key made  up of parent PKs as  as well as being FKs. |
| gsDEPARTMENT | department\_id | N/A | Each department has their own ID which is unique to the department |
| gsEMPLOYEE | employee\_id | department\_id | Each employee has their own ID which is unique |
| gsSALESPEFORMANCE | peformance\_id | employee\_id | Each employee has their own performance\_id which is unique to them |
|  |  |  |  |

ERD DIAGRAM:



**Relationships between Entities:**

A customer can place many orders (One-to-Many relationship). Each order belongs to a specific customer. Some customers may not place an order.

An order can include multiple items. The Item\_Order is a relationship table that links orders to the items being purchased. An order may consist of many items, and each item can appear in many orders so there's a many to many relationship between item and order.

The product manufacturing department is responsible for creating the product, other departments can promote the products so there is a One-to-Many relationship.

An employee's performance can be tracked through sales performance metrics so it is a One-to-Many relationship from Employee to SalesPerformance.

Each employee works in a department so there is a Many-to-One relationship. Departments have at least one employee and can have many employees

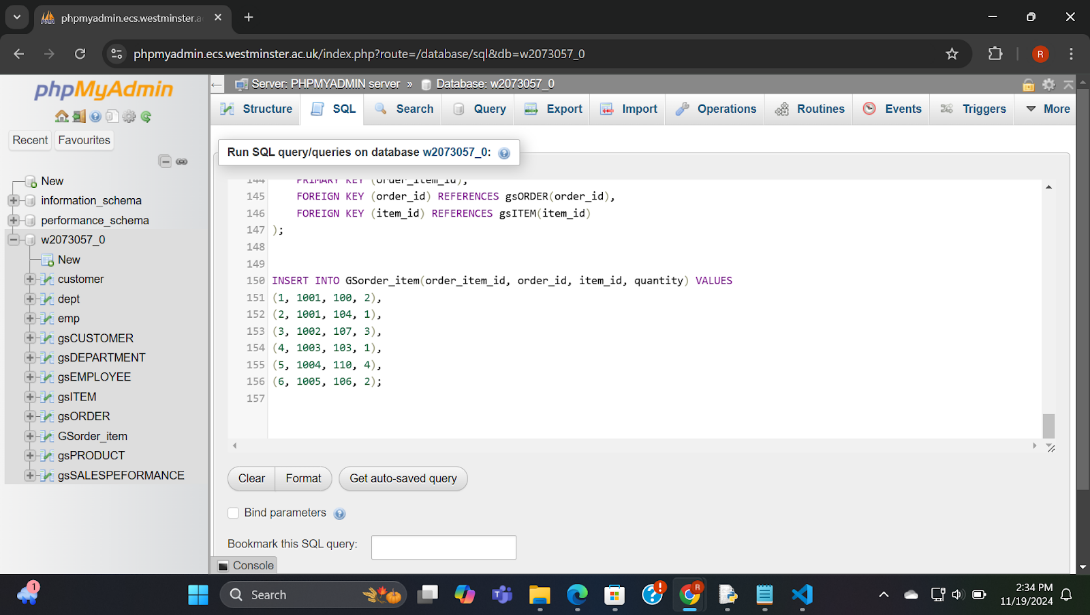
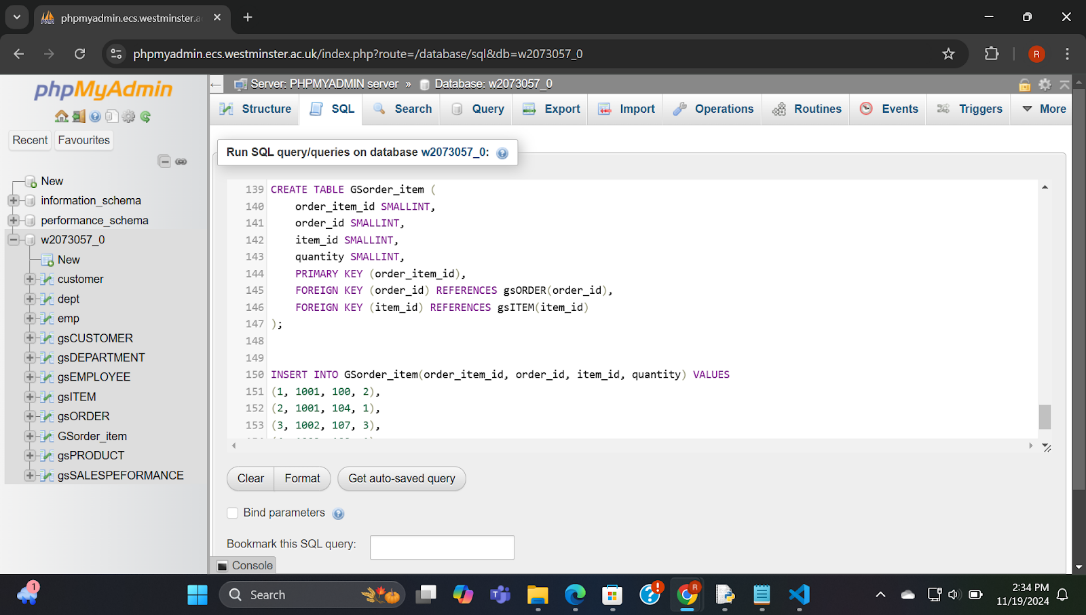
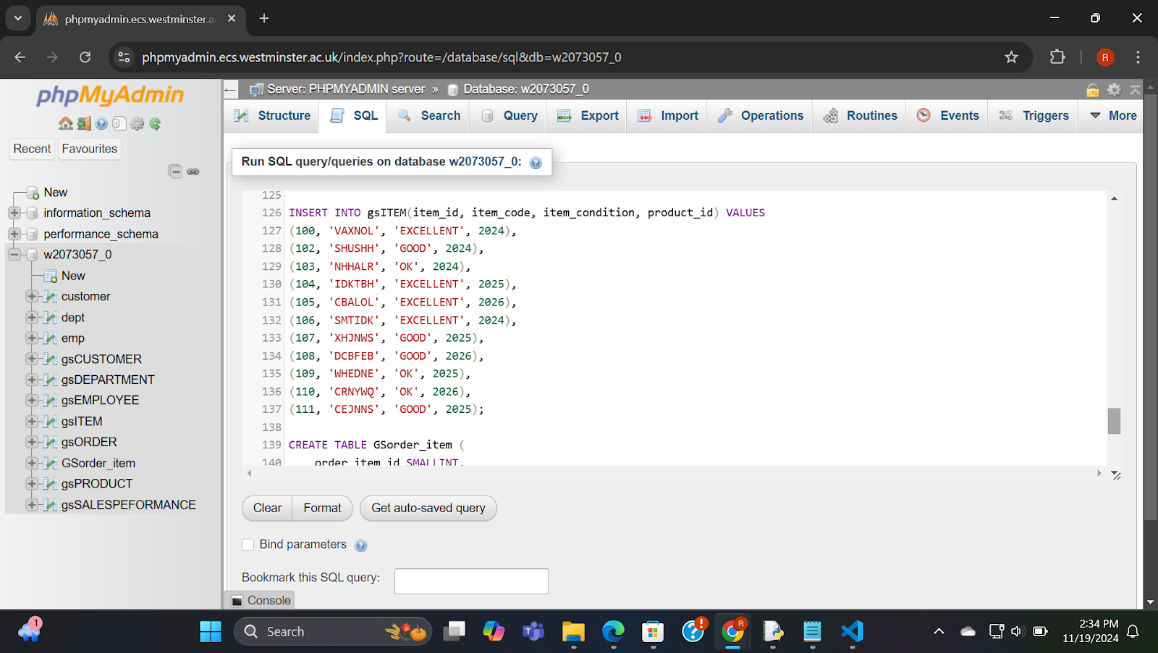
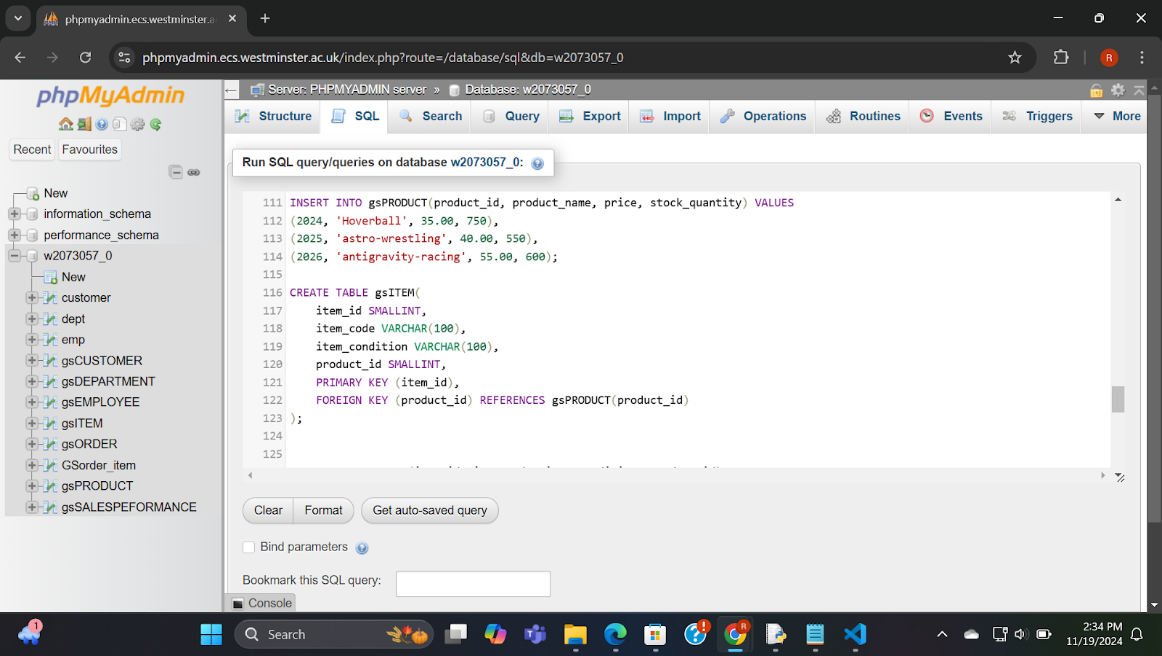
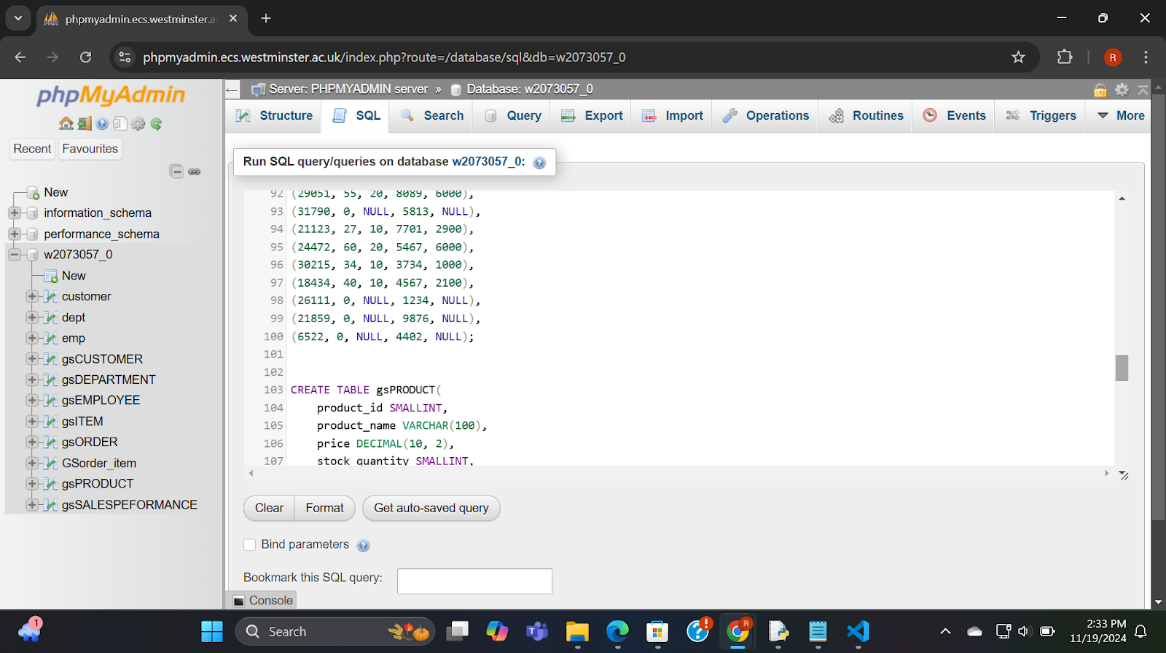
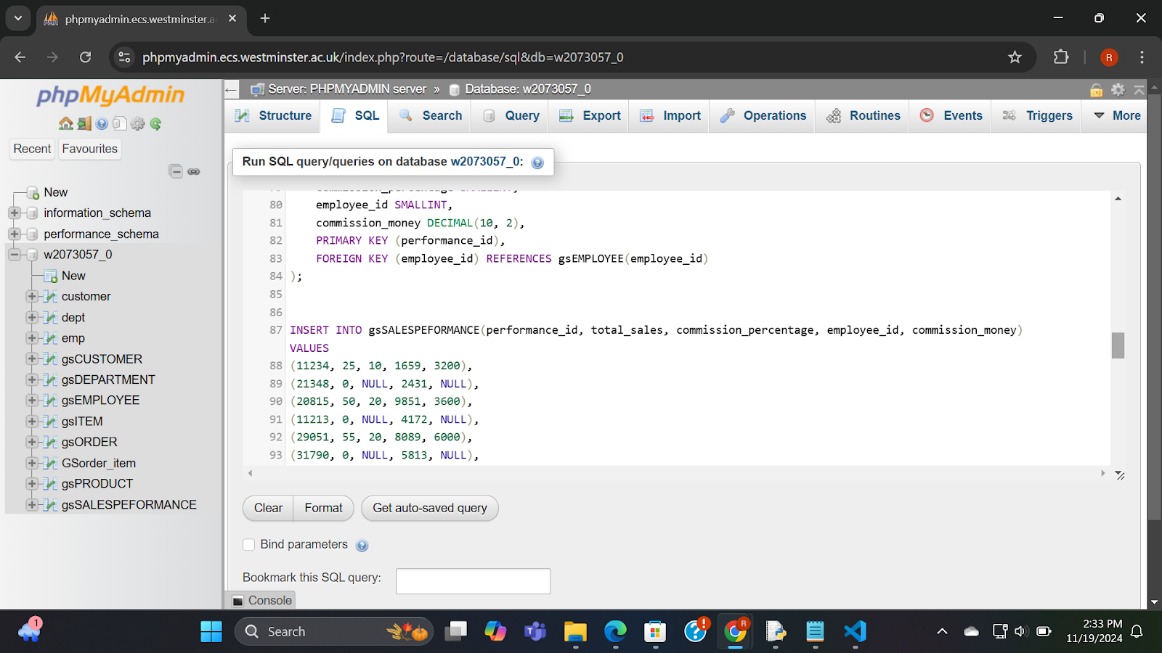
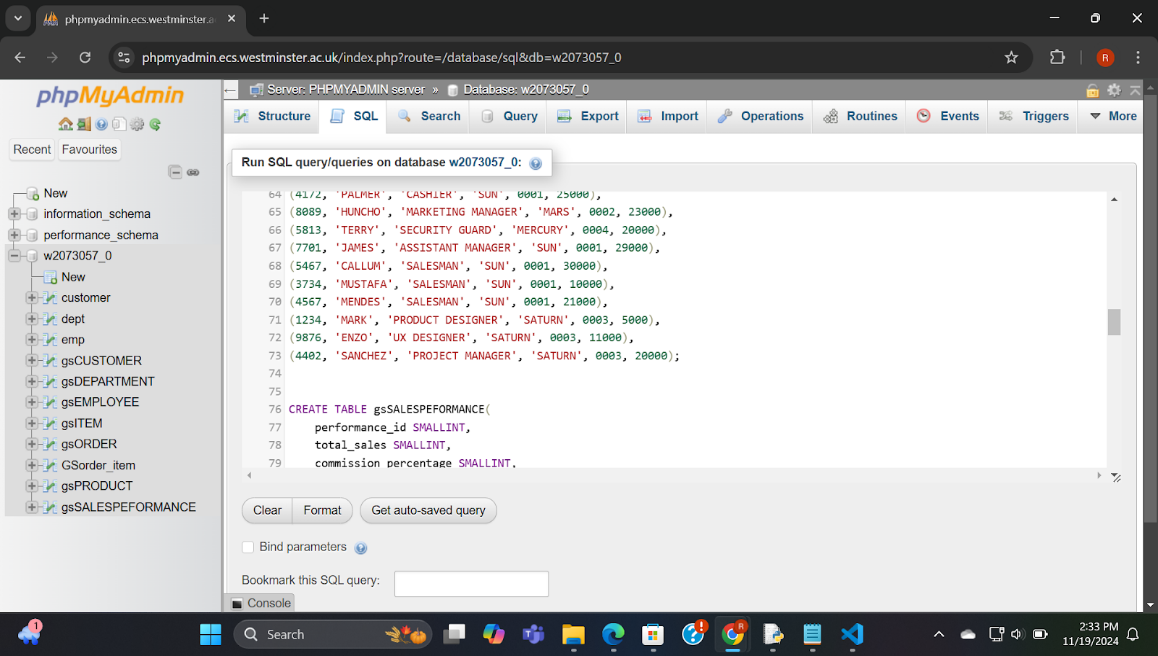
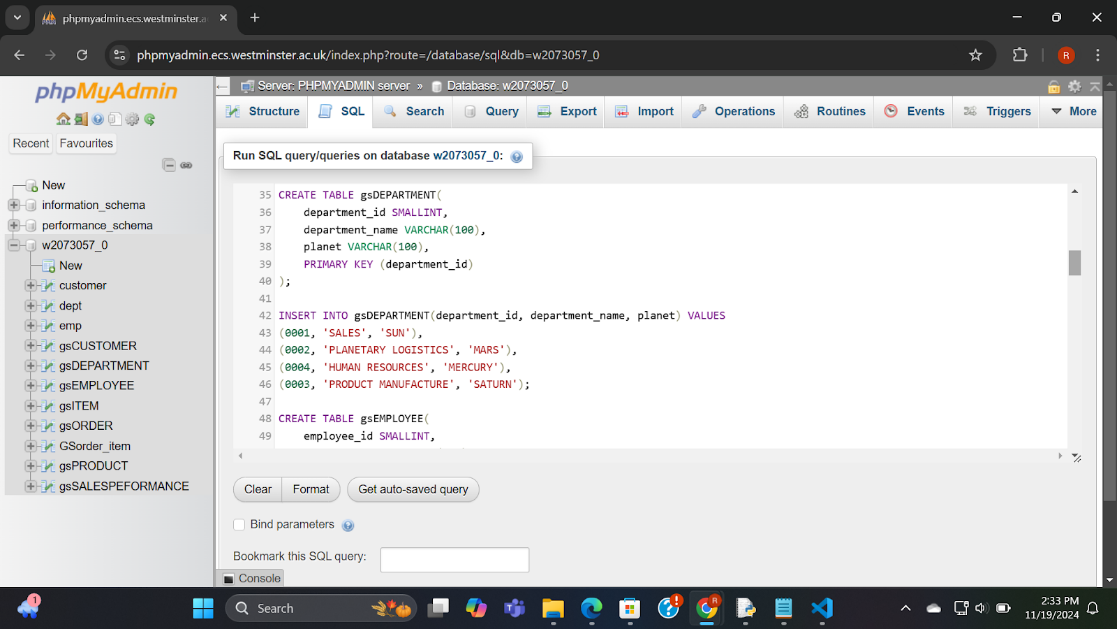
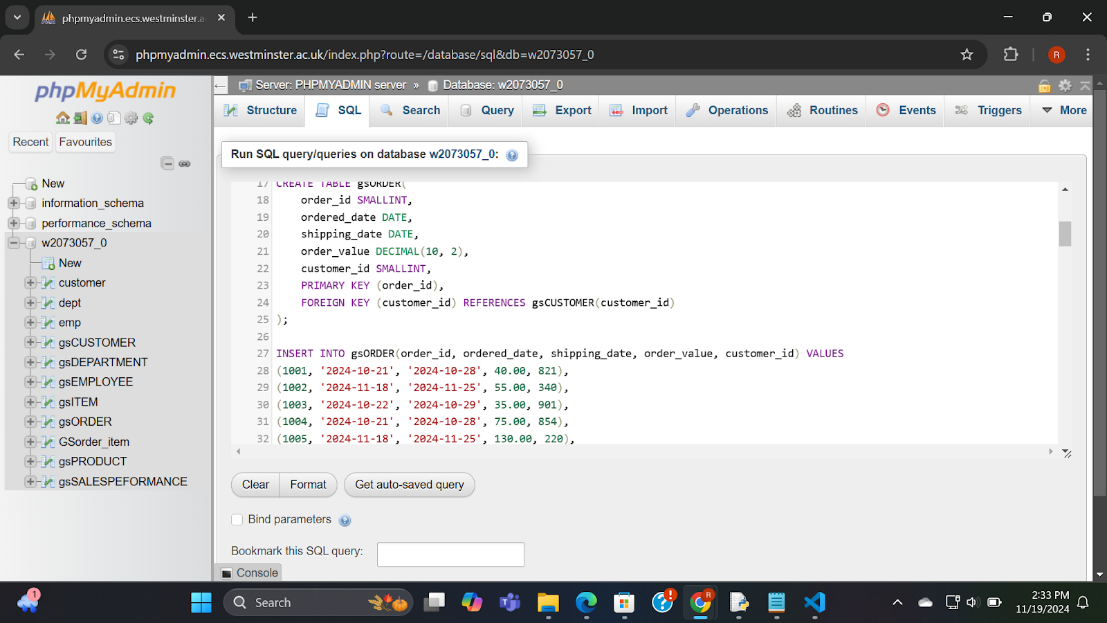
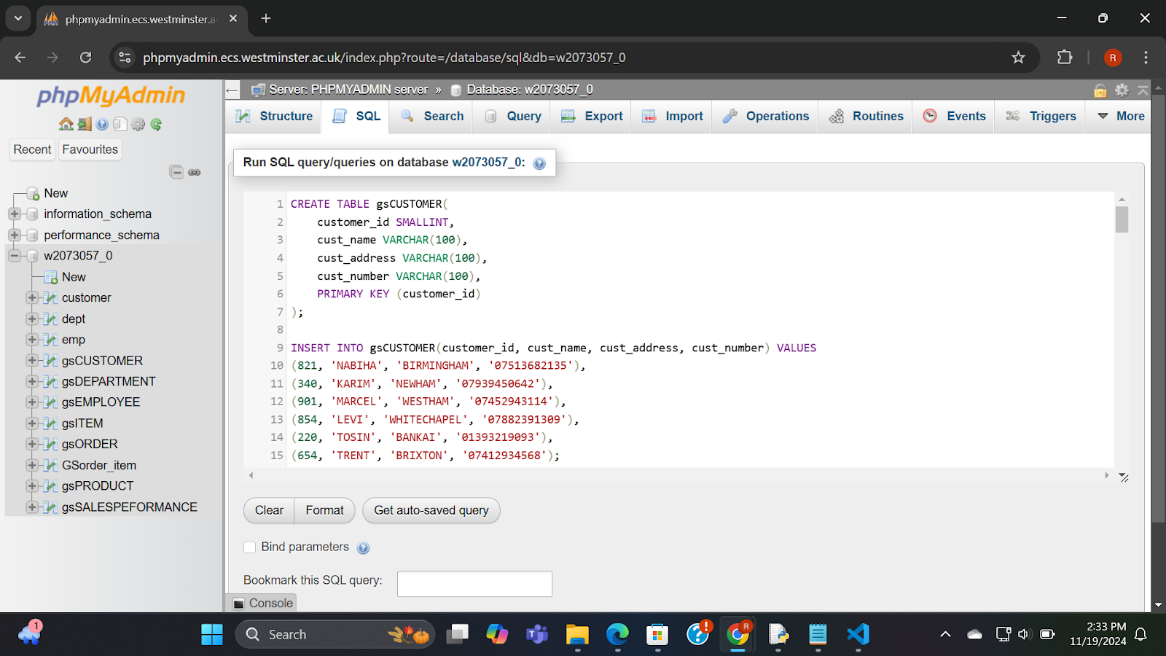
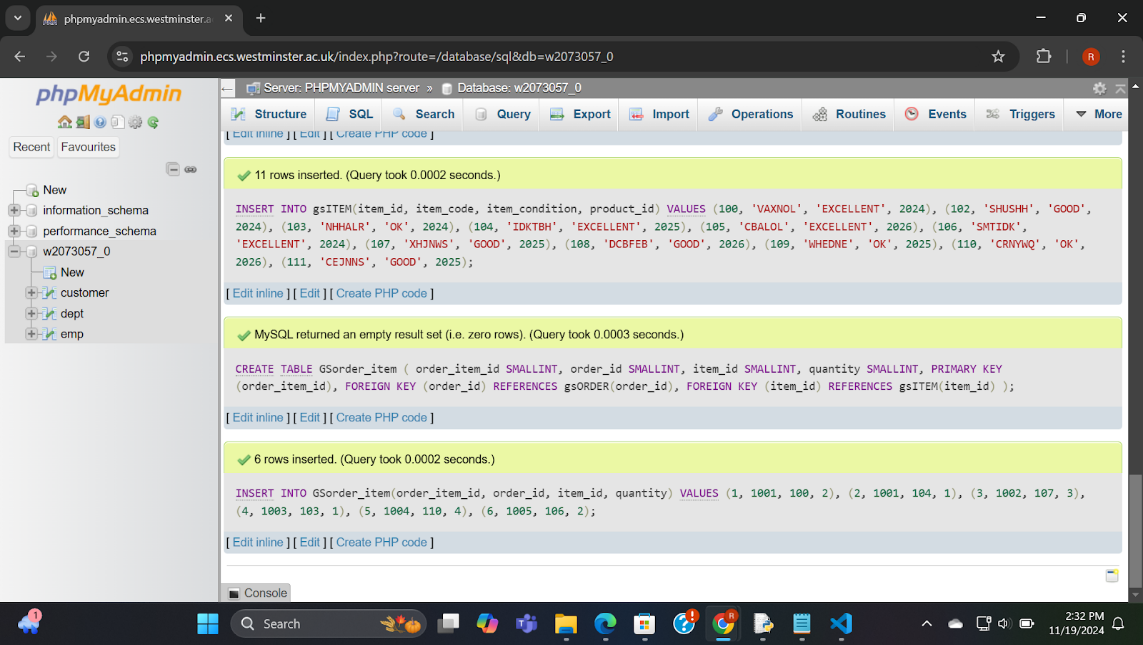
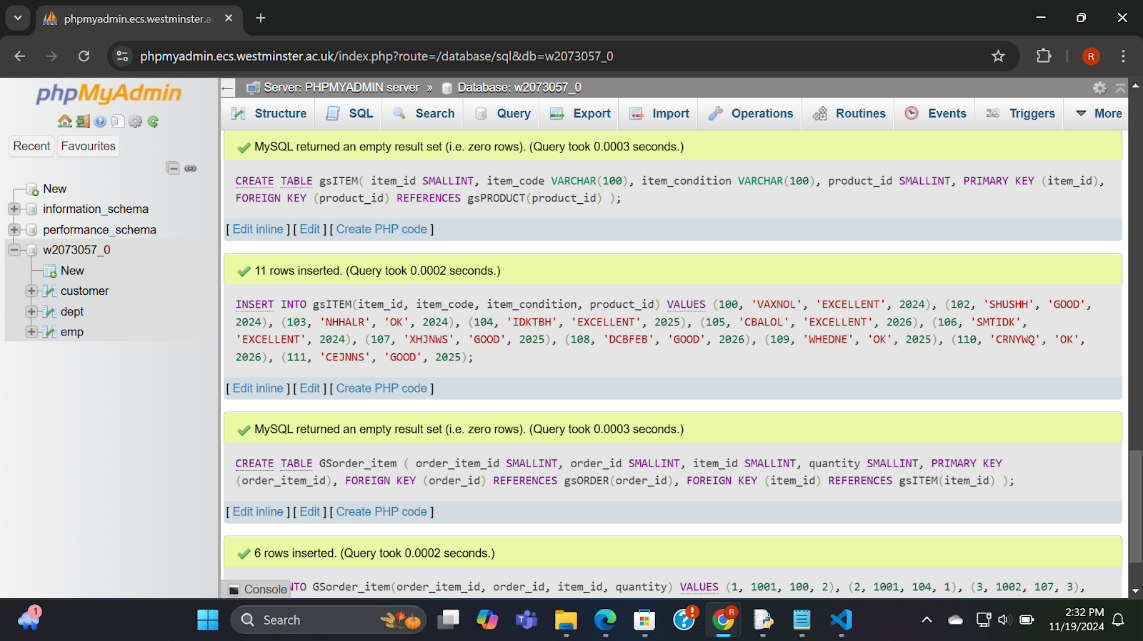
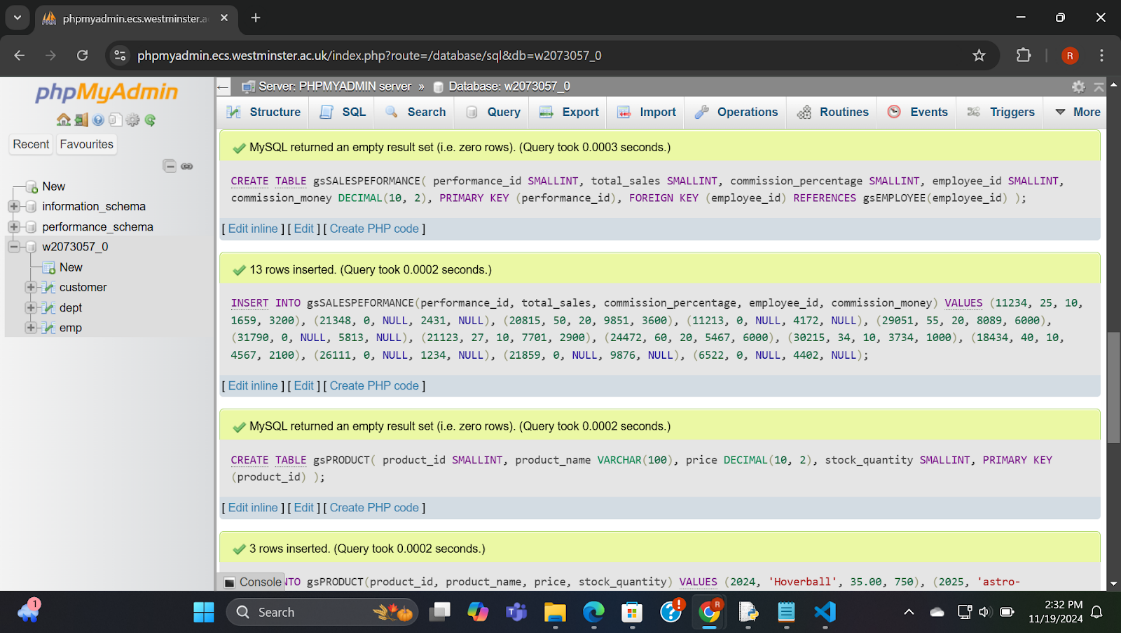
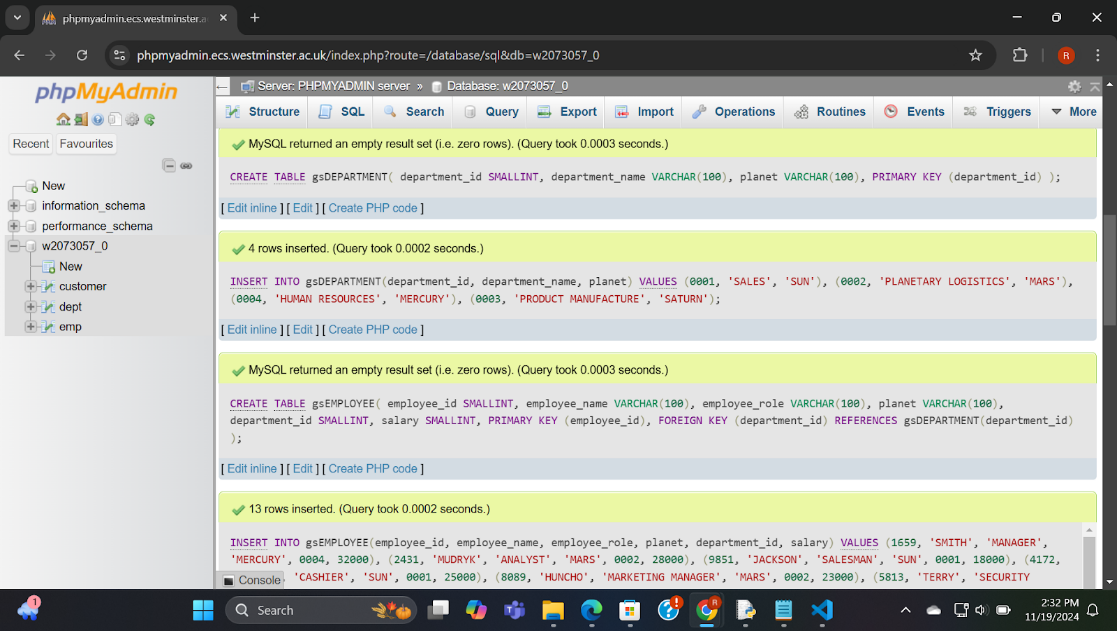
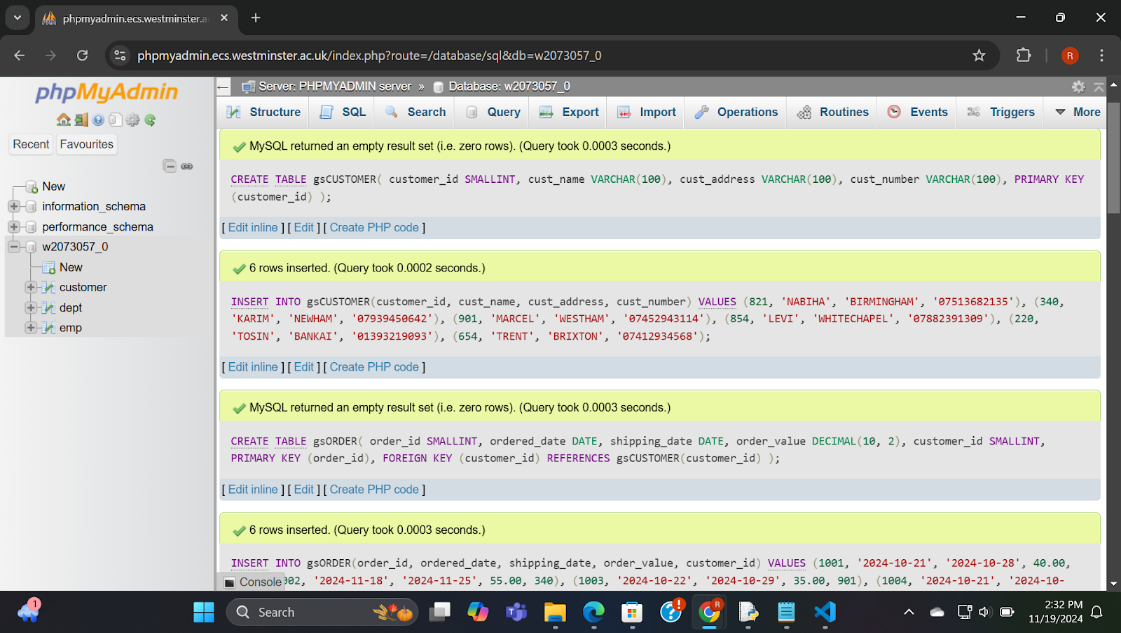
Employees process orders , employees can process multiple orders.

Employees interact with customers as some employees are salesmen and would promote the products and influence customers to buy the product.

Employees create the product , many employees work on a product.

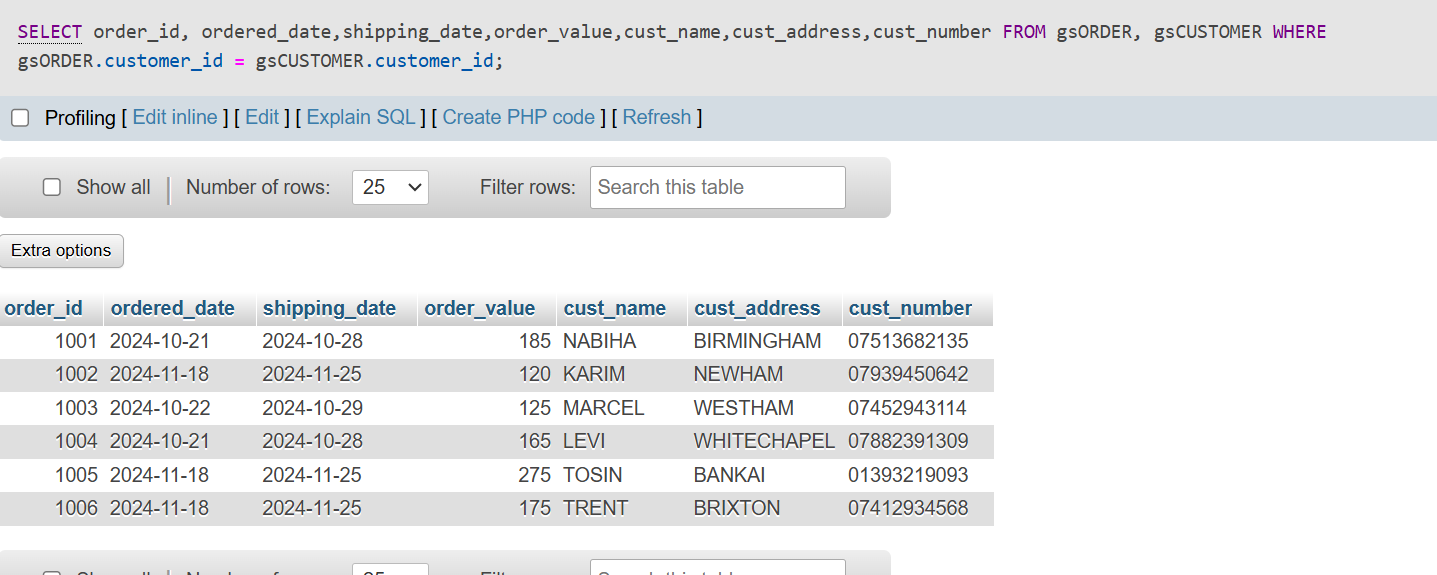
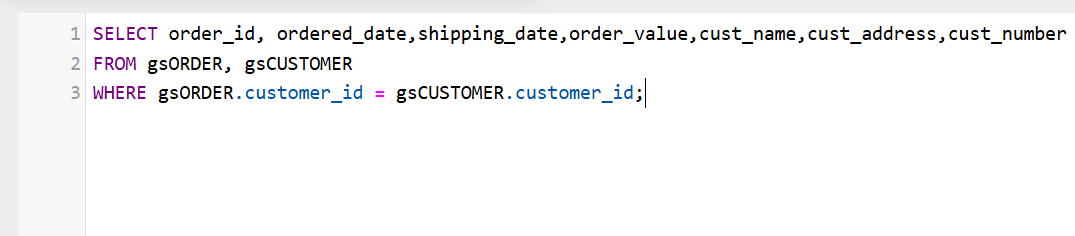
A product can have multiple items and an item can be associated with multiple products , since the relationship is many to many , we need to make another entity to link the two called product\_id.

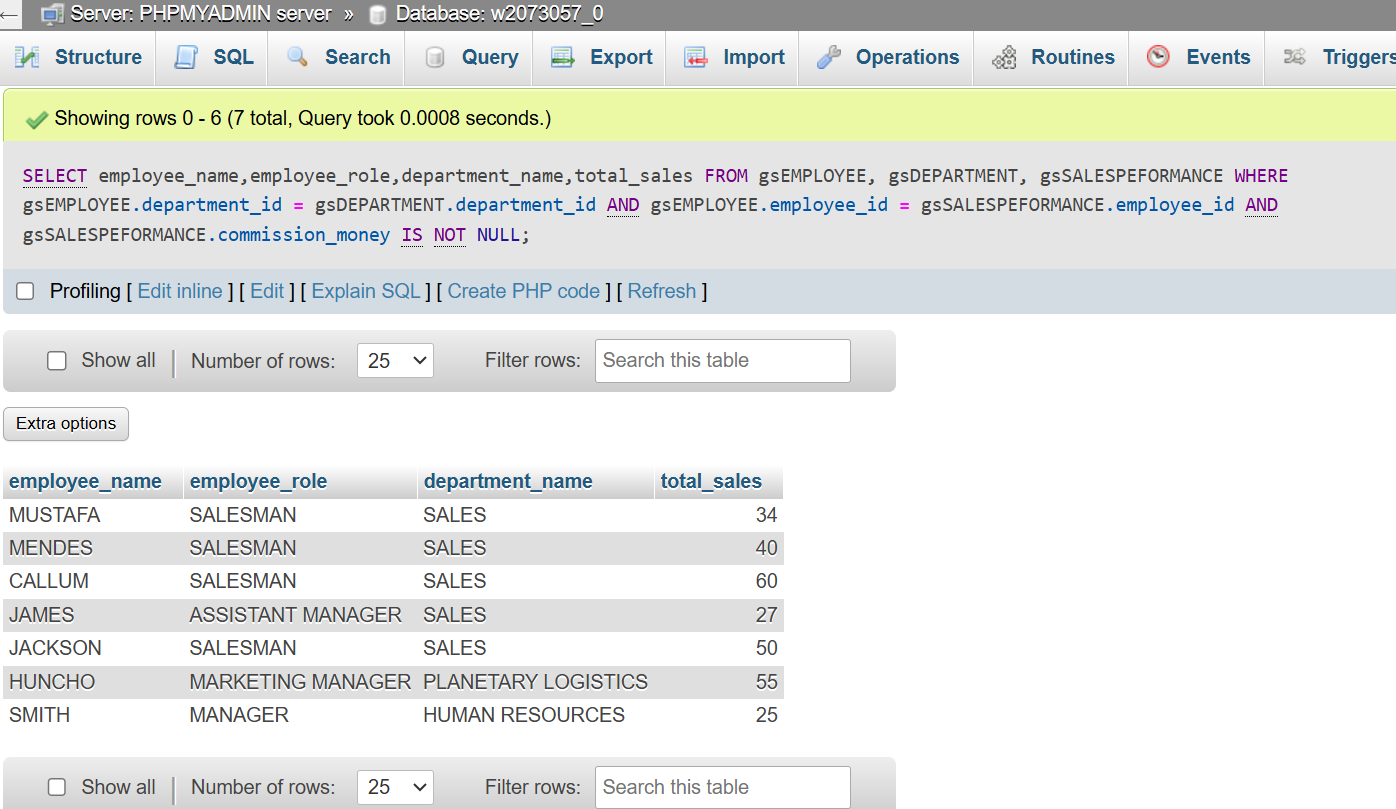
SQL TABLE CREATION:



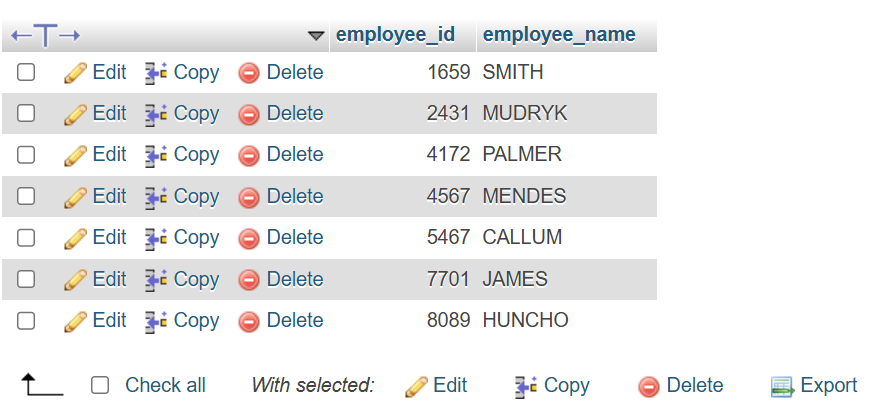
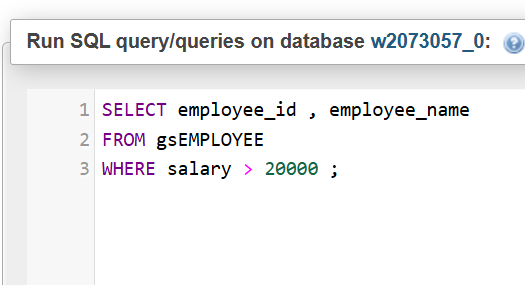
QUERIES :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Query NO | Purpose | Tables involved | SQL | Correct execution |
| 1 | Listing all order id with customer details | 2 tables - gsORDER,  gsCUSTOMER | SELECT order\_id, ordered\_date,shipping\_date,order\_value,cust\_name,cust\_address,cust\_number  FROM gsORDER, gsCUSTOMER  WHERE gsORDER.customer\_id = gsCUSTOMER.customer\_id; | YES/NO |
| 2 | This query lists the total sales for each employee along with their role and department name, only if the commission is not null. | 3 tables - gsEMPLOYEE ,gsDEPARTMENT,gsSALESPEFORMANCE | SELECT  employee\_name,employee\_role,department\_name,total\_sales  FROM gsEMPLOYEE, gsDEPARTMENT, gsSALESPEFORMANCE  WHERE gsEMPLOYEE.department\_id = gsDEPARTMENT.department\_id AND gsEMPLOYEE.employee\_id = gsSALESPEFORMANCE.employee\_id AND gsSALESPEFORMANCE.commission\_money IS NOT NULL; | YES/NO |
| 3 | Displays employee id and name for all employees with a salary greater than 20,000 | 1 table - gsEMPLOYEE | SELECT employee\_id , employee\_name  FROM gsEMPLOYEE  WHERE salary > 20000 ; | YES/NO |
| 4 | This query retrieves product names and their ordered quantities, showing the number of items ordered for each product. | 3- gsPRODUCT,gsITEM,GSorder\_item | SELECT product\_name,item\_code,quantity  FROM gsPRODUCT, gsITEM, GSorder\_item  WHERE gsPRODUCT.product\_id = gsITEM.product\_id AND gsITEM.item\_id = GSorder\_item.item\_id; | YES/NO |
| 5 | Query to calculate the total salary and total commission of employees in each department: | 3 tables- gsDEPARTMENT,gsEMPLOYEE, gsSALESPEFORMANCE | SELECT gsDEPARTMENT.department\_name, SUM(gsEMPLOYEE.salary) AS total\_salary,SUM(gsSALESPEFORMANCE.commission\_money) AS total\_commission  FROM gsDEPARTMENT  JOIN gsEMPLOYEE ON gsDEPARTMENT.department\_id = gsEMPLOYEE.department\_id  LEFT JOIN gsSALESPEFORMANCE ON gsEMPLOYEE.employee\_id = gsSALESPEFORMANCE.employee\_id  GROUP BY gsDEPARTMENT.department\_name; | YES/NO |
| 6 | This query returns the product name, stock quantity, and total number of items ordered, calculated from the item\_order table. | 3 tables - gsPRODUCT , gsITEM , GSorder\_item | SELECT product\_name,stock\_quantity, SUM(quantity) AS total\_ordered  FROM gsPRODUCT,gsITEM,GSorder\_item  WHERE gsPRODUCT.product\_id = gsITEM.product\_id AND gsITEM.item\_id = GSorder\_item.item\_id  GROUP BY gsPRODUCT.product\_name, gsPRODUCT.stock\_quantity; | YES/NO |
| 7 | This query lists employees who have earned commission, displaying their employee names, roles, and commission details. | 2 tables - gsEMPLOYEE, gsSALESPEFORMANCE | SELECT employee\_name,employee\_role,commission\_percentage,commission\_money  FROM gsEMPLOYEE, gsSALESPEFORMANCE  WHERE gsEMPLOYEE.employee\_id = gsSALESPEFORMANCE.employee\_id AND gsSALESPEFORMANCE.commission\_money IS NOT NULL; | YES/NO |
| 8 | Displays the average salary of a galactic sport employee | 1 table - gsEMPLOYEE | SELECT AVG(salary)  FROM gsEMPLOYEE ; | YES/NO |
| 9 | Shows the department name and id for the department that is represented by the planet “MARS” | 1 table- gsDEPARTMENT | SELECT department\_name , department\_id  FROM gsDEPARTMENT  WHERE planet = 'MARS'; | YES/NO |
| 10 | Shows the sum of all the employees salary | 1 table - gsEMPLOYEE | SELECT SUM(salary)  FROM gsEMPLOYEE ; | YES/NO |

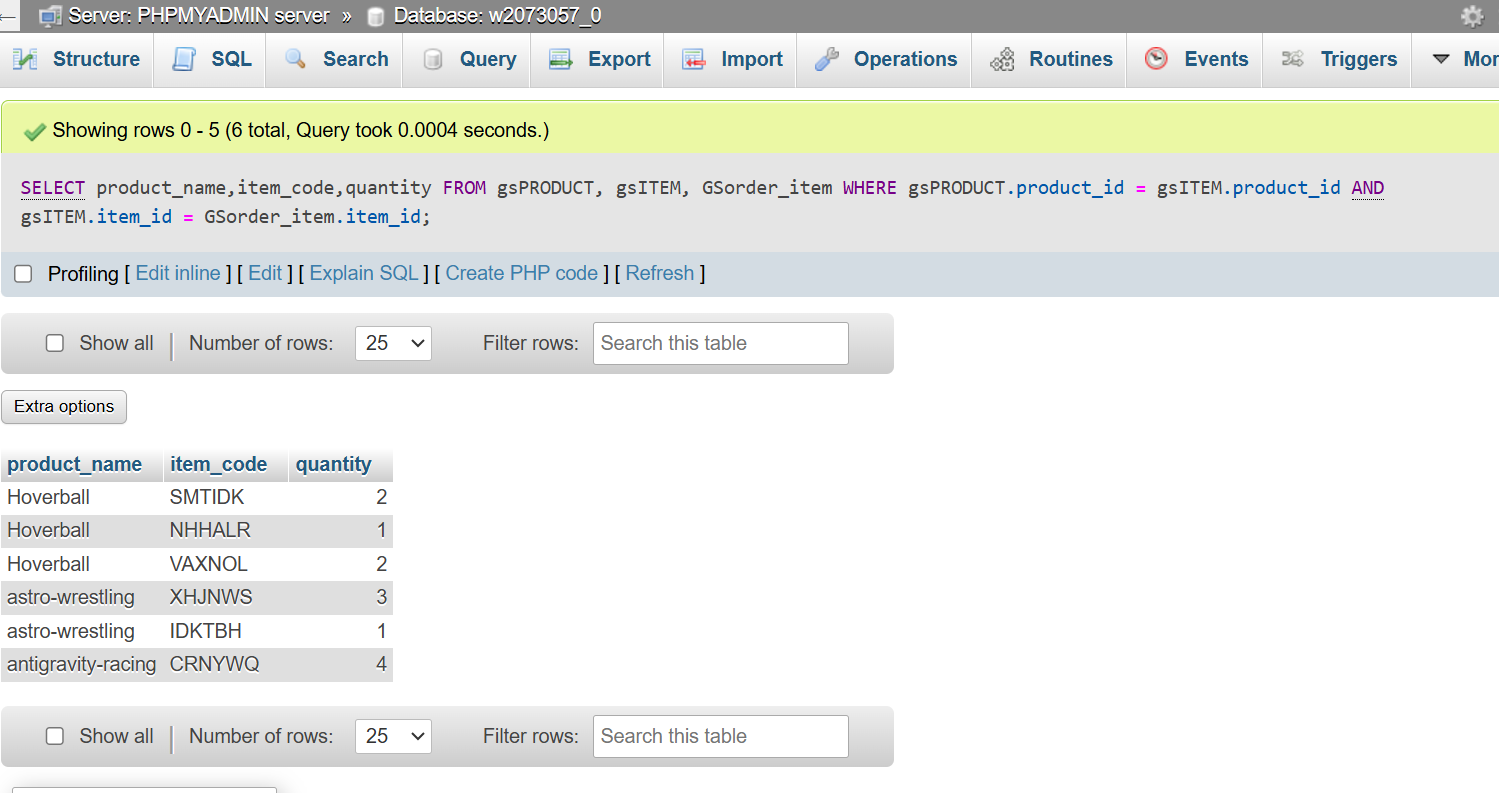
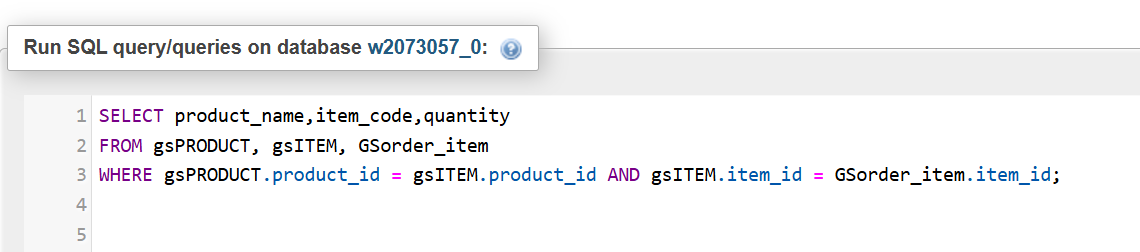
QUERY 1 AND OUTPUT:

QUERY 2 AND OUTPUT:

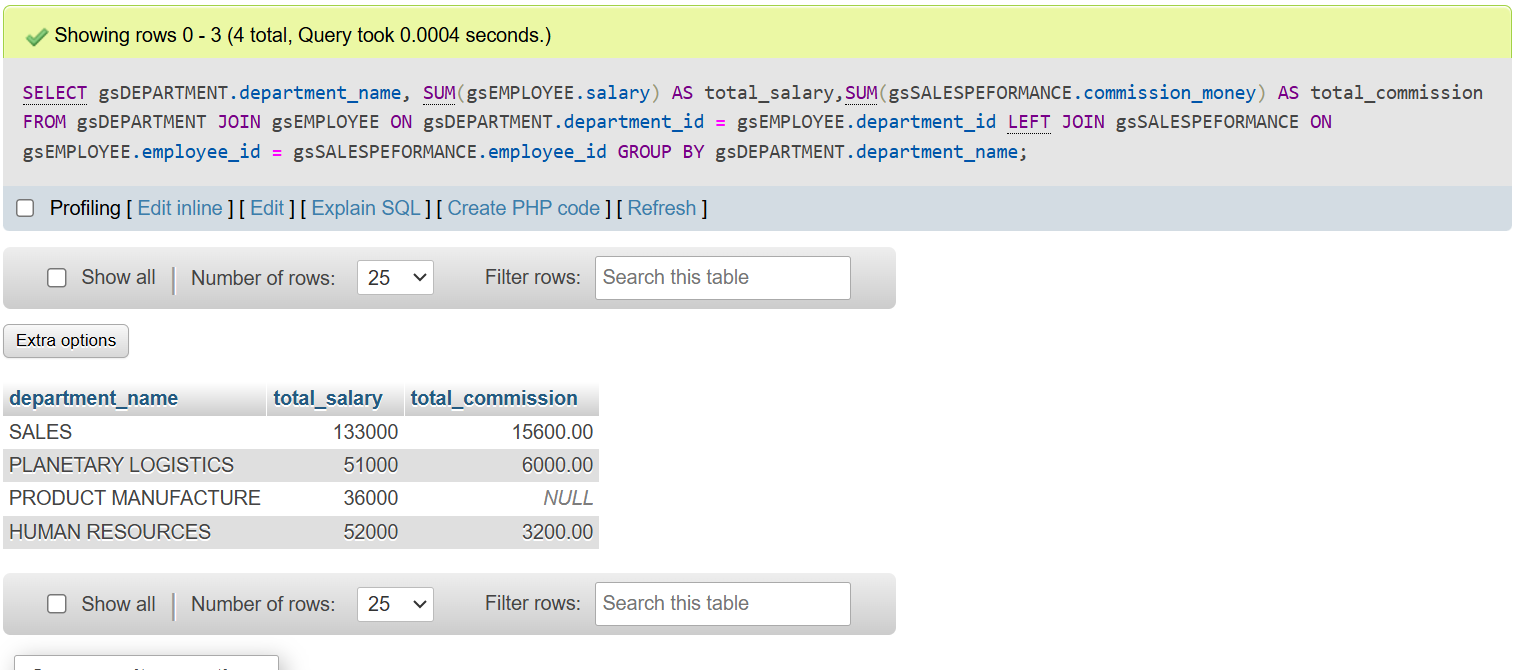
QUERY 3 AND OUTPUT:



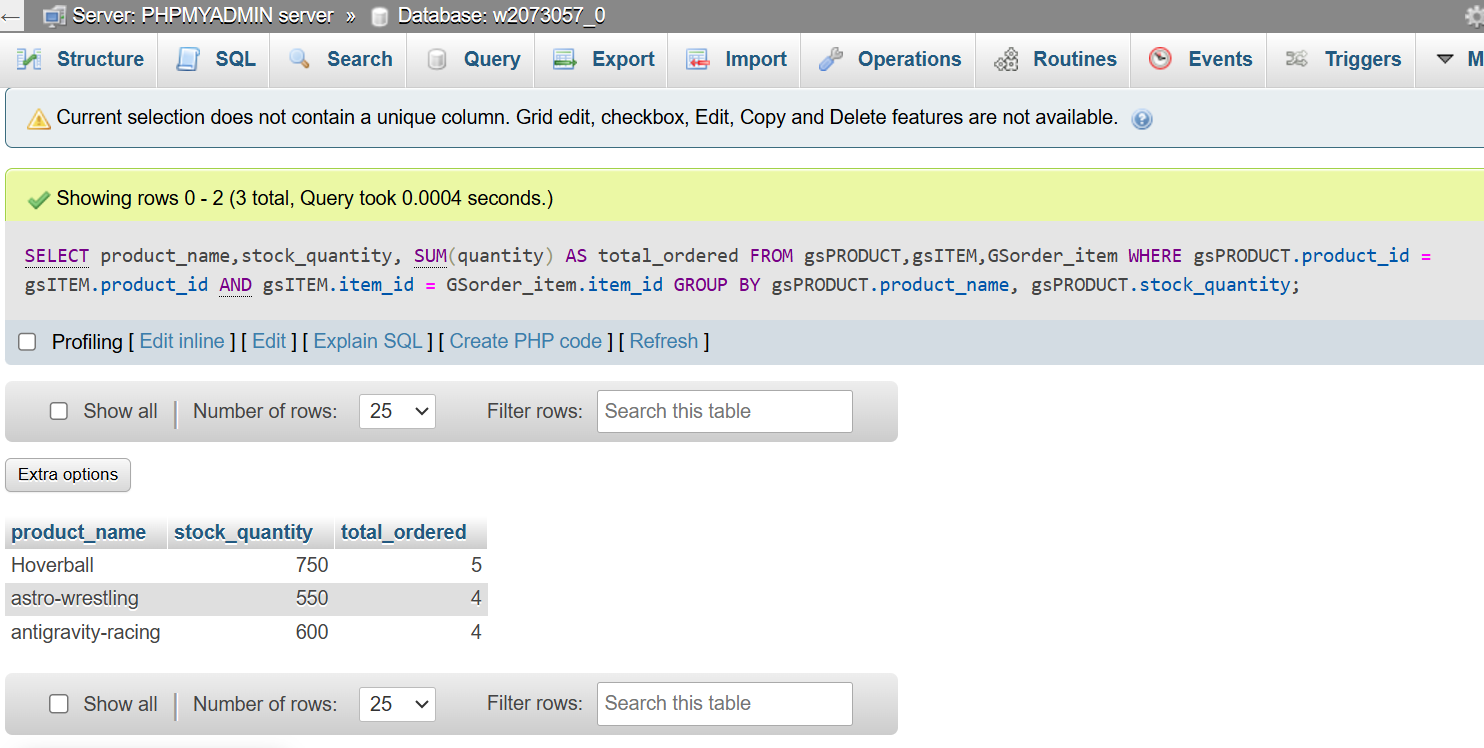
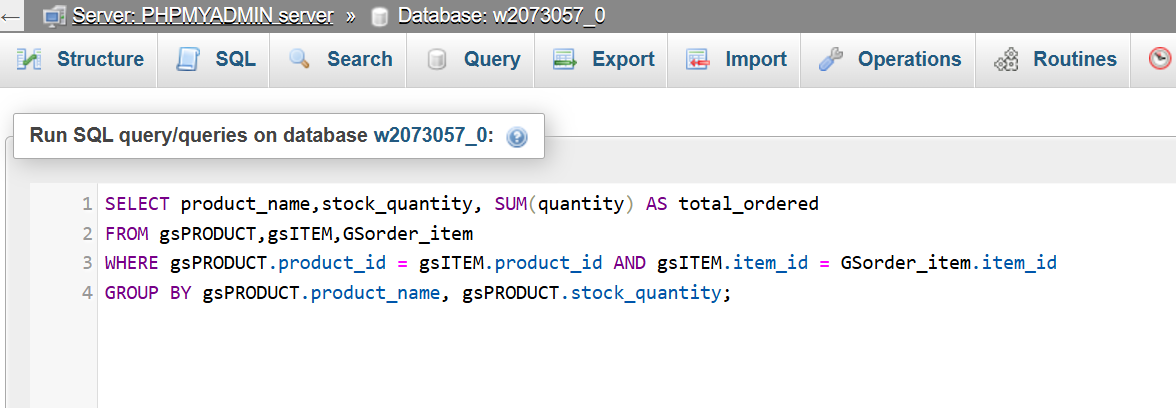
QUERY 4 AND OUTPUT:



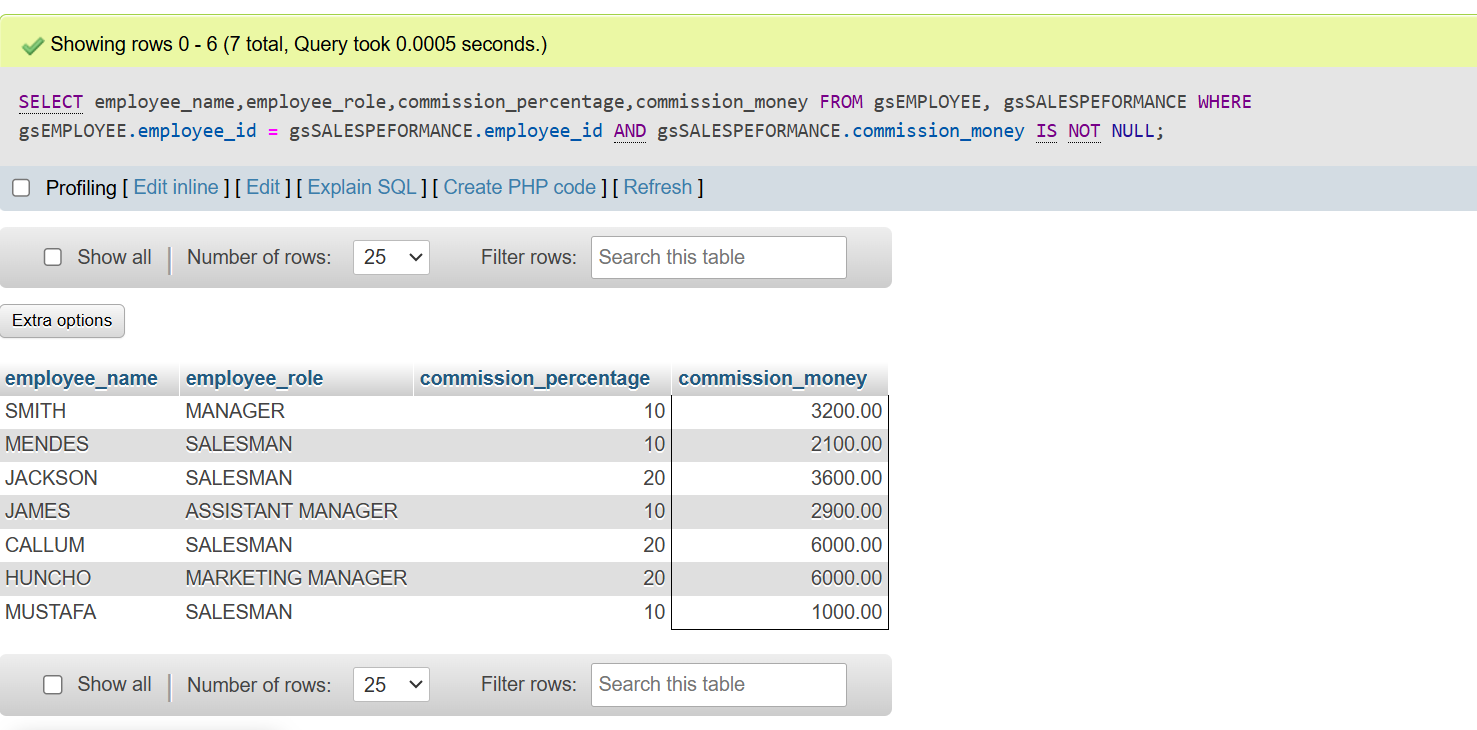
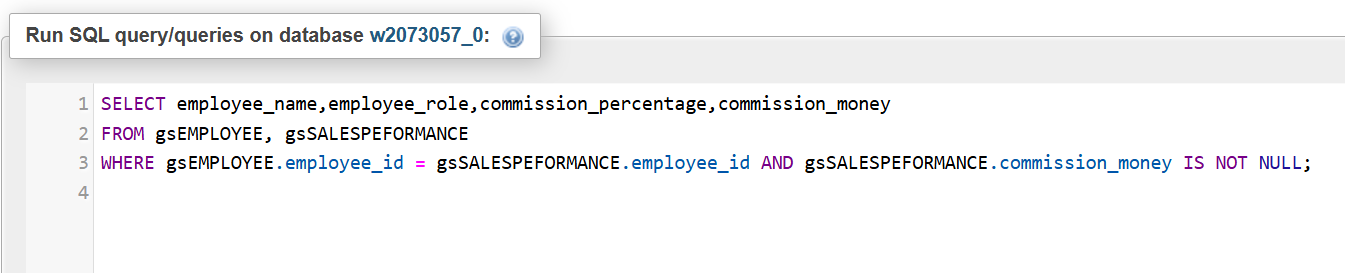
QUERY 5 AND OUTPUT:



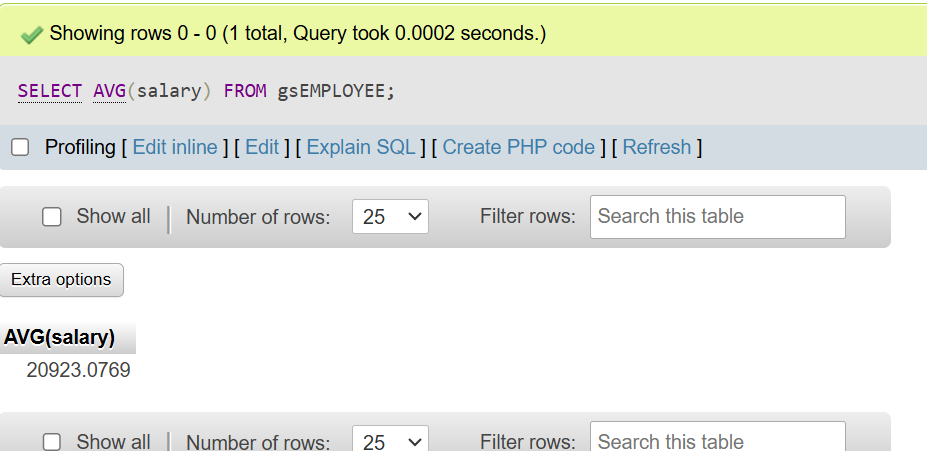
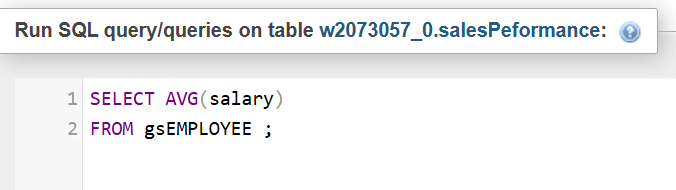
QUERY 6 AND OUTPUT:



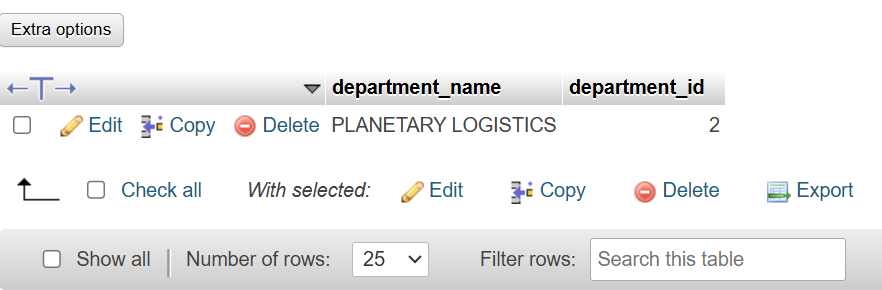
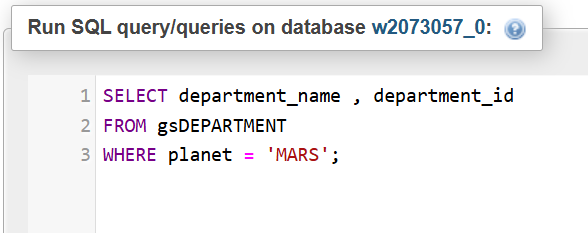
QUERY 7 AND OUTPUT:



QUERY 8 AND OUTPUT:



QUERY 9 AND OUTPUT:



QUERY 10 AND OUTPUT:

