Database Management for Retail Application

**Reports generated doing the analysis will contain the following:**

**--QUERIES**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

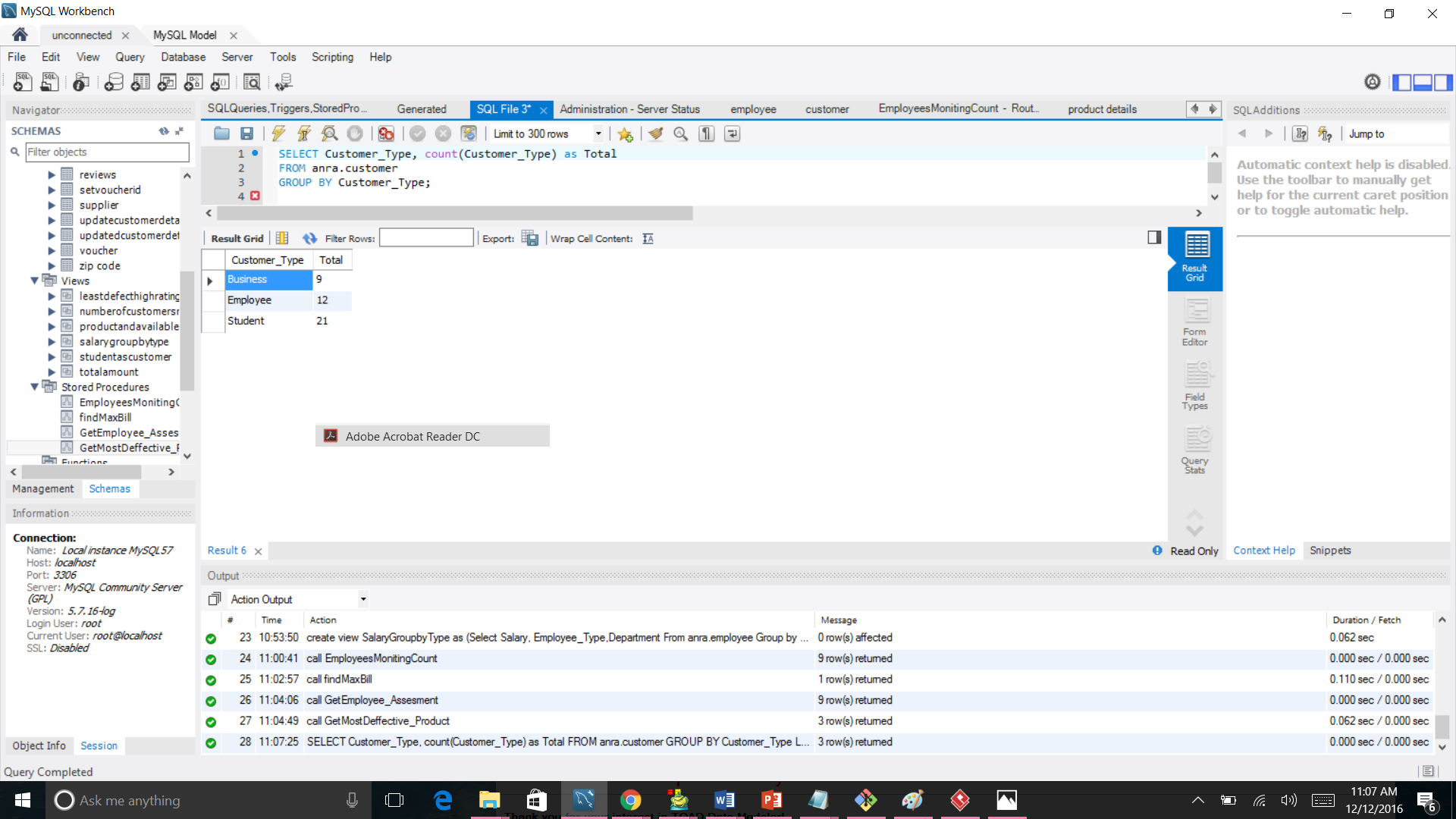
--Query1

--Find how many customers are there group by category

SELECT Customer\_Type, count(Customer\_Type) as Total

FROM anra.customer

GROUP BY Customer\_Type;



--Query2

--Find 2 appartments whose name is emerald or their id is 11003

SELECT Address\_ID, Appartment\_Name, Zipcode\_ID

FROM anra.address

WHERE (Appartment\_Name= 'Emerald' OR Zipcode\_ID = '11003')

ORDER BY Zipcode\_ID DESC

Limit 2;

--Query3

--Find the supplied product count and their group for each product

select anra.supplier.Supplier\_ID,anra.supplier.Supplier\_Name, Count(anra.product.Group\_ID) AS `Product Count`, anra.`product group`.Group\_Name

from

anra.supplier

inner join

anra.product

on anra.supplier.Supplier\_ID=anra.product.Supplier\_ID

inner join

anra.`product group`

on anra.product.Group\_ID=anra.`product group`.Group\_ID

Group by Supplier\_ID Asc

--Query4

--Total number of orders to be shipped immediate and is partially Shipped

SELECT Order\_ID, Order\_Date,`Status`,count(Order\_ID) as Total

FROM anra.orders

WHERE (Shippent\_Duration= 'Immediate' and`Status`='Partially Shipped')

ORDER BY Order\_ID DESC;

--Query5

--List of products by department which has high defect%

SELECT anra.reviews.Product\_ID,anra.product.Product\_Name,

MAX(anra.reviews.`Defect%`) As `Defect%`, anra.`product group`.Group\_Name

from

anra.reviews

inner join

anra.product

on anra.reviews.Product\_ID=anra.product.Product\_ID

inner join

anra.`product group`

on anra.product.Group\_ID=anra.`product group`.Group\_ID

Group by `product group`.Group\_ID

--Query6

--Total amount of revenue earned with respect to their purchasing modes

Select count(Payment\_Mode) As Total\_Cutomers, anra.payment.Payment\_Mode,Sum(anra.bill.Amount\_Paid)

As Total\_Amount

from anra.payment

inner join

anra.bill

on anra.payment.Payment\_ID=anra.bill.Payment\_ID

group by Payment\_Mode

--Query7

--Find The quantity of products available whose status is in progress and shipment duration is immediate

SELECT anra.product.Product\_Name,anra.product.Available\_Number,`order product`.Quantity,orders.Order\_Date, orders.`Status`,orders.Shippent\_Duration

from

anra.orders

inner join

anra.`order product`

on anra.orders.Order\_ID=anra.`order product`.Order\_ID

inner join

anra.product

on anra.`order product`.Product\_ID=anra.product.Product\_ID

where

orders.`Status`='In Progress' and orders.Shippent\_Duration='Immediate';

--Query8

--Find the names and defect% order by defect%

SELECT Product\_Name,`Defect%`

FROM anra.product

INNER JOIN anra.reviews

ON anra.reviews.Product\_ID=anra.product.Product\_ID

ORDER BY `Defect%` Desc;

--Query9

--Find Customers payment ID,mode, vocher applied and their visist number

SELECT anra.bill.Voucher\_id, anra.payment.Payment\_ID, anra.payment.Payment\_Mode, anra.payment.Visit\_Number

FROM anra.bill,anra.payment

WHERE anra.payment.Payment\_ID=anra.bill.Payment\_ID

AND anra.bill.Amount\_Paid> 1000;

--Query10

--Find product and their respective colour

SELECT Product\_Name, Colour

FROM anra.`product details`

INNER JOIN anra.product

ON anra.`product details`.Product\_ID=anra.product.Product\_ID order by colour

--Query11

--Find the product Names their respective groups

SELECT

Group\_Name, Product\_Name

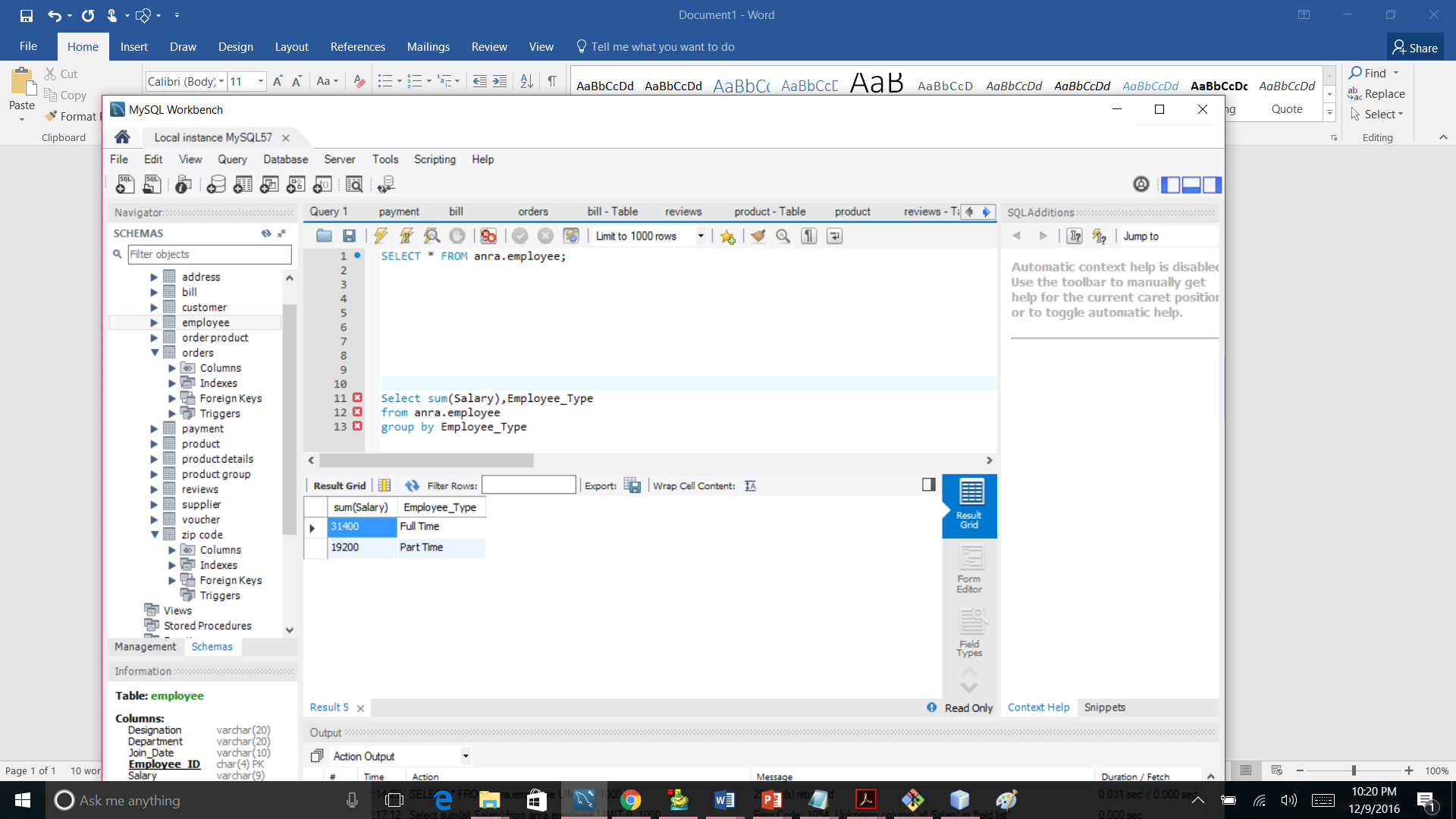
FROM anra.`product group`

INNER JOIN anra.product

ON anra.`product group`.Group\_ID=anra.product.Group\_ID Order by Group\_Name

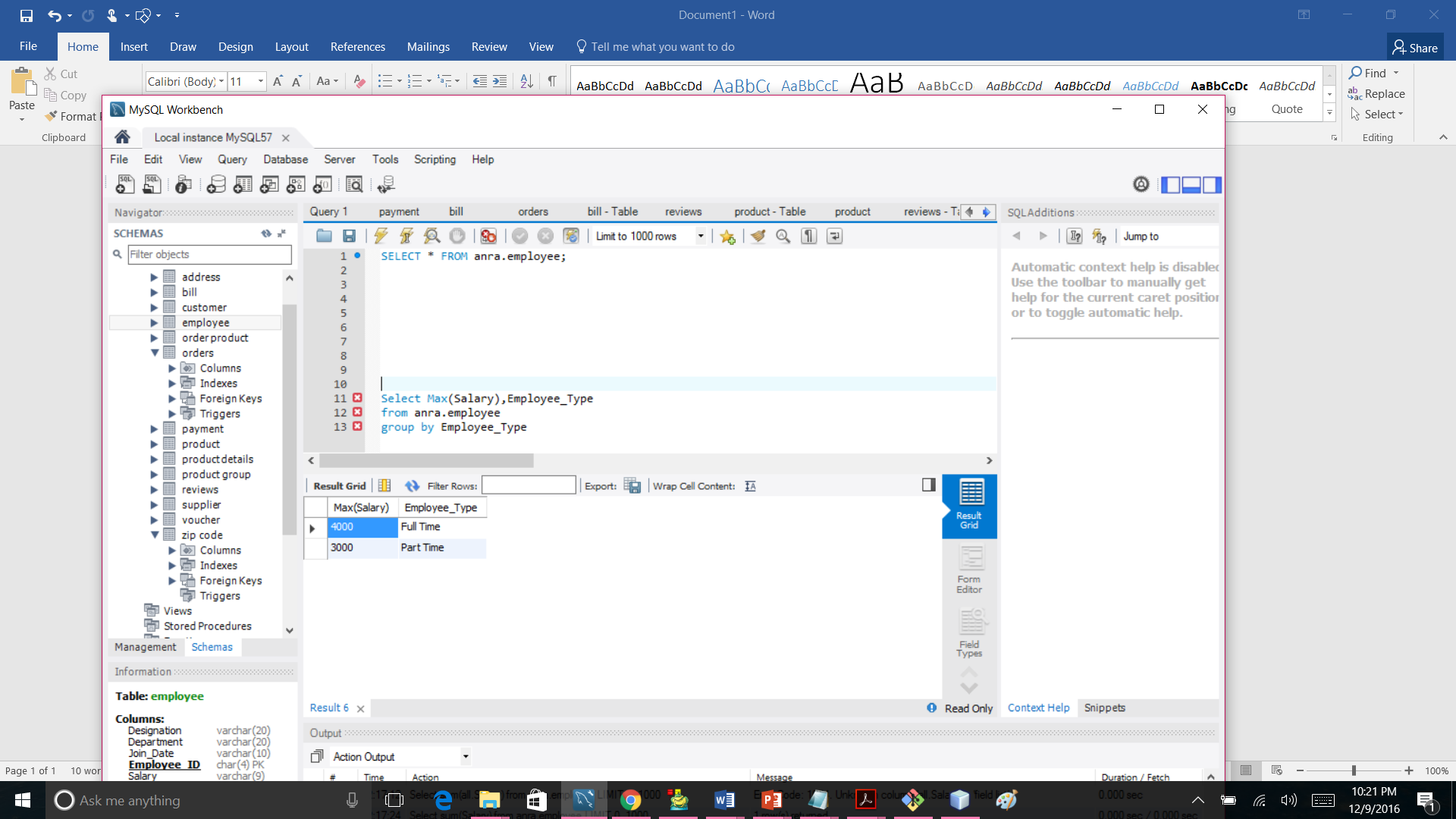
--Query12

--Find Total salaries payed to each employee type



--Query13

--Find Maximum salaries payed to each employee type



**TRIGGERS**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

--Trigger1

--Find Customer name and updated time on customers table

create table

UpdateCustomerDetails

(Customer\_id int, First\_Name varchar(20), update\_time Datetime)

delimiter \\

create trigger UpdateCustomerDetails\_trigger

after update on customer

for each row

begin

declare new\_date datetime;

set new\_date=now();

insert into UpdateCustomerDetails(Customer\_id,First\_Name,update\_time)

values(old.Customer\_ID,old.First\_Name, new\_date);

end \\

--when customers details are updated the trigger is set

update customer set Email\_Address='naynaa@gmail.com'

where Customer\_ID=10000

--updates can be seen in the newly created table

select \* from UpdateCustomerDetails

--Trigger2

--Find the newly changed colour for the product

Create table

AddedProductColour

(Product\_id char(5), Colour varchar(20))

delimiter \\

create trigger AddedProductColour\_trigger

after update on anra.`product details`

for each row

begin

insert into AddedProductColour(Product\_id, Colour )

values(old.Product\_id,Colour);

end \\

**--Stored\_Procedures**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

--1

--Stored procedure to find most deffective product from each product group

call GetMostDeffective\_Product()

delimiter//

Create Procedure GetMostDeffective\_Product()

Begin

SELECT anra.reviews.Product\_ID,anra.product.Product\_Name,MAX(anra.reviews.`Defect%`) As `Defect%`, anra.`product group`.Group\_Name

from

anra.reviews

inner join

anra.product

on anra.reviews.Product\_ID=anra.product.Product\_ID

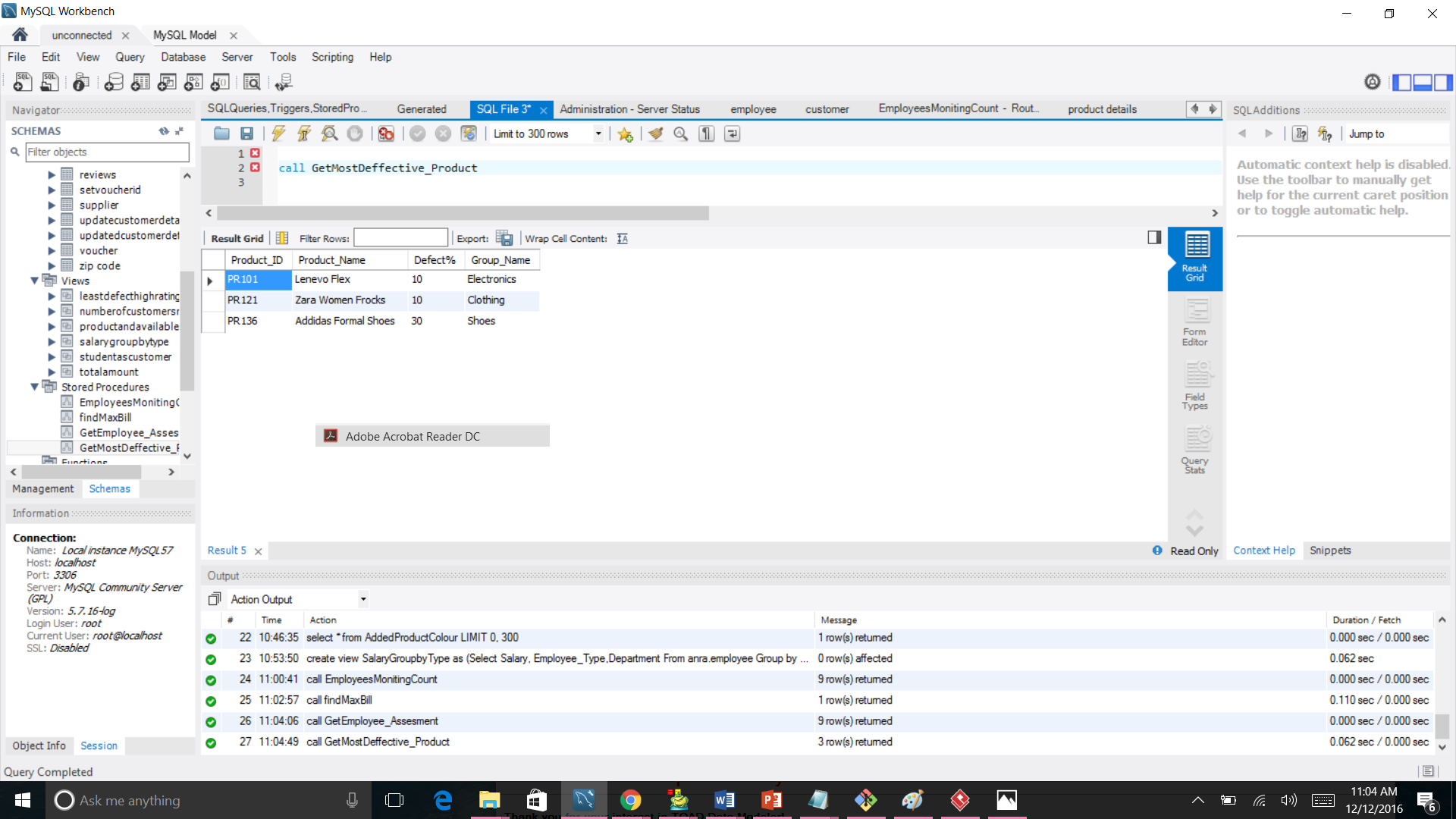
inner join

anra.`product group`

on anra.product.Group\_ID=anra.`product group`.Group\_ID

Group by `product group`.Group\_Name

end//



--2

--Stored procedure for employee assesment

call GetEmployee\_Assesment ()

delimiter//

Create Procedure GetEmployee\_Assesment ()

Begin

Select count(anra.customer.Customer\_ID) as Total\_Cutomers ,anra.customer.Employee\_ID,anra.employee.Designation,Salary

from anra.customer

Inner Join

anra.employee

on

anra.employee.Employee\_ID=anra.customer.Employee\_ID

group by Employee\_ID

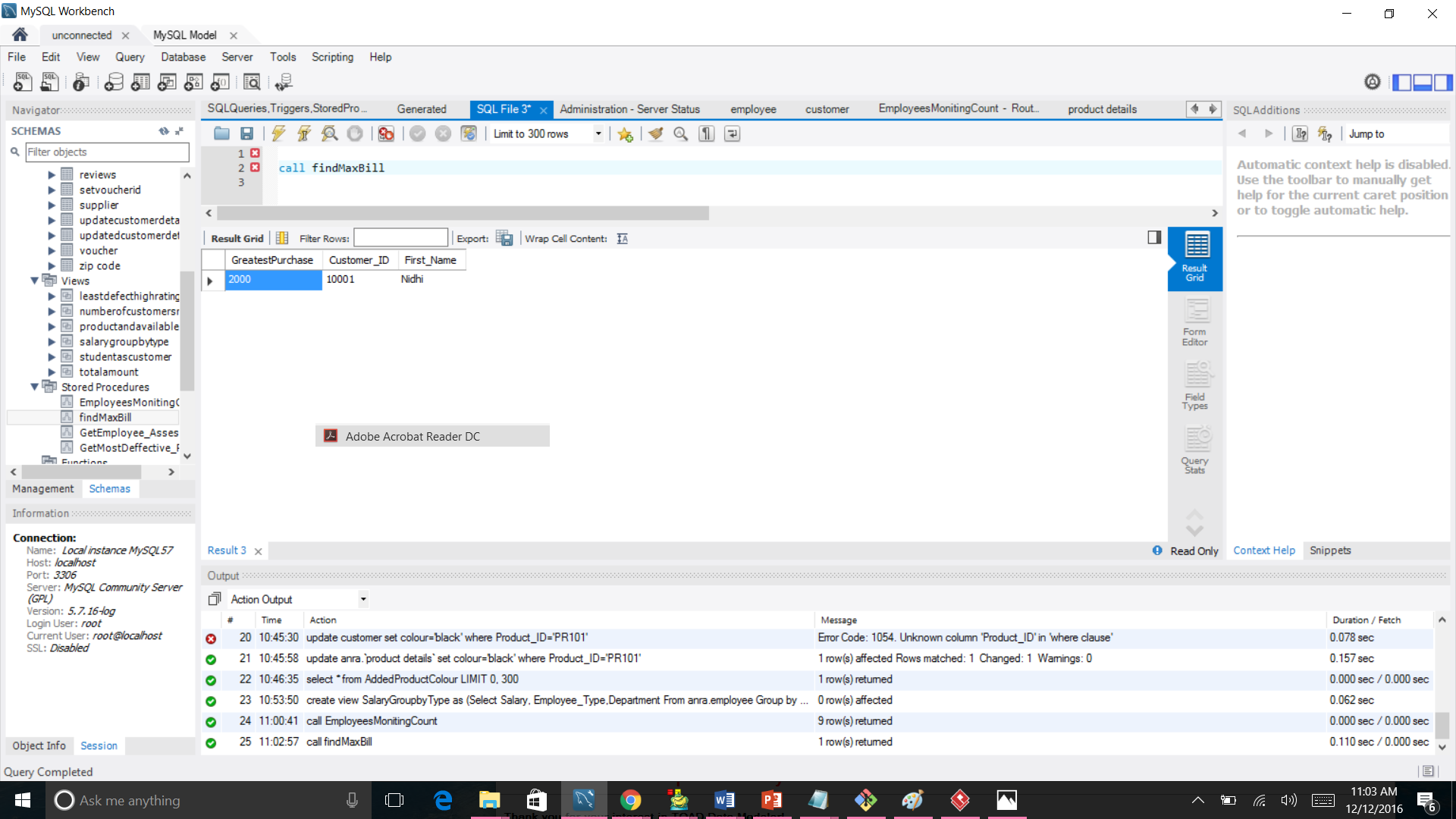
order by count(anra.customer.Customer\_ID) ;

end//

--3

--Stored procedure to find customer details of highest Amount\_Paid

call findMaxBill()



delimiter//

Create Procedure findMaxBill()

Begin

select Max(anra.bill.Amount\_Paid) As GreatestPurchase, anra.customer.Customer\_ID,anra.customer.First\_Name from anra.bill

inner join

anra.payment

on anra.payment.Payment\_ID=anra.bill.Payment\_ID

inner join

anra.Customer

on anra.payment.Customer\_ID=anra.customer.Customer\_ID;

end//

--4

-- Call EmployeesMonitingCount()

CREATE DEFINER=`root`@`localhost` PROCEDURE `EmployeesMonitingCount`()

BEGIN

Select count(anra.customer.Customer\_ID) as Total\_Customers, anra.customer.Employee\_ID,anra.employee.Designation

From anra.customer

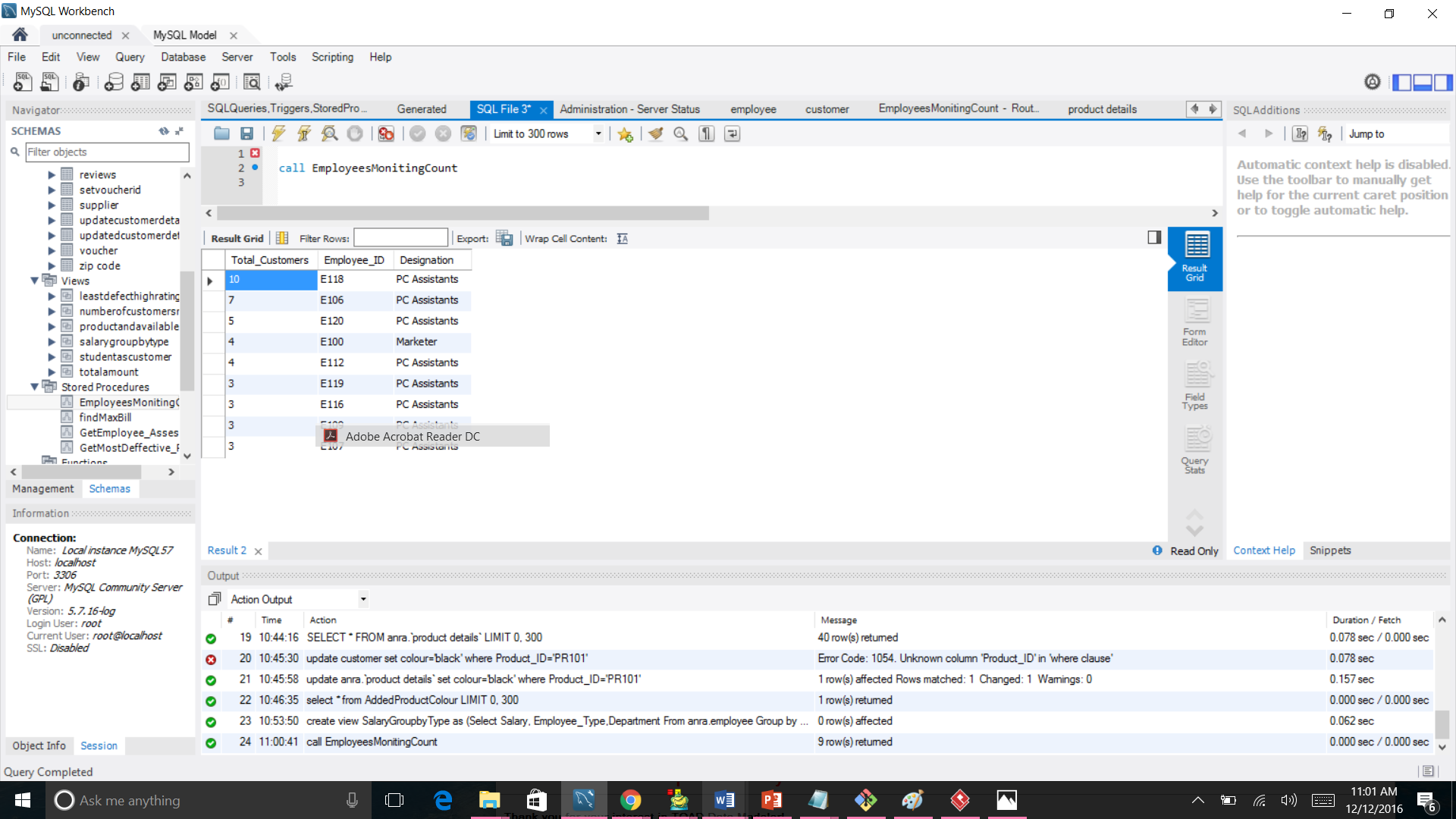
Inner Join

anra.employee

on anra.employee.Employee\_ID=anra.customer.Employee\_ID

group by anra.employee.Employee\_ID

order by count(anra.customer.Customer\_ID) desc;



END

**--Views**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_--1

--Student as customer views

create view StudentAsCustomer as ( select \* from customer where Customer\_Type = 'Student')

--2

--Product details with minimum defect% and high rating

create view LeastDefectHighRating as

(select product.Product\_Name, min(reviews.`Defect%`), max(reviews.Quality\_Rating)

from anra.reviews

inner join

anra.product

on anra.reviews.Product\_ID=anra.product.Product\_ID)

--3

--Products and the colours available

create view ProductandAvailableColours as

(SELECT Product\_Name, Colour

FROM anra.`product details`

INNER JOIN anra.product

ON anra.`product details`.Product\_ID=anra.product.Product\_ID order by colour)

--4

--Distinct employee departments whose salary is greater than 1200 limit to 4

create view DistinctEmployeeDepartments as

(Select DISTINCT anra.employee.Designation,anra.employee.Employee\_Name,anra.employee.Department

From anra.employee

Where anra.employee.salary>1200 limit 4)

--5

--Create view TotalAmount as Total amount of revenue earned with respect to their purchasing modes

create view as totalamount

(Select count(Payment\_Mode) As Total\_Cutomers, anra.payment.Payment\_Mode,Sum(anra.bill.Amount\_Paid)

As Total\_Amount

from anra.payment

inner join

anra.bill

on anra.payment.Payment\_ID=anra.bill.Payment\_ID

group by Payment\_Mode)

--6

--Create view to know the employee designation and id and total number of customers monitored by each employee in descending order

create view NumberOfCustomersMonitered as

(Select count(anra.customer.Customer\_ID) as Total\_Customers, anra.customer.Emploee\_ID,anra.emploee.Designation

From anra.customer

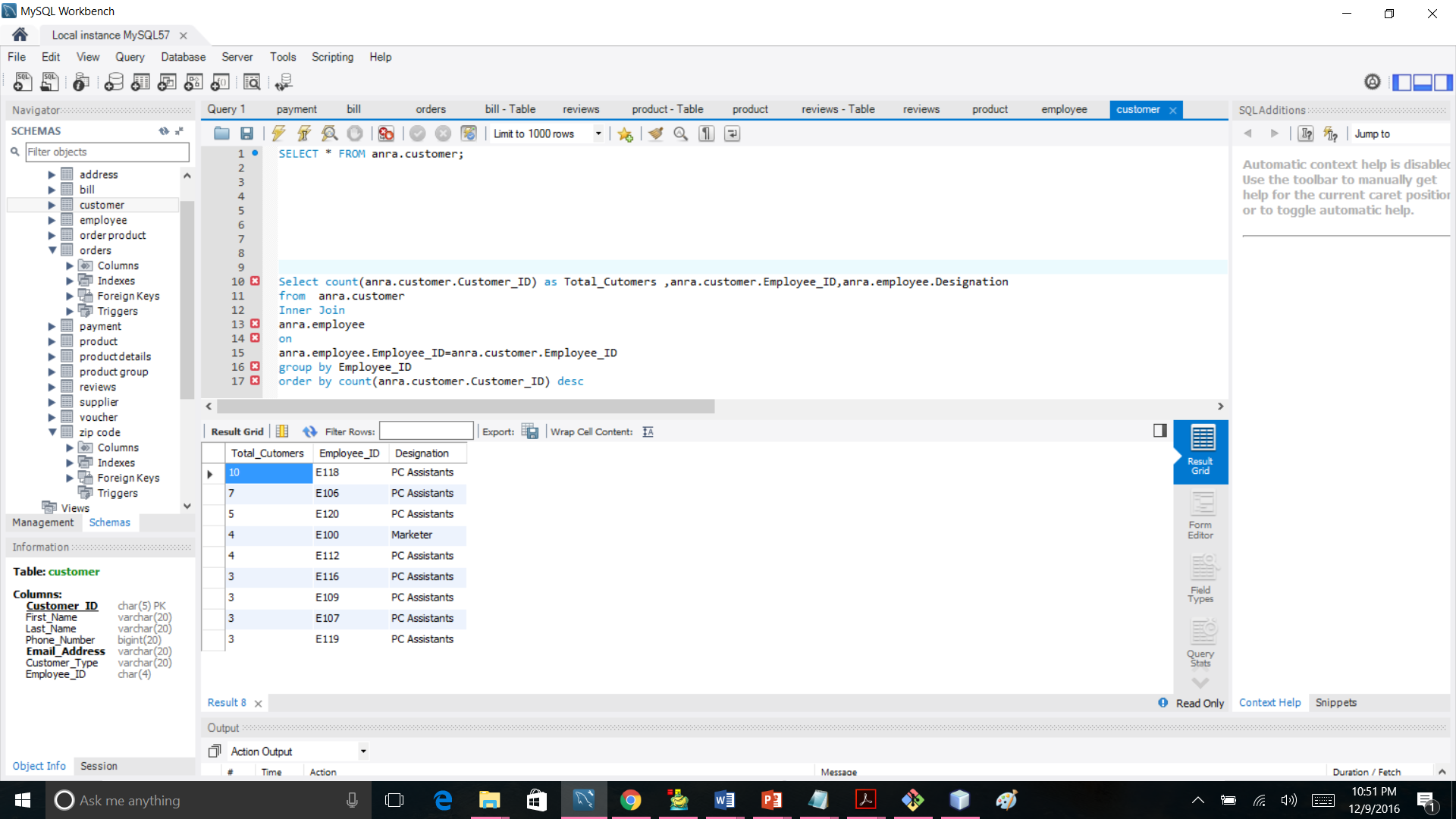
Inner Join

anra.employee

on anra.employee.Employee\_ID=anra.customer.Employee\_ID

group by Employee\_ID

order by count(anra.customers.Customer\_ID) desc)



--7

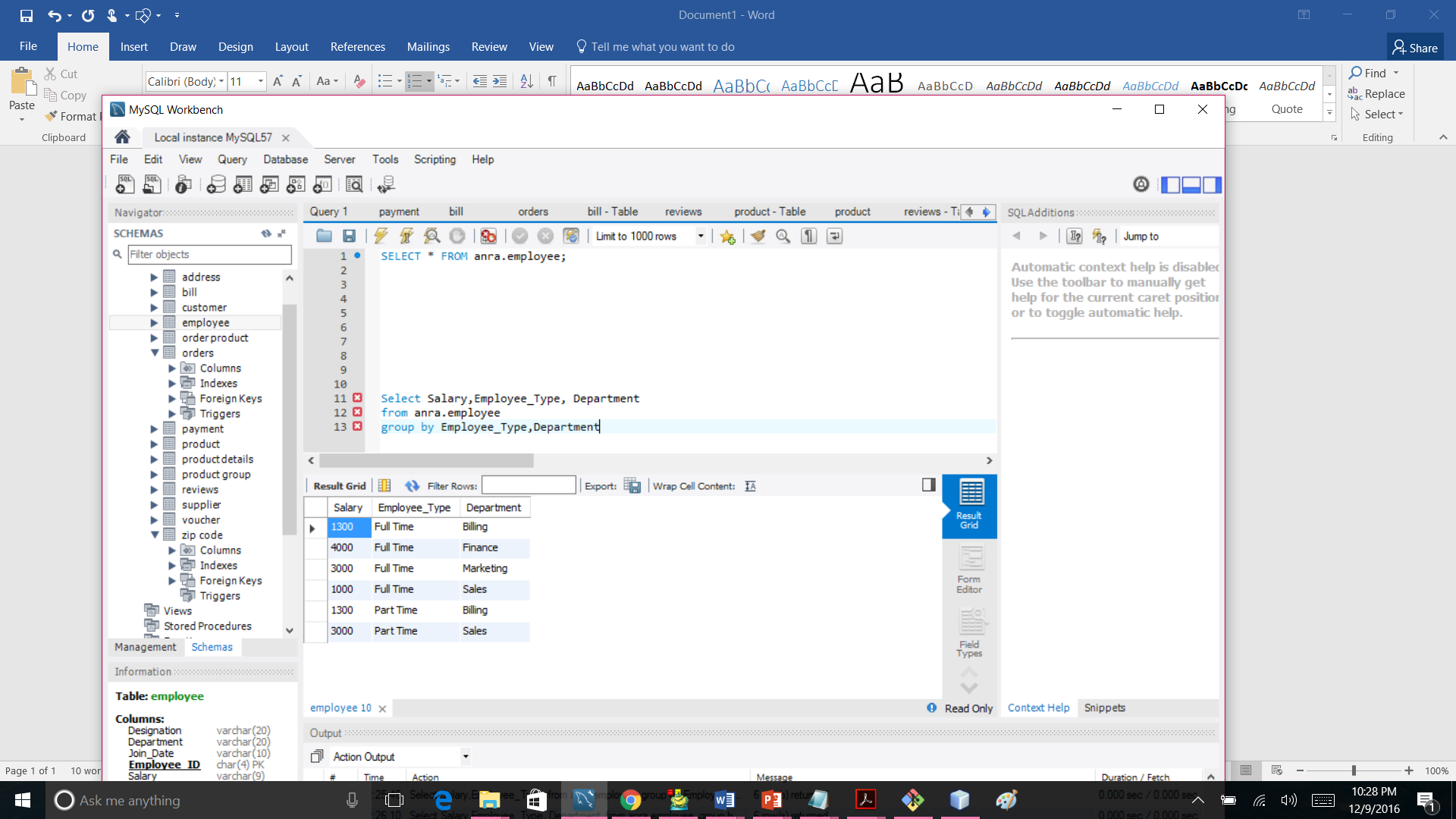
--View Salary of employee group by type and department

create view SalaryGroupbyType as

(Select Salary, Employee\_Type,Department

From anra.employee

Group by Employee\_Type,Department)



--8

-- Find The quantity of products available whose status is in progress and shipment duration is immediate

create view OrderStatusandAvailableQuantity as

(SELECT anra.product.Product\_Name,anra.product.Available\_Number,`order product`.Quantity,orders.Order\_Date, orders.`Status`,orders.Shippent\_Duration

from

anra.orders

inner join

anra.`order product`

on anra.orders.Order\_ID=anra.`order product`.Order\_ID

inner join

anra.product

on anra.`order product`.Product\_ID=anra.product.Product\_ID

where

orders.`Status`='In Progress' and orders.Shippent\_Duration='Immediate') 