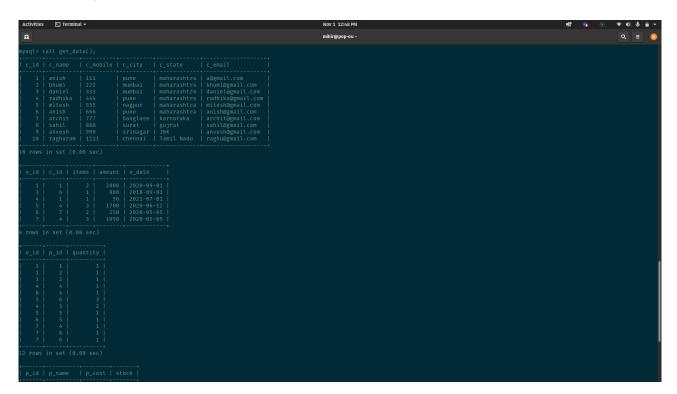
## **Assignment No 6**

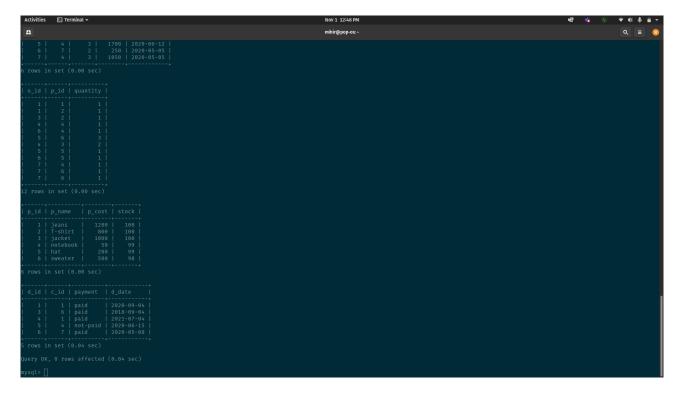
Q.Write and execute PL/SQL stored procedure and function to perform a suitable task on Order Management DB.



1) Stored procedure get\_data to fetch all tables..

```
DELIMITER $$
CREATE PROCEDURE get_data()
BEGIN
SELECT * from customer,
SELECT * from orders,
SELECT * from orders_product,
SELECT * from product,
END$$
DELIMITER;
```





2) Loop used to store all email ids in a stored procedure.

mysql> create procedure createmailList(INOUT emailList varchar(4000)) begin declare finished integer default 0; declare emailAddress varchar(100) default \*\*; declare curEmail cursor for select \_email from customer; declare continue handler for not found set finished=1; open curEmail; LOOP fetch curEmail into emailAddress; IF finished=1 THEN leave getEmail; END IF; set emailList = concar(emailList,"; emailAddress); end LOOP getEmail; close curEmail; END\$¶

```
mysql> call createEmailList(@emailList);
Query OK, 0 rows affected (0.00 sec)

mysql> set @emailList="";
Query OK, 0 rows affected (0.00 sec)

mysql> call createEmailList(@emailList);
Query OK, 0 rows affected (0.00 sec)

mysql> call createEmailList(@emailList);
Query OK, 0 rows affected (0.00 sec)

mysql> select @emailList;

| @emailList

| ;a@gmail.com;bhumi@gmail.com;daniel@gmail.com;radhika@gmail.com;mitesh@gmail.com;anish@gmail.com;archit@gmail.com;sahil@gmail.com;anvesh@gmail.com;raghu@gmail.com |
1 row in set (0.00 sec)
```

3) Procedure to insert a new product into table with error handling into the stored procedure.

mysql> create procedure InsertNewProduct(IN inp\_id int, IN inp\_name varchar(100), IN inp\_cost int, IN instock int) begin declare exit handler for 1062 select 'Duplicate keys error encountered d' Message; declare exit handler for SQLEXCEPTION select 'SQLEXCEPTION encountered into product( no is a superior of the second of t