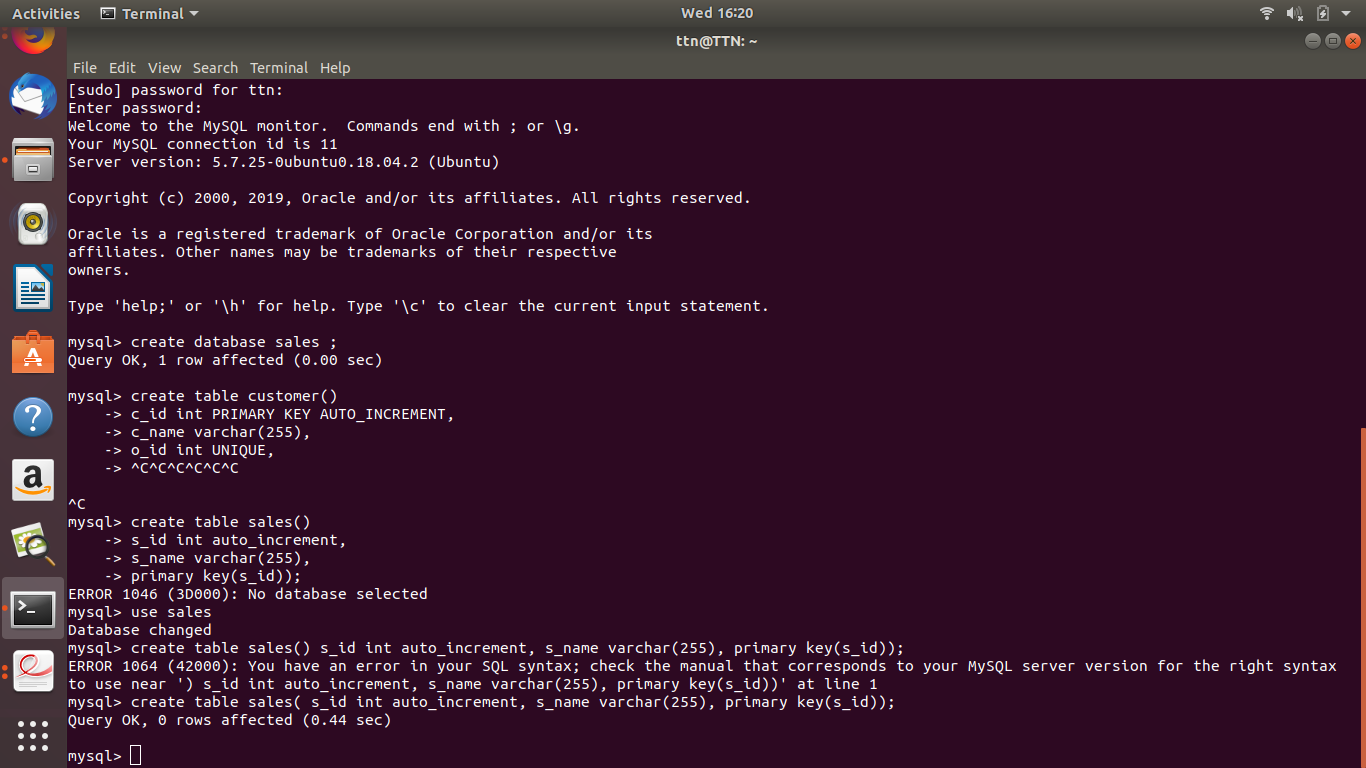
Assessment - 3

Problem Statement: There can be multiple customers, who can place multiple orders on the site. Now a sales person can handle these orders will distribute into multiple sales persons (One order will be assign to one salesperson only). So a sales person can have multiple orders of multiple customers

1. Create Database

1. Design Schema

Orders

|  |  |
| --- | --- |
| O\_id | Primary key |
| o\_name | Order details |
| c\_id | Customer id (foreign key) |
| S\_id | Sales id(foreign key) |

Customer

|  |  |
| --- | --- |
| c\_id | Primary key |
| c\_name | customer details |

Salesperson

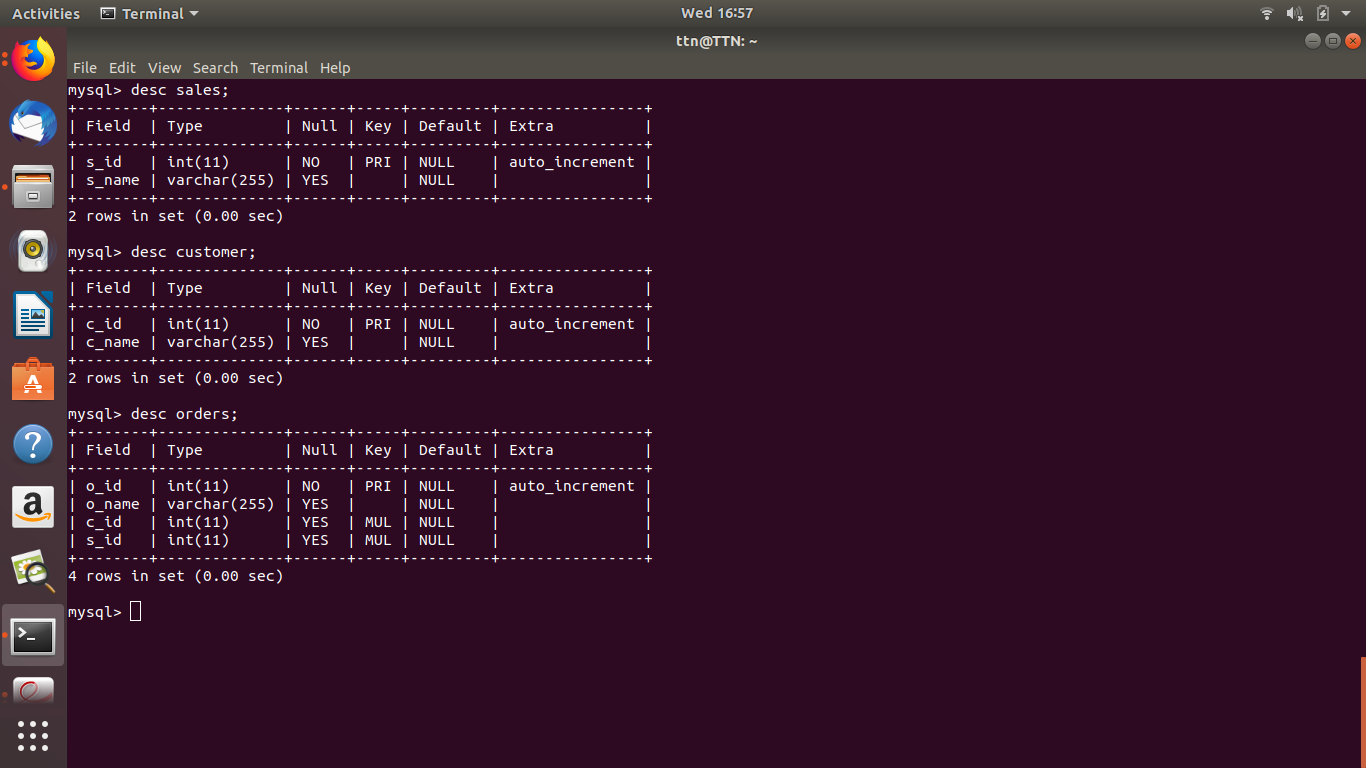
|  |  |
| --- | --- |
| s\_id | Primary key |
| s\_name | salesperson details |

1. Create tables

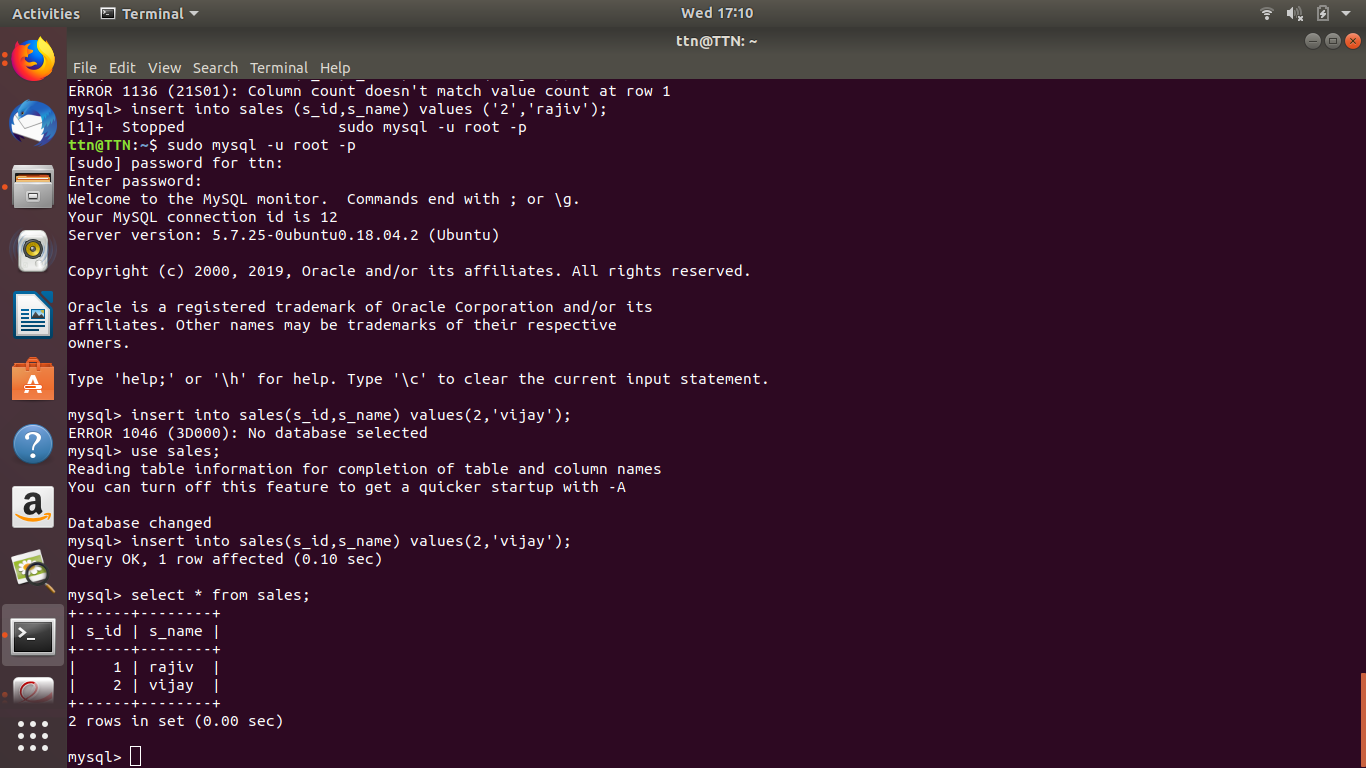
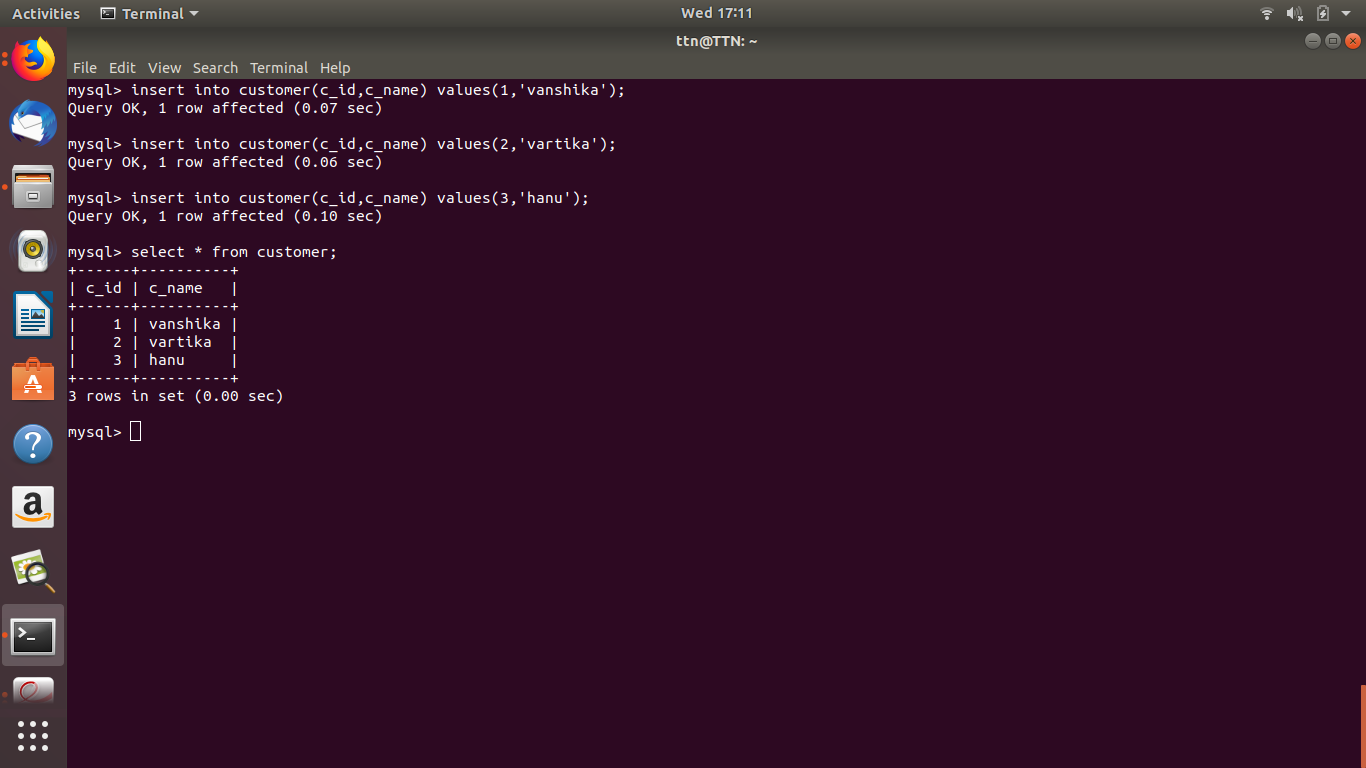
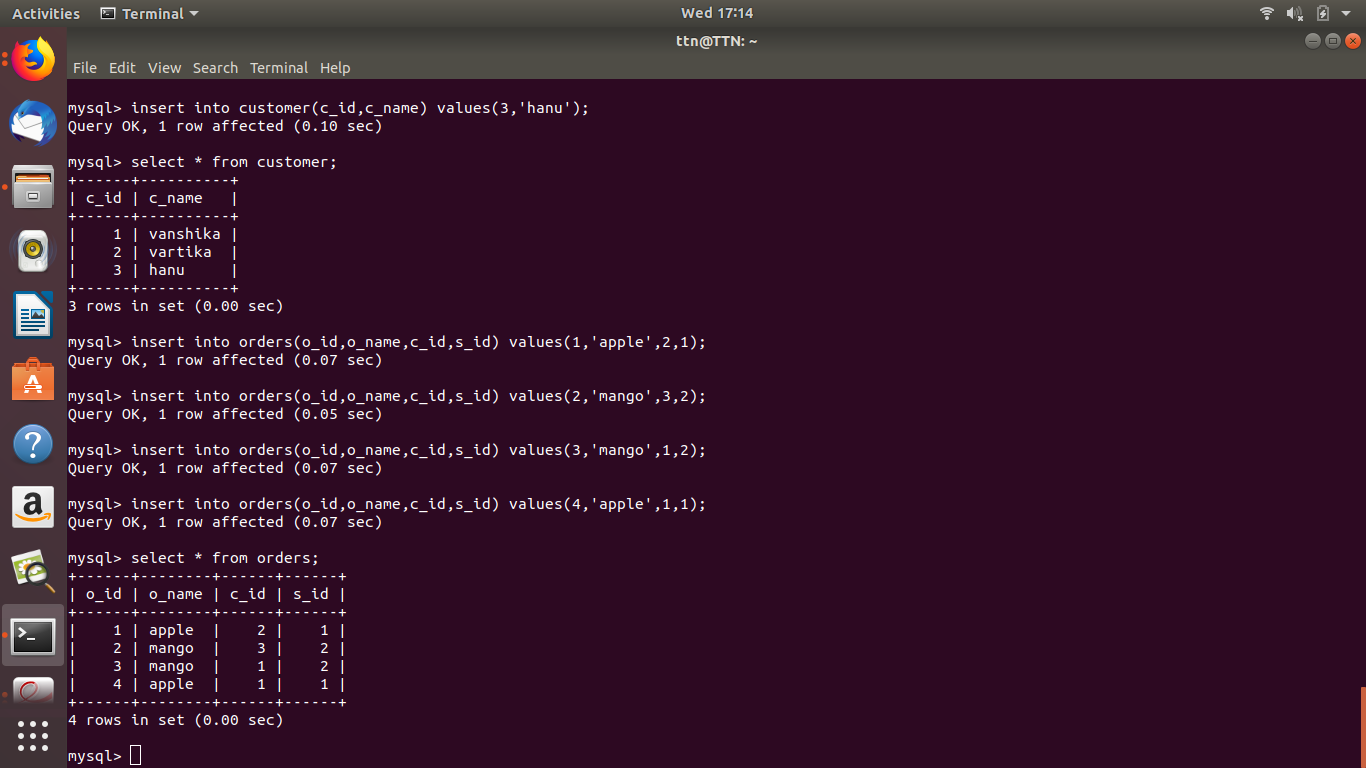
**Order table** - create table orders( o\_id int auto\_increment, o\_name varchar(255), c\_id int , s\_id int , primary key(o\_id), FOREIGN KEY (c\_id) REFERENCES customer(c\_id), FOREIGN KEY (s\_id) REFERENCES sales(s\_id));

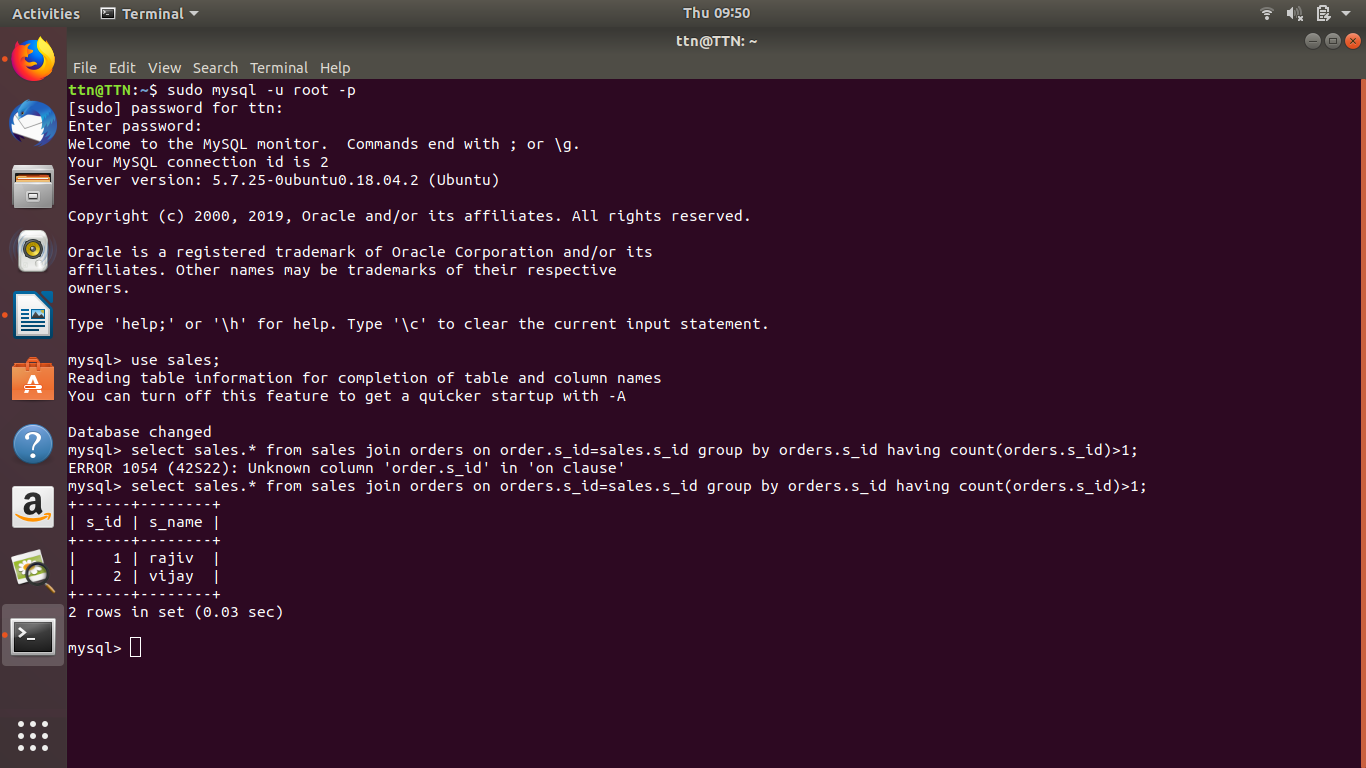
**Customer table -**  create table customer( c\_id int auto\_increment,c\_name varchar(255), primary key(c\_id));

**Sales table -** create table sales( s\_id int auto\_increment, s\_name varchar(255), primary key(s\_id));

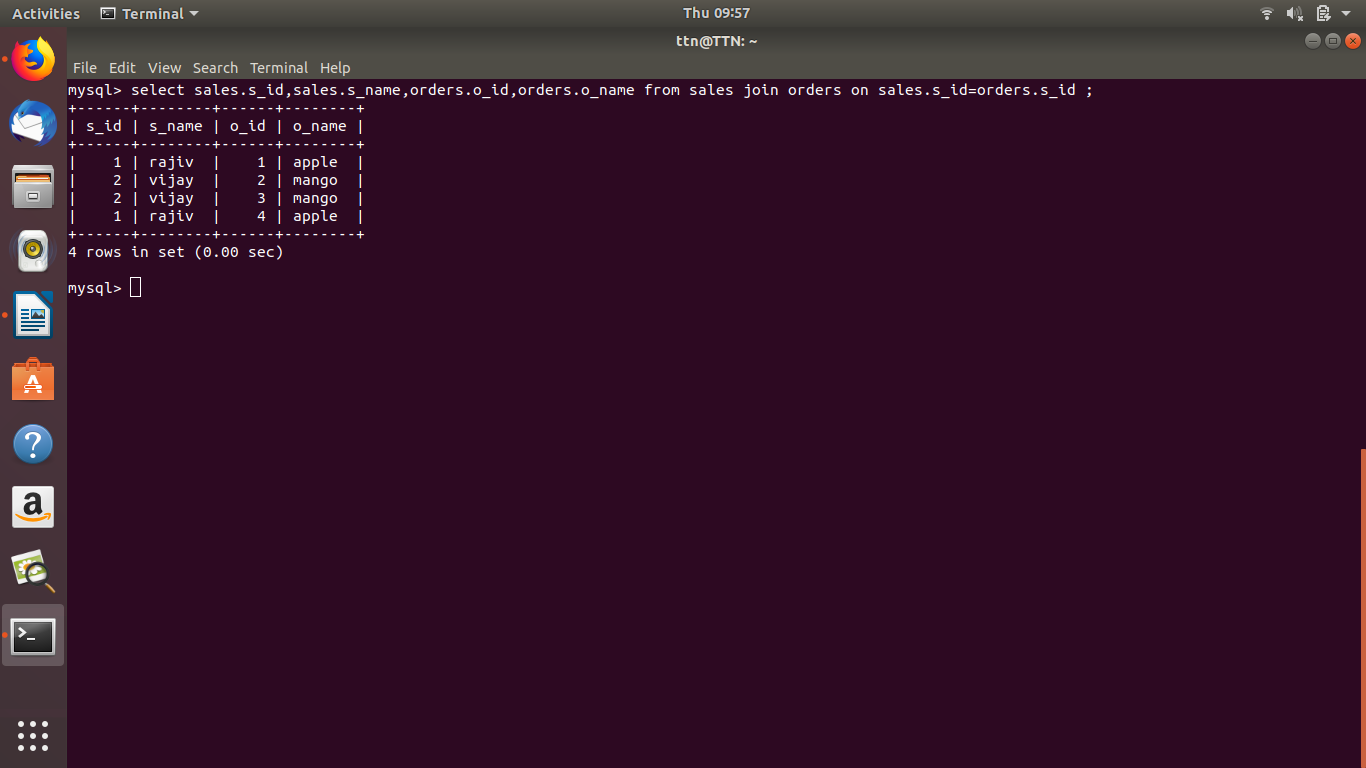
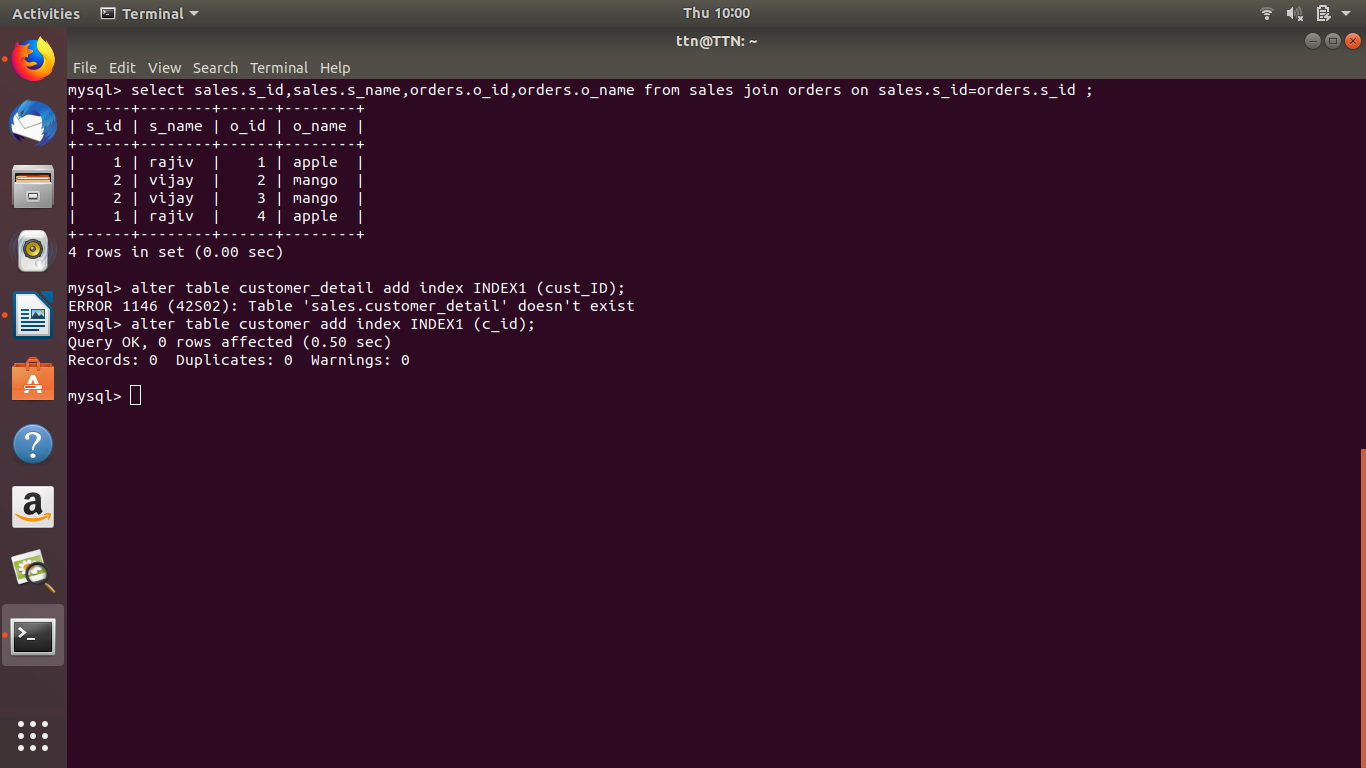


1. Insert sample data



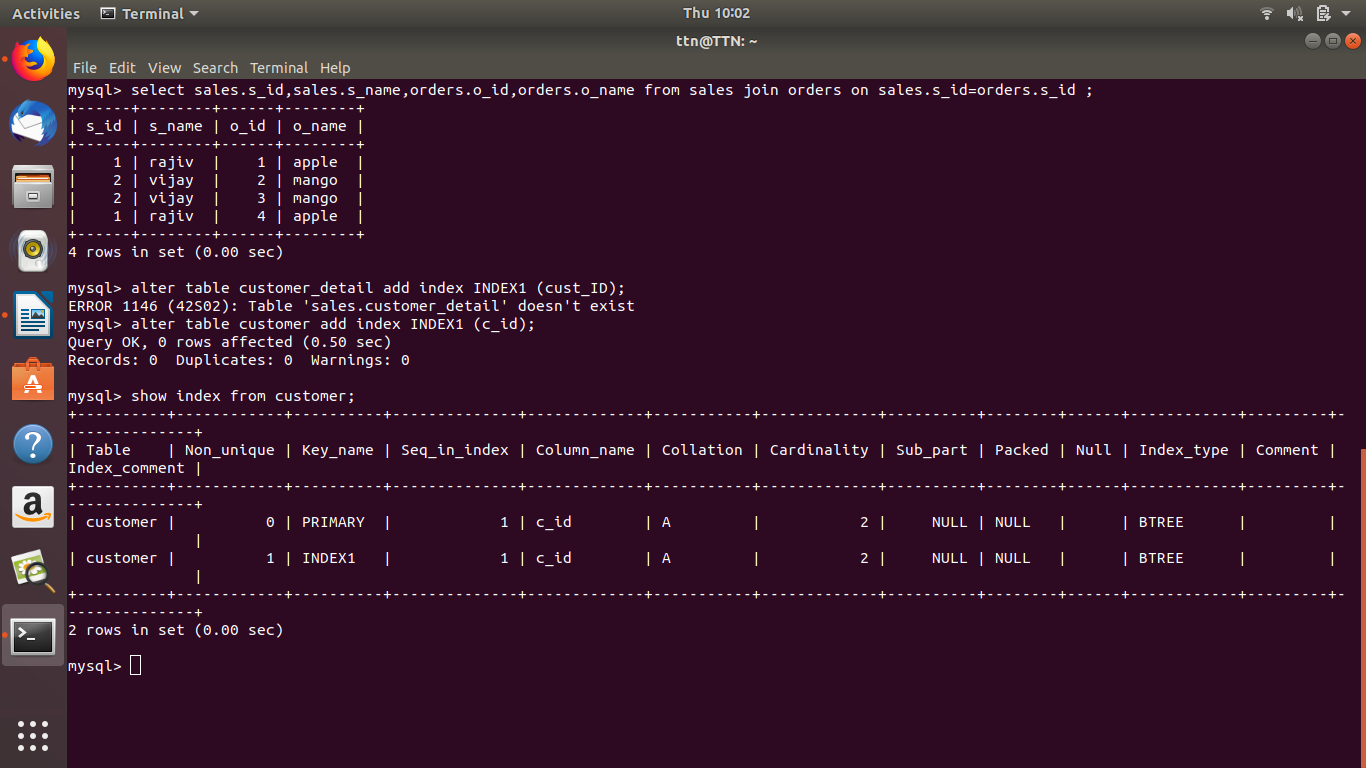
1. Find the sales person have multiple orders. - 

select sales.\* from sales join orders on orders.s\_id=sales.s\_id group by orders.s\_id having count(orders.s\_id)>1;

1. Find the all sales person details along with order details
2. Create index

**alter table customer add index INDEX1 (c\_id);**

1. How to show index on a table



1. Find the order number, sales person name, along with the customer to whom that order belongs to

