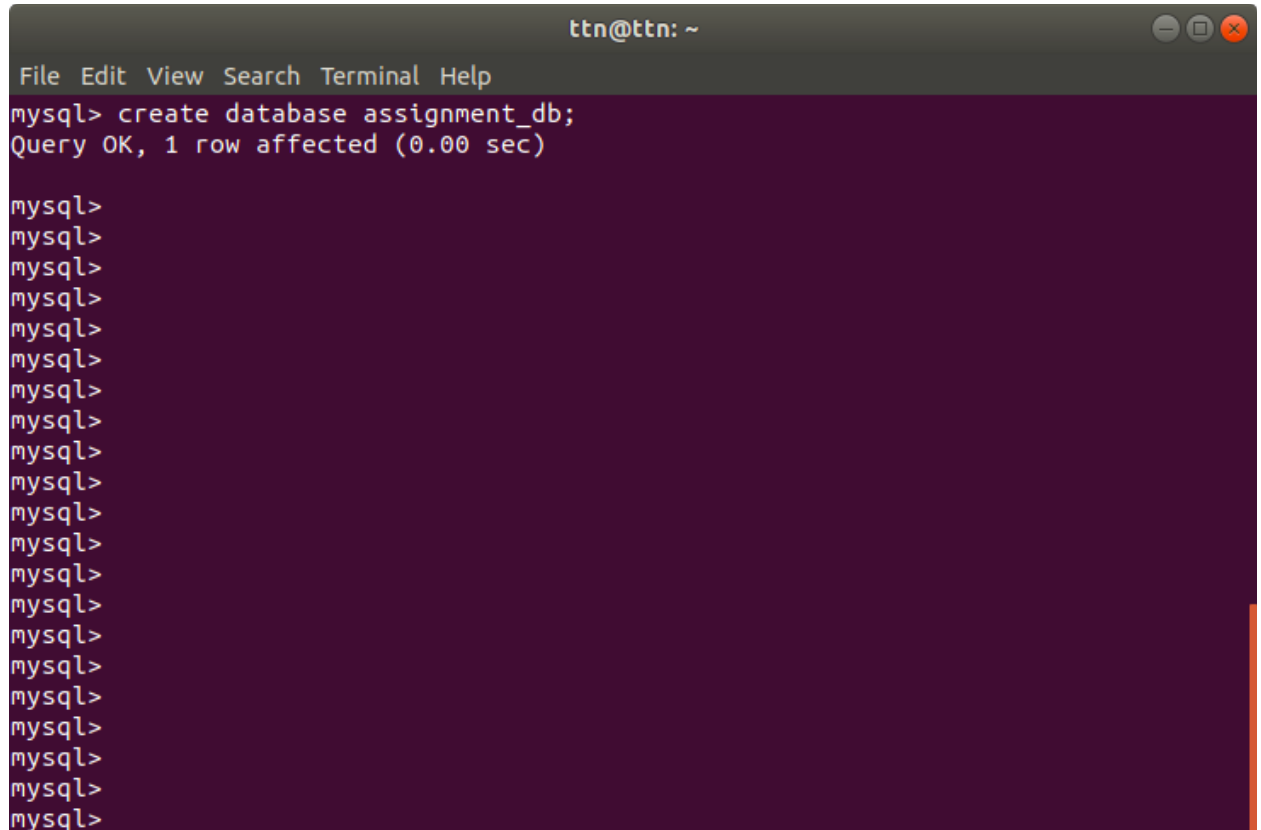


Problem Statement: There can be multiple customers, who can place multiple orders on the site. Now a sales person can handle these orders will distribute into multiple sales persons (One order will be assign to one salesperson only). So a sales person can have multiple orders of multiple customers

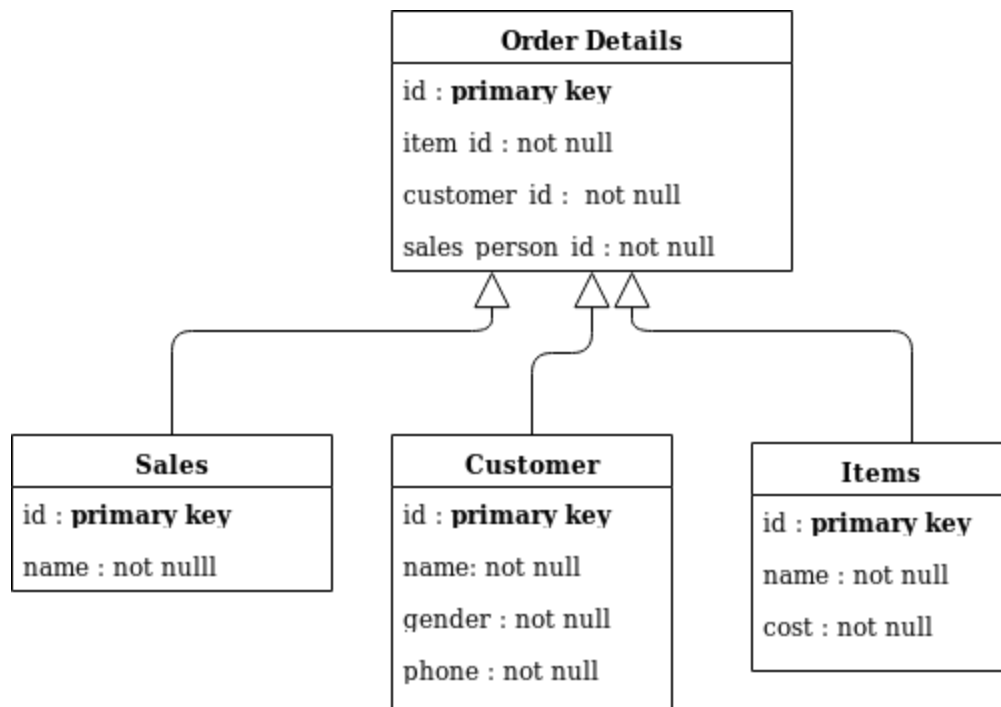
1. Create Database

create database database_name;

A screenshot of a terminal window titled 'ttn@ttn: ~'. The terminal shows a MySQL prompt 'mysql>' where the command 'create database assignment_db;' has been entered. The output of the command is 'Query OK, 1 row affected (0.00 sec)'. Below this, there are multiple 'mysql>' prompts, indicating the user is ready to enter more commands. The terminal window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The background is dark purple, and the text is light green.

2. Design Schema

->



b. Creating table Customer:

```
mysql> create table customer(id int(11) auto_increment not null, name varchar(255) not null, gender varchar(1) not null, phone varchar(11) not null,primary key(id));
Query OK, 0 rows affected (0.29 sec)
```

```
mysql> desc customer;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	varchar(255)	NO		NULL	
gender	varchar(1)	NO		NULL	
phone	varchar(11)	NO		NULL	

4 rows in set (0.00 sec)

c. Creating table Items:

```
mysql> create table items(id int(11) auto_increment not null, name varchar(100) not null, cost int(11) not null, primary key(id));
Query OK, 0 rows affected (0.32 sec)
```

```
mysql> desc items;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	varchar(100)	NO		NULL	
cost	int(11)	NO		NULL	

3 rows in set (0.01 sec)

```
mysql>
mysql>
mysql>
mysql>
mysql>
mysql>
mysql>
mysql>
mysql>
```

d. Creating table `order_details`:

```
ttn@ttn: ~  
File Edit View Search Terminal Help  
mysql> create table order_details(id int(11) auto_increment not null, item_id in  
t(11) not null, customer_id int(11) not null, sales_person_id int(11) not null,  
primary key(id));  
Query OK, 0 rows affected (0.34 sec)  
  
mysql> desc order_details;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| id | int(11) | NO | PRI | NULL | auto_increment |  
| item_id | int(11) | NO | | NULL | |  
| customer_id | int(11) | NO | | NULL | |  
| sales_person_id | int(11) | NO | | NULL | |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.00 sec)  
  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>
```

4. Insert sample data

Inserting sample data in:

a. Sales Table

```
mysql> insert into sales values(1, 'sp1');
Query OK, 1 row affected (0.09 sec)

mysql> insert into sales values(2, 'sp2');
Query OK, 1 row affected (0.10 sec)

mysql> insert into sales values(3, 'sp3');
Query OK, 1 row affected (0.09 sec)

mysql> insert into sales values(4, 'sp4');
Query OK, 1 row affected (0.08 sec)

mysql> insert into sales values(5, 'sp5');
Query OK, 1 row affected (0.09 sec)
```

```
mysql> select * from sales;
+----+-----+
| id | name |
+----+-----+
|  1 | sp1  |
|  2 | sp2  |
|  3 | sp3  |
|  4 | sp4  |
|  5 | sp5  |
+----+-----+
5 rows in set (0.00 sec)
```

b. Customer table

```
mysql> insert into customer values(1,'abc','m',12345);
Query OK, 1 row affected (0.09 sec)

mysql> insert into customer values(2,'def','m',1345);
Query OK, 1 row affected (0.09 sec)
```

```
mysql> insert into customer values(3,'desf','f',14345);
Query OK, 1 row affected (0.09 sec)

mysql> insert into customer values(4,'zsf','f',12345);
Query OK, 1 row affected (0.08 sec)

mysql> insert into customer values(5,'zaaf','f',14500);
Query OK, 1 row affected (0.08 sec)
```

```
mysql> select *from customer;
+----+-----+-----+-----+
| id | name | gender | phone |
+----+-----+-----+-----+
| 1  | abc  | m      | 12345 |
| 2  | def  | m      | 1345  |
| 3  | desf | f      | 14345 |
| 4  | zsf  | f      | 12345 |
| 5  | zaaf | f      | 14500 |
+----+-----+-----+-----+
5 rows in set (0.01 sec)
```

c. Items Table

```
mysql> insert into items values(1,'laptop',10000);
Query OK, 1 row affected (0.09 sec)

mysql> insert into items values(2,'computer',8000);
Query OK, 1 row affected (0.10 sec)

mysql> insert into items values(3,'mobile',4000);
Query OK, 1 row affected (0.09 sec)

mysql> insert into items values(4,'radio',1000);
Query OK, 1 row affected (0.08 sec)

mysql> insert into items values(5,'battery',1200);
Query OK, 1 row affected (0.04 sec)
```



```
mysql> select *from items;
+----+-----+-----+
| id | name   | cost |
+----+-----+-----+
| 1  | laptop | 10000 |
| 2  | computer | 8000 |
| 3  | mobile | 4000 |
| 4  | radio  | 1000 |
| 5  | battery | 1200 |
+----+-----+-----+
5 rows in set (0.00 sec)
```

d. Order_Details table

```
mysql> insert into order_details values(1,1,1,1);
Query OK, 1 row affected (0.09 sec)

mysql> insert into order_details values(2,2,1,1);
Query OK, 1 row affected (0.08 sec)

mysql> insert into order_details values(3,1,2,2);
Query OK, 1 row affected (0.09 sec)

mysql> insert into order_details values(4,5,3,4);
Query OK, 1 row affected (0.08 sec)

mysql> insert into order_details values(5,4,5,3);
Query OK, 1 row affected (0.09 sec)
```

```
mysql> select * from order_details;
+----+-----+-----+-----+
| id | item_id | customer_id | sales_person_id |
+----+-----+-----+-----+
| 1  | 1       | 1           | 1               |
| 2  | 2       | 1           | 1               |
| 3  | 1       | 2           | 2               |
| 4  | 5       | 3           | 4               |
| 5  | 4       | 5           | 3               |
+----+-----+-----+-----+
5 rows in set (0.00 sec)
```


5. Find the sales person have multiple orders.

```
-> select * from sales where id = (select sales_person_id from order_details group
by sales_person_id having count(sales_person_id)>1);
```

[illegible]

6. Find the all sales person details along with order details

-> select sales.name, customer.name, items.name, items.cost from sales join
order_details on sales.id=order_details.sales_person_id
join customer on order_details.customer_id=customer.id
join items on order_details.item_id=items.id;

```
ttn@ttn: ~  
File Edit View Search Terminal Help  
mysql> select sales.name, customer.name, items.name, items.cost from sales join  
order_details on sales.id=order_details.sales_person_id  
-> join customer on order_details.customer_id=customer.id  
-> join items on order_details.item_id=items.id;  
+-----+-----+-----+-----+  
| name | name | name | cost |  
+-----+-----+-----+-----+  
| sp1 | abc | laptop | 10000 |  
| sp1 | abc | computer | 8000 |  
| sp2 | def | laptop | 10000 |  
| sp4 | desf | battery | 1200 |  
| sp3 | zaaf | radio | 1000 |  
+-----+-----+-----+-----+  
5 rows in set (0.01 sec)  
  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>
```

Here first name column denotes sales person name, second name column denotes customer name and third name column denotes item name.

7. Create index

```
-> alter table table_name ADD INDEX index_name(column_list);
```

[illegible]

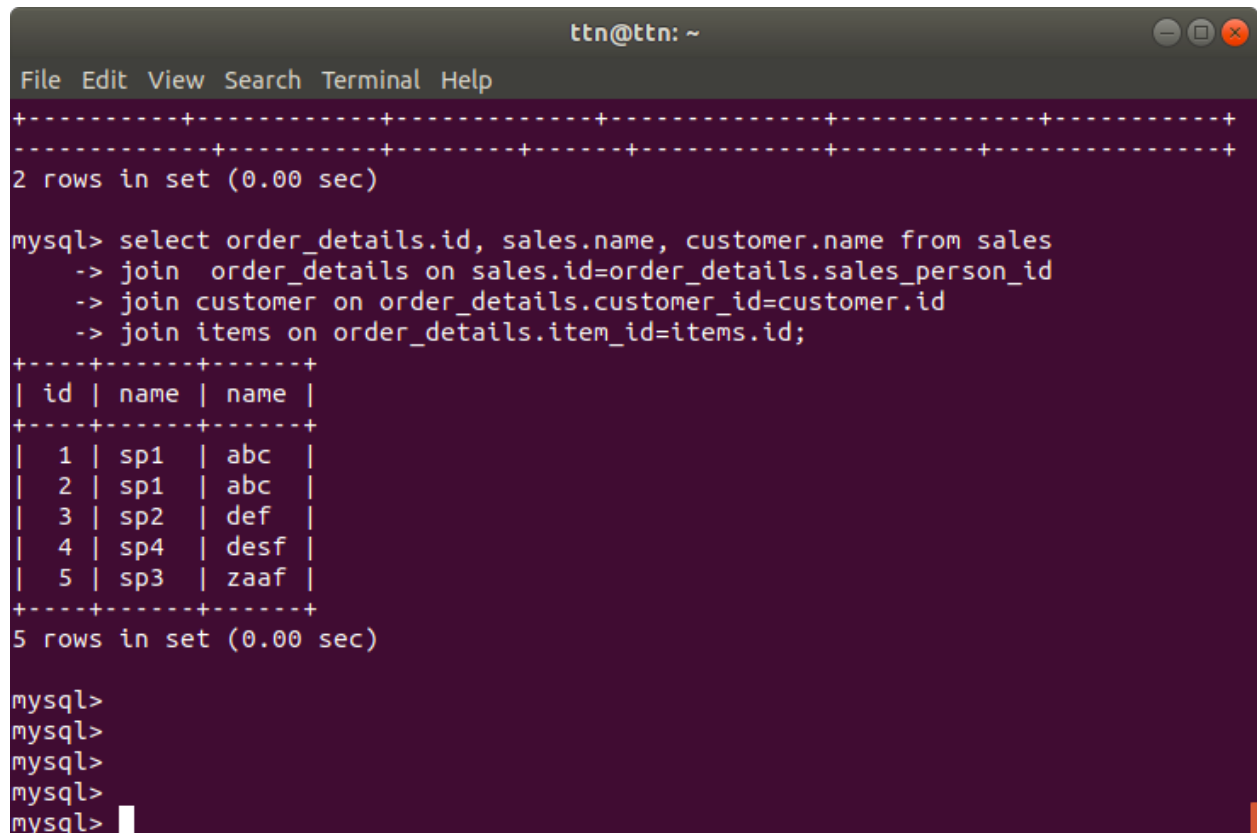
8. How to show index on a table

-> show index from table_name;

```
ttn@ttn: ~  
File Edit View Search Terminal Help  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql>  
mysql> show index from customer;  
+-----+-----+-----+-----+-----+-----+-----+  
| Table      | Non_unique | Key_name      | Seq_in_index | Column_name | Collation | Index_comment |  
| Cardinality | Sub_part  | Packed | Null | Index_type | Comment | Index_comment |  
+-----+-----+-----+-----+-----+-----+-----+  
| customer  |          0 | PRIMARY      |          1 | id          | A        |                |  
|          4 |          NULL | NULL |      | BTREE      |          |                |  
| customer  |          1 | phone_index   |          1 | phone       | A        |                |  
|          4 |          NULL | NULL |      | BTREE      |          |                |  
+-----+-----+-----+-----+-----+-----+  
2 rows in set (0.00 sec)  
  
mysql> 
```

9. Find the order number, sale person name, along with the customer to whom that order belongs to.

```
-> select order_details.id, sales.name, customer.name from sales
join order_details on sales.id=order_details.sales_person_id
join customer on order_details.customer_id=customer.id
join items on order_details.item_id=items.id;
```

A terminal window titled 'ttn@ttn: ~' with a menu bar (File, Edit, View, Search, Terminal, Help) and standard window controls. It shows the execution of a MySQL query. The first query returns 2 rows. The second query, which includes joins with 'order_details', 'customer', and 'items', returns 5 rows. The results are displayed in a table format with columns 'id', 'name', and 'name'.

```
ttn@ttn: ~
File Edit View Search Terminal Help
+-----+-----+-----+-----+-----+-----+-----+-----+
-+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select order_details.id, sales.name, customer.name from sales
-> join order_details on sales.id=order_details.sales_person_id
-> join customer on order_details.customer_id=customer.id
-> join items on order_details.item_id=items.id;
+-----+-----+-----+
| id | name | name |
+-----+-----+-----+
| 1 | sp1 | abc |
| 2 | sp1 | abc |
| 3 | sp2 | def |
| 4 | sp4 | desf |
| 5 | sp3 | zaaf |
+-----+-----+-----+
5 rows in set (0.00 sec)

mysql>
mysql>
mysql>
mysql>
mysql>
```

Here first name column denotes sales person name and second column denotes customer name.