**Database related command**

**CREATE** **DATABASE** testdb;

**SHOW** **DATABASES**;

**USE** testdb;

**DROP** **DATABASE** [**IF** **EXISTS**] database\_name;

CREATE TABLE IF NOT EXISTS `Members` (

`membership\_number` INT AUTOINCREMENT ,

`full\_names` VARCHAR(150) NOT NULL ,

`gender` VARCHAR(6) ,

`date\_of\_birth` DATE ,

`physical\_address` VARCHAR(255) ,

`postal\_address` VARCHAR(255) ,

`contact\_number` VARCHAR(75) ,

`email` VARCHAR(255) ,

PRIMARY KEY (`membership\_number`) );

**CREATE TABLE `zomato`.`category` ( `category\_id` INT NOT NULL AUTO\_INCREMENT , `category\_name` VARCHAR(255) NOT NULL , PRIMARY KEY (`category\_id`)) ENGINE = InnoDB;**

**CREATE TABLE `zomato`.`menu`  
( `menu\_id` INT NOT NULL AUTO\_INCREMENT ,  
 `menu\_name` VARCHAR(255) NOT NULL ,  
 `menu\_description` VARCHAR(700) NULL ,  
 `unit\_price` DECIMAL NOT NULL ,  
 PRIMARY KEY (`menu\_id`)) ENGINE = InnoDB;**

**CREATE TABLE IF NOT EXISTS `Product` (**

**`product\_id` INT AUTO\_INCREMENT ,**

**`product\_name` VARCHAR(150) NOT NULL ,**

**`description` VARCHAR(250) NOT NULL ,**

**`price\_per\_qty` DOUBLE NOT NULL,**

**`available\_qty` INT NOT NULL,**

**`category\_id\_fk` INT,**

**CONSTRAINT fk\_category**

**FOREIGN KEY(category\_id\_fk) REFERENCES Category(category\_id) ,**

**PRIMARY KEY (`product\_id`)**

**);**

**PRIMARY Key - Foreign Key**

CREATE TABLE CUSTOMERS(

ID INT NOT NULL,

NAME VARCHAR (20) NOT NULL,

AGE INT NOT NULL,

ADDRESS CHAR (25) ,

SALARY DECIMAL (18, 2),

PRIMARY KEY (ID)

);

CREATE TABLE ORDERS (

ID INT NOT NULL,

DATE DATETIME,

CUSTOMER\_ID INT FOREIGN KEY references CUSTOMERS(ID),

AMOUNT double,

PRIMARY KEY (ID)

);

If the ORDERS table has already been created and the foreign key has not yet been set, the use the syntax for specifying a foreign key by altering a table.

ALTER TABLE ORDERS

ADD FOREIGN KEY (Customer\_ID) REFERENCES CUSTOMERS (ID);

**DROP TABLE Orders; - deletes table structure and also data(rows)**

**TRUNCATE TABLE Orders; - deletes only data not table structure**