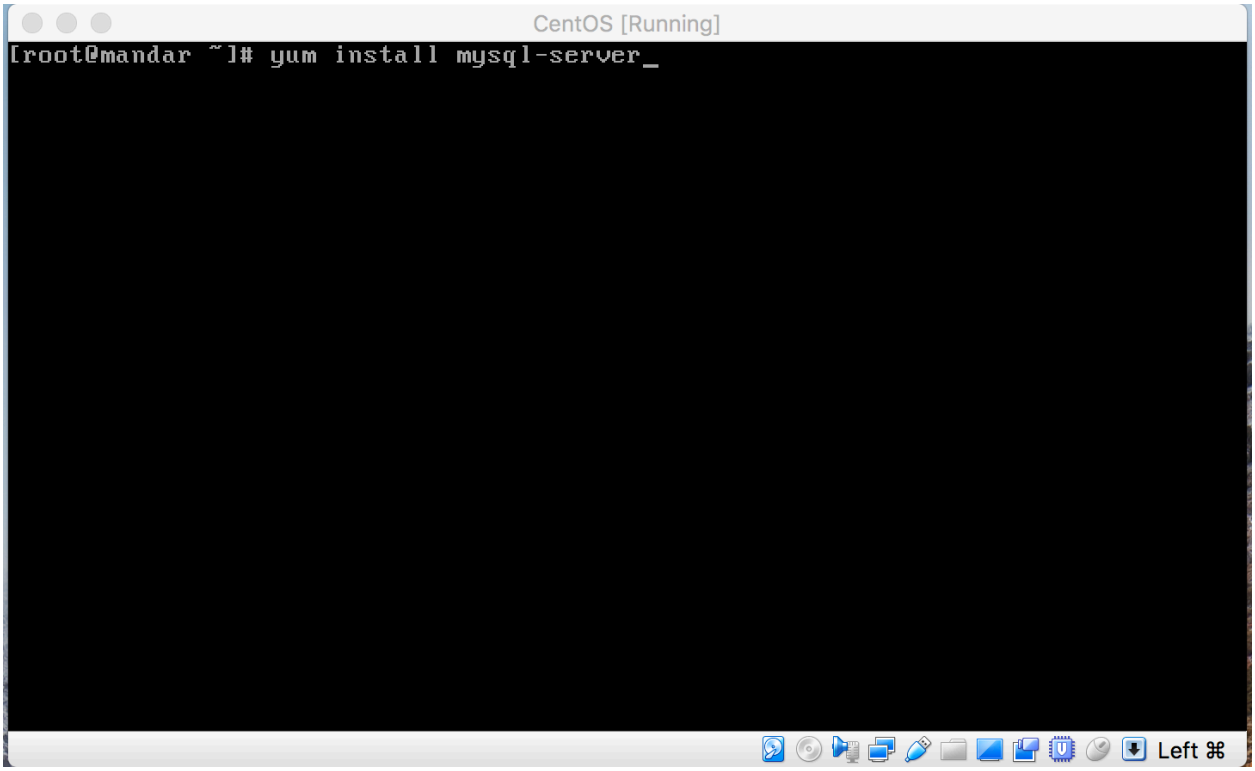
Learning SQL

1. Install My-SQL in CENTOS, create a user & grant all permissions to it. (10 Marks)

Ans. First we install MySQL server by using yum command and by using 'rpm' package as shown below:



CentOS [Running]

```
[root@mandar ~]# /sbin/service mysqld start_
```

Left ⌘

CentOS [Running]

```
[root@mandar ~]# md5sum mysql57-community-release-el7.9.noarch.rpm
md5sum: mysql57-community-release-el7.9.noarch.rpm: No such file or directory
[root@mandar ~]# md5sum mysql57-community-release-el7-9.noarch.rpm
1a29601dc380ef2c7bc25e2a0e25d31e  mysql57-community-release-el7-9.noarch.rpm
[root@mandar ~]# _
```

Left ⌘

CentOS [Running]

```
[root@mandar ~]# yum install wget_
```

Left ⌘

CentOS [Running]

```
Installing:
 wget           x86_64          1.14-15.el7_4.1           updates          547 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 547 k
Installed size: 2.0 M
Is this ok [y/d/N]: y
Downloading packages:
wget-1.14-15.el7_4.1.x86_64.rpm                                | 547 kB   00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : wget-1.14-15.el7_4.1.x86_64                                    1/1
  Verifying  : wget-1.14-15.el7_4.1.x86_64                                    1/1

Installed:
  wget.x86_64 0:1.14-15.el7_4.1

Complete!
[root@mandar ~]#
```

Left ⌘

CentOS [Running]

```
[root@mandar ~]# md5sum mysql57-community-release-el7.9.noarch.rpm
md5sum: mysql57-community-release-el7.9.noarch.rpm: No such file or directory
[root@mandar ~]# md5sum mysql57-community-release-el7-9.noarch.rpm
1a29601dc380ef2c7bc25e2a0e25d31e  mysql57-community-release-el7-9.noarch.rpm
[root@mandar ~]# yum install mysql-server_
```

Left ⌘

CentOS [Running]

```
  nss-softokn-freebl.x86_64 0:3.28.3-8.el7_4
  nss-sysinit.x86_64 0:3.28.4-15.el7_4
  nss-tools.x86_64 0:3.28.4-15.el7_4
  openssh.x86_64 0:7.4p1-13.el7_4
  openssh-clients.x86_64 0:7.4p1-13.el7_4
  openssh-server.x86_64 0:7.4p1-13.el7_4
  python-gobject-base.x86_64 0:3.22.0-1.el7_4.1
  python-perf.x86_64 0:3.10.0-693.17.1.el7
  selinux-policy.noarch 0:3.13.1-166.el7_4.7
  selinux-policy-targeted.noarch 0:3.13.1-166.el7_4.7
  sudo.x86_64 0:1.8.19p2-11.el7_4
  systemd.x86_64 0:219-42.el7_4.7
  systemd-libs.x86_64 0:219-42.el7_4.7
  systemd-sysv.x86_64 0:219-42.el7_4.7
  tuned.noarch 0:2.8.0-5.el7_4.2
  tzdata.noarch 0:2018c-1.el7
  util-linux.x86_64 0:2.23.2-43.el7_4.2
  wpa_supplicant.x86_64 1:2.6-5.el7_4.1
  yum.noarch 0:3.4.3-154.el7.centos.1

Replaced:
  grub2.x86_64 1:2.02-0.64.el7.centos grub2-tools.x86_64 1:2.02-0.64.el7.centos

Complete!
[root@mandar ~]# yum update
```

Left ⌘

```
CentOS [Running]
  Verifying  : perl-Net-Daemon-0.48-5.el7.noarch                          3/10
  Verifying  : perl-Data-Dumper-2.145-3.el7.x86_64                        4/10
  Verifying  : net-tools-2.0-0.22.20131004git.el7.x86_64                  5/10
  Verifying  : perl-PlRPC-0.2020-14.el7.noarch                            6/10
  Verifying  : 1:perl-Compress-Raw-Zlib-2.061-4.el7.x86_64                7/10
  Verifying  : perl-DBI-1.627-4.el7.x86_64                                8/10
  Verifying  : perl-IO-Compress-2.061-2.el7.noarch                        9/10
  Verifying  : mysql-community-client-5.6.39-2.el7.x86_64                10/10

Installed:
  mysql-community-server.x86_64 0:5.6.39-2.el7

Dependency Installed:
  mysql-community-client.x86_64 0:5.6.39-2.el7
  net-tools.x86_64 0:2.0-0.22.20131004git.el7
  perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.el7
  perl-Compress-Raw-Zlib.x86_64 1:2.061-4.el7
  perl-DBI.x86_64 0:1.627-4.el7
  perl-Data-Dumper.x86_64 0:2.145-3.el7
  perl-IO-Compress.noarch 0:2.061-2.el7
  perl-Net-Daemon.noarch 0:0.48-5.el7
  perl-PlRPC.noarch 0:0.2020-14.el7

Complete!
[root@mandar ~]# _
Left ⌘
```

Now, we will start mysqld service on our environment

```
CentOS [Running]
[root@mandar ~]# systemctl start mysqld
[root@mandar ~]# _
Left ⌘
```

CentOS [Running]

```
[root@mandar ~]# systemctl start mysqld
[root@mandar ~]# mysql_secure_installation
```

Left ⌘

CentOS [Running]

```
[root@mandar ~]# sudo mysql_secure_installation


NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user.  If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none): _
```

Left ⌘

CentOS [Running]

```
[root@mandar ~]# sudo mysql_secure_installation



NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user.  If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

You already have a root password set, so you can safely answer 'n'.

Change the root password? [Y/n] y_
```

Left ⌘

CentOS [Running]

```
[root@mandar ~]# sudo mysql_secure_installation



NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user.  If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

You already have a root password set, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
```

Left ⌘

CentOS [Running]

```
Remove anonymous users? [Y/n] y
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
 ... Success!

By default, MySQL comes with a database named 'test' that anyone can
access.  This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
 - Dropping test database...
ERROR 1008 (HY000) at line 1: Can't drop database 'test'; database doesn't exist
 ... Failed!  Not critical, keep moving...
 - Removing privileges on test database...
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n]
```

Left ⌘

CentOS [Running]

```
Remove test database and access to it? [Y/n] y
 - Dropping test database...
ERROR 1008 (HY000) at line 1: Can't drop database 'test'; database doesn't exist
 ... Failed!  Not critical, keep moving...
 - Removing privileges on test database...
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
 ... Success!



All done!  If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!


Cleaning up...
[root@mandar ~]# _
```

Left ⌘

```
CentOS [Running]
[root@mandar ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 45
Server version: 5.6.39 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Now, we have created a user 'mandar' and granting him all the privileges

```
CentOS [Running]
Enter password:
ERROR 1045 (28000): Access denied for user 'mandar'@'localhost' (using password:
 YES)
[root@mandar ~]# mysql -u root -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: Y
ES)
[root@mandar ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 48
Server version: 5.6.39 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE USER 'mandar'@'localhost' IDENTIFIED BY 'mandar';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

We can login into the new user ‘mandar’ which we have created above:

```
CentOS [Running]
[root@mandar ~]# mysql -u mandar -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 49
Server version: 5.6.39 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

2. Create the employee & department table & insert values in the table as shown below, meeting the following conditions

a. Make Employee_id & Department_id as Primary key of employee & department table respectively. (4 Marks)

For creating a table first, we have to create a database ‘andlab’ and then with ‘USE andlab’ command we can access the database and create the 2 tables in it.

```
CentOS [Running]
mysql> CREATE DATABASE ?
    -> Ctrl-C -- exit!
Aborted
[root@mandar ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 52
Server version: 5.6.39 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near '' at
line 1
mysql> CREATE DATABASE andlab;
Query OK, 1 row affected (0.00 sec)

mysql> _
Left ⌘
```

2.b.

```
CentOS [Running]

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE andlab;
ERROR 1007 (HY000): Can't create database 'andlab'; database exists
mysql> USE andlab;
Database changed
mysql> CREATE TABLE Employee (Employee_ID varchar (20) PRIMARY KEY, E_Name varch
ar (20), Phone varchar (20), Email varchar (30) DEFAULT 'UNKNOWN', Department_ID
 varchar (20));
Query OK, 0 rows affected (0.03 sec)

mysql> SHOW FIELDS FROM Employee;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Employee_ID   | varchar(20) | NO   | PRI | NULL    |       |
| E_Name        | varchar(20) | YES  |     | NULL    |       |
| Phone         | varchar(20) | YES  |     | NULL    |       |
| Email         | varchar(30) | YES  |     | UNKNOWN |       |
| Department_ID | varchar(20) | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> _
Left ⌘
```

2.c.

```
mysql> CREATE TABLE Department (Department_ID varchar (20) PRIMARY KEY, D_Name v
archar (20) NOT NULL, Location varchar (20));
Query OK, 0 rows affected (0.02 sec)

mysql> SHOW FIELDS FROM Department;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Department_ID | varchar(20) | NO   | PRI | NULL    |       |
| D_Name        | varchar(20) | NO   |     | NULL    |       |
| Location      | varchar(20) | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)

mysql>
```

By 'INSERT INTO' query we can fill the contents of the table according to given table:

```
CentOS [Running]
ALUES ('100', 'John', '6827235124', 'John21@gmail.com', '204');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Email, Department_ID) V
ALUES ('101', 'Michael', '8926465572', 'Michael@hotmail.com', '201');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Email, Department_ID) V
ALUES ('102', 'Chris', '6844375638', 'Chriss.freeman@yahoo.com', '201');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Email, Department_ID) V
ALUES ('103', 'Tom', '9648234733', 'Tom420@smu.edu', '203');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Email, Department_ID) V
ALUES ('104', 'Susana', '8463748391', 'Susana.ellis@gmail.com', '202');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Email, Department_ID) V
ALUES ('105', 'Richard', '6707631747', 'Richard1992@yahoo.com', '202');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Email, Department_ID) V
ALUES ('106', 'Katrine', '6826373721', 'coolKatrine.com', '201');
```

```
mysql> INSERT INTO Department (Department_ID, D_name, Location) VALUES ('201', '
Sales', 'Texas');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Department (Department_ID, D_name, Location) VALUES ('202', '
Technical', 'California');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Department (Department_ID, D_name, Location) VALUES ('203', '
Finance', 'Washington');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Department (Department_ID, D_name, Location) VALUES ('204', '
Management', 'New York');
Query OK, 1 row affected (0.00 sec)

mysql> _
```

We can able to display the contents of Employee and Department table using 'SELECT * FROM' query.

```
CentOS [Running]
+-------------+---------+------------+-------------------------+---------------
+
| Employee_ID | E_Name  | Phone      | Email                   | Department_ID
|
+-------------+---------+------------+-------------------------+---------------
+
| 100         | John    | 6827235124 | John21@gmail.com        | 204
|
| 101         | Michael | 8926465572 | Michael@hotmail.com     | 201
|
| 102         | Chris   | 6844375638 | Chriss.freeman@yahoo.com | 201
|
| 103         | Tom     | 9648234733 | Tom420@smu.edu          | 203
|
| 104         | Susana  | 8463748391 | Susana.ellis@gmail.com  | 202
|
| 105         | Richard | 6707631747 | Richard1992@yahoo.com   | 202
|
| 106         | Katrine | 6826373721 | coolKatrine.com         | 201
|
+-------------+---------+------------+-------------------------+---------------
+
7 rows in set (0.00 sec)

mysql> SELECT * FROM Employee;
```

```
mysql> SELECT * FROM Department;
+---------------+-------------+-------------+
| Department_ID | D_Name      | Location    |
+---------------+-------------+-------------+
| 201           | Sales       | Texas       |
| 202           | Technical   | California  |
| 203           | Finance     | Washington  |
| 204           | Management  | New York    |
+---------------+-------------+-------------+
4 rows in set (0.00 sec)

mysql> SELECT * FROM Department;_
```

2.d. For using FOREIGN KEY constraint, we have to make an index on the parent table, so we create and index on the field “Department ID” and add a Foreign Key constraint to “Department ID” of Employee Table with reference as “Department ID” of Department Table. We can see “MUL” created due to index in Employee Table.

```
mysql> CREATE INDEX root on Department (Department_ID);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE Employee ADD CONSTRAINT FOREIGN KEY (Department_ID) REFERENCE
S Department(Department_ID);
Query OK, 7 rows affected (0.05 sec)
Records: 7  Duplicates: 0  Warnings: 0

mysql>
```

```
mysql> CREATE INDEX root on Department (Department_ID);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE Employee ADD CONSTRAINT FOREIGN KEY (Department_ID) REFERENCE
S Department(Department_ID);
Query OK, 7 rows affected (0.05 sec)
Records: 7  Duplicates: 0  Warnings: 0

mysql> SHOW FIELDS FROM Employee;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Employee_ID   | varchar(20) | NO   | PRI | NULL    |       |
| E_Name        | varchar(20) | YES  |     | NULL    |       |
| Phone         | varchar(20) | YES  |     | NULL    |       |
| Email         | varchar(30) | YES  |     | UNKNOWN |       |
| Department_ID | varchar(20) | YES  | MUL | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql>
```

3. Using the above tables, do the following.

a. insert a record demonstrating that 2.b is working as expected. (5 Marks)

Here we did not give any value for Email so we have got UNKONOWN in front of 'Mandar' in the table.

```
CentOS [Running]
|
+-------------+---------+------------+-------------------------+---------------
+
| 100         | John    | 6827235124 | John21@gmail.com        | 204
|
| 101         | Michael | 8926465572 | Michael@hotmail.com     | 201
|
| 102         | Chris   | 6844375638 | Chriss.freeman@yahoo.com | 201
|
| 103         | Tom     | 9648234733 | Tom420@smu.edu          | 203
|
| 104         | Susana  | 8463748391 | Susana.ellis@gmail.com  | 202
|
| 105         | Richard | 6707631747 | Richard1992@yahoo.com   | 202
|
| 106         | Katrine | 6826373721 | coolKatrine.com         | 201
|
| 107         | Mandar  | 9892477674 | UNKNOWN                 | 202
|
+-------------+---------+------------+-------------------------+---------------
+
8 rows in set (0.00 sec)

mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Department_ID) VALUES (
'107', 'Mandar', '9892477674', '202');
```

b. insert a record demonstrating that 2.d is working as expected. (5 Marks)
We came across error that no valid entry can be made in the child table if the corresponding entry is not present in the parent table as '205' is not defined in parent table.

```
mysql> INSERT INTO Employee (Employee_ID, E_name, Phone, Department_ID) VALUES (
'108', 'MandarGhogale', '9969015289', '205');
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint f
ails (`andlab`.`Employee`, CONSTRAINT `Employee_ibfk_1` FOREIGN KEY (`Department
_ID`) REFERENCES `Department` (`Department_ID`))
mysql>
```

c. Fetch employees working in 'sales' department which has department = 201, display only the top 2 records orders by Employee id. (10 Marks)

```
mysql> SELECT Employee.Employee_ID, Employee.E_Name, Department.Department_ID, D
epartment.D_Name FROM Employee INNER JOIN Department ON Employee.Department_ID=D
epartment.Department_ID WHERE Department.Department_ID=201 ORDER BY Employee_ID
LIMIT 2;
+-------------+---------+---------------+--------+
| Employee_ID | E_Name  | Department_ID | D_Name |
+-------------+---------+---------------+--------+
| 101         | Michael | 201           | Sales  |
| 102         | Chris   | 201           | Sales  |
+-------------+---------+---------------+--------+
2 rows in set (0.00 sec)

mysql> _
```

d. Fetch all the names of the employees having 'n' as the second last alphabet in their name.

```
mysql> SELECT E_Name FROM Employee WHERE E_Name LIKE '%n_';
+---------+
| E_Name  |
+---------+
| Susana  |
| Katrine |
+---------+
2 rows in set (0.00 sec)

mysql> _
```

e. Fetch names of departments which has 2 or more employees. (hint: this has more 1 way to implement, easier way is to use subqueries) (10 marks)

```
mysql> SELECT Department.D_Name FROM Employee INNER JOIN Department ON Employee.
Department_ID=Department.Department_ID GROUP BY D_Name having count(D_Name)>=2;
+-----------+
| D_Name    |
+-----------+
| Sales     |
| Technical |
+-----------+
2 rows in set (0.00 sec)

mysql>
```

4. Answer the following in 1-2 lines:

a. What is dual table in database? Write any SELECT query to show current date using dual table & show output.

Dual table is special one row and one column table in Oracle database and has single VARCHAR2(1) column called DUMMY that has value of 'X'. Some queries which do not have any table name, so for the system to recognize it and send in system database as an output we use 'dual' in place of table name.
SELECT SYSDATE(); gives date and time of system, it is the variation of SELECT SYSDATE() FROM DUAL.

```
mysql> SELECT SYSDATE();
+---------------------+
| SYSDATE()           |
+---------------------+
| 2018-02-24 22:48:19 |
+---------------------+
1 row in set (0.00 sec)

mysql>
```

b. What are 2 ways in database to ensure values entered in phone column is of length 10.

1.

```
mysql> SELECT LENGTH(Phone) FROM Employee;
+----------------+
| LENGTH(Phone) |
+----------------+
|             10 |
|             10 |
|             10 |
|             10 |
|             10 |
|             10 |
|             10 |
|             10 |
+----------------+
8 rows in set (0.00 sec)

mysql>
```

2.

```
mysql> SELECT CHAR_LENGTH(Phone);
ERROR 1054 (42S22): Unknown column 'Phone' in 'field list'
mysql> SELECT CHAR_LENGTH(Phone) FROM Employee;
+--------------------+
| CHAR_LENGTH(Phone) |
+--------------------+
|                 10 |
|                 10 |
|                 10 |
|                 10 |
|                 10 |
|                 10 |
|                 10 |
|                 10 |
+--------------------+
8 rows in set (0.00 sec)

mysql> _
```

3.We can select length as 10 by making following subquery:

```
mysql> SELECT Phone FROM Employee GROUP BY Phone HAVING LENGTH(Phone)>10;
Empty set (0.00 sec)

mysql> SELECT Phone FROM Employee GROUP BY Phone HAVING LENGTH(Phone)<10;
Empty set (0.00 sec)

mysql> SELECT Phone FROM Employee GROUP BY Phone HAVING LENGTH(Phone)=10;
+------------+
| Phone      |
+------------+
| 6707631747 |
| 6826373721 |
| 6827235124 |
| 6844375638 |
| 8463748391 |
| 8926465572 |
| 9648234733 |
| 9892477674 |
+------------+
8 rows in set (0.00 sec)

mysql> _
```

c. What does ‘desc employee;’ command do?

‘desc employee’ query gives the description of ‘employee’ table in MYSQL environment in Linux like ‘SHOW FIELDS FROM Employee’ query. But in MYSQL tool, it will display more information about the table. Here, I have replaced ‘employee’ with ‘Employee’ to show output of the query as I am using table name as ‘Employee’.

```
mysql> desc Employee;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Employee_ID   | varchar(20) | NO   | PRI | NULL    |       |
| E_Name        | varchar(20) | YES  |     | NULL    |       |
| Phone         | varchar(20) | YES  |     | NULL    |       |
| Email         | varchar(30) | YES  |     | UNKNOWN |       |
| Department_ID | varchar(20) | YES  | MUL | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql>
```

5. List all possible pairs of employees (using table of 2nd question) in 2 columns, where 1st column has E_name of person_1 & 2nd column has E_name of person_2. Employee cannot pair with self. (hint: self- join)

We split single column into 2 logical columns 'Person_1' and 'Person_2' and then by using WHERE condition we have got the output excluding same name combination.

```
CentOS [Running]
| Michael | Richard |
| Chris   | Richard |
| Tom     | Richard |
| Susana  | Richard |
| Katrine | Richard |
| Mandar  | Richard |
| John    | Katrine |
| Michael | Katrine |
| Chris   | Katrine |
| Tom     | Katrine |
| Susana  | Katrine |
| Richard | Katrine |
| Mandar  | Katrine |
| John    | Mandar  |
| Michael | Mandar  |
| Chris   | Mandar  |
| Tom     | Mandar  |
| Susana  | Mandar  |
| Richard | Mandar  |
| Katrine | Mandar  |
+---------+---------+
56 rows in set (0.00 sec)

mysql> SELECT Person_1.E_NAme, Person_2.E_Name FROM Employee Person_1, Employee
Person_2 WHERE Person_1.E_Name<>Person_2.E_Name;
Left ⌘
```