

Student's Full Name: Yatender Singh Suman

Course Title: Data Warehousing and Analytics in the Cloud

Term name and year: Example, Fall 2023

Submission Week: [Week 3- Assignment 2]

Instructor's Name: Dr. Nayem Rahman

Date of Submission: 28 July 2024

Q1. Show the screenshot of a successful installation of MySQL Software and MySQL Workbench with the latest version on your machine. Show the screenshot of the database "Worker" created.

```
mysql> CREATE DATABASE IF NOT EXISTS Worker;
Query OK, 1 row affected (0.01 sec)

mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| Worker        |
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0.02 sec)

mysql> 
```

Q2.. Create the Department table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).
(a) Columns, Primary Key (PK), Data Type & length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.
(b) Show the table definition (DDL) that you implemented.

```

mysql> USE Worker;
Database changed
mysql>
mysql> CREATE TABLE Department (
->     DepartmentID TINYINT NOT NULL,
->     DepartmentNm CHAR(30) NOT NULL,
->     PRIMARY KEY (DepartmentID)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> show create table Department;
+-----+-----+
| Table      | Create Table           |
+-----+-----+
| Department | CREATE TABLE `Department` (
`DepartmentID` tinyint NOT NULL,
`DepartmentNm` char(30) NOT NULL,
PRIMARY KEY (`DepartmentID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
1 row in set (0.03 sec)

```

(c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.

(d) Retrieve the data from the Department table by using the SELECT * statement and order by PK column(s).

Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```

mysql>
mysql> INSERT INTO Department (DepartmentID, DepartmentNm) VALUES
-> (1, 'Research & Development'),
-> (2, 'Production'),
-> (3, 'IT Support'),
-> (4, 'Operations'),
-> (5, 'Customer Service'),
-> (6, 'Purchasing'),
-> (7, 'Sales & Marketing'),
-> (8, 'Human Resource Management'),
-> (9, 'Accounting and Finance'),
-> (10, 'Legal Department');
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Department ORDER BY DepartmentID;
+-----+-----+
| DepartmentID | DepartmentNm           |
+-----+-----+
|      1 | Research & Development |
|      2 | Production                |
|      3 | IT Support                 |
|      4 | Operations                 |
|      5 | Customer Service           |
|      6 | Purchasing                 |
|      7 | Sales & Marketing          |
|      8 | Human Resource Management  |
|      9 | Accounting and Finance     |
|     10 | Legal Department            |
+-----+-----+
10 rows in set (0.01 sec)

```

Q3. Create the Employee table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).

- (a) Columns, Primary Key (PK), Data Type and length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.
(b) Show the table definition (DDL) that you implemented (not in a graphical view).

```
mysql> CREATE TABLE Employee (
->     EmployeeID INTEGER NOT NULL,
->     DepartmentID TINYINT NOT NULL,
->     FirstName VARCHAR(20) NOT NULL,
->     LastName VARCHAR(20) NOT NULL,
->     Address VARCHAR(50) NOT NULL,
->     PhoneNumberTxt VARCHAR(15),
->     HireDate DATE NOT NULL,
->     PRIMARY KEY (EmployeeID),
->     FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
-> );
Query OK, 0 rows affected (0.05 sec)

mysql> show create table Employee;
+-----+-----+
| Table | Create Table
+-----+-----+
| Employee | CREATE TABLE `Employee` (
`EmployeeID` int NOT NULL,
`DepartmentID` tinyint NOT NULL,
`FirstName` varchar(20) NOT NULL,
`LastName` varchar(20) NOT NULL,
`Address` varchar(50) NOT NULL,
`PhoneNumberTxt` varchar(15) DEFAULT NULL,
`HireDate` date NOT NULL,
PRIMARY KEY (`EmployeeID`),
KEY `DepartmentID` (`DepartmentID`),
CONSTRAINT `Employee_ibfk_1` FOREIGN KEY (`DepartmentID`) REFERENCES `Department` (`DepartmentID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
(END)
```

- (c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.
(d) Retrieve the data from the Employee table by using the SELECT * statement and order by PK column(s). Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql>
mysql>
mysql> INSERT INTO Employee (EmployeeID, DepartmentID, FirstName, LastName, Address, PhoneNumberTxt, HireDate) VALUES
-> (1, 2, 'Andy', 'Wong', '345 South Street', '603-555-6880', '2001-01-15'),
-> (2, 3, 'Vivek', 'Pandey', '15 Mineral Drive', '603-555-4420', '2003-11-15'),
-> (3, 8, 'Kathy', 'Cooper', '15 Hatter Drive', '212-555-9630', '2011-11-18'),
-> (4, 1, 'John', 'Wilson', '560 Broadway', '518-555-6690', '2017-03-19'),
-> (5, 7, 'Nola', 'Davis', '15 Long Ave', '478-555-8822', '2016-03-23'),
-> (6, 9, 'Tom', 'Harper', '64 Highland Street', '212-555-7755', '2010-04-11');
Query OK, 6 rows affected (0.01 sec)
Records: 6  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Employee ORDER BY EmployeeID;
+-----+-----+-----+-----+-----+-----+-----+
| EmployeeID | DepartmentID | FirstName | LastName | Address | PhoneNumberTxt | HireDate |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 2 | Andy | Wong | 345 South Street | 603-555-6880 | 2001-01-15 |
| 2 | 3 | Vivek | Pandey | 15 Mineral Drive | 603-555-4420 | 2003-11-15 |
| 3 | 8 | Kathy | Cooper | 15 Hatter Drive | 212-555-9630 | 2011-11-18 |
| 4 | 1 | John | Wilson | 560 Broadway | 518-555-6690 | 2017-03-19 |
| 5 | 7 | Nola | Davis | 15 Long Ave | 478-555-8822 | 2016-03-23 |
| 6 | 9 | Tom | Harper | 64 Highland Street | 212-555-7755 | 2010-04-11 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

(e) mysql>

Q4. Create the Equipment table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).

- Columns, Primary Key (PK), Data Type & length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.
- Show the table definition (DDL) that you implemented.

```
mysql> CREATE TABLE Equipment (
->     EquipmentID INTEGER NOT NULL,
->     EquipmentName VARCHAR(30) NOT NULL,
->     EquipmentCostAmount DECIMAL(13, 2) NOT NULL,
->     Currency VARCHAR(3) NOT NULL DEFAULT 'USD',
->     PRIMARY KEY (EquipmentID)
-> );
Query OK, 0 rows affected (0.04 sec)

mysql> SHOW CREATE TABLE Equipment;
+-----+-----+
| Table | Create Table
+-----+-----+
| Equipment | CREATE TABLE `Equipment` (
`EquipmentID` int NOT NULL,
`EquipmentName` varchar(30) NOT NULL,
`EquipmentCostAmount` decimal(13,2) NOT NULL,
`Currency` varchar(3) NOT NULL DEFAULT 'USD',
PRIMARY KEY (`EquipmentID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
(END)
```

(c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.

(d) Retrieve the data from the Equipment table by using the SELECT * statement and order by PK column(s).

Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql> INSERT INTO Equipment (EquipmentID, EquipmentName, EquipmentCostAmount, Currency) VALUES
-> (1, 'Notebook Computers', 1200.00, 'USD'),
-> (2, 'Headsets', 150.00, 'USD'),
-> (3, 'Computer Monitor', 300.00, 'USD'),
-> (4, 'Multi-Function Printers', 500.00, 'USD'),
-> (5, 'Projector or Big Screen TV', 1000.00, 'USD'),
-> (6, 'Servers', 2500.00, 'USD'),
-> (7, 'Internet Modem', 200.00, 'USD'),
-> (8, 'Cell Phone', 800.00, 'USD');
Query OK, 8 rows affected (0.02 sec)
Records: 8  Duplicates: 0  Warnings: 0

mysql>
mysql> SELECT * FROM Equipment ORDER BY EquipmentID;
+-----+-----+-----+-----+
| EquipmentID | EquipmentName | EquipmentCostAmount | Currency |
+-----+-----+-----+-----+
| 1 | Notebook Computers | 1200.00 | USD
| 2 | Headsets | 150.00 | USD
| 3 | Computer Monitor | 300.00 | USD
| 4 | Multi-Function Printers | 500.00 | USD
| 5 | Projector or Big Screen TV | 1000.00 | USD
| 6 | Servers | 2500.00 | USD
| 7 | Internet Modem | 200.00 | USD
| 8 | Cell Phone | 800.00 | USD
+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

Q5. Create the EmployeeEquipment table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).

(a) Columns, Primary Key (PK), Data Type & length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.

(b) Show the table definition (DDL) that you implemented.

```
mysql> CREATE TABLE EmployeeEquipment (
->     EmployeeID INTEGER NOT NULL,
->     EquipmentID INTEGER NOT NULL,
->     PRIMARY KEY (EmployeeID, EquipmentID),
->     FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID),
->     FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID)
-> );
Query OK, 0 rows affected (0.04 sec)

mysql> SHOW CREATE TABLE EmployeeEquipment;
+-----+-----+
| Table | Create Table |
+-----+-----+
| EmployeeEquipment | CREATE TABLE `EmployeeEquipment` (
`EmployeeID` int NOT NULL,
`EquipmentID` int NOT NULL,
PRIMARY KEY (`EmployeeID`, `EquipmentID`),
KEY `EquipmentID` (`EquipmentID`),
CONSTRAINT `EmployeeEquipment_ibfk_1` FOREIGN KEY (`EmployeeID`) REFERENCES `Employee`(`EmployeeID`),
CONSTRAINT `EmployeeEquipment_ibfk_2` FOREIGN KEY (`EquipmentID`) REFERENCES `Equipment`(`EquipmentID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
1 row in set (0.00 sec)
```

(c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.

(d) Retrieve the data from the EmployeeEquipment table by using the SELECT * statement and order by PK column(s).

Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql> INSERT INTO EmployeeEquipment (EmployeeID, EquipmentID) VALUES
-> (1, 1), -- Andy Wong needs Notebook Computers
-> (2, 1), -- Vivek Pandey needs Notebook Computers
-> (2, 2), -- Vivek Pandey needs Headsets
-> (2, 3), -- Vivek Pandey needs Computer Monitor
-> (3, 1), -- Kathy Cooper needs Notebook Computers
-> (3, 2), -- Kathy Cooper needs Headsets
-> (3, 3), -- Kathy Cooper needs Computer Monitor
-> (4, 1), -- John Wilson needs Notebook Computers
-> (4, 3), -- John Wilson needs Computer Monitor
-> (5, 1), -- Nola Davis needs Notebook Computers
-> (5, 2), -- Nola Davis needs Headsets
-> (6, 1), -- Tom Harper needs Notebook Computers
-> (6, 3); -- Tom Harper needs Computer Monitor
Query OK, 13 rows affected (0.00 sec)
Records: 13  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM EmployeeEquipment ORDER BY EmployeeID, EquipmentID;
+-----+-----+
| EmployeeID | EquipmentID |
+-----+-----+
|      1 |       1 |
|      2 |       1 |
|      2 |       2 |
|      2 |       3 |
|      3 |       1 |
|      3 |       2 |
|      3 |       3 |
|      4 |       1 |
|      4 |       3 |
|      5 |       1 |
|      5 |       2 |
|      6 |       1 |
|      6 |       3 |
+-----+-----+
13 rows in set (0.01 sec)
```

Q6. Create the Training table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).

(a) Columns, Primary Key (PK), Data Type & length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.

(b) Show the table definition (DDL) that you implemented.

```
mysql> CREATE TABLE Training (
->     TrainingID INTEGER NOT NULL,
->     TrainingName VARCHAR(50) NOT NULL,
->     PRIMARY KEY (TrainingID)
-> );
Query OK, 0 rows affected (0.04 sec)

mysql> SHOW CREATE TABLE Training;
+-----+-----+
| Table | Create Table |
+-----+-----+
| Training | CREATE TABLE `Training` (
`TrainingID` int NOT NULL,
`TrainingName` varchar(50) NOT NULL,
PRIMARY KEY (`TrainingID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
(END)
```

(c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.

(d) Retrieve the data from the Training table by using the SELECT * statement and order by PK column(s).

Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql> INSERT INTO Training (TrainingID, TrainingName) VALUES
-> (1, 'COVID-19 Awareness and Protection Training'),
-> (2, 'Code of Conduct Training'),
-> (3, 'Safety Training'),
-> (4, 'Intro to Python'),
-> (5, 'Machine Learning'),
-> (6, 'Microsoft Certifications'),
-> (7, 'Security and Privacy'),
-> (8, 'Product Knowledge'),
-> (9, 'Sales Skills'),
-> (10, 'Employee Relations'),
-> (11, 'Travel and Expense Management');
Query OK, 11 rows affected (0.01 sec)
Records: 11  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Training ORDER BY TrainingID;
+-----+-----+
| TrainingID | TrainingName |
+-----+-----+
| 1 | COVID-19 Awareness and Protection Training |
| 2 | Code of Conduct Training |
| 3 | Safety Training |
| 4 | Intro to Python |
| 5 | Machine Learning |
| 6 | Microsoft Certifications |
| 7 | Security and Privacy |
| 8 | Product Knowledge |
| 9 | Sales Skills |
| 10 | Employee Relations |
| 11 | Travel and Expense Management |
+-----+-----+
11 rows in set (0.00 sec)
```

Q7. Create the EmployeeTraining table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).

(a) Columns, Primary Key (PK), Data Type & length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.

(b) Show the table definition (DDL) that you implemented.

```
mysql> CREATE TABLE EmployeeTraining (
->     EmployeeID INTEGER NOT NULL,
->     TrainingID INTEGER NOT NULL,
->     PRIMARY KEY (EmployeeID, TrainingID),
->     FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID),
->     FOREIGN KEY (TrainingID) REFERENCES Training(TrainingID)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> SHOW CREATE TABLE EmployeeTraining;
+-----+-----+
| Table | Create Table |
+-----+-----+
| EmployeeTraining | CREATE TABLE `EmployeeTraining` (
`EmployeeID` int NOT NULL,
`TrainingID` int NOT NULL,
PRIMARY KEY (`EmployeeID`,`TrainingID`),
KEY `TrainingID` (`TrainingID`),
CONSTRAINT `EmployeeTraining_ibfk_1` FOREIGN KEY (`EmployeeID`) REFERENCES `Employee` (`EmployeeID`),
CONSTRAINT `EmployeeTraining_ibfk_2` FOREIGN KEY (`TrainingID`) REFERENCES `Training` (`TrainingID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
(END)
```

(c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.

(d) Retrieve the data from the EmployeeTraining table by using the SELECT * statement and order by PK column(s).

Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql> INSERT INTO EmployeeTraining (EmployeeID, TrainingID) VALUES
-> (1, 2), -- Andy Wong needs Code of Conduct Training
-> (1, 3), -- Andy Wong needs Safety Training
-> (2, 2), -- Vivek Pandey needs Code of Conduct Training
-> (2, 6), -- Vivek Pandey needs Microsoft Certifications
-> (2, 7), -- Vivek Pandey needs Security and Privacy
-> (3, 2), -- Kathy Cooper needs Code of Conduct Training
-> (3, 10), -- Kathy Cooper needs Employee Relations
-> (4, 2), -- John Wilson needs Code of Conduct Training
-> (4, 4), -- John Wilson needs Intro to Python
-> (4, 5), -- John Wilson needs Machine Learning
-> (5, 2), -- Nola Davis needs Code of Conduct Training
-> (5, 8), -- Nola Davis needs Product Knowledge
-> (5, 9), -- Nola Davis needs Sales Skills
-> (6, 2), -- Tom Harper needs Code of Conduct Training
-> (6, 11); -- Tom Harper needs Travel and Expense Management
Query OK, 15 rows affected (0.00 sec)
Records: 15  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM EmployeeTraining ORDER BY EmployeeID, TrainingID;
+-----+-----+
| EmployeeID | TrainingID |
+-----+-----+
|      1 |      2 |
|      1 |      3 |
|      2 |      2 |
|      2 |      6 |
|      2 |      7 |
|      3 |      2 |
|      3 |     10 |
|      4 |      2 |
|      4 |      4 |
|      4 |      5 |
|      5 |      2 |
|      5 |      8 |
|      5 |      9 |
|      6 |      2 |
|      6 |     11 |
+-----+-----+
15 rows in set (0.01 sec)
```

```
mysql> █
```

Q8. Create the Trainer table in the Worker database (table must be based on Physical Model Provided in the Assignment folder).

(a) Columns, Primary Key (PK), Data Type & length, and NULL/NOT NULL need to be implemented, as provided in the Physical Model.

(b) Show the table definition (DDL) that you implemented.

```
mysql> CREATE TABLE Trainer (
    ->     TrainerID INTEGER NOT NULL,
    ->     TrainingID INTEGER NOT NULL,
    ->     TrainerFirstName VARCHAR(20) NOT NULL,
    ->     TrainerLastName VARCHAR(20),
    ->     PRIMARY KEY (TrainerID),
    ->     FOREIGN KEY (TrainingID) REFERENCES Training(TrainingID)
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql> SHOW CREATE TABLE Trainer;
+-----+-----+
| Table | Create Table
+-----+-----+
| Trainer | CREATE TABLE `Trainer` (
`TrainerID` int NOT NULL,
`TrainingID` int NOT NULL,
`TrainerFirstName` varchar(20) NOT NULL,
`TrainerLastName` varchar(20) DEFAULT NULL,
PRIMARY KEY (`TrainerID`),
KEY `TrainingID` (`TrainingID`),
CONSTRAINT `Trainer_ibfk_1` FOREIGN KEY (`TrainingID`) REFERENCES `Training` (`TrainingID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+-----+
(END)
```

(c) Insert the complete set of data provided in the Excel file (uploaded in the Assignment folder) and show the insert statements used.

(d) Retrieve the data from the Trainer table by using the SELECT * statement and order by PK column(s). Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql> INSERT INTO Trainer (TrainerID, TrainingID, TrainerFirstName, TrainerLastName) VALUES
    -> (1, 1, 'James', 'Smith'),
    -> (2, 1, 'Johnny', 'Khor'),
    -> (3, 2, 'Michael', 'Smith'),
    -> (4, 3, 'Maria', 'Garcia'),
    -> (5, 4, 'John', NULL),
    -> (6, 4, 'Paul', 'Deitel'),
    -> (7, 5, 'Mike', 'Taylor'),
    -> (8, 5, 'Avinash', 'Navlani'),
    -> (9, 6, 'Robert', 'Smith'),
    -> (10, 7, 'Maria', 'Rodriguez'),
    -> (11, 8, 'Mike', 'Donlon'),
    -> (12, 9, 'Kathy', 'Corby'),
    -> (13, 10, 'Mary', 'Garcia'),
    -> (14, 10, 'Vanessa', NULL),
    -> (15, 11, 'Jordan', NULL),
    -> (16, 11, 'Maria', 'Hernandez');
Query OK, 16 rows affected (0.01 sec)
Records: 16  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM Trainer ORDER BY TrainerID;
+-----+-----+-----+-----+
| TrainerID | TrainingID | TrainerFirstName | TrainerLastName |
+-----+-----+-----+-----+
|      1 |       1 |      James |      Smith |
|      2 |       1 |     Johnny |      Khor |
|      3 |       2 |    Michael |      Smith |
|      4 |       3 |      Maria |      Garcia |
|      5 |       4 |      John |        NULL |
|      6 |       4 |      Paul |      Deitel |
|      7 |       5 |      Mike |      Taylor |
|      8 |       5 |    Avinash |      Navlani |
|      9 |       6 |    Robert |      Smith |
|     10 |       7 |      Maria | Rodriguez |
|     11 |       8 |      Mike |      Donlon |
|     12 |       9 |     Kathy |      Corby |
|     13 |      10 |      Mary |      Garcia |
|     14 |      10 |    Vanessa |        NULL |
|     15 |      11 |    Jordan |        NULL |
|     16 |      11 |      Maria | Hernandez |
+-----+-----+-----+-----+
16 rows in set (0.00 sec)
```

```
mysql>
```

Q9. Retrieve the data from the Trainer table by using the SELECT * statement with filter, WHERE TrainerLastName IS NULL. Show the output. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql>
mysql> SELECT * FROM Trainer WHERE TrainerLastName IS NULL ORDER BY TrainerID;
+-----+-----+-----+-----+
| TrainerID | TrainingID | TrainerFirstName | TrainerLastName |
+-----+-----+-----+-----+
|      5   |      4   |    John        |     NULL       |
|     14   |     10   | Vanessa       |     NULL       |
|     15   |     11   | Jordan        |     NULL       |
+-----+-----+-----+-----+
3 rows in set (0.01 sec)

mysql>
```

Q10. By using the SHOW tables statements, show the list of tables you have created in the Worker database. Show the screenshot of the execution of the above statements and results. Make sure you show the print screen of the complete set of the rows and columns. The rows must be ordered by PK column(s).

```
mysql> SHOW TABLES;
+-----+
| Tables_in_Worker |
+-----+
| Department
| Employee
| EmployeeEquipment
| EmployeeTraining
| Equipment
| Trainer
| Training
+-----+
7 rows in set (0.00 sec)

mysql>
```

Q11. Write a single-row subquery to display EmployeeID, FirstName, LastName, and HireDate of employees hired after employee Vivek Pandey. Sort the results by EmployeeID. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
+-----+
| EmployeeID | FirstName | LastName | HireDate |
+-----+
|      3 | Kathy     | Cooper   | 2011-11-18 |
|      4 | John      | Wilson   | 2017-03-19 |
|      5 | Nola      | Davis    | 2016-03-23 |
|      6 | Tom       | Harper   | 2010-04-11 |
+-----+
4 rows in set (0.02 sec)

mysql> ■
```

Q12. Write a query to display FirstName, LastName, and TrainingName for employee Tom Harper. Sort the results by TrainingName. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
+-----+
| FirstName | LastName | TrainingName |
+-----+
| Tom       | Harper   | Code of Conduct Training |
| Tom       | Harper   | Travel and Expense Management |
+-----+
2 rows in set (0.02 sec)

mysql> ■
```

Q13. Write a query to display the complete list of Trainings, and trainers (first and last name) available for each training. Sort the output by TrainingName and Trainers' first and last name. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
+-----+
| TrainingName           | TrainerFirstName | TrainerLastName |
+-----+
| Code of Conduct Training | Michael        | Smith          |
| COVID-19 Awareness and Protection Training | James          | Smith          |
| COVID-19 Awareness and Protection Training | Johnny         | Khor           |
| Employee Relations     | Mary           | Garcia          |
| Employee Relations     | Vanessa        | NULL            |
| Intro to Python         | John           | NULL            |
| Intro to Python         | Paul           | Deitel          |
| Machine Learning        | Avinash        | Navlani          |
| Machine Learning        | Mike           | Taylor          |
| Microsoft Certifications | Robert        | Smith          |
| Product Knowledge       | Mike           | Donlon          |
| Safety Training          | Maria          | Garcia          |
| Sales Skills             | Kathy          | Corby           |
| Security and Privacy    | Maria          | Rodriguez        |
| Travel and Expense Management | Jordan        | NULL            |
| Travel and Expense Management | Maria          | Hernandez        |
+-----+
16 rows in set (0.01 sec)

mysql> ■
```

Q14. Write a multiple-row subquery to display EmployeeID, FirstName, LastName, and HireDate of employees who work for the following departments: Accounting and Finance, IT Support, and Production. Sort the results by EmployeeID. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql> SELECT E.EmployeeID, E.FirstName, E.LastName, E.HireDate
-> FROM Employee E
-> INNER JOIN Department D ON E.DepartmentID = D.DepartmentID
-> WHERE D.DepartmentNm IN ('Accounting and Finance', 'IT Support', 'Production')
-> ORDER BY E.EmployeeID;
+-----+-----+-----+-----+
| EmployeeID | FirstName | LastName | HireDate |
+-----+-----+-----+-----+
|       1 | Andy     | Wong    | 2001-01-15 |
|       2 | Vivek    | Pandey  | 2003-11-15 |
|       6 | Tom      | Harper  | 2010-04-11 |
+-----+-----+-----+-----+
3 rows in set (0.02 sec)

mysql>
```

Q15. Write a query to display the EmployeeID, FirstName, LastName, EquipmentName, and EquipmentCostAmount for one of the employees. Sort the results by EmployeeID. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql>
mysql>
mysql> SELECT E.EmployeeID, E.FirstName, E.LastName, D.DepartmentNm AS DepartmentName, T.TrainingName
-> FROM Employee E
-> INNER JOIN Department D ON E.DepartmentID = D.DepartmentID
-> INNER JOIN EmployeeTraining ET ON E.EmployeeID = ET.EmployeeID
-> INNER JOIN Training T ON ET.TrainingID = T.TrainingID
-> ORDER BY E.EmployeeID, T.TrainingName;
+-----+-----+-----+-----+-----+
| EmployeeID | FirstName | LastName | DepartmentName | TrainingName |
+-----+-----+-----+-----+-----+
|       1 | Andy     | Wong    | Production     | Code of Conduct Training |
|       1 | Andy     | Wong    | Production     | Safety Training          |
|       2 | Vivek    | Pandey  | IT Support     | Code of Conduct Training |
|       2 | Vivek    | Pandey  | IT Support     | Microsoft Certifications |
|       2 | Vivek    | Pandey  | IT Support     | Security and Privacy   |
|       3 | Kathy    | Cooper  | Human Resource Management | Code of Conduct Training |
|       3 | Kathy    | Cooper  | Human Resource Management | Employee Relations        |
|       4 | John     | Wilson  | Research & Development | Code of Conduct Training |
|       4 | John     | Wilson  | Research & Development | Intro to Python           |
|       4 | John     | Wilson  | Research & Development | Machine Learning          |
|       5 | Nola     | Davis   | Sales & Marketing | Code of Conduct Training |
|       5 | Nola     | Davis   | Sales & Marketing | Product Knowledge         |
|       5 | Nola     | Davis   | Sales & Marketing | Sales Skills              |
|       6 | Tom      | Harper  | Accounting and Finance | Code of Conduct Training |
|       6 | Tom      | Harper  | Accounting and Finance | Travel and Expense Management |
+-----+-----+-----+-----+-----+
15 rows in set (0.01 sec)

mysql>
```

Q16. Write a query to display the TrainingID, TrainingName, TrainerID, TrainerFirstName, and TrainerLastName with the trainers who did not provide their last name. Sort the results by TrainingID and TrainerID. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql>
mysql>
mysql> SELECT T.TrainingID, T.TrainingName, TT.TrainerID, TT.TrainerFirstName, TT.TrainerLastName
-> FROM Trainer TT
-> INNER JOIN Training T ON T.TrainingID = TT.TrainingID
-> WHERE TT.TrainerLastName IS NULL
-> ORDER BY T.TrainingID, TT.TrainerID;
+-----+-----+-----+-----+-----+
| TrainingID | TrainingName | TrainerID | TrainerFirstName | TrainerLastName |
+-----+-----+-----+-----+-----+
|       4 | Intro to Python |       5 | John           | NULL            |
|      10 | Employee Relations |      14 | Vanessa        | NULL            |
|      11 | Travel and Expense Management |      15 | Jordan         | NULL            |
+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)

mysql>
```

Q17. Write a query to display the distinct list of equipments used by the current employees. Sort the output by EquipmentName. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql>
mysql> SELECT DISTINCT E.EquipmentName
      -> FROM Equipment E
      -> INNER JOIN EmployeeEquipment EE ON E.EquipmentID = EE.EquipmentID
      -> ORDER BY E.EquipmentName;
+-----+
| EquipmentName      |
+-----+
| Computer Monitor  |
| Headsets           |
| Notebook Computers |
+-----+
3 rows in set (0.01 sec)
```

Q18. Write a query to display the FirstName, LastName, TrainingName, and trainer(s) (with first and last name in two separate columns) for one of the employees. Sort the results by TrainingName and TrainerFirstName. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql>
mysql>
mysql> SELECT E.FirstName, E.LastName, T.TrainingName, TR.TrainerFirstName, TR.TrainerLastName
      -> FROM Employee E
      -> INNER JOIN EmployeeTraining ET ON E.EmployeeID = ET.EmployeeID
      -> INNER JOIN Training T ON ET.TrainingID = T.TrainingID
      -> INNER JOIN Trainer TR ON T.TrainingID = TR.TrainingID
      -> WHERE E.FirstName = 'Tom' AND E.LastName = 'Harper'
      -> ORDER BY T.TrainingName, TR.TrainerFirstName;
+-----+-----+-----+-----+-----+
| FirstName | LastName | TrainingName      | TrainerFirstName | TrainerLastName |
+-----+-----+-----+-----+-----+
| Tom       | Harper   | Code of Conduct Training | Michael        | Smith          |
| Tom       | Harper   | Travel and Expense Management | Jordan         | NULL           |
| Tom       | Harper   | Travel and Expense Management | Maria          | Hernandez      |
+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

Q19. Write a query to display the EmployeeID, FirstName, LastName, DepartmentID, DepartmentName, EquipmentID, EquipmentName for all employees. Sort the results by EmployeeID, DepartmentID, and EquipmentID. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql>
mysql>
mysql> SELECT E.EmployeeID, E.FirstName, E.LastName, D.DepartmentID, D.DepartmentNm AS DepartmentName, EE.EquipmentID, EQ.EquipmentName
      -> FROM Employee E
      -> INNER JOIN Department D ON E.DepartmentID = D.DepartmentID
      -> INNER JOIN EmployeeEquipment EE ON E.EmployeeID = EE.EmployeeID
      -> INNER JOIN Equipment EQ ON EE.EquipmentID = EQ.EquipmentID
      -> ORDER BY E.EmployeeID, D.DepartmentID, EE.EquipmentID;
+-----+-----+-----+-----+-----+-----+
| EmployeeID | FirstName | LastName | DepartmentID | DepartmentName      | EquipmentID | EquipmentName |
+-----+-----+-----+-----+-----+-----+
| 1 | Andy    | Wong     | 2 | Production      | 1 | Notebook Computers |
| 2 | Vivek   | Pandey   | 3 | IT Support       | 1 | Notebook Computers |
| 2 | Vivek   | Pandey   | 3 | IT Support       | 2 | Headsets          |
| 2 | Vivek   | Pandey   | 3 | IT Support       | 3 | Computer Monitor |
| 3 | Kathy   | Cooper   | 8 | Human Resource Management | 1 | Notebook Computers |
| 3 | Kathy   | Cooper   | 8 | Human Resource Management | 2 | Headsets          |
| 3 | Kathy   | Cooper   | 8 | Human Resource Management | 3 | Computer Monitor |
| 4 | John    | Wilson   | 1 | Research & Development | 1 | Notebook Computers |
| 4 | John    | Wilson   | 1 | Research & Development | 3 | Computer Monitor |
| 5 | Nola   | Davis    | 7 | Sales & Marketing  | 1 | Notebook Computers |
| 5 | Nola   | Davis    | 7 | Sales & Marketing  | 2 | Headsets          |
| 6 | Tom    | Harper   | 9 | Accounting and Finance | 1 | Notebook Computers |
| 6 | Tom    | Harper   | 9 | Accounting and Finance | 3 | Computer Monitor |
+-----+-----+-----+-----+-----+-----+
13 rows in set (0.01 sec)
```

Q20. Write a query to display the EmployeeID, FirstName, LastName, DepartmentID, DepartmentName, TrainingID, TrainingName for all employees. Sort the results by EmployeeID, DepartmentID, and TrainingID. Make sure you show the print screen of the complete set of the rows, and columns as specified.

```
mysql>
mysql>
mysql> SELECT E.EmployeeID, E.FirstName, E.LastName, D.DepartmentID, D.DepartmentNm AS DepartmentName, ET.TrainingID, T.TrainingName
-> FROM Employee E
-> INNER JOIN Department D ON E.DepartmentID = D.DepartmentID
-> INNER JOIN EmployeeTraining ET ON E.EmployeeID = ET.EmployeeID
-> INNER JOIN Training T ON ET.TrainingID = T.TrainingID
-> ORDER BY E.EmployeeID, D.DepartmentID, ET.TrainingID;
+-----+-----+-----+-----+-----+-----+-----+
| EmployeeID | FirstName | LastName | DepartmentID | DepartmentName | TrainingID | TrainingName |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Andy | Wong | 2 | Production | 2 | Code of Conduct Training
| 1 | Andy | Wong | 2 | Production | 3 | Safety Training
| 2 | Vivek | Pandey | 3 | IT Support | 2 | Code of Conduct Training
| 2 | Vivek | Pandey | 3 | IT Support | 6 | Microsoft Certifications
| 2 | Vivek | Pandey | 3 | IT Support | 7 | Security and Privacy
| 3 | Kathy | Cooper | 8 | Human Resource Management | 2 | Code of Conduct Training
| 3 | Kathy | Cooper | 8 | Human Resource Management | 10 | Employee Relations
| 4 | John | Wilson | 1 | Research & Development | 2 | Code of Conduct Training
| 4 | John | Wilson | 1 | Research & Development | 4 | Intro to Python
| 4 | John | Wilson | 1 | Research & Development | 5 | Machine Learning
| 5 | Nola | Davis | 7 | Sales & Marketing | 2 | Code of Conduct Training
| 5 | Nola | Davis | 7 | Sales & Marketing | 8 | Product Knowledge
| 5 | Nola | Davis | 7 | Sales & Marketing | 9 | Sales Skills
| 6 | Tom | Harper | 9 | Accounting and Finance | 2 | Code of Conduct Training
| 6 | Tom | Harper | 9 | Accounting and Finance | 11 | Travel and Expense Management
+-----+-----+-----+-----+-----+-----+
15 rows in set (0.01 sec)
```