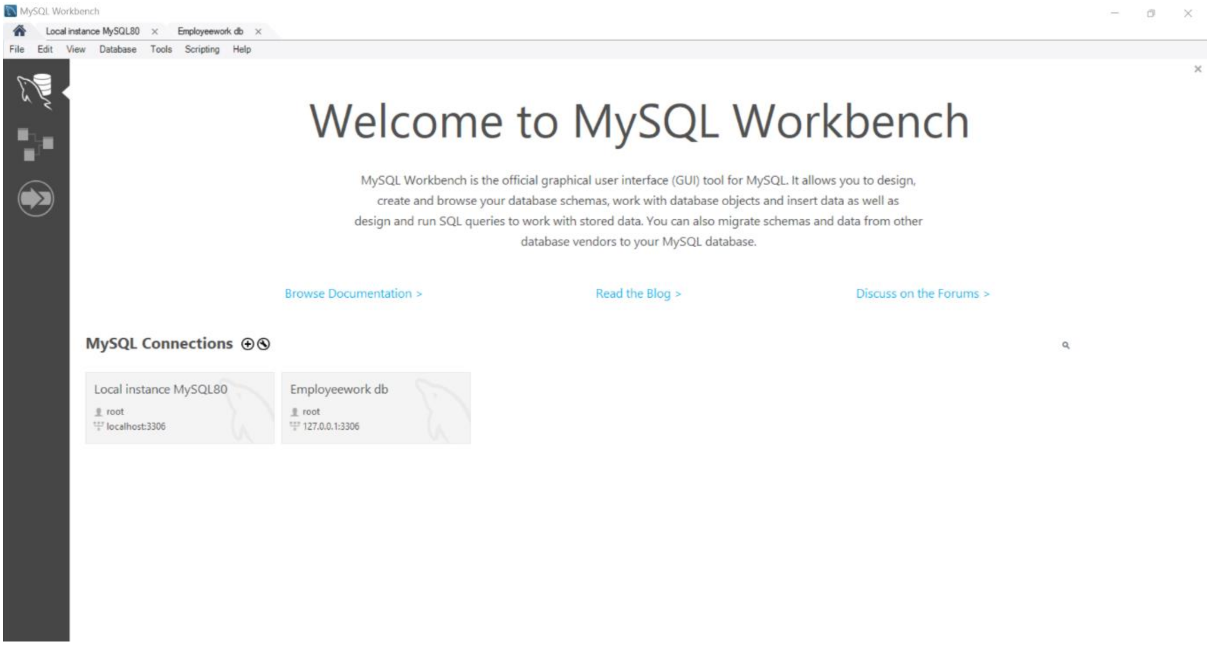
Assignment-3 Data Warehousing and Analytics in the cloud

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.

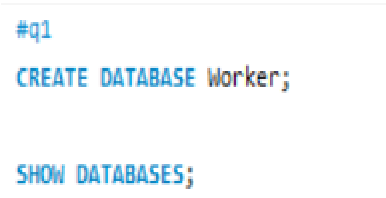


Welcome to MySQL Workbench

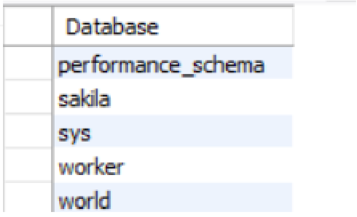
MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Since I have already MySQL installed in my computer. I have attached the screenshot of the Homepage of MySQL Workbench.

I have created a connection as “Employeework db”.



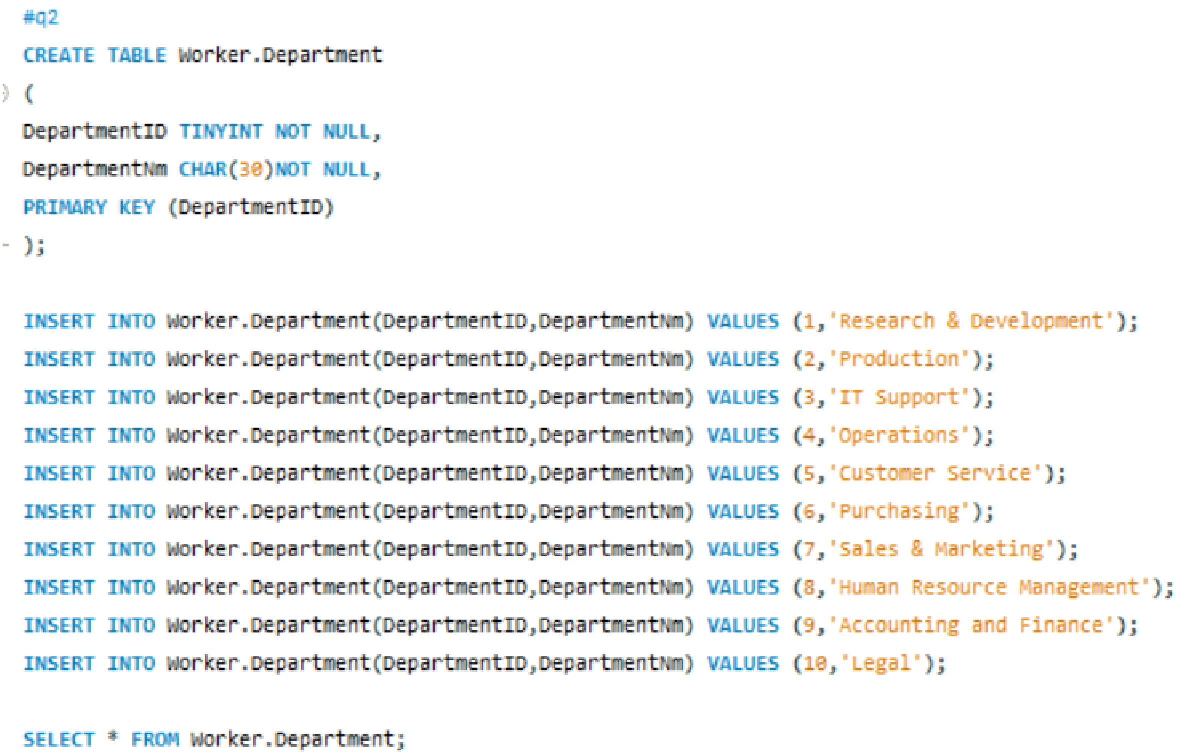
```
#q1
CREATE DATABASE Worker;


SHOW DATABASES;
```



Database
performance_schema
sakila
sys
worker
world

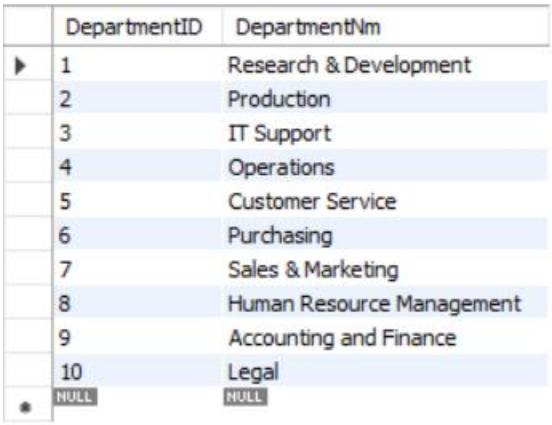
Created a database “worker” and as it is displayed in the list of databases that are available.

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. (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).



```
#q2
CREATE TABLE Worker.Department
(
DepartmentID TINYINT NOT NULL,
DepartmentNm CHAR(30)NOT NULL,
PRIMARY KEY (DepartmentID)
);

INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (1,'Research & Development');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (2,'Production');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (3,'IT Support');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (4,'Operations');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (5,'Customer Service');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (6,'Purchasing');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (7,'Sales & Marketing');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (8,'Human Resource Management');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (9,'Accounting and Finance');
INSERT INTO Worker.Department(DepartmentID,DepartmentNm) VALUES (10,'Legal');

SELECT * FROM Worker.Department;
```

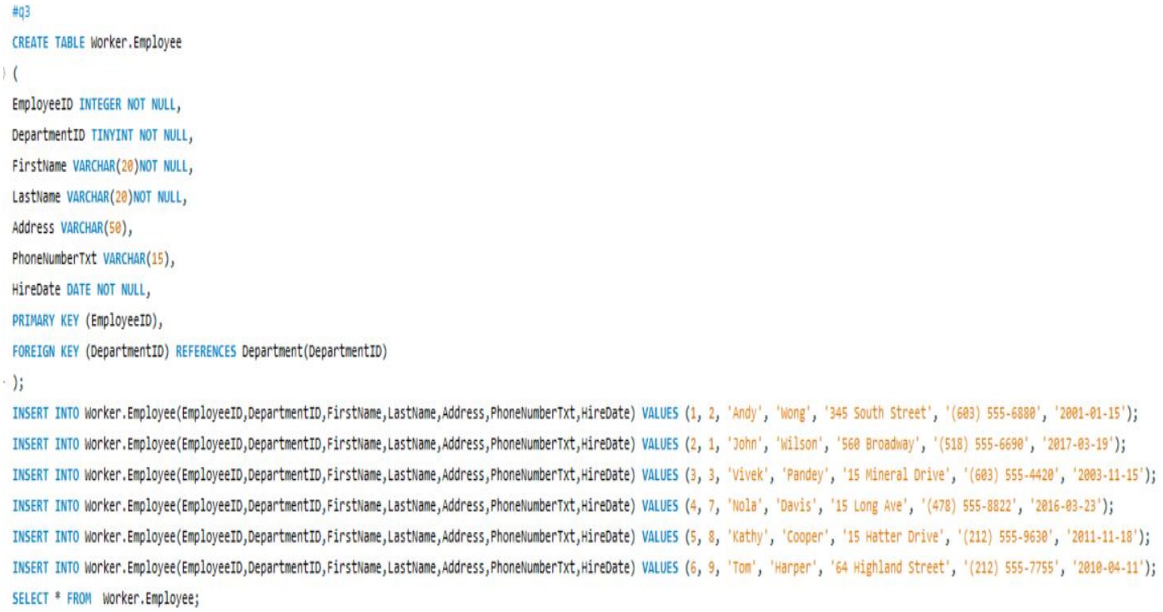
As you can see above created a table with DepartmentID and DepartmentNm which are available in the specific organisation



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
NULL	NULL

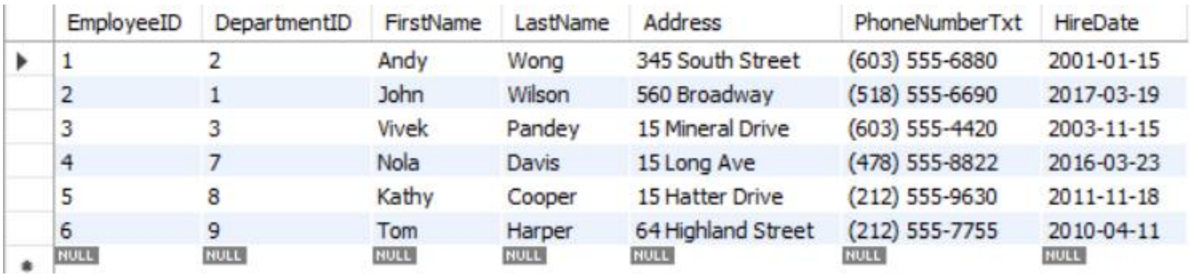
From the above code the table is displayed as above.

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). (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).



```
#q3
CREATE TABLE Worker.Employee
(
EmployeeID INTEGER NOT NULL,
DepartmentID TINYINT NOT NULL,
FirstName VARCHAR(20)NOT NULL,
LastName VARCHAR(20)NOT NULL,
Address VARCHAR(50),
PhoneNumberTxt VARCHAR(15),
HireDate DATE NOT NULL,
PRIMARY KEY (EmployeeID),
FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
);
INSERT INTO Worker.Employee(EmployeeID,DepartmentID,FirstName,LastName,Address,PhoneNumberTxt,HireDate) VALUES (1, 2, 'Andy', 'Wong', '345 South Street', '(603) 555-6880', '2001-01-15');
INSERT INTO Worker.Employee(EmployeeID,DepartmentID,FirstName,LastName,Address,PhoneNumberTxt,HireDate) VALUES (2, 1, 'John', 'Wilson', '560 Broadway', '(518) 555-6690', '2017-03-19');
INSERT INTO Worker.Employee(EmployeeID,DepartmentID,FirstName,LastName,Address,PhoneNumberTxt,HireDate) VALUES (3, 3, 'Vivek', 'Pandey', '15 Mineral Drive', '(603) 555-4420', '2003-11-15');
INSERT INTO Worker.Employee(EmployeeID,DepartmentID,FirstName,LastName,Address,PhoneNumberTxt,HireDate) VALUES (4, 7, 'Nola', 'Davis', '15 Long Ave', '(478) 555-8822', '2016-03-23');
INSERT INTO Worker.Employee(EmployeeID,DepartmentID,FirstName,LastName,Address,PhoneNumberTxt,HireDate) VALUES (5, 8, 'Kathy', 'Cooper', '15 Hatter Drive', '(212) 555-9630', '2011-11-18');
INSERT INTO Worker.Employee(EmployeeID,DepartmentID,FirstName,LastName,Address,PhoneNumberTxt,HireDate) VALUES (6, 9, 'Tom', 'Harper', '64 Highland Street', '(212) 555-7755', '2010-04-11');
SELECT * FROM  Worker.Employee;
```

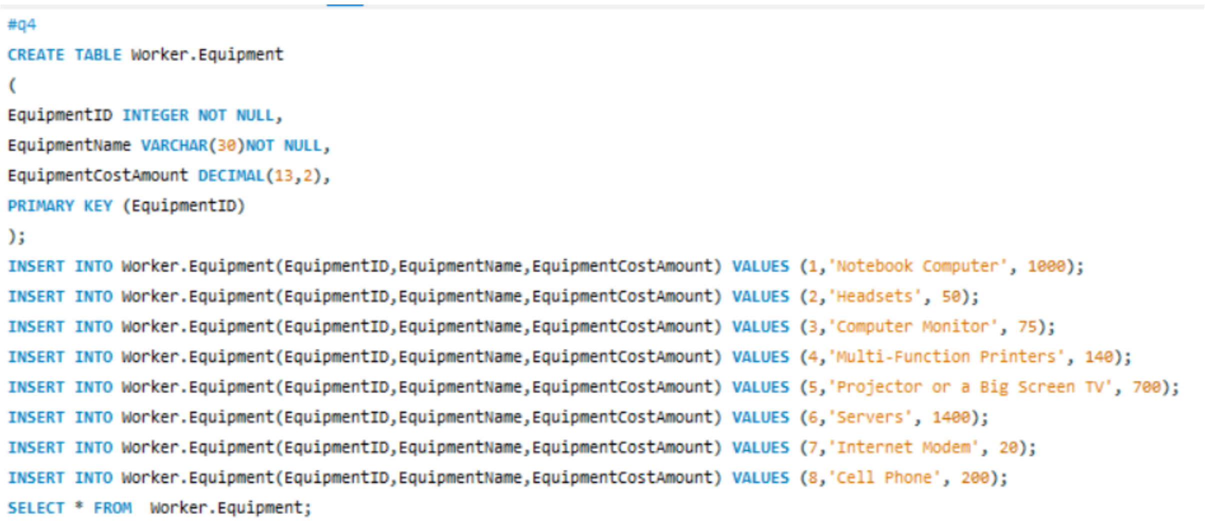
As you can see the we are creating the table with the Employee details such as employee id, first name, last name, address etc.. and filling all the details



EmployeeID	DepartmentID	FirstName	LastName	Address	PhoneNumberTxt	HireDate
1	2	Andy	Wong	345 South Street	(603) 555-6880	2001-01-15
2	1	John	Wilson	560 Broadway	(518) 555-6690	2017-03-19
3	3	Vivek	Pandey	15 Mineral Drive	(603) 555-4420	2003-11-15
4	7	Nola	Davis	15 Long Ave	(478) 555-8822	2016-03-23
5	8	Kathy	Cooper	15 Hatter Drive	(212) 555-9630	2011-11-18
6	9	Tom	Harper	64 Highland Street	(212) 555-7755	2010-04-11
NULL	NULL	NULL	NULL	NULL	NULL	NULL

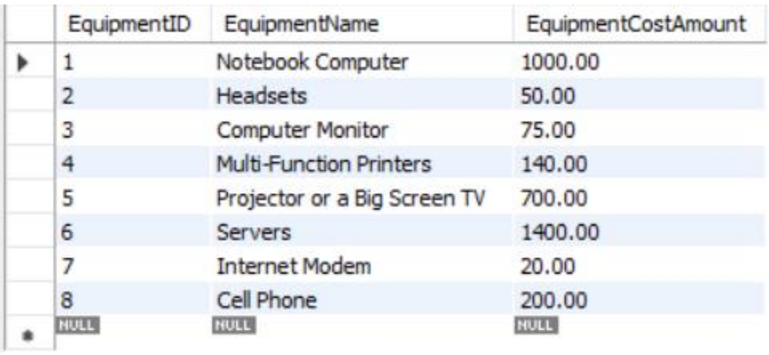
From the above code the employee table is displayed above

Q4. Create the Equipment 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. (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).



```
#Q4
CREATE TABLE Worker.Equipment
(
EquipmentID INTEGER NOT NULL,
EquipmentName VARCHAR(30)NOT NULL,
EquipmentCostAmount DECIMAL(13,2),
PRIMARY KEY (EquipmentID)
);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (1,'Notebook Computer', 1000);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (2,'Headsets', 50);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (3,'Computer Monitor', 75);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (4,'Multi-Function Printers', 140);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (5,'Projector or a Big Screen TV', 700);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (6,'Servers', 1400);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (7,'Internet Modem', 20);
INSERT INTO Worker.Equipment(EquipmentID,EquipmentName,EquipmentCostAmount) VALUES (8,'Cell Phone', 200);
SELECT * FROM  Worker.Equipment;
```

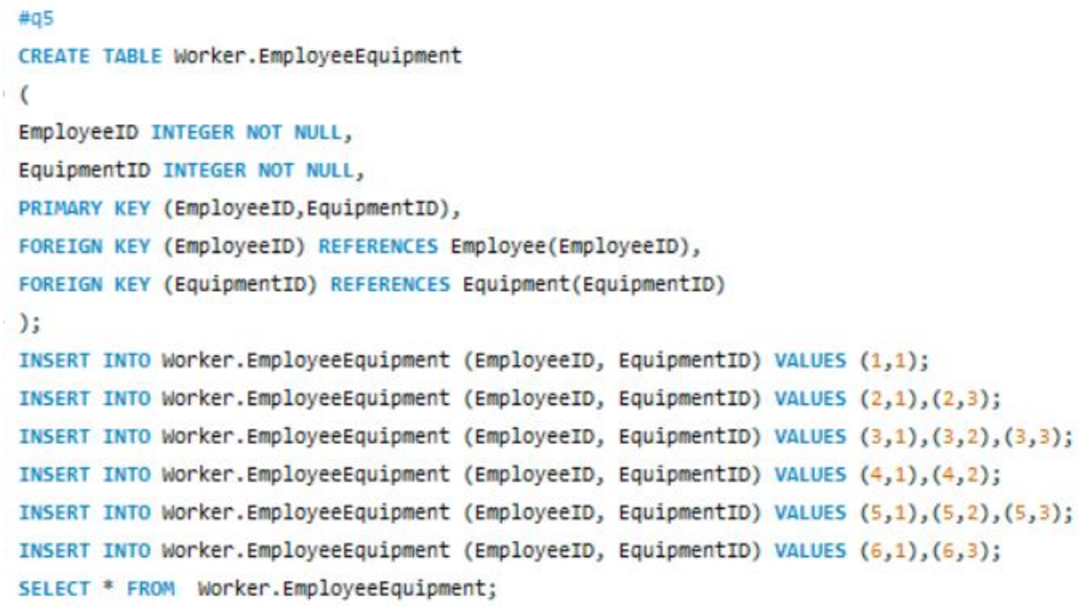
As you can see, we have created a table for the equipment with the equipment id, name and cost.



	EquipmentID	EquipmentName	EquipmentCostAmount
▶	1	Notebook Computer	1000.00
	2	Headsets	50.00
	3	Computer Monitor	75.00
	4	Multi-Function Printers	140.00
	5	Projector or a Big Screen TV	700.00
	6	Servers	1400.00
	7	Internet Modem	20.00
	8	Cell Phone	200.00
*	NULL	NULL	NULL

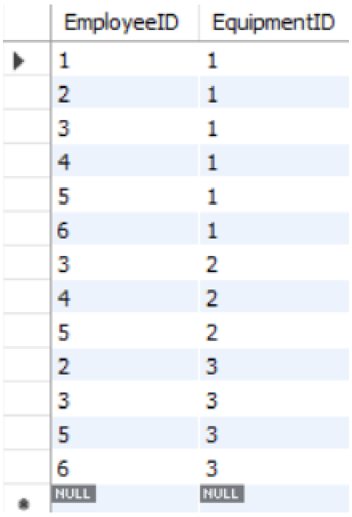
From the above code the equipment table is displayed.

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. (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).



```
#q5
CREATE TABLE Worker.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)
);
INSERT INTO Worker.EmployeeEquipment (EmployeeID, EquipmentID) VALUES (1,1);
INSERT INTO Worker.EmployeeEquipment (EmployeeID, EquipmentID) VALUES (2,1),(2,3);
INSERT INTO Worker.EmployeeEquipment (EmployeeID, EquipmentID) VALUES (3,1),(3,2),(3,3);
INSERT INTO Worker.EmployeeEquipment (EmployeeID, EquipmentID) VALUES (4,1),(4,2);
INSERT INTO Worker.EmployeeEquipment (EmployeeID, EquipmentID) VALUES (5,1),(5,2),(5,3);
INSERT INTO Worker.EmployeeEquipment (EmployeeID, EquipmentID) VALUES (6,1),(6,3);
SELECT * FROM  Worker.EmployeeEquipment;
```

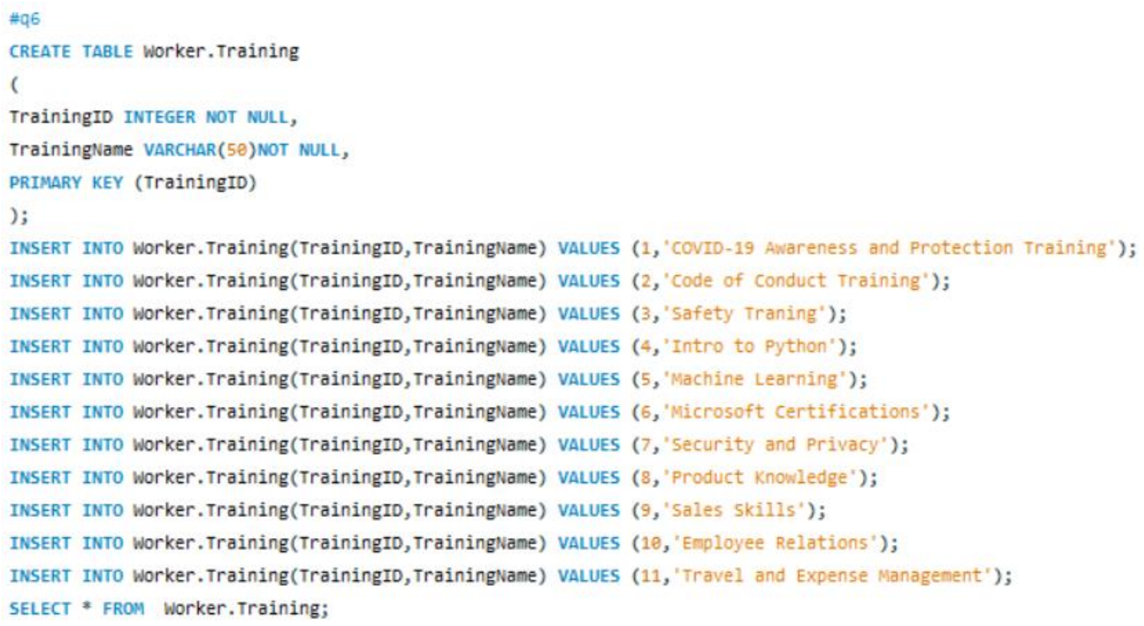
We are writing a code for junction entity between Employee and Equipment tables.



EmployeeID	EquipmentID
1	1
2	1
3	1
4	1
5	1
6	1
3	2
4	2
5	2
2	3
3	3
5	3
6	3
NULL	NULL

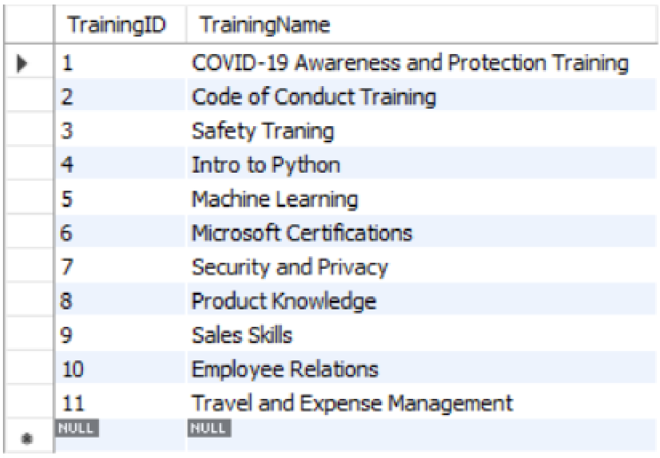
Above is the table of junction entity as Employee id and Equipment id.

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. (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).



```
#q6
CREATE TABLE Worker.Training
(
TrainingID INTEGER NOT NULL,
TrainingName VARCHAR(50)NOT NULL,
PRIMARY KEY (TrainingID)
);
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (1,'COVID-19 Awareness and Protection Training');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (2,'Code of Conduct Training');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (3,'Safety Traning');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (4,'Intro to Python');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (5,'Machine Learning');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (6,'Microsoft Certifications');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (7,'Security and Privacy');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (8,'Product Knowledge');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (9,'Sales Skills');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (10,'Employee Relations');
INSERT INTO Worker.Training(TrainingID,TrainingName) VALUES (11,'Travel and Expense Management');
SELECT * FROM  Worker.Training;
```

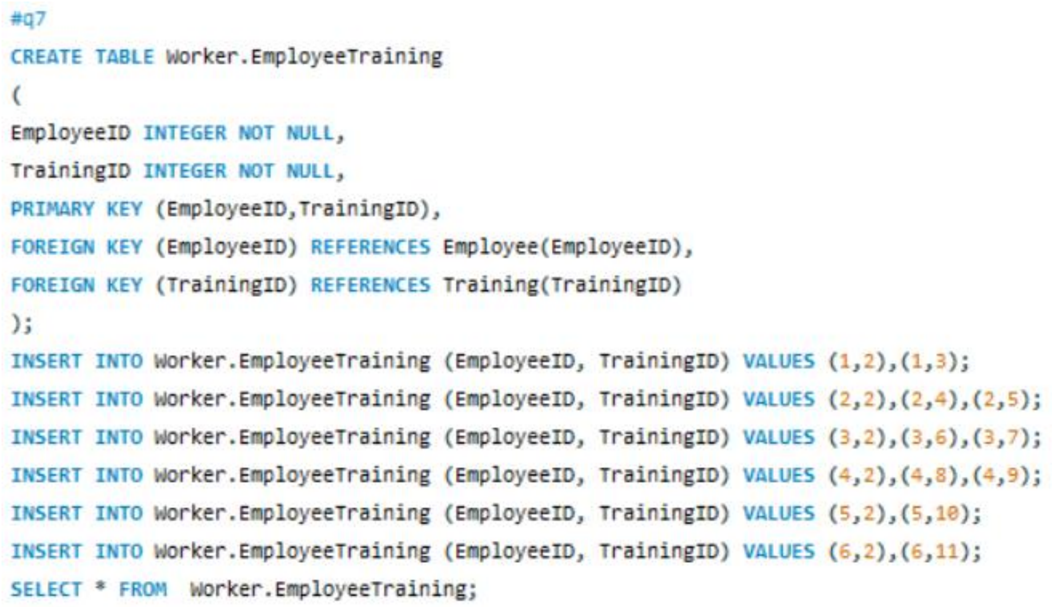
The above code represents the training table creation such as training id and training name.



TrainingID	TrainingName
1	COVID-19 Awareness and Protection Training
2	Code of Conduct Training
3	Safety Traning
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
NULL	NULL

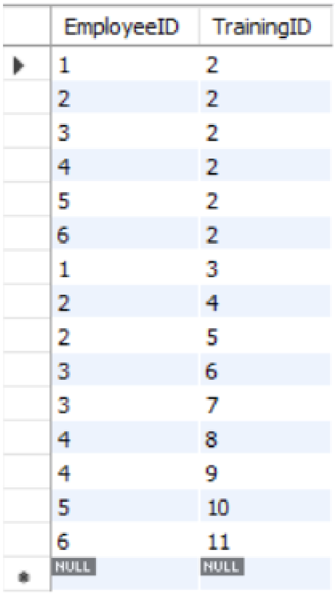
The table represents the training id and training name of the training given in the organisation.

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. (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).



```
#q7
CREATE TABLE Worker.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)
);
INSERT INTO Worker.EmployeeTraining (EmployeeID, TrainingID) VALUES (1,2),(1,3);
INSERT INTO Worker.EmployeeTraining (EmployeeID, TrainingID) VALUES (2,2),(2,4),(2,5);
INSERT INTO Worker.EmployeeTraining (EmployeeID, TrainingID) VALUES (3,2),(3,6),(3,7);
INSERT INTO Worker.EmployeeTraining (EmployeeID, TrainingID) VALUES (4,2),(4,8),(4,9);
INSERT INTO Worker.EmployeeTraining (EmployeeID, TrainingID) VALUES (5,2),(5,10);
INSERT INTO Worker.EmployeeTraining (EmployeeID, TrainingID) VALUES (6,2),(6,11);
SELECT * FROM  Worker.EmployeeTraining;
```

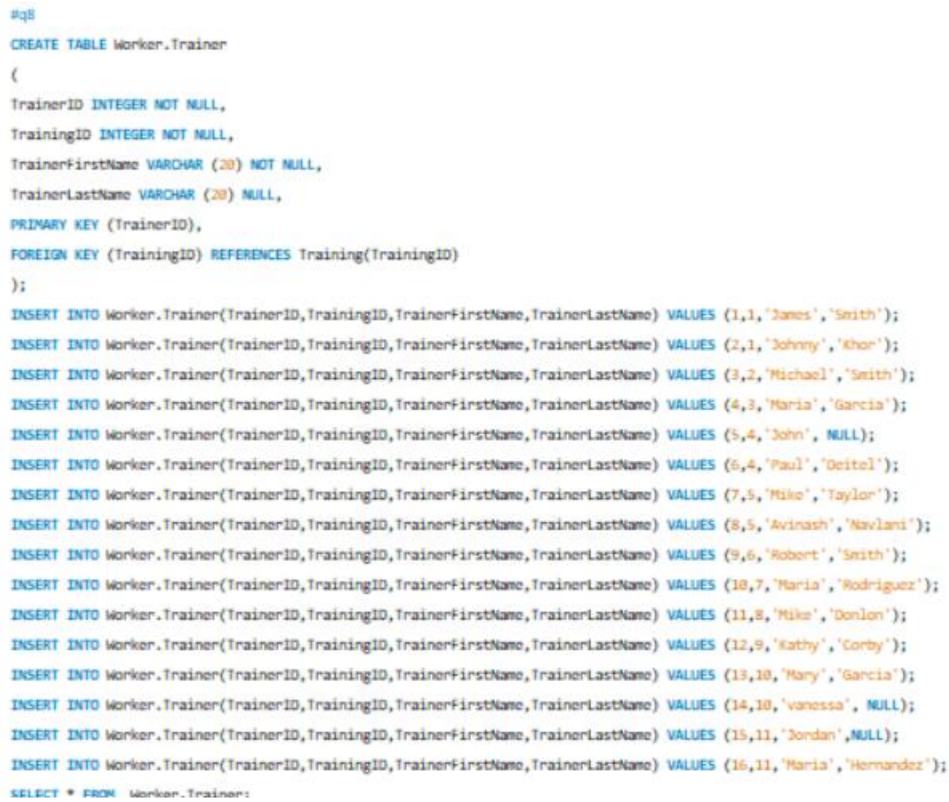
As we are creating a junction entity for the Employee table and Training table.



EmployeeID	TrainingID
1	2
2	2
3	2
4	2
5	2
6	2
1	3
2	4
2	5
3	6
3	7
4	8
4	9
5	10
6	11
NULL	NULL

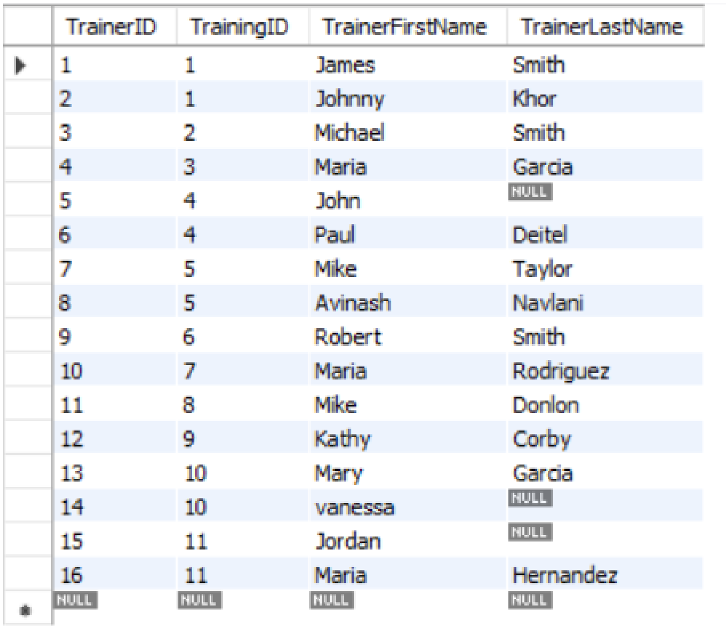
The table represents the Employee and Training ID.

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. (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).



```
#q8
CREATE TABLE Worker.Trainer
(
TrainerID INTEGER NOT NULL,
TrainingID INTEGER NOT NULL,
TrainerFirstName VARCHAR (20) NOT NULL,
TrainerLastName VARCHAR (20) NULL,
PRIMARY KEY (TrainerID),
FOREIGN KEY (TrainingID) REFERENCES Training(TrainingID)
);
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (1,1,'James','Smith');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (2,1,'Johnny','Khor');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (3,2,'Michael','Smith');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (4,3,'Maria','Garcia');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (5,4,'John', NULL);
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (6,4,'Paul','Deitel');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (7,5,'Mike','Taylor');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (8,5,'Avinash','Navlani');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (9,6,'Robert','Smith');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (10,7,'Maria','Rodriguez');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (11,8,'Mike','Donlon');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (12,9,'Kathy','Corby');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (13,10,'Mary','Garcia');
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (14,10,'vanessa', NULL);
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (15,11,'Jordan',NULL);
INSERT INTO Worker.Trainer(TrainerID,TrainingID,TrainerFirstName,TrainerLastName) VALUES (16,11,'Maria','Hernandez');
SELECT * FROM  Worker.Trainer;
```

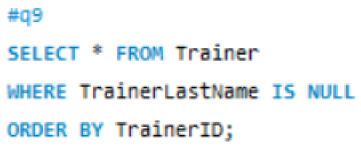
Creating a code for the table of the Trainers.



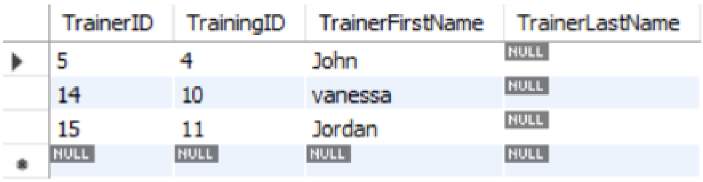
	TrainerID	TrainingID	TrainerFirstName	TrainerLastName
▶	1	1	James	Smith
	2	1	Johnny	Khor
	3	2	Michael	Smith
	4	3	Maria	Garcia
	5	4	John	NULL
	6	4	Paul	Deitel
	7	5	Mike	Taylor
	8	5	Avinash	Navlani
	9	6	Robert	Smith
	10	7	Maria	Rodriguez
	11	8	Mike	Donlon
	12	9	Kathy	Corby
	13	10	Mary	Garcia
	14	10	vanessa	NULL
	15	11	Jordan	NULL
	16	11	Maria	Hernandez
*	NULL	NULL	NULL	NULL

The trainer table looks as above.

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).



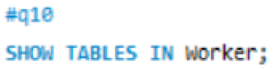
```
#q9
SELECT * FROM Trainer
WHERE TrainerLastName IS NULL
ORDER BY TrainerID;
```



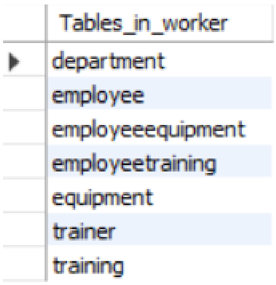
	TrainerID	TrainingID	TrainerFirstName	TrainerLastName
▶	5	4	John	NULL
	14	10	vanessa	NULL
	15	11	Jordan	NULL
*	NULL	NULL	NULL	NULL

John, Vanessa and Jordan doesn’t have the last name.

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.



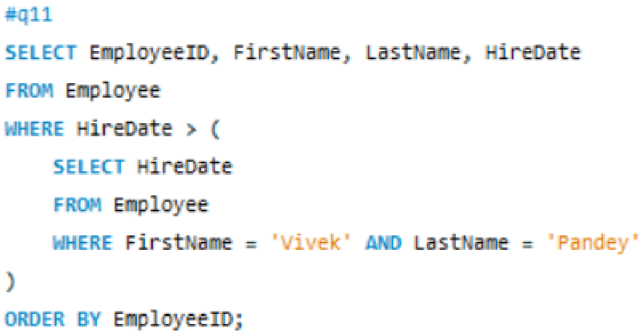
```
#q10
SHOW TABLES IN Worker;
```



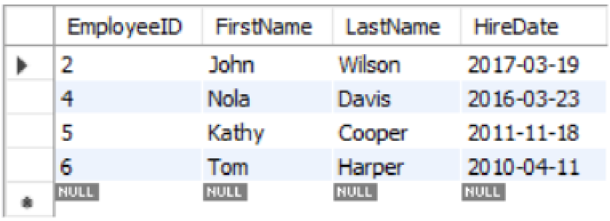
	Tables_in_worker
▶	department
	employee
	employeeequipment
	employeetraining
	equipment
	trainer
	training

These are the list of tables available in the Worker Database.

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.



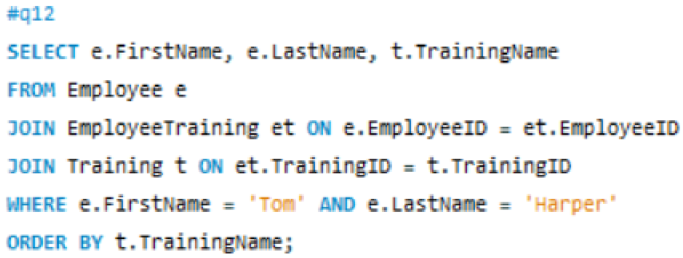
```
#q11
SELECT EmployeeID, FirstName, LastName, HireDate
FROM Employee
WHERE HireDate > (
    SELECT HireDate
    FROM Employee
    WHERE FirstName = 'Vivek' AND LastName = 'Pandey'
)
ORDER BY EmployeeID;
```



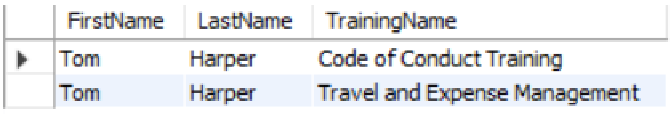
EmployeeID	FirstName	LastName	HireDate
2	John	Wilson	2017-03-19
4	Nola	Davis	2016-03-23
5	Kathy	Cooper	2011-11-18
6	Tom	Harper	2010-04-11
NULL	NULL	NULL	NULL

The above table represents the list of employees hired after Vivek Pandey.

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.



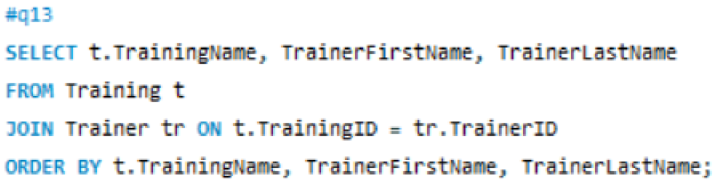
```
#q12
SELECT e.FirstName, e.LastName, t.TrainingName
FROM Employee e
JOIN EmployeeTraining et ON e.EmployeeID = et.EmployeeID
JOIN Training t ON et.TrainingID = t.TrainingID
WHERE e.FirstName = 'Tom' AND e.LastName = 'Harper'
ORDER BY t.TrainingName;
```



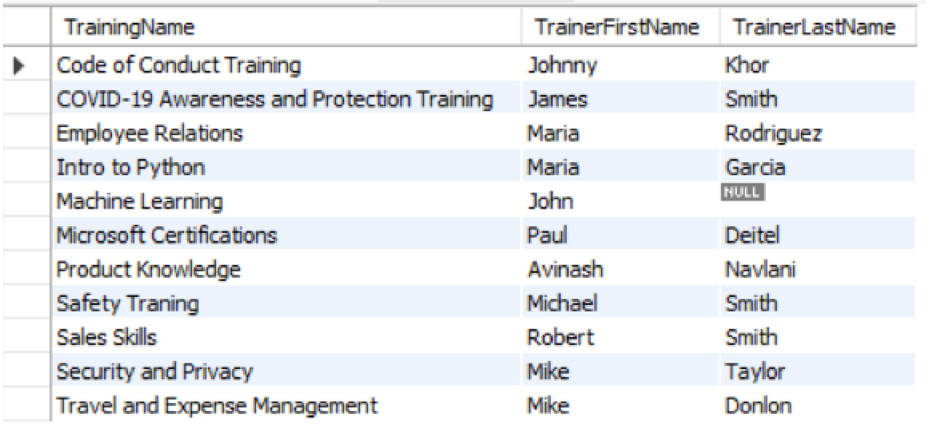
FirstName	LastName	TrainingName
Tom	Harper	Code of Conduct Training
Tom	Harper	Travel and Expense Management

The list of training taken by Tom Harper.

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.



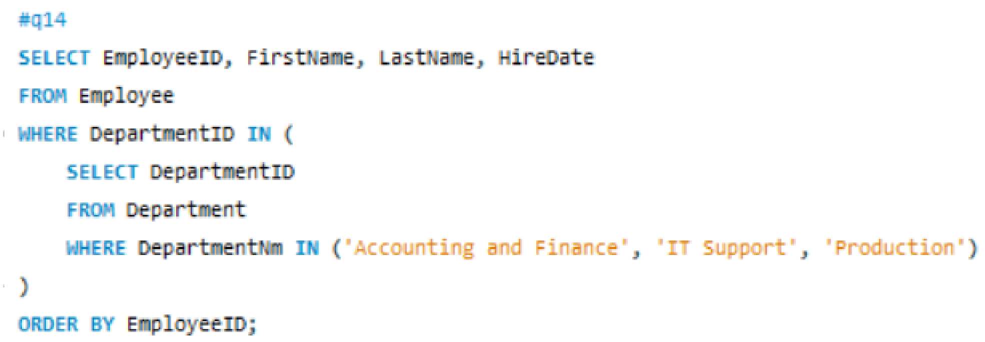
```
#q13
SELECT t.TrainingName, TrainerFirstName, TrainerLastName
FROM Training t
JOIN Trainer tr ON t.TrainingID = tr.TrainerID
ORDER BY t.TrainingName, TrainerFirstName, TrainerLastName;
```



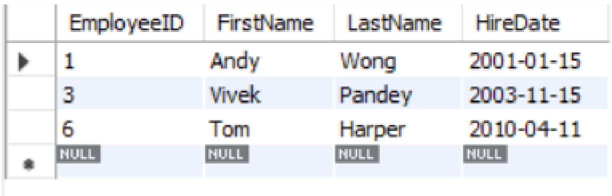
TrainingName	TrainerFirstName	TrainerLastName
Code of Conduct Training	Johnny	Khor
COVID-19 Awareness and Protection Training	James	Smith
Employee Relations	Maria	Rodriguez
Intro to Python	Maria	Garcia
Machine Learning	John	NULL
Microsoft Certifications	Paul	Deitel
Product Knowledge	Avinash	Navlani
Safety Traning	Michael	Smith
Sales Skills	Robert	Smith
Security and Privacy	Mike	Taylor
Travel and Expense Management	Mike	Donlon

The above is the list of training and trainer provided by the organisation.

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.



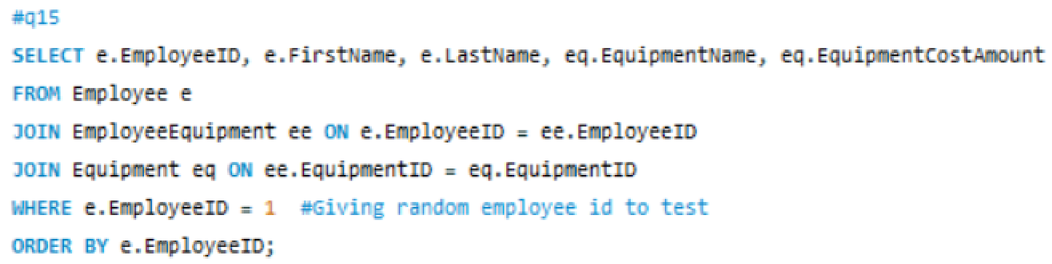
```
#q14
SELECT EmployeeID, FirstName, LastName, HireDate
FROM Employee
WHERE DepartmentID IN (
    SELECT DepartmentID
    FROM Department
    WHERE DepartmentNm IN ('Accounting and Finance', 'IT Support', 'Production')
)
ORDER BY EmployeeID;
```



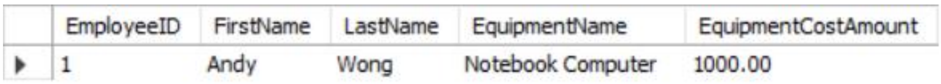
	EmployeeID	FirstName	LastName	HireDate
▶	1	Andy	Wong	2001-01-15
	3	Vivek	Pandey	2003-11-15
	6	Tom	Harper	2010-04-11
*	NULL	NULL	NULL	NULL

The list of employee who work in Accounting and Finance, IT Support and Production are as displayed above.

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.



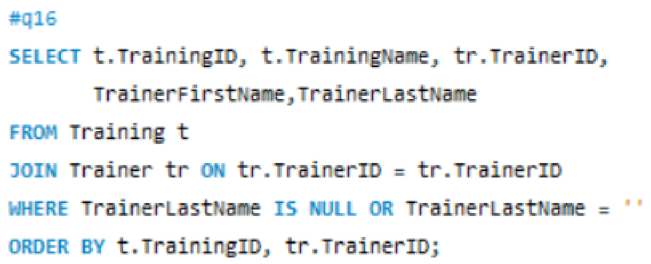
```
#q15
SELECT e.EmployeeID, e.FirstName, e.LastName, eq.EquipmentName, eq.EquipmentCostAmount
FROM Employee e
JOIN EmployeeEquipment ee ON e.EmployeeID = ee.EmployeeID
JOIN Equipment eq ON ee.EquipmentID = eq.EquipmentID
WHERE e.EmployeeID = 1  #Giving random employee id to test
ORDER BY e.EmployeeID;
```



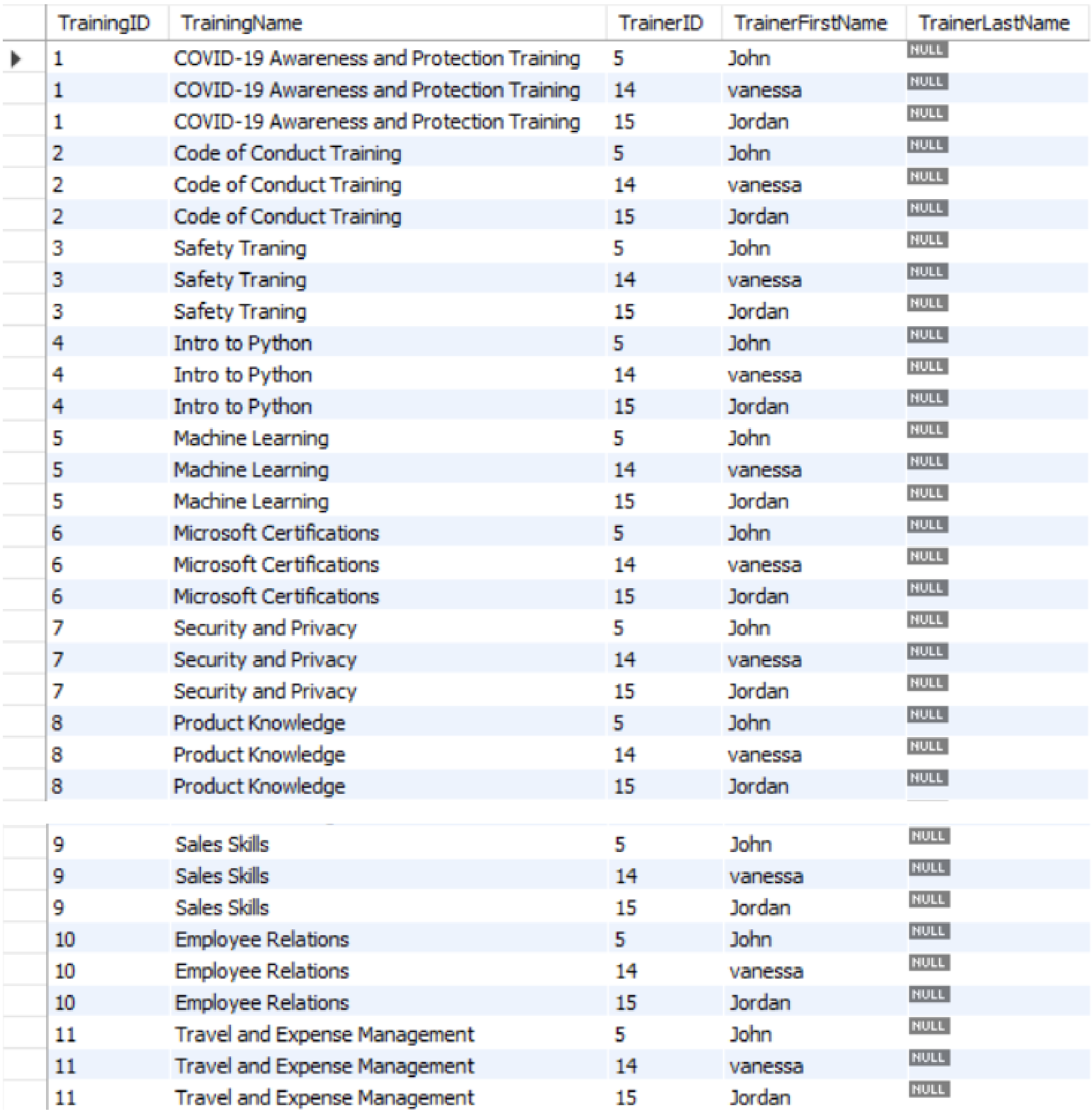
	EmployeeID	FirstName	LastName	EquipmentName	EquipmentCostAmount
▶	1	Andy	Wong	Notebook Computer	1000.00

Here we are giving the random employee Id as 1 and the results are displayed as follows for the specified employee.

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.



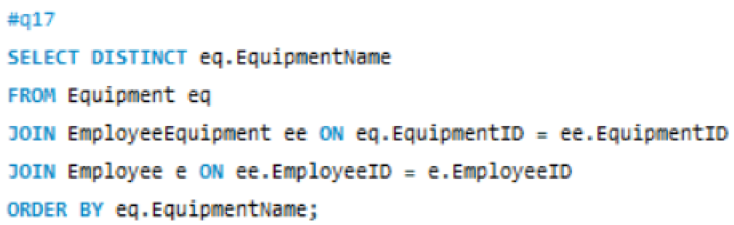
```
#q16
SELECT t.TrainingID, t.TrainingName, tr.TrainerID,
       TrainerFirstName,TrainerLastName
FROM Training t
JOIN Trainer tr ON tr.TrainerID = tr.TrainerID
WHERE TrainerLastName IS NULL OR TrainerLastName = ''
ORDER BY t.TrainingID, tr.TrainerID;
```



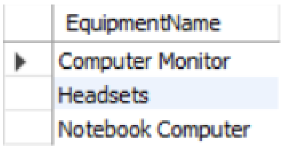
TrainingID	TrainingName	TrainerID	TrainerFirstName	TrainerLastName
1	COVID-19 Awareness and Protection Training	5	John	NULL
1	COVID-19 Awareness and Protection Training	14	vanessa	NULL
1	COVID-19 Awareness and Protection Training	15	Jordan	NULL
2	Code of Conduct Training	5	John	NULL
2	Code of Conduct Training	14	vanessa	NULL
2	Code of Conduct Training	15	Jordan	NULL
3	Safety Traning	5	John	NULL
3	Safety Traning	14	vanessa	NULL
3	Safety Traning	15	Jordan	NULL
4	Intro to Python	5	John	NULL
4	Intro to Python	14	vanessa	NULL
4	Intro to Python	15	Jordan	NULL
5	Machine Learning	5	John	NULL
5	Machine Learning	14	vanessa	NULL
5	Machine Learning	15	Jordan	NULL
6	Microsoft Certifications	5	John	NULL
6	Microsoft Certifications	14	vanessa	NULL
6	Microsoft Certifications	15	Jordan	NULL
7	Security and Privacy	5	John	NULL
7	Security and Privacy	14	vanessa	NULL
7	Security and Privacy	15	Jordan	NULL
8	Product Knowledge	5	John	NULL
8	Product Knowledge	14	vanessa	NULL
8	Product Knowledge	15	Jordan	NULL
9	Sales Skills	5	John	NULL
9	Sales Skills	14	vanessa	NULL
9	Sales Skills	15	Jordan	NULL
10	Employee Relations	5	John	NULL
10	Employee Relations	14	vanessa	NULL
10	Employee Relations	15	Jordan	NULL
11	Travel and Expense Management	5	John	NULL
11	Travel and Expense Management	14	vanessa	NULL
11	Travel and Expense Management	15	Jordan	NULL

The list of training given by the trainer who doesn’t have last name are as above.

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.



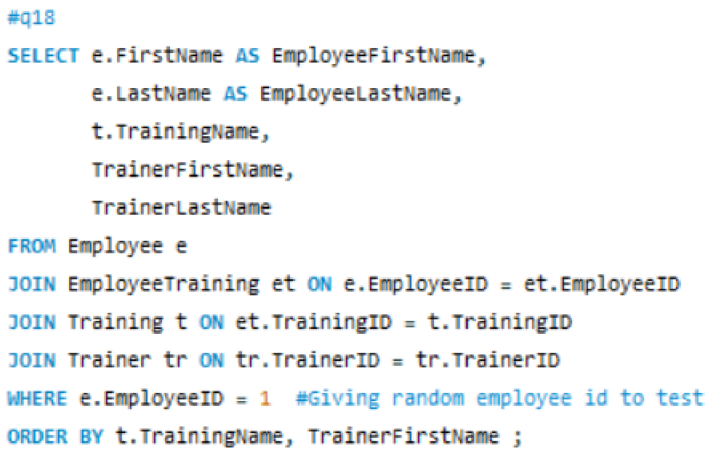
```
#q17
SELECT DISTINCT eq.EquipmentName
FROM Equipment eq
JOIN EmployeeEquipment ee ON eq.EquipmentID = ee.EquipmentID
JOIN Employee e ON ee.EmployeeID = e.EmployeeID
ORDER BY eq.EquipmentName;
```



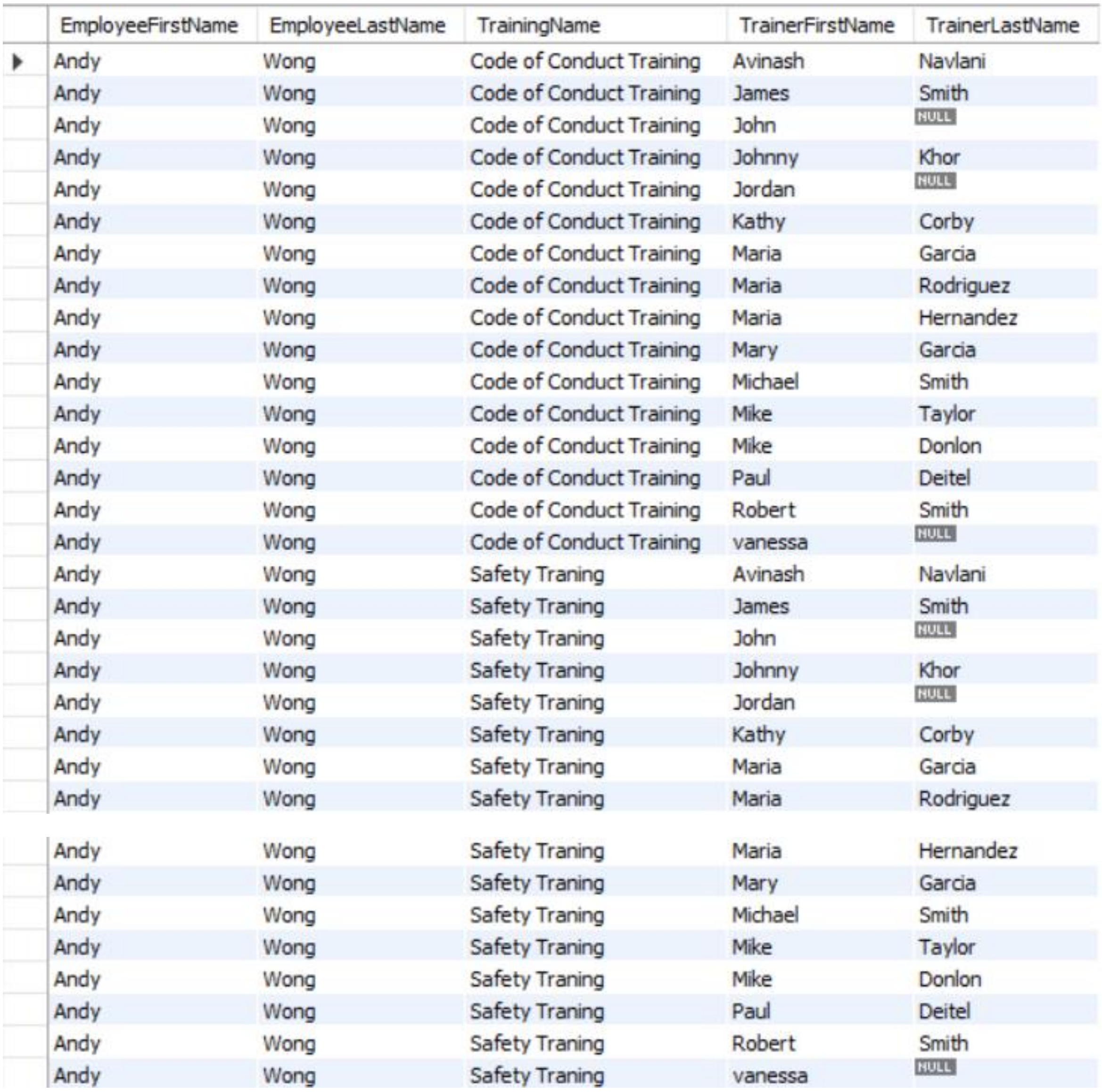
EquipmentName
Computer Monitor
Headsets
Notebook Computer

The list of equipment used by current employees in the organisation are as above.

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.



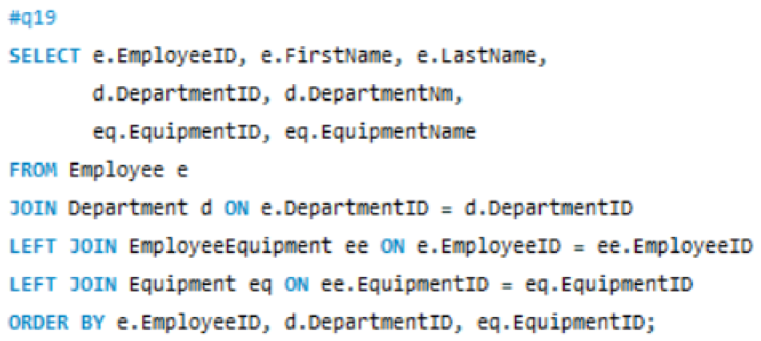
```
#q18
SELECT e.FirstName AS EmployeeFirstName,
       e.LastName AS EmployeeLastName,
       t.TrainingName,
       TrainerFirstName,
       TrainerLastName
FROM Employee e
JOIN EmployeeTraining et ON e.EmployeeID = et.EmployeeID
JOIN Training t ON et.TrainingID = t.TrainingID
JOIN Trainer tr ON tr.TrainerID = tr.TrainerID
WHERE e.EmployeeID = 1  #Giving random employee id to test
ORDER BY t.TrainingName, TrainerFirstName ;
```



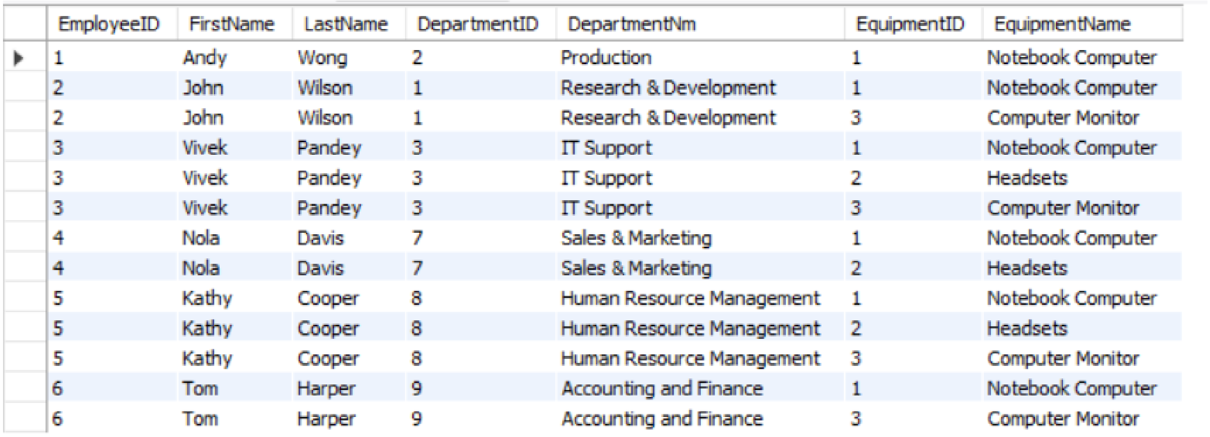
EmployeeFirstName	EmployeeLastName	TrainingName	TrainerFirstName	TrainerLastName
Andy	Wong	Code of Conduct Training	Avinash	Navlani
Andy	Wong	Code of Conduct Training	James	Smith
Andy	Wong	Code of Conduct Training	John	NULL
Andy	Wong	Code of Conduct Training	Johnny	Khor
Andy	Wong	Code of Conduct Training	Jordan	NULL
Andy	Wong	Code of Conduct Training	Kathy	Corby
Andy	Wong	Code of Conduct Training	Maria	Garcia
Andy	Wong	Code of Conduct Training	Maria	Rodriguez
Andy	Wong	Code of Conduct Training	Maria	Hernandez
Andy	Wong	Code of Conduct Training	Mary	Garcia
Andy	Wong	Code of Conduct Training	Michael	Smith
Andy	Wong	Code of Conduct Training	Mike	Taylor
Andy	Wong	Code of Conduct Training	Mike	Donlon
Andy	Wong	Code of Conduct Training	Paul	Deitel
Andy	Wong	Code of Conduct Training	Robert	Smith
Andy	Wong	Code of Conduct Training	vanessa	NULL
Andy	Wong	Safety Traning	Avinash	Navlani
Andy	Wong	Safety Traning	James	Smith
Andy	Wong	Safety Traning	John	NULL
Andy	Wong	Safety Traning	Johnny	Khor
Andy	Wong	Safety Traning	Jordan	NULL
Andy	Wong	Safety Traning	Kathy	Corby
Andy	Wong	Safety Traning	Maria	Garcia
Andy	Wong	Safety Traning	Maria	Rodriguez
Andy	Wong	Safety Traning	Maria	Hernandez
Andy	Wong	Safety Traning	Mary	Garcia
Andy	Wong	Safety Traning	Michael	Smith
Andy	Wong	Safety Traning	Mike	Taylor
Andy	Wong	Safety Traning	Mike	Donlon
Andy	Wong	Safety Traning	Paul	Deitel
Andy	Wong	Safety Traning	Robert	Smith
Andy	Wong	Safety Traning	vanessa	NULL

Here we are taking a random employee id as 1 and list of training and trainer with respect to it.

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.



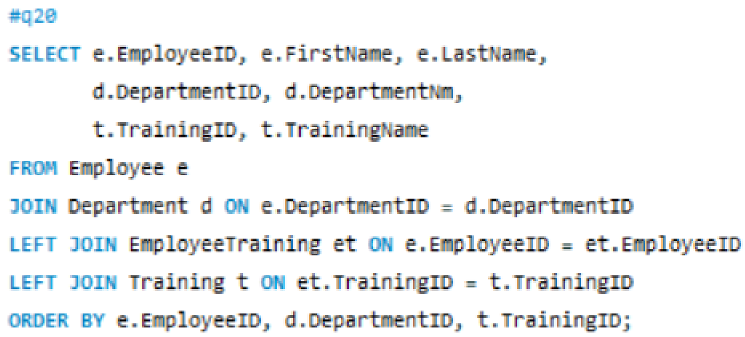
```
#q19
SELECT e.EmployeeID, e.FirstName, e.LastName,
       d.DepartmentID, d.DepartmentNm,
       eq.EquipmentID, eq.EquipmentName
FROM Employee e
JOIN Department d ON e.DepartmentID = d.DepartmentID
LEFT JOIN EmployeeEquipment ee ON e.EmployeeID = ee.EmployeeID
LEFT JOIN Equipment eq ON ee.EquipmentID = eq.EquipmentID
ORDER BY e.EmployeeID, d.DepartmentID, eq.EquipmentID;
```



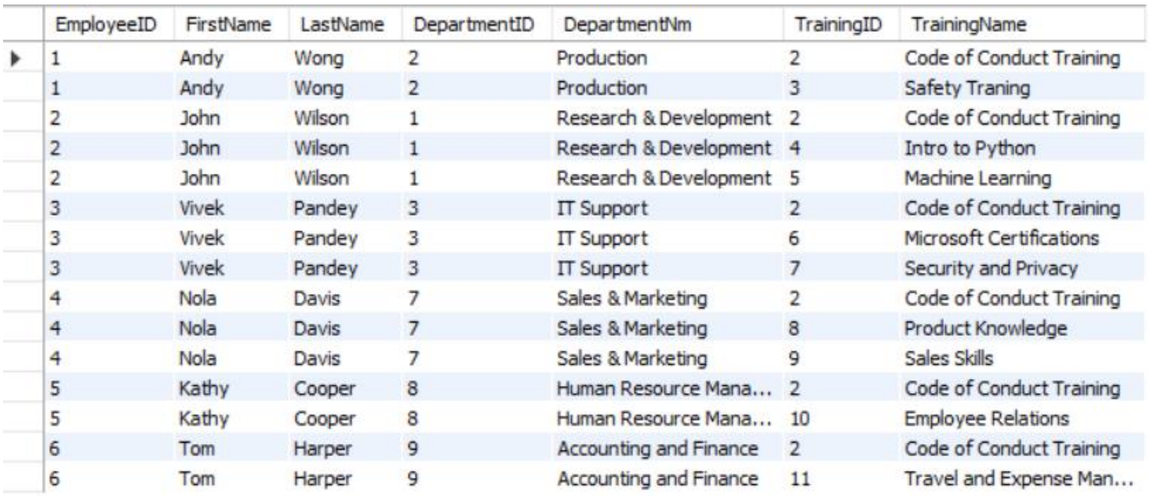
EmployeeID	FirstName	LastName	DepartmentID	DepartmentNm	EquipmentID	EquipmentName
1	Andy	Wong	2	Production	1	Notebook Computer
2	John	Wilson	1	Research & Development	1	Notebook Computer
2	John	Wilson	1	Research & Development	3	Computer Monitor
3	Vivek	Pandey	3	IT Support	1	Notebook Computer
3	Vivek	Pandey	3	IT Support	2	Headsets
3	Vivek	Pandey	3	IT Support	3	Computer Monitor
4	Nola	Davis	7	Sales & Marketing	1	Notebook Computer
4	Nola	Davis	7	Sales & Marketing	2	Headsets
5	Kathy	Cooper	8	Human Resource Management	1	Notebook Computer
5	Kathy	Cooper	8	Human Resource Management	2	Headsets
5	Kathy	Cooper	8	Human Resource Management	3	Computer Monitor
6	Tom	Harper	9	Accounting and Finance	1	Notebook Computer
6	Tom	Harper	9	Accounting and Finance	3	Computer Monitor

List of equipment used by all the employee with respect to their department are displayed as above.

20. 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.



```
#q20
SELECT e.EmployeeID, e.FirstName, e.LastName,
       d.DepartmentID, d.DepartmentNm,
       t.TrainingID, t.TrainingName
FROM Employee e
JOIN Department d ON e.DepartmentID = d.DepartmentID
LEFT JOIN EmployeeTraining et ON e.EmployeeID = et.EmployeeID
LEFT JOIN Training t ON et.TrainingID = t.TrainingID
ORDER BY e.EmployeeID, d.DepartmentID, t.TrainingID;
```



EmployeeID	FirstName	LastName	DepartmentID	DepartmentNm	TrainingID	TrainingName
1	Andy	Wong	2	Production	2	Code of Conduct Training
1	Andy	Wong	2	Production	3	Safety Traning
2	John	Wilson	1	Research & Development	2	Code of Conduct Training
2	John	Wilson	1	Research & Development	4	Intro to Python
2	John	Wilson	1	Research & Development	5	Machine Learning
3	Vivek	Pandey	3	IT Support	2	Code of Conduct Training
3	Vivek	Pandey	3	IT Support	6	Microsoft Certifications
3	Vivek	Pandey	3	IT Support	7	Security and Privacy
4	Nola	Davis	7	Sales & Marketing	2	Code of Conduct Training
4	Nola	Davis	7	Sales & Marketing	8	Product Knowledge
4	Nola	Davis	7	Sales & Marketing	9	Sales Skills
5	Kathy	Cooper	8	Human Resource Mana…	2	Code of Conduct Training
5	Kathy	Cooper	8	Human Resource Mana…	10	Employee Relations
6	Tom	Harper	9	Accounting and Finance	2	Code of Conduct Training
6	Tom	Harper	9	Accounting and Finance	11	Travel and Expense Man…

List of Training for all the employee with respect to the Department.