

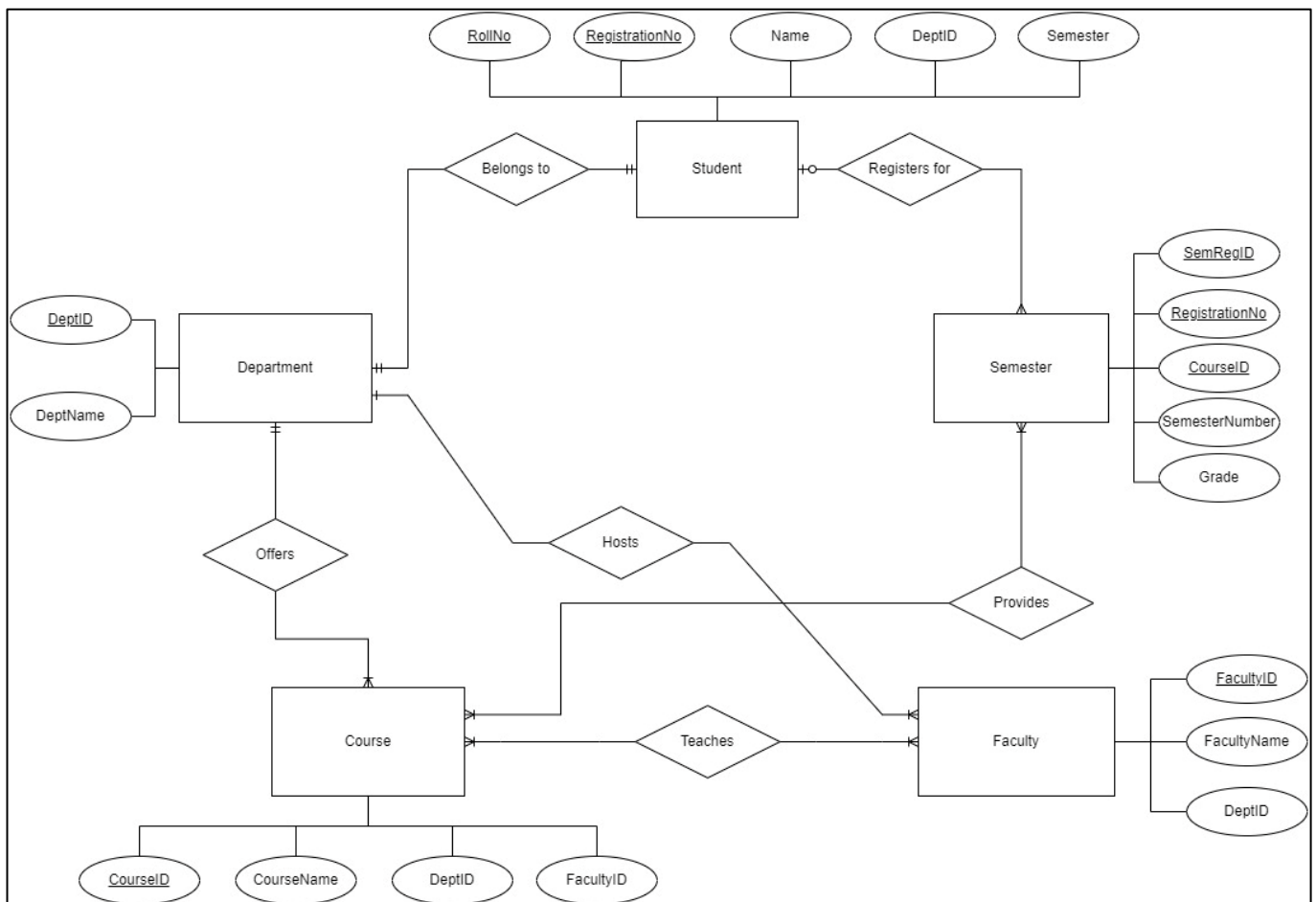
NATIONAL INSTITUTE OF TECHNOLOGY,  
DURGAPUR  
SOFTWARE ENGINEERING LABORATORY  
ASSIGNMENT 2

Name – Shubhabrata Ghosh  
Roll – 20CS8003  
Regn. – 20U10008  
Sec – X      Sub- CSS751  
Dept- CSE(2020-2024)

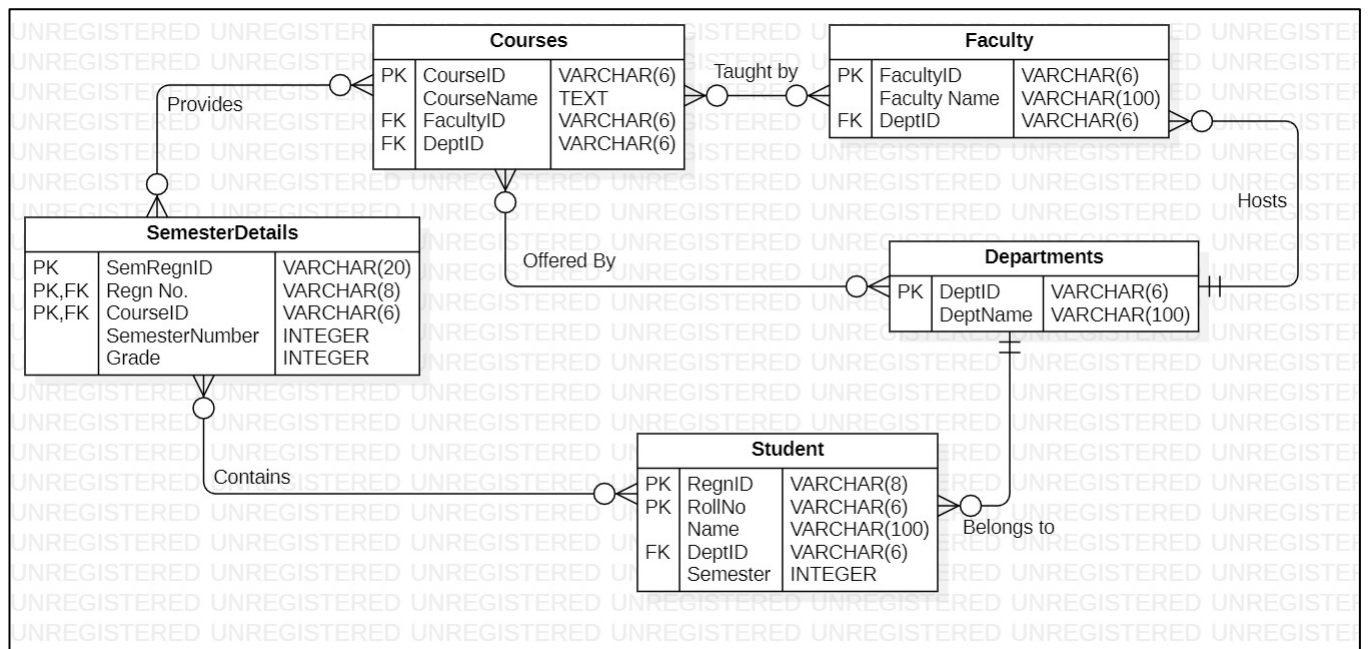
## Question:

Q1) NIT Durgapur wants to automate the Student Registration System and seek for intranet based solution. The Candidates need to register for some specific semester may go online with the system and provide the necessary data (roll no., name, department, course, semester and last semester grade point etc.) as input data. The system must able check the input data entered by student from the pre-existing student database and then automatically generate the registration slip for the specific semester as student copy. The system also will maintain a log of the students already registered and it will restrict any duplicate registration. System Administrator can view and generate report of the list of registered students and list of unregistered students. Draw Suitable ER/EER Diagram, Convert it into suitable Relational Model and Data Definition Language of SQL.

## ER:



## UML:



## DDL Code:

```
SET FOREIGN_KEY_CHECKS = 0;
DROP TABLE IF EXISTS `Student`;
DROP TABLE IF EXISTS `Departments`;
DROP TABLE IF EXISTS `SemesterDetails`;
DROP TABLE IF EXISTS `Courses`;
DROP TABLE IF EXISTS `Faculty`;
SET FOREIGN_KEY_CHECKS = 1;
```

```
CREATE TABLE `Student` (
  `RegnID` VARCHAR(8) NOT NULL,
  `RollNo` VARCHAR(6) NOT NULL,
  `Name` VARCHAR(100) NOT NULL,
  `DeptID` VARCHAR(6) NOT NULL,
  `Semester` INTEGER NOT NULL,
  PRIMARY KEY (`RegnID`, `RollNo`)
);
```

```
CREATE TABLE `Departments` (
  `DeptID` VARCHAR(6) NOT NULL,
  `DeptName` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`DeptID`)
);
```

```
CREATE TABLE `SemesterDetails` (
  `SemRegnID` VARCHAR(20) NOT NULL,
  `Regn No.` VARCHAR(8) NOT NULL,
```

```
`CourseID` VARCHAR(6) NOT NULL,  
`SemesterNumber` INTEGER NOT NULL,  
`Grade` INTEGER NOT NULL,  
PRIMARY KEY (`SemRegnID`, `Regn No.`, `CourseID`)  
);
```

```
CREATE TABLE `Courses` (  
  `CourseID` VARCHAR(6) NOT NULL,  
  `CourseName` TEXT NOT NULL,  
  `FacultyID` VARCHAR(6) NOT NULL,  
  `DeptID` VARCHAR(6) NOT NULL,  
  PRIMARY KEY (`CourseID`)  
);
```

```
CREATE TABLE `Faculty` (  
  `FacultyID` VARCHAR(6) NOT NULL,  
  `Faculty Name` VARCHAR(100) NOT NULL,  
  `DeptID` VARCHAR(6) NOT NULL,  
  PRIMARY KEY (`FacultyID`)  
);
```

```
ALTER TABLE `Student` ADD FOREIGN KEY (`DeptID`) REFERENCES  
`Departments`(`DeptID`);  
ALTER TABLE `SemesterDetails` ADD FOREIGN KEY (`Regn No.`) REFERENCES  
`Student`(`RegnID`);  
ALTER TABLE `SemesterDetails` ADD FOREIGN KEY (`CourseID`) REFERENCES  
`Courses`(`CourseID`);  
ALTER TABLE `Courses` ADD FOREIGN KEY (`FacultyID`) REFERENCES  
`Faculty`(`FacultyID`);  
ALTER TABLE `Courses` ADD FOREIGN KEY (`DeptID`) REFERENCES  
`Departments`(`DeptID`);  
ALTER TABLE `Faculty` ADD FOREIGN KEY (`DeptID`) REFERENCES  
`Departments`(`DeptID`);
```

## Output:

UnivDDL x Administration - Server Status

Limit to 1000 rows

```
1 • SET FOREIGN_KEY_CHECKS = 0;
2 • DROP TABLE IF EXISTS `Student`;
3 • DROP TABLE IF EXISTS `Departments`;
4 • DROP TABLE IF EXISTS `SemesterDetails`;
5 • DROP TABLE IF EXISTS `Courses`;
6 • DROP TABLE IF EXISTS `Faculty`;
7 • SET FOREIGN_KEY_CHECKS = 1;
8
9 • CREATE TABLE `Student` (
10     `RegnID` VARCHAR(8) NOT NULL,
11     `RollNo` VARCHAR(6) NOT NULL,
12     `Name` VARCHAR(100) NOT NULL,
13     `DeptID` VARCHAR(6) NOT NULL,
```

Output

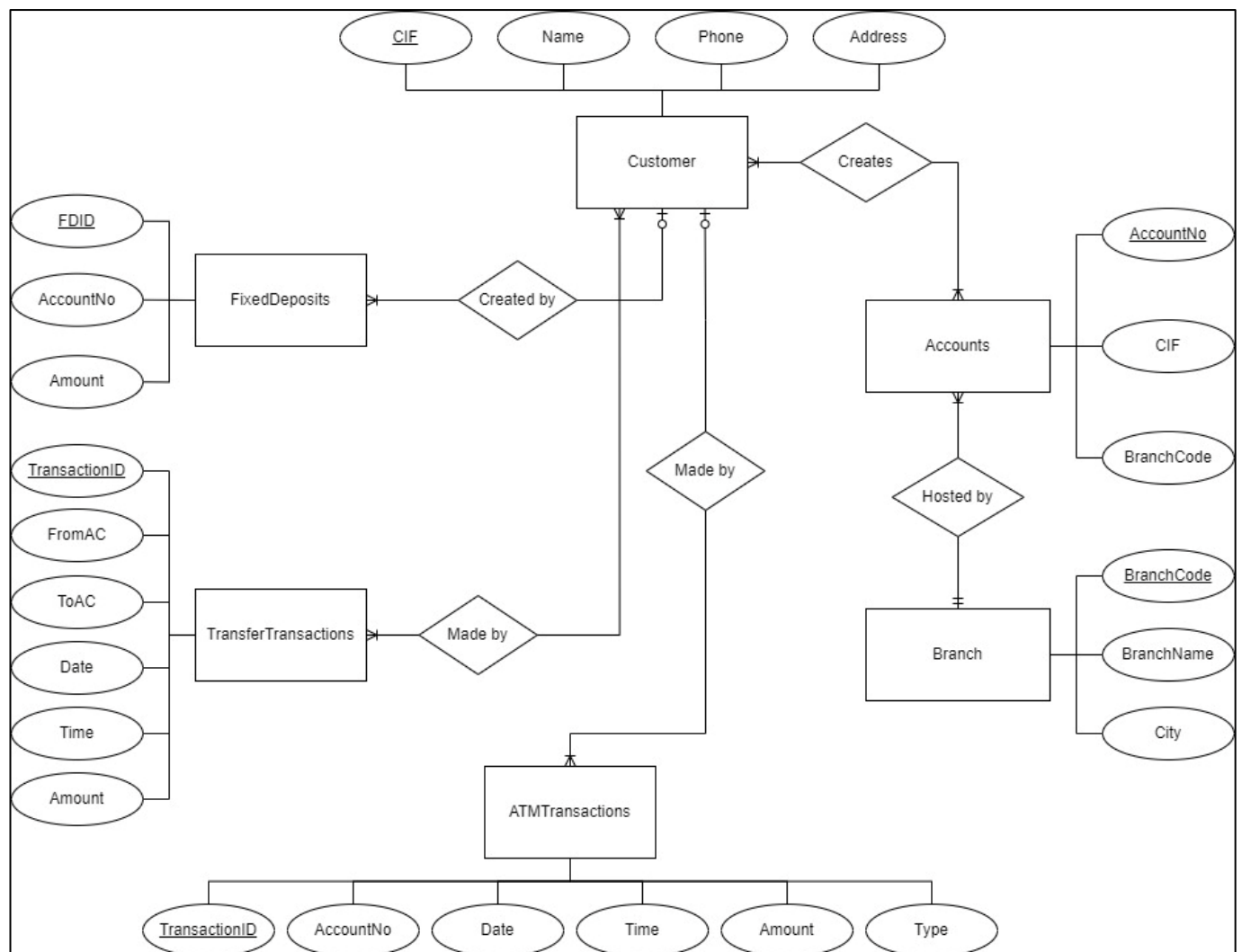
Action Output

#	Time	Action	Message
✓ 9	23:14:42	CREATE TABLE `Departments` ( `DeptID` VARCHAR(6) NOT NULL, `DeptNam...	0 row(s) affected
✓ 10	23:14:42	CREATE TABLE `SemesterDetails` ( `SemRegnID` VARCHAR(20) NOT NULL, ...	0 row(s) affected
✓ 11	23:14:42	CREATE TABLE `Courses` ( `CourseID` VARCHAR(6) NOT NULL, `CourseNam...	0 row(s) affected
✓ 12	23:14:42	CREATE TABLE `Faculty` ( `FacultyID` VARCHAR(6) NOT NULL, `Faculty Nam...	0 row(s) affected
✓ 13	23:14:42	ALTER TABLE `Student` ADD FOREIGN KEY (`DeptID`) REFERENCES `Departme...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 14	23:14:42	ALTER TABLE `SemesterDetails` ADD FOREIGN KEY (`Regn No.`) REFERENCES `...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 15	23:14:42	ALTER TABLE `SemesterDetails` ADD FOREIGN KEY (`CourseID`) REFERENCES `...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 16	23:14:42	ALTER TABLE `Courses` ADD FOREIGN KEY (`FacultyID`) REFERENCES `Faculty`...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 17	23:14:42	ALTER TABLE `Courses` ADD FOREIGN KEY (`DeptID`) REFERENCES `Departme...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 18	23:14:42	ALTER TABLE `Faculty` ADD FOREIGN KEY (`DeptID`) REFERENCES `Departmen...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

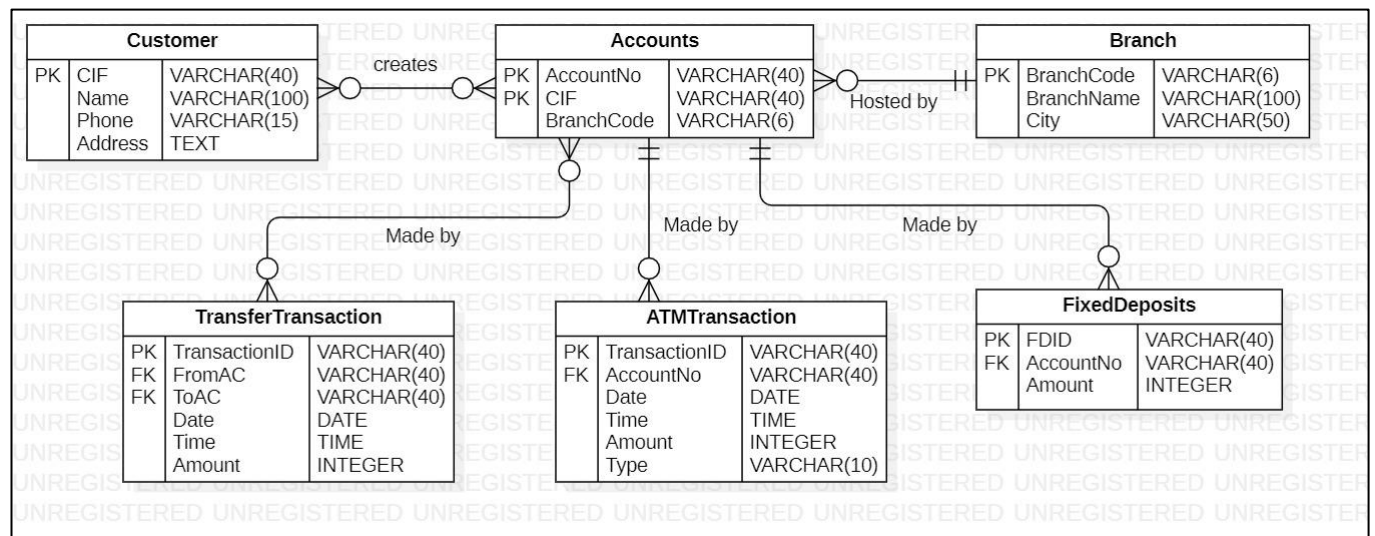
## Question

Q2) XYZ Bank is a facilitator for public banking system with many branches in different region. They would like to automate the banking Processing System. The bank facilitates deposit, withdrawal and fixed deposit system from its saving accounts. Customer may have joint account as well as many account in any branch of the bank. Withdrawal or deposit can be done from any branch of the bank. The bank also has facility of ATM. On any transaction, the system will able to maintain the transaction record in some log. Using which, system administrator can generate report for account wise transaction per day. Draw Suitable ER/EER Diagram, Convert it into suitable Relational Model and Data Definition Language of SQL.

ER:



## UML:



## DDL Code:

```

SET FOREIGN_KEY_CHECKS = 0;
DROP TABLE IF EXISTS `Customer`;
DROP TABLE IF EXISTS `Accounts`;
DROP TABLE IF EXISTS `TransferTransaction`;
DROP TABLE IF EXISTS `Branch`;
DROP TABLE IF EXISTS `ATMTransaction`;
DROP TABLE IF EXISTS `FixedDeposits`;
SET FOREIGN_KEY_CHECKS = 1;

```

```

CREATE TABLE `Customer` (
  `CIF` VARCHAR(40) NOT NULL,
  `Name` VARCHAR(100) NOT NULL,
  `Column1` INTEGER NOT NULL,
  `Phone` VARCHAR(15) NOT NULL,
  `Address` TEXT NOT NULL,
  PRIMARY KEY (`CIF`)
);

```

```

CREATE TABLE `Accounts` (
  `AccountNo` VARCHAR(40) NOT NULL,
  `CIF` VARCHAR(40) NOT NULL,
  `BranchCode` VARCHAR(6) NOT NULL,
  PRIMARY KEY (`AccountNo`, `CIF`)
);

```

```

CREATE TABLE `TransferTransaction` (
  `TransactionID` VARCHAR(40) NOT NULL,
  `FromAC` VARCHAR(40) NOT NULL,
  `ToAC` VARCHAR(40) NOT NULL,
  `Date` DATE NOT NULL,

```



```
    `Time` TIME NOT NULL,  
    `Amount` INTEGER NOT NULL,  
    PRIMARY KEY (`TransactionID`)  
);
```

```
CREATE TABLE `Branch` (  
    `BranchCode` VARCHAR(6) NOT NULL,  
    `BranchName` VARCHAR(100) NOT NULL,  
    `City` VARCHAR(50) NOT NULL,  
    PRIMARY KEY (`BranchCode`)  
);
```

```
CREATE TABLE `ATMTransaction` (  
    `TransactionID` VARCHAR(40) NOT NULL,  
    `AccountNo` VARCHAR(40) NOT NULL,  
    `Date` DATE NOT NULL,  
    `Time` TIME NOT NULL,  
    `Amount` INTEGER NOT NULL,  
    `Type` VARCHAR(10) NOT NULL,  
    PRIMARY KEY (`TransactionID`)  
);
```

```
CREATE TABLE `FixedDeposits` (  
    `FDID` VARCHAR(40) NOT NULL,  
    `AccountNo` VARCHAR(40) NOT NULL,  
    `Amount` INTEGER NOT NULL,  
    PRIMARY KEY (`FDID`)  
);
```

```
ALTER TABLE `TransferTransaction` ADD FOREIGN KEY (`FromAC`) REFERENCES  
`Accounts`(`AccountNo`);  
ALTER TABLE `TransferTransaction` ADD FOREIGN KEY (`ToAC`) REFERENCES  
`Accounts`(`AccountNo`);  
ALTER TABLE `ATMTransaction` ADD FOREIGN KEY (`AccountNo`) REFERENCES  
`Accounts`(`AccountNo`);  
ALTER TABLE `FixedDeposits` ADD FOREIGN KEY (`AccountNo`) REFERENCES  
`Accounts`(`AccountNo`);
```



## Output:

BankDDL x

Limit to 1000 rows

```
1 • SET FOREIGN_KEY_CHECKS = 0;
2 • DROP TABLE IF EXISTS `Customer`;
3 • DROP TABLE IF EXISTS `Accounts`;
4 • DROP TABLE IF EXISTS `TransferTransaction`;
5 • DROP TABLE IF EXISTS `Branch`;
6 • DROP TABLE IF EXISTS `ATMTransaction`;
7 • DROP TABLE IF EXISTS `FixedDeposits`;
8 • SET FOREIGN_KEY_CHECKS = 1;
9
10 • CREATE TABLE `Customer` (
11     `CIF` VARCHAR(40) NOT NULL,
12     `Name` VARCHAR(100) NOT NULL,
13     `Column1` INTEGER NOT NULL,
```

Output

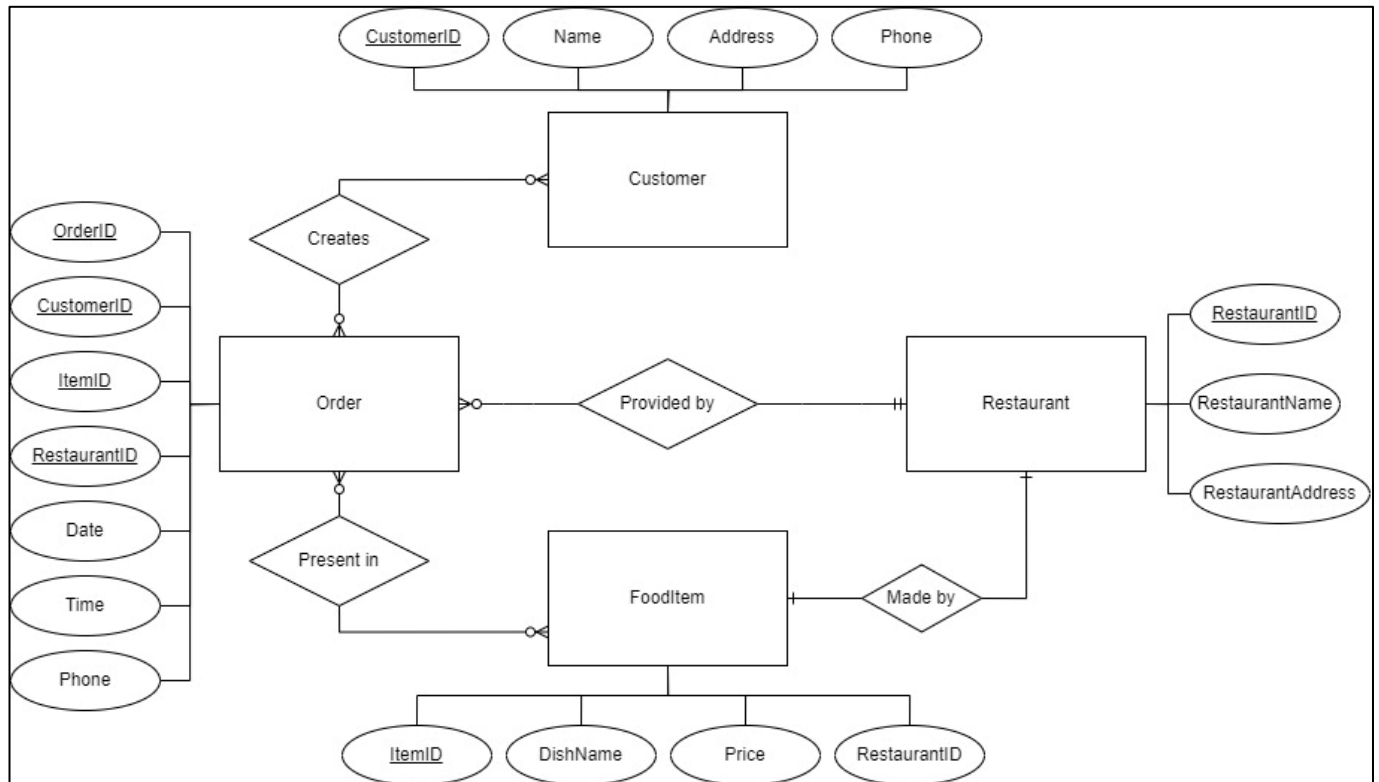
Action Output

#	Time	Action	Message
✓ 9	23:17:38	CREATE TABLE `Customer` ( `CIF` VARCHAR(40) NOT NULL, `Name` VARCH...	0 row(s) affected
✓ 10	23:17:38	CREATE TABLE `Accounts` ( `AccountNo` VARCHAR(40) NOT NULL, `CIF` V...	0 row(s) affected
✓ 11	23:17:38	CREATE TABLE `TransferTransaction` ( `TransactionID` VARCHAR(40) NOT NU...	0 row(s) affected
✓ 12	23:17:39	CREATE TABLE `Branch` ( `BranchCode` VARCHAR(6) NOT NULL, `BranchN...	0 row(s) affected
✓ 13	23:17:39	CREATE TABLE `ATMTransaction` ( `TransactionID` VARCHAR(40) NOT NULL, ...	0 row(s) affected
✓ 14	23:17:39	CREATE TABLE `FixedDeposits` ( `FDID` VARCHAR(40) NOT NULL, `Account...	0 row(s) affected
✓ 15	23:17:39	ALTER TABLE `TransferTransaction` ADD FOREIGN KEY (`FromAC`) REFERENCE...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 16	23:17:39	ALTER TABLE `TransferTransaction` ADD FOREIGN KEY (`ToAC`) REFERENCES `...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 17	23:17:39	ALTER TABLE `ATMTransaction` ADD FOREIGN KEY (`AccountNo`) REFERENCE...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 18	23:17:39	ALTER TABLE `FixedDeposits` ADD FOREIGN KEY (`AccountNo`) REFERENCES `...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

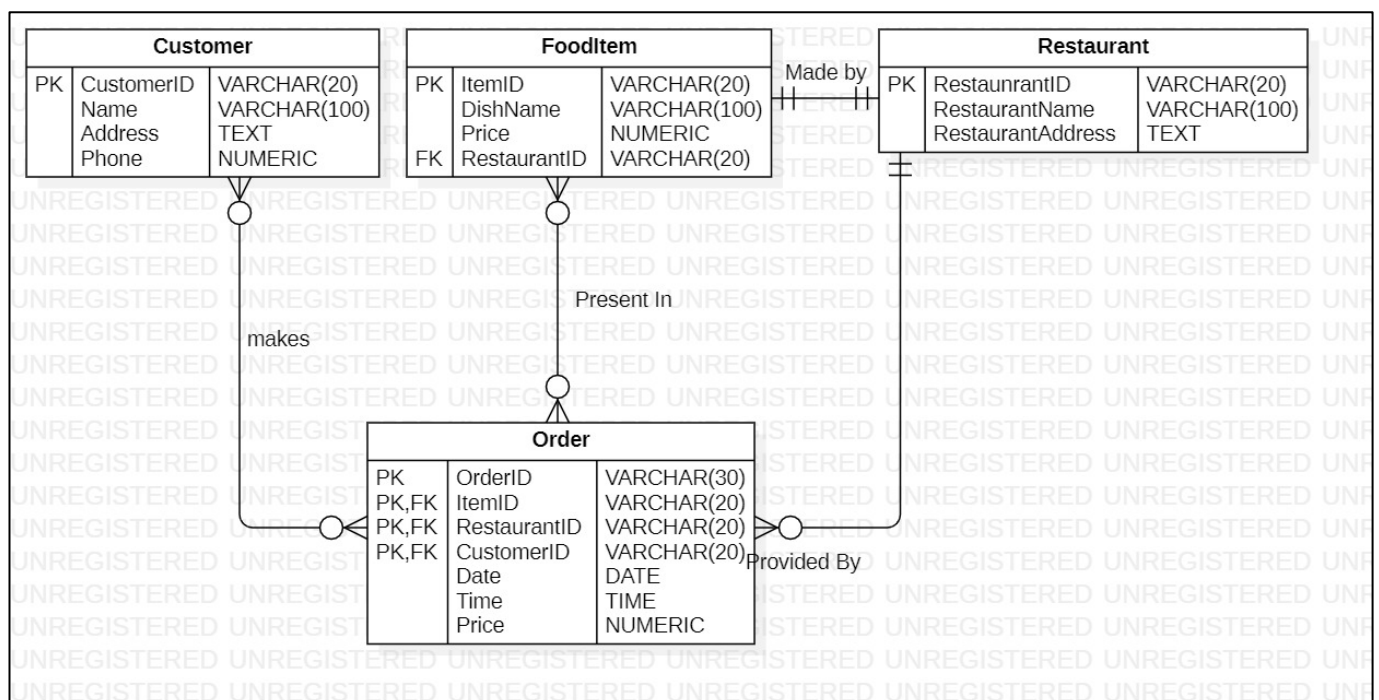
## Question:

Case Study: Online Food Ordering System A restaurant chain wants to develop an online platform that allows customers to browse their menu, place orders for food items, and have the food delivered to their specified location. The system should also enable the restaurant staff to manage orders, update the menu, and track delivery status. Draw Suitable ER Diagram, Convert it into suitable Relational Model.

## ER:



## UML:



## DDL Code:

```
SET FOREIGN_KEY_CHECKS = 0;
DROP TABLE IF EXISTS `Customer`;
DROP TABLE IF EXISTS `Restaurant`;
DROP TABLE IF EXISTS `Menu`;
DROP TABLE IF EXISTS `FoodItem`;
DROP TABLE IF EXISTS `Order`;
SET FOREIGN_KEY_CHECKS = 1;

CREATE TABLE `Customer` (
  `CustomerID` VARCHAR(20) NOT NULL,
  `Name` VARCHAR(100) NOT NULL,
  `Address` TEXT NOT NULL,
  `Phone` NUMERIC NOT NULL,
  PRIMARY KEY (`CustomerID`)
);

CREATE TABLE `Restaurant` (
  `RestaunrantID` VARCHAR(20) NOT NULL,
  `RestaurantName` VARCHAR(100) NOT NULL,
  `RestaurantAddress` TEXT NOT NULL,
  `Column1` INTEGER NOT NULL,
  PRIMARY KEY (`RestaunrantID`)
);

CREATE TABLE `FoodItem` (
  `ItemID` VARCHAR(20) NOT NULL,
  `DishName` VARCHAR(100) NOT NULL,
  `Price` NUMERIC NOT NULL,
  `RestaurantID` VARCHAR(20) NOT NULL,
  `Column1` INTEGER NOT NULL,
  `Column2` INTEGER NOT NULL,
  PRIMARY KEY (`ItemID`)
);

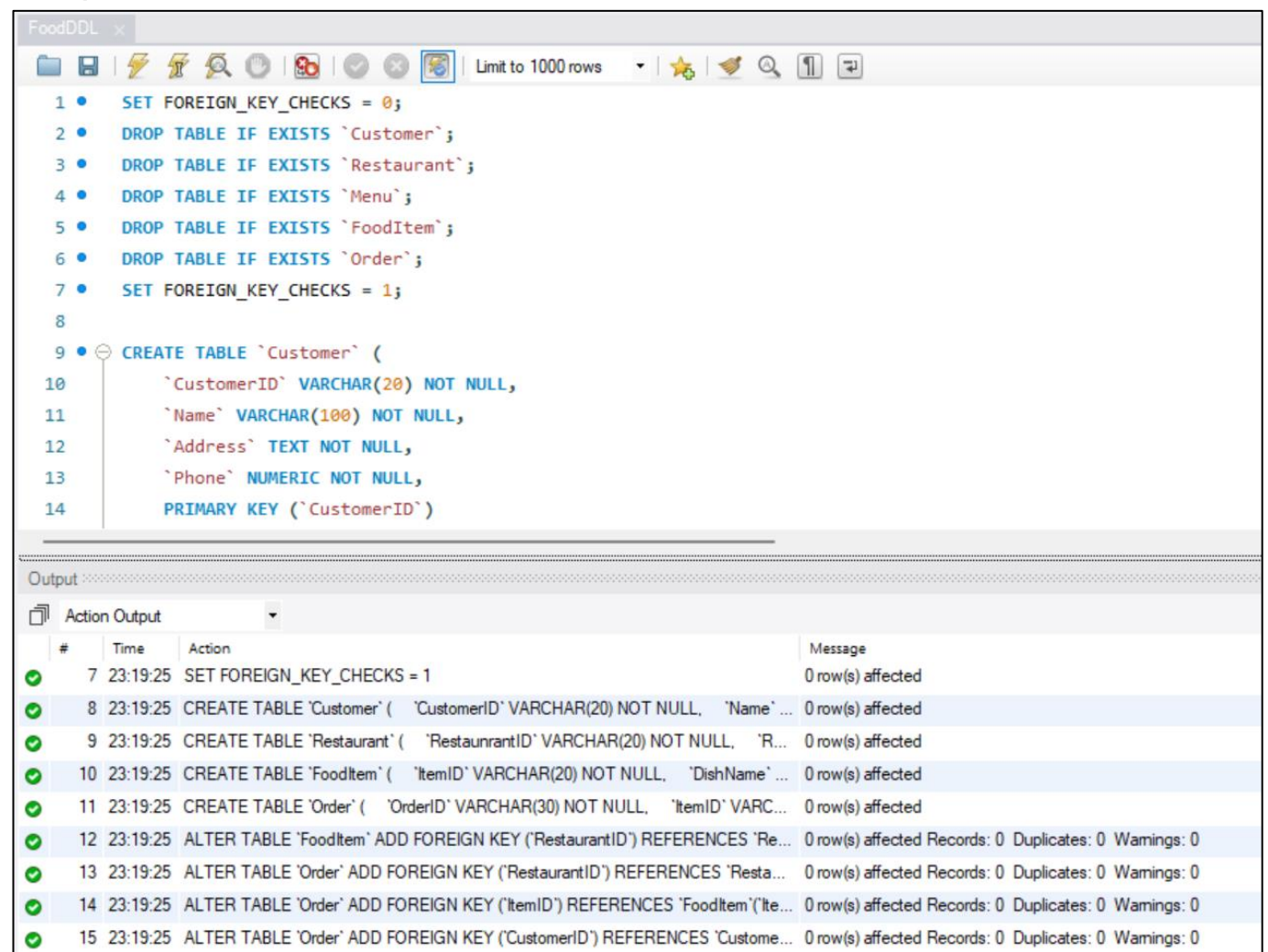
CREATE TABLE `Order` (
  `OrderID` VARCHAR(30) NOT NULL,
  `ItemID` VARCHAR(20) NOT NULL,
  `RestaurantID` VARCHAR(20) NOT NULL,
  `CustomerID` VARCHAR(20) NOT NULL,
  `Date` DATE NOT NULL,
  `Time` TIME NOT NULL,
  `Price` NUMERIC NOT NULL,
  PRIMARY KEY (`OrderID`, `ItemID`, `RestaurantID`, `CustomerID`)
);
```

```

ALTER TABLE `FoodItem` ADD FOREIGN KEY (`RestaurantID`) REFERENCES
`Restaurant`(`RestaunrantID`);
ALTER TABLE `Order` ADD FOREIGN KEY (`RestaurantID`) REFERENCES
`Restaurant`(`RestaunrantID`);
ALTER TABLE `Order` ADD FOREIGN KEY (`ItemID`) REFERENCES
`FoodItem`(`ItemID`);
ALTER TABLE `Order` ADD FOREIGN KEY (`CustomerID`) REFERENCES
`Customer`(`CustomerID`);

```

## Output:



The screenshot displays a SQL script in an IDE window titled 'FoodDDL'. The script performs the following actions:

1. SET FOREIGN\_KEY\_CHECKS = 0;
2. DROP TABLE IF EXISTS `Customer`;
3. DROP TABLE IF EXISTS `Restaurant`;
4. DROP TABLE IF EXISTS `Menu`;
5. DROP TABLE IF EXISTS `FoodItem`;
6. DROP TABLE IF EXISTS `Order`;
7. SET FOREIGN\_KEY\_CHECKS = 1;
8. (Empty line)
9. CREATE TABLE `Customer` (
  - 10. `CustomerID` VARCHAR(20) NOT NULL,
  - 11. `Name` VARCHAR(100) NOT NULL,
  - 12. `Address` TEXT NOT NULL,
  - 13. `Phone` NUMERIC NOT NULL,
  - 14. PRIMARY KEY (`CustomerID`)

The output pane at the bottom shows the execution results for steps 7 through 15:

#	Time	Action	Message
7	23:19:25	SET FOREIGN_KEY_CHECKS = 1	0 row(s) affected
8	23:19:25	CREATE TABLE `Customer` ( `CustomerID` VARCHAR(20) NOT NULL, `Name` ...	0 row(s) affected
9	23:19:25	CREATE TABLE `Restaurant` ( `RestaunrantID` VARCHAR(20) NOT NULL, `R...	0 row(s) affected
10	23:19:25	CREATE TABLE `FoodItem` ( `ItemID` VARCHAR(20) NOT NULL, `DishName` ...	0 row(s) affected
11	23:19:25	CREATE TABLE `Order` ( `OrderID` VARCHAR(30) NOT NULL, `ItemID` VARC...	0 row(s) affected
12	23:19:25	ALTER TABLE `FoodItem` ADD FOREIGN KEY (`RestaurantID`) REFERENCES `Re...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
13	23:19:25	ALTER TABLE `Order` ADD FOREIGN KEY (`RestaurantID`) REFERENCES `Resta...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
14	23:19:25	ALTER TABLE `Order` ADD FOREIGN KEY (`ItemID`) REFERENCES `FoodItem`(`ite...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
15	23:19:25	ALTER TABLE `Order` ADD FOREIGN KEY (`CustomerID`) REFERENCES `Custome...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0