Domas Budrys Assignment 7 Script

Create Table: CREATE TABLE Employee (EMP ID **INT** NOT NULL, VARCHAR(255) NOT NULL, EMP FNAME EMP LNAME NOT NULL, VARCHAR(255) EMP_STREET VARCHAR(255) NULL, EMP CITY NULL, VARCHAR(255) EMP STATE VARCHAR(255) NULL, EMP ZIP VARCHAR(255) NULL, NOT NULL EMP START DATEDATE CONSTRAINT Employee PK PRIMARY KEY (EMP ID)); CREATE TABLE Area (AREA ID **INT** NOT NULL, AREA NAME VARCHAR(255) NOT NULL, AREA SUPERVISOR EMPLOYEE ID INT NOT NULL, CONSTRAINT Area PK PRIMARY KEY (AREA ID)); CREATE TABLE [Table](TABLE ID **INT** NOT NULL, AREA ID **INT** NOT NULL, TABLE_SEATS INT NOT NULL, CONSTRAINT Table PK PRIMARY KEY (TABLE ID), CONSTRAINT Table FK1 FOREIGN KEY (AREA ID) REFERENCES Area(AREA ID)

);

```
CREATE TABLE Customer (
CUST ID
                      INT
                                            NOT NULL,
CUST LAST NAME
                      VARCHAR(255)
                                      NOT NULL,
CUST NUMBER OF GUESTS
                                 INT
                                                       NOT NULL,
CONSTRAINT Customer PK
                                 PRIMARY KEY (CUST ID)
);
CREATE TABLE Assignment (
EMP ID
                      INT
                                            NOT NULL,
TABLE ID
                INT
                                      NOT NULL,
CONSTRAINT Assignment PK
                                 PRIMARY KEY (EMP ID, TABLE ID),
CONSTRAINT Assignment FK1
                                 FOREIGN KEY (EMP ID)
                                       REFERENCES Employee(EMP ID),
CONSTRAINT Assignment FK2
                                 FOREIGN KEY (TABLE ID)
                                       REFERENCES [TABLE](TABLE ID)
);
CREATE TABLE Seating (
CUST_ID
                      INT
                                            NOT NULL,
TABLE ID
                INT
                                      NOT NULL,
SEATING COST
                DECIMAL(10,2)
                                 NOT NULL,
SEATING DATE
                DATE
                                 NOT NULL,
SEATING TIP
                                      NOT NULL,
                      DECIMAL(10,2)
CONSTRAINT Seating_PK
                           PRIMARY KEY (CUST ID, TABLE ID),
CONSTRAINT Seating FK1
                           FOREIGN KEY (CUST ID)
                                      REFERENCES Customer(CUST ID),
CONSTRAINT Seating FK2
                           FOREIGN KEY (TABLE ID)
                                       REFERENCES [TABLE](TABLE ID)
);
```

```
Question 1:
```

```
CREATE PROCEDURE Customer_Waited @waited VARCHAR(100)
AS
IF NOT EXISTS(
SELECT C.CUST ID, C.CUST LAST NAME, C.CUST NUMBER OF GUESTS
FROM Customer AS C
JOIN Seating AS S
ON C.CUST ID = S.CUST ID
JOIN Assignment AS A
ON S.TABLE_ID = A.TABLE_ID
JOIN Employee AS E
ON A.EMP ID = E.EMP ID
WHERE E.EMP ID = @waited
PRINT 'Employee ID does not exists'
ELSE IF NOT EXISTS (
SELECT C.CUST_ID, C.CUST_LAST_NAME, C.CUST_NUMBER_OF_GUESTS
FROM Customer AS C
JOIN Seating AS S
ON C.CUST ID = S.CUST ID
JOIN Assignment AS A
ON S.TABLE ID = A.TABLE ID
WHERE A.EMP ID = @waited
```

PRINT 'Have not waited any tables'

ELSE

SELECT C.CUST_ID, C.CUST_LAST_NAME, C.CUST_NUMBER_OF_GUESTS FROM Customer AS C
JOIN Seating AS S
ON C.CUST_ID = S.CUST_ID

JOIN Assignment AS A
ON S.TABLE_ID = A.TABLE_ID

WHERE A.EMP_ID = @waited

Exec Customer_Waited 1

Question 2:

Exec Seat_Number 1

```
CREATE PROCEDURE Seat_Number @number INT AS

IF EXISTS (
SELECT *
FROM [Table] AS T

WHERE T.TABLE_SEATS < @number
)

PRINT 'There is no table with that many seats'

ELSE

SELECT T.TABLE_ID, A.AREA_ID, A.AREA_NAME
FROM [Table] AS T
JOIN Area AS A
ON T.AREA_ID = A.AREA_ID

WHERE T.TABLE_SEATS >= @number
```

Question 3:

```
CREATE PROCEDURE Cust Emp @Cust INT, @Emp INT
AS
IF NOT EXISTS (
SELECT S.SEATING DATE
FROM Seating AS S
JOIN Customer AS C
ON C.CUST ID = S.CUST ID
JOIN Assignment AS A
ON S.TABLE ID = A.TABLE ID
JOIN Employee AS E
ON A.EMP ID = E.EMP ID
WHERE C.CUST ID = @Cust
AND
E.EMP ID = @Emp
PRINT 'Employee have not resverd this customer yet'
ELSE
SELECT S.SEATING DATE
FROM Seating AS S
JOIN Customer AS C
ON C.CUST_ID = S.CUST_ID
JOIN Assignment AS A
ON S.TABLE ID = A.TABLE ID
JOIN Employee AS E
ON A.EMP ID = E.EMP ID
WHERE C.CUST ID = @Cust
AND
```

 $E.EMP_ID = @Emp$

Trigger 1:

```
CREATE TRIGGER Seating_Table_Error ON Seating FOR INSERT, UPDATE AS
```

```
IF EXISTS (
SELECT *
FROM Seating AS S
JOIN [Table] AS T
On S.TABLE_ID = T.TABLE_ID

WHERE T.TABLE_ID < @seat_num
)

BEGIN
ROLLBACK
PRINT 'Cannot be assigned'
END
GO
```

Trigger 2:

```
CREATE TRIGGER Max_Guests ON Customer FOR INSERT AS
```

```
DECLARE @guest_num INT

IF EXISTS (
SELECT *
FROM Customer AS C
JOIN Seating AS S
On S.CUST_ID = C.CUST_ID

WHERE C.CUST_NUMBER_OF_GUESTS > @guest_num
)

BEGIN
ROLLBACK
PRINT 'Cannot be inserted, table does not have enough seats'
END
GO
```