Paula Hanna
Rafael Bechara

Homework 5

1)
-An attribute-based constraint is a constraint applied on a certain attribute to check its validity when inserting or updating tables, example:

```
CREAT TABLE Book(

PhysicalBook_ID INT NOT NULL CHECK(PhysicalBook_ID > 0),

);
```

-A tuple-based constraint is constraint is similar to attribute-based except that it is applied on tuples, I.e., it checks the validity of multiple attributes, example:

```
CREATE TABLE Borrowed_Book(

Date_of_Borrowing DATE NOT NULL,

Expiring_Date DATE NOT NULL,

CHECK(Date_of_Borrowing = CURRENT_DATE OR Expiring_Date >
Date_of_Borrowing )

);
```

2)

-Attribute-based check constraint can solve the problem of inserting invalid data, for example if you have an attribute-based check such as (Age INT (CHECK Age > 0)) will assure that there is no invalid Age data such age negative age will be inserted in the relation or any wrong updates can be done either.

-Tuple-based constraint can solve more complicated problems where the validation of data is based on tuples (multiple attributes), for example if I want a whole tuple to meet a certain criterion in order to get inserted in the table,

Example:

```
CREAT TABLE Employee(
        Employee_ID varchar(225) PRIMARY KEY,
        Age INT NOT NULL,
        JobType varchar(225) NOT NULL,
        CHECK (Age > 18 or JobType = ‘Talent’)
);
```

For that example, using a tuple-based check will assure that all the under 18 employees has a job title as Talent. Solved the problem by checking either of the two attributes Age or JobType.

-Triggers can solve problems that attribute-based and tuple-based constraint cannot, for example if I want to do a certain action only when updating existing data. Flor example:

```
 CREAT TRIGGER employeeSalaries(
        AFTER UPDATE OF Salary ON Employee
        REFRENCEING
        OLD ROW AS OldTuple
        NEW ROW AS NewTuple
        FOR EACH ROW
        WHEN (OldTuple.Salary > NewTuple.Salary)
        UPDATE
        SET Salary = OldTuple.Salary
        WHERE Employee_ID = NewTuple. HERE Employee_ID
);
```

Here the trigger will assure that the salaries of the employees cannot deacreas.