Worksheet : 1.4

Student Name: Garv Khurana **UID:** 21BCS6615
Branch: CSE AIML **Section/Group AIML-9** "A"
Semester: 3rd
Subject Name: DBMS **Subject Code: 21CSH-243**

1. Aim/Overview of the practical:

- To manipulate restrictions on the table.

2. Concept:

Constraints in SQL:

- SQL constraints are used to specify rules for the data in a table.
  Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

Commonly used constraints in SQL

- **NOT NULL:** Ensures that a column cannot have a NULL value
- **UNIQUE**: Ensures that all values in a column are different
- **PRIMARY KEY:** A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
- **FOREIGN KEY**: Prevents actions that would destroy links between tables
- **CHECK**: Ensures that the values in a column satisfies a specific condition
- **DEFAULT**: Sets a default value for a column if no value is specified
- **CREATE INDEX**: Used to create and retrieve data from the database very quickly

3. Code and Output

- **Creating a persons tabel with person id as primary key.**

```
CREATE TABLE Persons (
 ID INTEGER NOT NULL,
 LastName TEXT NOT NULL,
 FirstName TEXT,
 Age INTEGER,
 PRIMARY KEY (ID)
);
```

DATABASE SCHEMA

Persons	0 rows
ID (PK)	INTEGER
LastName	TEXT
FirstName	TEXT
Age	INTEGER

- **Inserting elements into it**

```
CREATE TABLE Persons (
 ID INTEGER NOT NULL,
 LastName TEXT NOT NULL,
 FirstName TEXT,
 Age INTEGER,
 PRIMARY KEY (ID)
);
INSERT INTO PERSONS VALUES (1, 'Hansen', 'Ola','3o');
INSERT INTO PERSONS VALUES (2, 'Svendson', 'Tove','23');
INSERT INTO PERSONS VALUES (3, 'Pettersen', 'Kari','20');
SELECT * FROM PERSONS;
```

QUERY RESULTS

ID	LastName	FirstName	Age
1	Hansen	Ola	3o
2	Svendson	Tove	23
3	Pettersen	Kari	20

- **Creating another table orders with persons id as foreign key and order id as primary**

```
CREATE TABLE Orders (
 OrderID INTEGER NOT NULL,
 OrderNumber INTEGER NOT NULL,
 PersonID INTEGER,
 PRIMARY KEY (OrderID),
 FOREIGN KEY (PersonID) REFERENCES Persons(ID)
);
```

Orders	4 rows
OrderID (PK)	INTEGER
OrderNumber	INTEGER
ID	INTEGER

- **Inserting elements into it**

```
CREATE TABLE Orders (
 OrderID INTEGER NOT NULL,
 OrderNumber INTEGER NOT NULL,
 ID INTEGER,
 PRIMARY KEY (OrderID),
 FOREIGN KEY (ID) REFERENCES Persons(ID)
);
INSERT INTO Orders VALUES (1, 77895,3);
INSERT INTO Orders VALUES (2, 77895,3);
INSERT INTO Orders VALUES (3, 77895,2);
INSERT INTO Orders VALUES (4, 77895,1);
SELECT * FROM Orders;
```

QUERY RESULTS

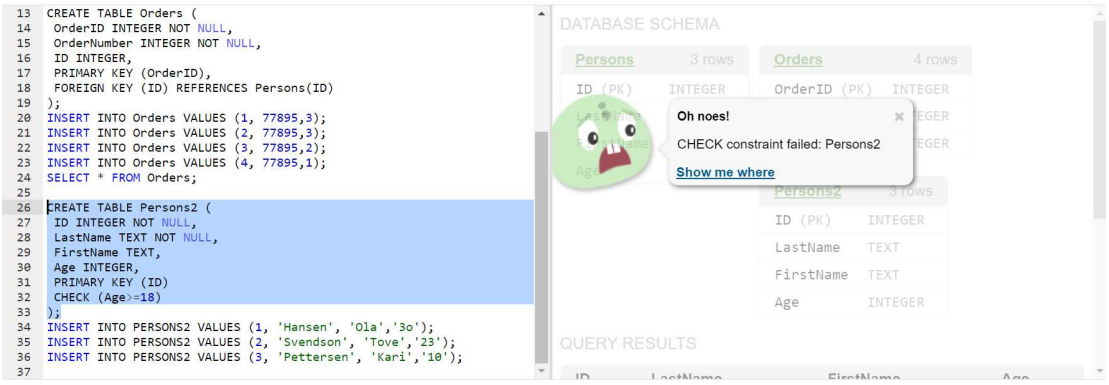
ID	LastName	FirstName	Age
1	Hansen	Ola	3o
2	Svendson	Tove	23
3	Pettersen	Kari	20

OrderID	OrderNumber	ID
1	77895	3
2	77895	3
3	77895	2
4	77895	1

- **Using Check Constraint**

```
CREATE TABLE Persons2 (
 ID INTEGER NOT NULL,
 LastName TEXT NOT NULL,
 FirstName TEXT,
 Age INTEGER,
 PRIMARY KEY (ID)
 CHECK (Age>=18)
);
```

If Constraint failed:



Learning Outcome:

1. Understand the concept of constraints
2. Understand the types of constraints
3. Understand the use of forgin key
4. Understand the use of primary key
5. Understand the use of check

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Viva		
2.	Worksheet		
3.	Performance		