Database concept2

Lab Instructor :shatha Alsafi

SQL Constraints

- SQL constraints are used to specify rules for data in a table.

- Constraints can be specified when the table is created with the CREATE TABLE statement, or after the table is created with the ALTER TABLE statement.

Syntax

CREATE TABLE table_name (
    column1 datatype constraint,
    column2 datatype constraint,
    column3 datatype constraint)

For many columns:

CREATE TABLE table_name (
    column1 datatype constraint,
    column2 datatype constraint,
 CONSTRAINT *CONSTRAINT Name constrantType* (column1, column2)

The following constraints are commonly used 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.
- **FOREIGN KEY** – Uniquely identifies a row/record in another table
- **CHECK** – Ensures that all values in a column satisfies a specific condition
- **DEFAULT** – Sets a default value for a column when no value is specified

Examples of constraints

```
create database sales
use sales
create table person(
id int primary key,
lastName nvarchar(50) unique,
firstName nvarchar(50) not null ,
age int check (age>18),
addressof nvarchar(50) default 'Mukalla'
)
```

Examples of constraints

```
create table orders(
orderId int,
orderName nvarchar(50),
orderDate date,
orderMount int,
personId int,
CONSTRAINT PK_order PRIMARY KEY (orderId),
CONSTRAINT FK_order foreign KEY (personId)
references person(id),
CONSTRAINT CK_order check  (orderMount>0),
)
```

Add constraints by Alter

▸ alter table orders
add primary key (orderId)

▸ alter table orders
add foreign key (personId) references person(id) ,
check (orderMount >1)

▸ alter table orders
alter column orderName nvarchar(50) not null

▸ alter table orders
add constraint DF_order default getdate() for
orderDate

Drop constraints

▸ alter table orders
drop constraint PK_order,
 constraint FK_order,
 constraint CK_order