29/8/25

EXERCISE 12

Intro to Constraints; NOT NULL and UNIQUE Constraints

Global Fast Foods has been very successful this past year and has opened several new stores. They need to add a table to their database to store information about each of their store's locations. The owners want to make sure that all entries have an identification number, date opened, address, and city and that no other entry in the table can have the same email address. Based on this information, answer the following questions about the global_locations table. Use the table for your answers.

Global Fast Foods global_locations Table

NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
Id						
name						
date_opened						
address						-
city						
zip/postal code						
phone						
email						
manager_id						
Emergency contact						

1. What is a "constraint" as it relates to data integrity? A constraint is a rule applied to a database column to enforce data integrity. It ensure data entered into a table meet specific condition for example: UNIQUE, NOT NULL

2. What are the limitations of constraints that may be applied at the column level and at the table level? column-level constraint can only apply to a single column. Table-level constraints can apply to one or more column, but they cannot define column-specific attributes

3. Why is it important to give meaningful names to constraints?

Identify which constraint caused an error
Maintain or modify the database later
Debug and document the database design effectively

4. Based on the information provided by the owners, choose a datatype for each column. Indicate the length, precision, and scale for each NUMBER datatype.

5. Use "(nullable)" to indicate those columns that can have null values.

→ zip / postal code (nullable)
→ phone (nullable)
→ manger_id (nullable)
→ Emergency contact (nullable)

6. Write the CREATE TABLE statement for the Global Fast Foods locations table to define the constraints at the column level.

```
CREATE TABLE global_locations (id NUMBER(4) PRIMARY KEY,
name VARCHAR (20) NOT NULL,
date_opened DATE NOT NULL,
address VARCHAR (30) NOT NULL
city VARCHAR (20) NOT NULL
zip_postal VARCHAR2 (20),
phone VARCHAR2(15),
email VARCHAR2(80) UNIQUE NOT
manager_id NUMBER (4) NULL,
contact VARCHAR (40);
```

7. Execute the CREATE TABLE statement in Oracle Application Express.

you would own the table
SQL code in SQL command or SQL workshop inside
Oracle APEX. (Just type and click Run)

8. Execute a DESCRIBE command to view the Table Summary information.

DESC global_locations;

This command display the column names, datatypes, and nullability

9. Rewrite the CREATE TABLE statement for the Global Fast Foods locations table to define the UNIQUE constraints at the table level. Do not execute this statement.

NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
id	number	4				
loc_name	varchar2	20			X	
	date					
address	varchar2	30				
city	varchar2	20			X	
zip_postal	varchar2	20			X	
phone	varchar2	15			X	
email	varchar2	80			X	
manager_id	number	4			X	
contact	varchar2	40			X	

PRIMARY KEY, FOREIGN KEY, and CHECK Constraints

1. What is the purpose of a
   - PRIMARY KEY - a set of fields that uniquely identifies each record in table
   - FOREIGN KEY - a set of field in one table that refers to the primary key in another table
   - CHECK CONSTRAINT

   Used to limit the value range that can be placed in a column

2. Using the column information for the animals table below, name constraints where applicable at the table level, otherwise name them at the column level. Define the primary key (animal_id). The license_tag_number must be unique. The admit_date and vaccination_date columns cannot contain null values.

animal_id NUMBER(6)
name VARCHAR2(25)
license_tag_number NUMBER(10)
admit_date DATE
adoption_id NUMBER(5),
vaccination_date DATE

```
Create table animals (
animal_id NUMBER (6) PRIMARY KEY
name VARCHAR (25),
license_tag_number_ NUMBER (10) UNIQUE
admit table Date not null
adaption_id NUMBER (5),
vaccination_date DATE NOT NULL);
```

3. Create the animals table. Write the syntax you will use to create the table.

```
create table animals(
animal_id INT PRIMARY KEY,
ADMIT_DATE DATE,
ADOPTION_ID INT,);
```

4. Enter one row into the table. Execute a SELECT * statement to verify your input. Refer to the graphic below for input.

ANIMAL_ID	NAME	LICENSE_TAG_NUMBER	ADMIT_DATE	ADOPTION_ID	VACCINATION_DATE
101	Spot	35540	10-Oct-2004	205	12-Oct-2004

```
Insert into animals (ANIMAL_ID, NAME, LICENSE_TAG,
ADMIT_DATE, ADOPTION_ID, VACCINATION_DATE)
Values (101, 'spot', 35540, 2004-10-10, 20, '2004-10-12'),
```

5. Write the syntax to create a foreign key (adoption_id) in the animals table that has a corresponding primary-key reference in the adoptions table. Show both the column-level and table-level syntax. Note that because you have not actually created an adoptions table, no adoption_id primary key exists, so the foreign key cannot be added to the animals table.

```
Create key animal_id,
ADMIT_DATE DATE,

Vaccination_date DATE NOT NULL;
```

6. What is the effect of setting the foreign key in the ANIMAL table as:

a. ON DELETE CASCADE When a row in parent table is deleted, all corresponding rows in child table are also automatically deleted
b. ON DELETE SET NULL b) when a row in parent table is deleted, the foreign key values in corresponding rows of table are set to NULL

7. What are the restrictions on defining a CHECK constraint?

check constraint cannot explain subqueries
check constraint cannot reference column from other table.

Evaluation Procedure	Marks awarded
Query(5)	5
Execution (5)	5
Viva(5)	5
Total (15)	15
Faculty Signature	