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 rule enforced on data in a table to ensure accuracy & reliability (data integrity)
2. What are the limitations of constraints that may be applied at the column level and at the table level? column level constraints can only reference the column they are defined on table level constraints can find at the end if the create table can reference multiple columns which is necessary for composite keys or complex check constraints.
3. Why is it important to give meaningful names to constraints?
It makes debugging basis. An error message like ORA-02291: integrity constraint (scott.FK-EMP_DEPTNO) violated is unhelpful, where ORA-02291: integrity constraint (SCOTT.FK-EMP_DEPTNO) violated tells you the exact problem
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.
id number 4, name varchar=20, date-opend date, address varchar=30, city varchar=20, Phone varchar=15, email varchar=80, manager-id number 4, contact varchar=40
5. Use "(nullable)" to indicate those columns that can have null values.
Zip-portal, phone, email, manager_id, contact

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) not null, name varchar(20), date-opened date not null, address varchar2(30 not null, city varchar(20) not null, zip portal varchar2(20), phone varchar2(5), email varchar-(80) unique, manager-id Number(4), contact varchar 2(40)

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

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

desc global-locations;

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				
zip_postal	varchar2	20			X	
phone	varchar2	15			X	
email	varchar2	80			X	
manager_id	number	4			X	
contact	varchar2	40			X	

create table global-locations :(id number(4) not null, name varchar(20), date-opened date not null, address varchar2 :(30) not null, city varchar :(20) not null, zip-portal varchar(20), place varchar :(15), email varchar :(80), manager_id number(4), contact varchar (10), constraint glob-lo email_unique (email);

PRIMARY KEY, FOREIGN KEY, and CHECK Constraints

1. What is the purpose of a
   - PRIMARY KEY -> uniquely identifies record
   - FOREIGN KEY -> Links Tables
   - CHECK CONSTRAINT -> Restricts valid data

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

animal_id -> primary key
license_tag_number -> UNIQUE
admit_date and vaccination_date -> NOT NULL

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

```
create table animals( animal_id NUMBER(5) CONSTRAINT pk_animal
PRIMARY KEY, name varchar(25), license_tag_number NUMBER(10)
CONSTRAINT uq_licence UNIQUE, admit_date DATE CONSTRAINT
NOT NULL, adoption_id NUMBER(5), vaccination_date DATE
```

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_number,
admit_data, adoption_id, vaccination_date)
VALUES(101, 'SPOT', 35540, '10-OCT-2004', 205, '12-OCT-2004');
SELECT * FROM animals;
```

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 table animals(animal_id NUMBER(6) PRIMARY KEY,
name VARCHAR(25), license_tag_number(10) UNIQUE,
admit_data DATE NOT NULL, adoption_id NUMBER(5) REFERENCES
adoptions(adoption_id), vaccination_date DATE NOT NULL);
```

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

a. ON DELETE CASCADE → deletes related child rows automatically
b. ON DELETE SET NULL → sets foreign key values in child rows to Null

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

1.) CHECK constraint can only refer to columns within same table but it cannot refer columns in ~~an~~ other tables.
2.) It cannot include subqueries.
3.) It must be a boolean expression that evaluates to TRUE OR FALSE.
4.) It cannot use functions that return non-deterministic value (like SYSDATE, USER, or Random values)
5.) Multiple check constraints can exist on the same table, but each must have a unique name.

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