

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 that ensure data accuracy and integrity in a table

2. What are the limitations of constraints that may be applied at the column level and at the table level?

column level constraint is apply to one column
table-level constraint is apply to many

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

meaningful name make constraints' easy to identify and manage

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 (6) NOT NULL

name - VARCHAR(50) NOT NULL

date opened - DATE NOT NULL

address - VARCHAR(200) NOT NULL

city - VARCHAR(30) NOT NULL

phone - VARCHAR(15) NOT NULL

zip-postal - code

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

manager_id - VARCHAR(20) NOT NULL
Emergency contact - VARCHAR(20) NOT NULL

Nullable columns:

zip - postal - code, phone, manager_id,
Emergency - 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) primary key loc_name VARCHAR2(30) NOT NULL
date_opened DATE NOT NULL address VARCHAR2(30) NOT NULL
city VARCHAR2(20) NOT NULL zip_postal VARCHAR2(20) NOT NULL
email VARCHAR2(80) NOT NULL contact VARCHAR2(40) NOT NULL
);

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

execute the SQL 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)

loc_name VARCHAR2(30) NOT NULL,

date_opened DATE NOT NULL,

address VARCHAR2(30) NOT NULL,

city VARCHAR2(20) NOT NULL,

zip_postal VARCHAR2(20), phone VARCHAR2(15),
manager_id number(4),
contact VARCHAR2(40)

constraint pk_global_id primary key (id)

constraint uq_global_email unique (email)

);

125

PRIMARY KEY, FOREIGN KEY, and CHECK Constraints

1. What is the purpose of a

- PRIMARY KEY - uniquely identifies each row in a table
- FOREIGN KEY - links one table to another, using a related column, ensures value meet specific condition
- CHECK CONSTRAINT

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,

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(6) primary key
name varchar2(25), license_tag_number number(10)
admit_date date not null, adoption_id number(5)
vaccination_date date not null;

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 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.

adoption_id number(5)

REFERENCES adoptions

(adoption_id)

Constraint

foreign key (adoption_id)

REFERENCES

adoptions (adoption_id)

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

- a. ON DELETE CASCADE
- b. ON DELETE SET NULL

ON DELETE CASCADE → when the referenced (parent) record is deleted all related record in the Animal table are also deleted automatically.

ON DELETE SET NULL → when the referenced (parent) record is deleted the foreign key field in the Animal is set to null.

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

The Condition must be a Boolean expression.
 It cannot reference other table or subquery.
 It applies only to the current row's column.
 It cannot use aggregate function!
 (like sum, avg)

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