Programming 14 +

Due on December 2, 2024

14-1 Intro to Constraints; NOT NULL and UNIQUE Constraints

Vocabulary

Every value in a column or set of columns (a composite key) must be unique

For every row entered into the table, there must be a value for that column

Constraint ensures that the column contains no null values and uniquely identifies each row of the table

Specifies a condition for a column that must be true for each row of data

Identifies that table and column in the parent table

An integrity constraint that requires every value in a column or set of columns be unique

Designates a column (child table) that establishes a relationship between a primary key in the same table and a different table (parent table)

References one or more columns and is defined separately from the definitions of the columns in the table Database rule.

Database rule that references a single column

1. What is a “constraint” as it relates to data integrity?
2. What are the limitations of constraints that may be applied at the column level and at the table level?
3. Why is it important to give meaningful names to constraints?
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.
6. Write the CREATE TABLE statement for the Global Fast Foods locations table to define the constraints at the column level.
7. Execute the CREATE TABLE statement in Oracle Application Express.
8. Execute a DESCRIBE command to view the Table Summary information.
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.

• 14-2 PRIMARY KEY, FOREIGN KEY, and CHECK Constraints

Vocabulary

Allows a foreign key row that is referenced to a primary key row to be deleted

Explicitly defines a condition that must be met

A column or set of columns that uniquely identifies each row in a table

Constraint ensures that the column contains no null values

Allows a child row to remain in a table with null values when a parent record has been deleted

Establishes a relationship between the foreign key column and a primary key or unique key in the same table or a different table

1.What is the purpose of a

a.PRIMARY KEY

b.FOREIGN KEY

c.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.

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

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

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.

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

a. ON DELETE CASCADE

b. ON DELETE SET NULL

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

• 14-3 Managing Constraints

To deactivate an integrity constraint

Disables dependent integrity constraints

To add, modify, or drop columns from a table

To activate an integrity constraint currently disabled

Removes a constraint from a table

Allows user to delete a column from a table

Defines the actions the database server takes when a user attempts to delete or update a key to which existing foreign keys point

Using Oracle Application Express, click the SQL Workshop tab in the menu bar. Click the Object Browser and verify that you have a table named copy\_d\_clients and a table named copy\_d\_events. If you don’t have these tables in your schema, create them before completing the exercises below. Here is how the original tables are related. The d\_clients table has a primary key client\_number. This has a primary-key constraint and it is referenced in the foreign-key constraint on the d\_events table.

1.What are four functions that an ALTER statement can perform on constraints?

2.Since the tables are copies of the original tables, the integrity rules are not passed onto the new tables; only the column datatype definitions remain. You will need to add a PRIMARY KEY constraint to the copy\_d\_clients table. Name the primary key copy\_d\_clients\_pk . What is the syntax you used to create the PRIMARY KEY constraint to the copy\_d\_clients.table?

3.Create a FOREIGN KEY constraint in the copy\_d\_events table. Name the foreign key

copy\_d\_events\_fk. This key references the copy\_d\_clients table client\_number column. What is the syntax you used to create the FOREIGN KEY constraint in the copy\_d\_events table?

4.Use a SELECT statement to verify the constraint names for each of the tables. Note that the tablenames must be capitalized.

a.The constraint name for the primary key in the copy\_d\_clients table is

B. The constraint name for the foreign key in the copy\_d\_events table is

5.Drop the PRIMARY KEY constraint on the copy\_d\_clients table. Explain your results.

6.Add the following event to the copy\_d\_events table. Explain your results.

7.Create an ALTER TABLE query to disable the primary key in the copy\_d\_clients table. Then add the values from #6 to the copy\_d\_events table. Explain your results.

8.Repeat question 6: Insert the new values in the copy\_d\_events table. Explain your results.

9.Enable the primary-key constraint in the copy\_d\_clients table. Explain your results.

10. If you wanted to enable the foreign-key column and reestablish the referential integrity between these two tables, what must be done?

11. Why might you want to disable and then re-enable a constraint?

12. Query the data dictionary for some of the constraints that you have created. How does the data dictionary identify each constraint type?

• 15-1 Creating Views

A subset of data from one or more tables that is generated from a query and stored as a virtual table

Name of view

Creates a view regardless of whether or not the base tables exist

Derives data from a table, no functions or groups, performs DML operations through the view

Creates the view only if the base table exists

Statement used to create a new view

Specifies a name for each expression selected by the view’s query

A complete SELECT statement

Derives data from more than one table, contains functions or groups of data, and does not always allow DML operations through the view

Re-creates the view if it already exists

1.What are three uses for a view from a DBA’s perspective?

2.Create a simple view called view\_d\_songs that contains the ID, title, and artist from the DJs on Demand table for each “New Age” type code. In the subquery, use the alias “Song Title” for the title column.

3. SELECT \* FROM view\_d\_songs. What was returned?

4.REPLACE view\_d\_songs. Add type\_code to the column list. Use aliases for all columns.

5.Jason Tsang, the disk jockey for DJs on Demand, needs a list of the past events and those planned for the coming months so he can make arrangements for each event’s equipment setup. As the company manager, you do not want him to have access to the price that clients paid for their events. Create a view for Jason to use that displays the name of the event, the event date, and the theme description. Use aliases for each column name.

6.It is company policy that only upper-level management be allowed access to individual employee salaries. The department managers, however, need to know the minimum, maximum, and average salaries, grouped by department. Use the Oracle database to prepare a view that displays the needed information for department managers.

• 15-2 DML Operations and Views

A pseudocolumn which assigns a sequential value starting with 1 to each of the rows returned from the subquery

Specifies that INSERTS and UPDATES performed through the view can’t create rows which the view cannot select

Ensures that no DML operations can be performed on this view

Use the DESCRIBE statement to verify that you have tables named copy\_d\_songs, copy\_d\_events, copy\_d\_cds, and copy\_d\_clients in your schema. If you don't, write a query to create a copy of each.

1.Query the data dictionary USER\_UPDATABLE\_COLUMNS to make sure the columns in the base tables will allow UPDATE, INSERT, or DELETE. Use a SELECT statement. All table names in the data dictionary are stored in uppercase.

2.Use the CREATE or REPLACE option to create a view of all the columns in the copy\_d\_songs table called view\_copy\_d\_songs.

3.Use view\_copy\_d\_songs to INSERT the following data into the underlying copy\_d\_songs table.

4.Create a view based on the DJs on Demand COPY\_D\_CDS table. Name the view

read\_copy\_d\_cds. Select all columns to be included in the view. Add a WHERE clause to restrict the year to 2000. Add the WITH READ ONLY option.

5.Using the read\_copy\_d\_cds view, execute a DELETE FROM read\_copy\_d\_cds WHERE cd\_number = 90;

6.Use REPLACE to modify read\_copy\_d\_cds. Replace the READ ONLY option with WITH CHECK OPTION CONSTRAINT ck\_read\_copy\_d\_cds. Execute a SELECT \* statement to verify that the view exists.

7. Use the read\_copy\_d\_cds view to delete any CD of year 2000 from the underlying copy\_d\_cds.

8. Use the read\_copy\_d\_cds view to delete cd\_number 90 from the underlying copy\_d\_cds table.

9.Use the read\_copy\_d\_cds view to delete year 2001 records.

10. Execute a SELECT \* statement for the base table copy\_d\_cds. What rows were deleted?

11. What are the restrictions on modifying data through a view?

12. What is Moore’s Law? Do you consider that it will continue to apply indefinitely? Support your opinion with research from the internet.

13. What is the “singularity” in terms of computing?

• 15-3 Managing Views

Asks for the N largest or smallest values in a column

Removes a view

Subquery with an alias that can be used within a SQL statement

1.Create a view from the copy\_d\_songs table called view\_copy\_d\_songs that includes only the title and artist. Execute a SELECT \* statement to verify that the view exists.

2.Issue a DROP view\_copy\_d\_songs. Execute a SELECT \* statement to verify that the view has been deleted.

3.Create a query that selects the last name and salary from the Oracle database. Rank the salaries from highest to lowest for the top three employees.

4.Construct an inline view from the Oracle database that lists the last name, salary, department ID, and maximum salary for each department. Hint: One query will need to calculate maximum salary by department ID.

5.Create a query that will return the staff members of Global Fast Foods ranked by salary from lowest to highest.

Extension Exercises

1.Create a new table called my\_departments and add all columns and all rows to it using a subquery from the Oracle departments table. Do a SELECT \* from my\_departments to confirm that you have all the columns and rows.

2. To view any constraints that may affect the my\_departments table, DESCRIBE my\_departments to check if any constraints were carried over from the departments table. If there are constraints on my\_departments, use an ALTER TABLE command to DISABLE all constraints on my\_departments.

3.Create a view called view\_my\_departments that includes: department\_id and department\_name.

4.Add the following data to the my\_departments table using view\_my\_departments.

5.Create or enable the department\_id column as the primary key.

6.Enter a new department named Human Resources into the my\_departments table using view\_my\_departments. Do not add a new department ID.

7.Add the Human Resources department, department ID 220, to my\_departments using

view\_my\_departments.

8.Verify that the new additions to my\_departments were added using view\_my\_departments.

9.Modify view\_my\_departments to include location ID. Do a SELECT \* command to show what columns are present and a DESCRIBE command to view the columns and associated constraints.

10. Make location\_id a NOT NULL column in the my\_departments table.

11. Using the Oracle database, create a complex view between locations and departments with only

the following columns: department\_name, street\_address, city, and state. Include only U.S. cities. Verify that the view was created using a SELECT \* statement.

16-1 Working With Sequences

Command that automatically generates sequential numbers

Generates a numeric value

Returns the next available sequence value

Specifies the interval between sequence numbers

Specifies a maximum value of 10^27 for an ascending sequence and -1 for a descending sequence (default)

returns the current sequence value

specifies the minimum sequence value

specifies whether the sequence continues to generate values after reaching its maximum or minimum values

specifies a minimum value of 1 for an ascending sequence and – (10^26) for a descending sequence (default)

specifies a maximum or default value the sequence can generate

specifies the first sequence number to be generated

specifies how many values the Server pre-allocates and keeps in memory

1.Using CREATE TABLE AS subquery syntax, create a seq\_d\_songs table of all the columns in the

DJs on Demand database table d\_songs. Use the SELECT \* in the subquery to make sure that you have copied all of the columns.

2.Because you are using copies of the original tables, the only constraints that were carried over were the NOT NULL constraints. Create a sequence to be used with the primary-key column of the seq\_d\_songs table. To avoid assigning primary-key numbers to these tables that already exist, the sequence should start at 100 and have a maximum value of 1000. Have your sequence increment by 2 and have NOCACHE and NOCYCLE. Name the sequence seq\_d\_songs\_seq.

3.Query the USER\_SEQUENCES data dictionary to verify the seq\_d\_songs\_seq SEQUENCE settings.

4.Insert two rows into the seq\_d\_songs table. Be sure to use the sequence that you created for the ID column. Add the two songs shown in the graphic

5.Write out the syntax for seq\_d\_songs\_seq to view the current value for the sequence. Use the DUAL table. (Oracle Application Developer will not run this query.)

6.What are three benefits of using SEQUENCEs?

7.What are the advantages of caching sequence values?

8.Name three reasons why gaps may occur in a sequence?

1.Create a table called “students”. You can decide which columns belong in that table and what datatypes these columns require. (The students may create a table with different columns; however, the important piece that must be there is the student\_id column with a numeric datatype.

This column length must allow the sequence to fit, e.g. a column length of 4 with a sequence that starts with 1 and goes to 10000000 will not work after student #9999 is entered.)

2.Create a sequence called student\_id\_seq so that you can assign unique student\_id numbers for all students that you add to your table.

3.Now write the code to add students to your STUDENTS table, using your sequence “database object.”

• 16-2 Indexes and Synonyms

Confirms the existence of indexes from the USER\_INDEXES

data dictionary view

Schema object that speeds up retrieval of rows

To refer to a table by another name to simplify access

An index that you create on multiple columns in a table

The Oracle Server automatically creates this index when you define a column in a table to have a PRIMARY KEY or a UNIQUE

KEY constraint

Stores the indexed values and uses the index based on a

SELECT statement to retrieve the data

Removes an index

Gives alternative names to objects

1.What is an index and what is it used for?

2.What is a ROWID, and how is it used?

3.When will an index be created automatically?

4.Create a nonunique index (foreign key) for the DJs on Demand column (cd\_number) in the D\_TRACK\_LISTINGS table. Use the Oracle Application Developer SQL Workshop Data Browser to confirm that the index was created.

5.Use the join statement to display the indexes and uniqueness that exist in the data dictionary for the DJs on Demand D\_SONGS table.

6.Use a SELECT statement to display the index\_name, table\_name, and uniqueness from the data dictionary USER\_INDEXES for the DJs on Demand D\_EVENTS table.

7.Write a query to create a synonym called dj\_tracks for the DJs on Demand d\_track\_listings table.

8.Create a function-based index for the last\_name column in DJs on Demand D\_PARTNERS table

that makes it possible not to have to capitalize the table name for searches. Write a SELECT statement that would use this index.

9.Create a synonym for the D\_TRACK\_LISTINGS table. Confirm that it has been created by querying the data dictionary.

10. Drop the synonym that you created in question 9.

• 17-1 Controlling User Access

1.What are system privileges concerned with?

2.What are object privileges concerned with?

3.What is another name for object security?

4.What commands are necessary to allow Scott access to the database with a password of tiger?

5.What are the commands to allow Scott to SELECT from and UPDATE the d\_clients table?

6.What is the command to allow everybody the ability to view the d\_songs table?

7.Query the data dictionary to view the object privileges granted to you the user.

8.What privilege should a user be given to create tables?

9. If you create a table, how can you pass along privileges to other users just to view your table?

10. What syntax would you use to grant another user access to your copy\_employees table?

11. How can you find out what privileges you have been granted for columns in the tables belonging to others?

• 17-2 Creating and Revoking Object Privileges

1.What is a role?

2.What are the advantages of a role to a DBA?

3.Give the ability to another user in your class to look at one of your tables. Give him the right to let other students have that ability.

4.You are the DBA. You are creating many users who require the same system privileges. What should you use to make your job easier?

5.What is the syntax to accomplish the following?

a.Create a role of manager that has the privileges to select, insert, and update and delete from the employees table

b.Create a role of clerk that just has the privileges of select and insert on the employees

table

c.Grant the manager role to user scott

d.Revoke the ability to delete from the employees table from the manager role

6.What is the purpose of a database link?

• 17-3 Regular Expressions

1. Working with the employees table, and using regular expressions, write a query that returns employees whose first names start with a “S” (uppercase) followed by either a “t” (lowercase) or “h” (lowercase).

2.Investigate the LOCATIONS table.

a.Describe the table.

b.Perform a select that returns all rows and all columns of that table.

c.Write a query using regular expressions that removes the spaces in the street\_address column in the LOCATIONS table.