

21.9.25

EXERCISE 13

Creating Views

1. What are three uses for a view from a DBA's perspective?
 1. Security - Restrict access to specific columns or rows of data (User see only what they need).
 2. Simplification - Simplify complex SQL queries by storing them as a single view.
 3. Data Consistency - Provide a consistent, underlying interface even if underlying tables or structures change.
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.

```
CREATE VIEW view-d-songs AS
SELECT id,
       title AS 'Song Title',
       artist
  FROM dis-on-demand
 WHERE type-code = 'New Age'
```
3. SELECT * FROM view_d_songs. What was returned?

This query will return all rows from the view-d-songs showing only 3 columns: song title, artist, and id.
4. REPLACE view_d_songs. Add type_code to the column list. Use aliases for all columns.

```
CREATE OR REPLACE VIEW view-d-songs AS
SELECT id AS 'Song ID',
       title AS 'Song Title',
       artist AS 'Artist Name',
       type-code AS 'Type code',
  FROM dis-on-demand
 WHERE type-code = 'New Age'
```

Or use alias after the CREATE statement as shown.

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.

```
CREATE VIEW view-event-list AS SELECT  
event-name AS "Event Name",  
event-name AS "Event Date",  
theme-description AS "Theme Description" FROM events;
```

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.

```
CREATE VIEW v_perdept_salaries AS  
SELECT department-id AS "Department ID",  
MIN(salary) AS "Minimum salary",  
MAX(salary) AS "Maximum salary",  
AVG(salary) AS "Average salary",  
FROM employees  
GROUP BY departmentid;
```

DML Operations and Views

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. All table names in the data dictionary are stored in uppercase.

```
SELECT table-name, column-name,  
      nullable, update, delete  
  FROM user-updatable-columns  
 WHERE table-name = 'COPY-D-SONGS';  
      'COPY-D-EVENTS', 'COPY-D-CDS', 'COPY-D-CLIENTS'
```

Use the same syntax but change table_name of the other tables.

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.

```
CREATE OR REPLACE VIEW  
  View-copy-d-songs AS  
  SELECT * FROM copy-d-songs;
```

3. Use view_copy_d_songs to INSERT the following data into the underlying copy_d_songs table. Execute a SELECT * from copy_d_songs to verify your DML command. See the graphic.

ID	TITLE	DURATION	ARTIST	TYPE_CODE
88	Mello Jello	2	The Who	4

```
INSERT INTO VIEW_copy-d-songs (id, title, duration, artist,  
type_code) VALUES (88, 'Mello Jello', 2, 'The Who', 4);  
SELECT * FROM copy-d-songs.
```

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.

CREATE OR REPLACE VIEW read-copy-d-cds
AS SELECT *
FROM copy-d-cds
WHERE year = 2000
WITH READ ONLY;

5. Using the read_copy_d_cds view, execute a DELETE FROM read_copy_d_cds WHERE cd_number = 90;

SELECT 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.

CREATE OR REPLACE VIEW read-copy-d-cds
AS SELECT *
FROM copy-d-cds
WHERE year = 2000
WITH CHECK OPTION CONSTRAINT ck-read-copy-d-cds;
SELECT * FROM read-copy-d-cds;

7. Use the read_copy_d_cds view to delete any CD of year 2000 from the underlying copy_d_cds.

SELECT FROM read-copy-d-cds;

8. Use the read_copy_d_cds view to delete cd_number 90 from the underlying copy_d_cds table.

DELETE FROM read-copy-d-cds
WHERE cd-number = 90;

9. Use the read_copy_d_cds view to delete year 2001 records.

DELETE FROM read-copy-d-cds
WHERE year = 2001;

10. Execute a SELECT * statement for the base table copy_d_cds. What rows were deleted?

SELECT * FROM copy-d-cds;

All rows from copy-d-cds where year = 2000 were deleted.

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

1. The view must be updateable. 2- If the view has with READ ONLY, no INSERT, UPDATE OR DELETE is allowed. 3. If the view has with check option, any inserted or updated row must satisfy the views WHERE Condition. i.e. View must include the primary key of the table.

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

Moore's Law states that the number of transistors on a microchip double approximately every 18-20 months, which results in computers becoming cheaper and faster over time likely NO, & Transistors cannot continue shrinking forever, & Heat and power constraints limit further scaling.

13. What is the "singularity" in terms of computing?

The technological singularity is a theoretical point in the future when artificial intelligence surpasses human intelligence to the extent that it can improve itself without human input. At this point, technological growth would become unpredictable and extremely rapid - leading to cognitive changes in civilization.

Managing Views

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.

```
CREATE OR REPLACE VIEW  
view-copy-d-songs AS  
SELECT title, artist  
FROM copy-d-songs;  
SELECT * FROM view-copy-d-songs;
```

2. Issue a DROP view_copy_d_songs. Execute a SELECT * statement to verify that the view has been deleted.

```
DROP VIEW view-copy-d-songs;  
SELECT * FROM view-copy-d-songs;
```

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.

```
SELECT last_name, salary  
FROM employees  
ORDER BY salary DESC  
FETCH FIRST 3 ROWS ONLY;
```

```
SELECT last_name, salary  
FROM (SELECT last_name, salary  
FROM employees  
ORDER BY salary DESC  
WHERE ROWNUM <= 3);
```

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.

```
SELECT e.last_name, e.salary,  
d.department_id, d.max_salary  
FROM employees e  
JOIN (SELECT department_id, MAX(salary) AS max_salary  
FROM employees GROUP BY department_id) d
```

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

```
SELECT *  
FROM global-fast-foods-staff  
ORDER BY salary ASC;
```

Indexes and Synonyms

1. What is an index and what is it used for?
An index is a database object that improves the speed of data retrieval. It works like the index in a book - allowing the database to find rows faster without scanning the entire table.
2. What is a ROWID, and how is it used?
Row ID is a unique address for each row in an Oracle table #1, tells the exact physical storage location of the row on disk, uniquelyoccus specific rows. Identifies duplicate rows.
3. When will an index be created automatically?
Oracle automatically creates an index when:
A primary key or unique constraint is created on a column.
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 Express SQL Workshop Data Browser to confirm that the index was created.
`CREATE INDEX idx_cd_number
ON d-track-listings (cd_number);`
`SELECT index_name, table_name
FROM user_indexes
WHERE table_name = 'D-TRACK-LISTINGS';`
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.
`SELECT i.index_name, i.table_name, i.uniqueness
FROM user_indexes i
JOIN user_index_columns ic
ON i.index_name = ic.index_name
WHERE i.table_name = 'D-SONGS';`
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.
`SELECT index_name, table_name, uniqueness
FROM user_indexes
WHERE table_name = 'D-EVENTS';`
7. Write a query to create a synonym called dj_tracks for the DJs on Demand d_track_listings table.
`CREATE SYNONYM dj_tracks
FOR d-track-listings;
SELECT * FROM dj-tracks;`
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.
`CREATE INDEX ne_upper-last-name
ON department(CUPPER(last_name));`
`SELECT * FROM d-partners
WHERE UPPER(last_name) = UPPER ('Swanson');`

9. Create a synonym for the D_TRACK_LISTINGS table. Confirm that it has been created by querying the data dictionary.

~~CREATE SYNONYM track-listing-syn FOR D-track-listing;~~

~~SELECT SYNONYM_NAME, TABLE_OWNER, TABLE_NAME~~

~~FROM USER-SYNONYMS WHERE SYNONYM_NAME = 'TRACK-LISTING-SYN';~~

10. Drop the synonym that you created in question

~~DROP SYNONYM Track-listings-syn;~~

~~SELECT SYNONYM_NAME~~

~~FROM USER-SYNONYMS~~

~~WHERE SYNONYM_NAME = 'TRACK-LISTING-SYN';~~

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