

EXERCISE 13

23/4/25

Creating Views

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

- Restrict access to sensitive data
- Simplify complex queries
- Present data in a specific format for users

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 d_songs  
Where type_code = 'New Age';
```

3. SELECT * FROM view_d_songs. What was returned?

→ Returns ID, Song Title, And Artist for all New Age Songs

4. REPLACE view_d_songs. Add type_code to the column list. Use aliases for all columns.

```
Create or replace View view_d_songs (Song-ID, Song-Title, Song-Artist,  
Type-Code) As  
Select id, title, artist, type_code  
From d_songs  
Where type_code = 'New Age';
```

Or use alias after the CREATE statement as shown.

(K1)

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-events-jason As  
Select event_name As "Event Name", event_date As "Event Date", theme_desc  
As "Theme Description" From d_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 view-dept-salaries As  
Select dept_id,  
Min(salary) As min_salary,  
Max(salary) As max_salary,  
Avg(salary) As avg_salary  
From employees  
Group by dept_id;
```

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, insertable, updatable, deletable
From user_updatable_columns
Where table_name = 'COPY_D_SONGS';

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 What	4

Insert into view_copy_d_songs Values (88, 'Mello Jello', 2, 'The What', 4);

Select * From copy_d_songs;

(43)

4. Create a view based on the CDs 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 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;

Error: Cannot delete because the view is read only.

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.

• Deleting rows with year = 2000 → allowed

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

• Deleting cd_number = 90 (year 2000) → allowed

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

• Deleting year = 2001 → not allowed (violates check option)

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

Shows that only rows from year 2000 were deleted

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

- View must reference one base table (for DML)
- Cannot modify columns derived from expressions or group functions
- Read-only and check option restrict certain dml actions.

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

- Transistor counts double ^{roughly} every 2 years, increasing performance.
- Modern limit of physics mean it won't continue indefinitely

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

A future point where AI surpasses human intelligence, leading to exponential technological growth.

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 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 Rank() Over (Order by Salary Desc) As  
rank From employees Where Rank() <= 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, e.dept_id, m.max_salary From employees e Join  
(Select dept_id, Max (Salary) As max_salary From employees Group by dept_id) m  
On e.dept_id = m.dept_id;
```

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

```
Select name, salary, Rank() Over (Order by salary Asc) As rank  
From staff;
```

Indexes and Synonyms

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

A data structure that improves query performance by speeding up data retrieval

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

A unique physical address of a row in a table; used internally for quick access

3. When will an index be created automatically?

Automatically created for Primary key and Unique constraints

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);

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.uniqueness, c.column_name
FROM user_indexes i
JOIN user_ind_columns c ON i.index_name = c.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;

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 idx-lower-lastname ON d-partners (LOWER(last_name));
SELECT * FROM d-partners WHERE LOWER(last_name) = 'smith';

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

Select synonym-name, table-name From user-synonyms
Where synonym-name = 'DJ-TRACKS';

10. Drop the synonym that you created in question

~~Drop Synonym dj-tracks;~~

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