

2/9/25

### 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 unchanging interface over 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";

From DJs-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 view  
Showing only \* id  
\* Song title

but only for records where type\_code = 'New Age';

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 DJs-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\_date 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 view\_dept\_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 department\_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_Song";  
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 What	4

Insert into view\_copy\_d\_Songs (id, title, duration, artist, typecode) values (88, 'Mello Jello', 2, 'The What', 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 updatable 2. If the view has with read only, no insert, update or delete is allowed. 3. If the view data has with check option, any inserted or updated row must satisfy the views where condition. 4. 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.

Moore's law : Moore's law states that the number of transistors on a microchip doubles approximately every 18-20 months, which results in computers becoming faster and cheaper 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 the 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 major changes in civilization.

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 read\_copy\_d\_cds

AS Select \*

From copy\_d\_cds

Where year=2000

With Read Only;

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

Select From read\_copy\_d\_cds  
Where cd-number=90;

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.

Create or replace view read\_copy\_d\_cds

AS Select \*

From copy\_d\_cds

Where year=2000

With check option constraint

Cr\_read\_copy\_d\_cds;

Select \* From read\_copy\_d\_cds;

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 From read\_copy\_d\_cds;

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

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

Rowid is a unique address for each row in an Oracle table. It tells the exact physical storage location of the row on disk.

Quickly access specific rows. Identify 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 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;

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 idx\_upper\_last-name  
On D\_Partners (upper(last\_name));

Select \* From dt\_partners

Where upper(last\_name) = upper('Smith');

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 synonyms\_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	RJM