

EXERCISE 13

Creating Views

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

1. Security - use strict access of specific columns or rows of data

2. Simplification - simplify complex SQL queries by storing entire of a single view.

3. Data consistency - provides a consistent, unchanging integer even if underlying table of structure changes

- 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 DJs-on-demand WHERE type_code = 'New Age'

- SELECT * FROM view_d_songs. What was returned?

This query will return all from the view-d-song view showing only :id
song title set only for delete where type_code = "new age".
• artist.

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

CREATE OR REPLACE VIEW view_d_song 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_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 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;
```

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, updateable, deleteable
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 (id, title, duration,  
artist, type code)  
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;

```
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.

```
CREATE OR REPLACE VIEW read_copy_d_cds  
AS SELECT *  
FROM copy_d_cds  
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?

You can't change or delete data through a view
if the view is based on multiple tables.

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

in power about every two years. It says computer
double your is now by down now.

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

The "singularity" is when computers become
smarter than humans. At that point. This called
cause very fast danger in technology.

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
From empolye
order by salary DESC

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.department-id
d-max.salary
From employee
join

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

Select, From staff, red. cap ?

Indexes and Synonyms

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

A tool that speeds up searching for data in table
It is used for finding data.

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

The physical address of a row in the disk.

3. When will an index be created automatically?

When you set a column as Primary Key.

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 d-track-listings_cd_ix

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

Where i_table_name = 'D_SONGS'

Order by i_index_name, column_name

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

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 d_partners.lname (b)
on D_partners (upper(last_name))

Select
first_name;
last_name;
partner_id
from D_partners
where upper(last_name)

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

Create synonym dt-tracks
for D-tracks-listings

10. Drop the synonym that you created in question

Select
synonym-name,
table-owner,
table-name

From user_synonyms
where synonym-name = 'dj_tracks'