EXERCISE 13
Creating Views

1. What are three uses for a view from a DBA's perspective?
   - Restricting data access
   - Simplifying complex queries
   - presenting different data views

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_Song AS select id, title "Song Title", artist
From d_Songs where type_code = (Select type_code from d_types
where type_description = 'New age');
```

3. SELECT * FROM view_d_songs. What was returned?

```
Select * from view_d_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_Name, performer,
Type_Code) AS select id, title, artistic, type_code From d_songs
where type_code = (Select type_code from d_types where type_description
= '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 dj-event_view (Event_Name, Event_Date, Theme_Description) As select e.name, e.event_date, t.theme_description from dj_events e Join dj_themes t on e.theme_code = t.theme_code);

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 dept_Sal_Summary (Department_ID, min_salary, max_salary, avg_salary) As select department_id, Min (Salary), max (Salary), Avg (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 Column_name, updatable, insertable, 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 (ID, Title, duration, artist, type_code)
values (88, 'Mello Jello', '2', 'The What', 4);

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 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 where year = 2000 with check option
constraint ck_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.

delete from read_copy_d_cds where year='2000';

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;

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

- group functions (AVG, SUM, MAX, etc).
- group by clause.
- DISTINCT keyword.
- columns containing expressions.
- DML operations are denied if the view was created with Read Only.

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 is the observation that the number of transistors on an integrated circuit doubles approximately every 2 years

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

The technological singularity is a theoretical point in time wh AI surpasses human intelligence.

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 (Title, Artist)
as select title, artist from 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_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_num from (select last_name,
Salary, Rank() over (order by Salary DESC) as rank_num
from employees) where rank_num <=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.department_salary From
employee e, (select department_id, max (salary) as
max_department_salary from employees group by department
_id) max_sal where e.department_id = max_sal.department_id;
```

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

```
Select last_name, salary, Rank() over (order by salary Asc)
as salary_rank from f_staffs.
```

Indexes and Synonyms

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

an index is a schema object used by the oracle server to speed up the retrieval of rows by providing a rapid access path to locate data quickly

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

A ROWID is the unique physical address of a row in an Oracle database.

3. When will an index be created automatically?

A unique index is created automatically whenever you define a primary key or a unique constraint on a column or set of columns in a table definition.

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 INDEXd_trackcd_fk_idx ON D_TRACK_LISTINGS(cd_numba);
```

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 ix.index_name, ic.column_name, ix.uniqueness
From user_INDEXES ix JOIN USER_IND_COLUMNS ic ON
ix.index_name = ic.index_name
where ix.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 partno_lname_fbi ON D_Partners(upper(last_name));

Select statement_name, table_name from user_Synonyms
where Synonym_name = 'DTL_SYN';
```

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

```
create Synonym dtl_Syn For D_track_listing;
Select * from D_partners where upper(last_name) = 'Smith';
```

10. Drop the synonym that you created in question

```
Drop Synonym dtl_Syn;
```

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