Exercise 13 --- Creating views

```
CREATE VIEW view_d_songs AS
SELECT
    id,
    title AS "Song Title",
    artist
FROM
    DJs_on_Demand
WHERE
    type_code = 'New Age';
```

Output

SQL query successfully executed. However, the result set is empty.

Dropped and then :

```
CREATE VIEW view_d_songs AS
SELECT
    id AS "Song ID",
    title AS "Song Title",
    artist AS "Artist Name",
    type_code AS "Music Type"
FROM
    DJs_on_Demand
WHERE
    type_code = 'New Age';
```

Output

SQL query successfully executed. However, the result set is empty.

```sql
CREATE VIEW view_jason_events AS
SELECT
    event_name AS "Event Name",
    event_date AS "Event Date",
    theme_description AS "Theme Description"
FROM
    events;
```

Output

SQL query successfully executed. However, the result set is empty.

```sql
CREATE  VIEW dept_salary 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;
```

Output

SQL query successfully executed. However, the result set is empty.

Managing Views

1)

```
CREATE OR REPLACE VIEW view_copy_d_songs AS
SELECT title, artist
FROM copy_d_songs;

SELECT * FROM view_copy_d_songs;
```

2)

```
DROP VIEW view_copy_d_songs;

SELECT * FROM view_copy_d_songs;
```

3)

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

4)

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

5)

```
SELECT last_name, salary, RANK() OVER (ORDER BY salary ASC) AS salary_rank
FROM employees
```

```
WHERE department_id IN (
   SELECT department_id
   FROM departments
   WHERE department_name = 'Global Fast Foods'
)
ORDER BY salary ASC;
```

INDEXES

4.

```
CREATE INDEX idx_cd_number
ON d_track_listings(cd_number);
```

Verify in SQL Workshop Data Browser:

```
SELECT index_name, table_name, uniqueness
FROM user_indexes
WHERE table_name = 'D_TRACK_LISTINGS';
```

5.

```
SELECT index_name, table_name, uniqueness
FROM user_indexes
WHERE table_name = 'D_SONGS';
```

6.

```
SELECT index_name, table_name, uniqueness
FROM user_indexes
WHERE table_name = 'D_EVENTS';
```

7.

```
CREATE SYNONYM dj_tracks FOR d_track_listings;
```

Verify creation:

```
SELECT synonym_name, table_name
```

```
FROM user_synonyms

WHERE synonym_name = 'DJ_TRACKS';
```

8.

```
CREATE INDEX idx_last_name_upper

ON d_partners (UPPER(last_name));
```

Use the index in a query:

```
SELECT *

FROM d_partners

WHERE UPPER(last_name) = 'SMITH';
```

9.

```
CREATE SYNONYM syn_d_track_listings FOR d_track_listings;
```

Verify creation:

```
SELECT synonym_name, table_name

FROM user_synonyms

WHERE synonym_name = 'SYN_D_TRACK_LISTINGS';
```

10.

```
DROP SYNONYM syn_d_track_listings;
```

Verify deletion:

```
SELECT synonym_name

FROM user_synonyms

WHERE synonym_name = 'SYN_D_TRACK_LISTINGS';
```