1. Confirms the existence of indexes from the USER_INDEXES data dictionary view: Query USER_INDEXES
2. Schema object that speeds up retrieval of rows: Index
3. To refer to a table by another name to simplify access: Synonym
4. An index created on multiple columns in a table: Composite Index
5. Automatically created index for PRIMARY KEY or UNIQUE KEY: Implicit Index
6. Stores indexed values and retrieves data based on a SELECT statement: B-Tree Index
7. Removes an index: DROP INDEX
8. Gives alternative names to objects: Synonym

1. An index is a schema object that improves the performance of data retrieval operations. It speeds up searches by creating a data structure that allows the database to locate rows more efficiently.
2. ROWID is a unique identifier for each row in a table, representing the physical storage location. It is used for fast row access and can optimize updates and deletes.
3. An index is automatically created when a column is defined as a PRIMARY KEY or UNIQUE KEY.

4.
   ```
   CREATE INDEX idx_cd_number ON d_track_listings (cd_number);
   ```

5.
   ```
   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.
   ```
   SELECT index_name, table_name, uniqueness
   FROM user_indexes
   WHERE table_name = 'D_EVENTS';
   ```

7.
   ```
   CREATE SYNONYM dj_tracks FOR d_track_listings;
   ```

8.
   ```
   CREATE INDEX idx_lower_last_name
   ON d_partners (LOWER(last_name));
   ```

9.
   ```
   SELECT synonym_name, table_name
   FROM user_synonyms
   WHERE synonym_name = 'DJ_TRACKS';
   ```

10.
    ```
    DROP SYNONYM dj_tracks;
    ```