AIM: i) Create primary and secondary indexes on a column?

ii) Retrieve data using an indexes?

iii) Insert data and update indexes?

iv) Delete data and impact on indexes?

DESCRIPTION:

Index

Index is a database structure that improves the speed of data retrieval operations on a table at the cost of additional storage space and slower write operations.

Primary Index and Secondary Index

An index on a set of fields that includes the primary key is called primary index and other indexes are called secondary indexes.

A primary index is guaranteed not to contain duplicates, but an index on other fields can contain duplicates. A secondary index contains duplicates.

when indexes speed up reads dramatically, they make DELETE operations substantially more expensive.

To create a secondary index for the table

Syntax:

CREATE INDEX index_name

ON table_name(columnname)

To drop secondary index

Syntax:

DROP INDEX index_name

1. To create Primary Index StudentID:

```
CREATE TABLE StudentsF (
  StudentID INT PRIMARY KEY,
  FirstName VARCHAR2(50),
  LastName VARCHAR2(50),
  EnrollmentDate DATE
);
```

Table created. 0.15 seconds

To create Secondary Index LastName:

```
CREATE INDEX idx_LastName ON StudentsF(LastName);
```

Index created.

0.02 seconds

2. Insert Data using an index

```
INSERT INTO StudentsF VALUES(101, 'John', 'Doe', TO_DATE('15-Aug-2025', 'DD-Mon-YYYY'));
INSERT INTO StudentsF VALUES(102, 'Jane', 'Smith', TO_DATE('16-Aug-2025', 'DD-Mon-YYYY'));
INSERT INTO StudentsF VALUES(103, 'Ravi', 'Kumar', TO_DATE('25-Oct-2025', 'DD-Mon-YYYY'));
```

These rows automatically update both indexes StudentID and LastName.

3. Retrieve Data Using an Index:

Using the primary index on StudentID.

```
SELECT * FROM StudentsF WHERE StudentID = 102;
```

STUDENTID	FIRSTNAME	LASTNAME	ENROLLMENTDATE
102	Jane	Smith	16-AUG-25

Using the secondary index idx_LastName.

```
SELECT * FROM StudentsF WHERE LastName = 'Doe';
```

STUDENTID	FIRSTNAME	LASTNAME	ENROLLMENTDATE
101	John	Doe	15-AUG-25

4. Delete Data using primary index

DELETE FROM StudentsF WHERE StudentID = 103;

1 row(s) deleted.

Delete Data using secondary index

DELETE FROM StudentsF WHERE LastName = 'Smith';

1 row(s) deleted.

View Index Details

desc idx_LastName

Object Type **INDEX** Object **IDX_LASTNAME**

Drop Index

DROP INDEX idx_LastName;

Index dropped.

0.19 seconds

Impact on Indexes

Database indexes significantly affect performance

- Indexes dramatically speed up SELECT queries, especially with WHERE clauses, JOIN operations, and ORDER BY statements.
- Instead of scanning entire tables, the database can quickly locate specific rows.