Below are examples of different types of indexes in PostgreSQL, along with sample DDL, sample data insertions, and explanations of their use cases.

**1. B-tree Index (Default Index Type)**

**Example:**

sql

Copy code

-- Create a sample table

CREATE TABLE employees (

employee\_id SERIAL PRIMARY KEY,

name VARCHAR(100),

department VARCHAR(50),

salary NUMERIC

);

-- Create a B-tree index on the department column (default index type)

CREATE INDEX idx\_department ON employees (department);

-- Insert sample data

INSERT INTO employees (name, department, salary) VALUES

('Alice Johnson', 'HR', 50000),

('Bob Smith', 'Finance', 60000),

('Charlie Brown', 'Engineering', 70000),

('David Wilson', 'HR', 52000),

('Eve Davis', 'Engineering', 75000);

**2. Hash Index**

**Example:**

sql

Copy code

-- Create a hash index on the name column

CREATE INDEX idx\_name\_hash ON employees USING HASH (name);

-- Insert additional sample data

INSERT INTO employees (name, department, salary) VALUES

('Frank Miller', 'Finance', 58000),

('Grace Lee', 'HR', 51000);

**3. GIN (Generalized Inverted Index)**

**Example:**

sql

Copy code

-- Create a table with an array column

CREATE TABLE products (

product\_id SERIAL PRIMARY KEY,

name VARCHAR(100),

tags TEXT[]

);

-- Create a GIN index on the tags column

CREATE INDEX idx\_tags ON products USING GIN (tags);

-- Insert sample data

INSERT INTO products (name, tags) VALUES

('Laptop', ARRAY['electronics', 'computers']),

('Chair', ARRAY['furniture', 'office']),

('Desk', ARRAY['furniture', 'office']),

('Smartphone', ARRAY['electronics', 'mobile', 'gadgets']);

**4. GiST (Generalized Search Tree)**

**Example:**

sql

Copy code

-- Create a table with a geometric data type

CREATE TABLE locations (

location\_id SERIAL PRIMARY KEY,

name VARCHAR(100),

geom GEOMETRY(Point, 4326)

);

-- Create a GiST index on the geom column

CREATE INDEX idx\_geom ON locations USING GiST (geom);

-- Insert sample data

INSERT INTO locations (name, geom) VALUES

('Point A', ST\_SetSRID(ST\_MakePoint(1.0, 1.0), 4326)),

('Point B', ST\_SetSRID(ST\_MakePoint(2.0, 2.0), 4326)),

('Point C', ST\_SetSRID(ST\_MakePoint(3.0, 3.0), 4326));

**5. BRIN (Block Range INdex)**

**Example:**

sql

Copy code

-- Create a table with a date column

CREATE TABLE sales (

sale\_id SERIAL PRIMARY KEY,

product\_name VARCHAR(100),

sale\_amount NUMERIC,

sale\_date DATE

);

-- Create a BRIN index on the sale\_date column

CREATE INDEX idx\_sale\_date ON sales USING BRIN (sale\_date);

-- Insert sample data

INSERT INTO sales (product\_name, sale\_amount, sale\_date) VALUES

('Product A', 100, '2024-01-01'),

('Product B', 200, '2024-01-02'),

('Product C', 300, '2024-02-01'),

('Product D', 400, '2024-02-02'),

('Product E', 500, '2024-03-01');

**6. SP-GiST (Space-Partitioned Generalized Search Tree)**

**Example:**

sql

Copy code

-- Create a table with a point data type

CREATE TABLE points (

point\_id SERIAL PRIMARY KEY,

point GEOGRAPHY(Point)

);

-- Create an SP-GiST index on the point column

CREATE INDEX idx\_point ON points USING SPGIST (point);

-- Insert sample data

INSERT INTO points (point) VALUES

(ST\_GeogFromText('POINT(-73.935242 40.730610)'), -- New York

ST\_GeogFromText('POINT(-118.243683 34.052235)'), -- Los Angeles

ST\_GeogFromText('POINT(-0.127758 51.507351')); -- London

**Summary of Different Index Types**

1. **B-tree**: Default type used for equality and range queries.
   * Example: Index on the department column in the employees table.
2. **Hash**: Optimized for equality comparisons.
   * Example: Hash index on the name column in the employees table.
3. **GIN**: Suitable for indexing composite types, arrays, and full-text search.
   * Example: GIN index on the tags array column in the products table.
4. **GiST**: Supports geometric data types and various queries.
   * Example: GiST index on the geom column in the locations table.
5. **BRIN**: Efficient for large ordered datasets, summarizing data for range queries.
   * Example: BRIN index on the sale\_date column in the sales table.
6. **SP-GiST**: Designed for non-balanced and partitioned data, useful for spatial queries.
   * Example: SP-GiST index on the point column in the points table.

These examples illustrate how to create different types of indexes in PostgreSQL, including sample data for each scenario to demonstrate their functionality and purpose.