

44

45 **SELECT** * **FROM** DEPARTMENTS;46 **SELECT** * **FROM** EMPLOYEEDETAILS;

Data Output Messages Notifications



SQL

	departmentid [PK] integer	departmentname character varying (30)
--	-------------------------------------	---

1	101	IT
---	-----	----

2	102	MARKETING
---	-----	-----------

3	103	SALES
---	-----	-------

4	104	HR
---	-----	----

Query Query History

```
54      (1007, 'PROJECT_AE', NULL),
55      (1008, 'PROJECT_AF', 102),
56      (1009, 'PROJECT_AG', NULL),
57      (1010, 'PROJECT_AH', 104);
58
59  SELECT * FROM DEPARTMENTS;
60  SELECT * FROM EMPLOYEEDETAILS;
61  SELECT * FROM PROJECTS;
```

Data Output Messages Notifications

SQL

	projectid [PK] integer	projectname character varying (60)	departmentid integer
1	1001	PROJECT_ALPHA	101
2	1002	PROJECT_BETA	103
3	1003	PROJECT_GAMMA	101
4	1004	PROJECT_AB	102
5	1005	PROJECT_AC	104
6	1006	PROJECT_AD	[null]
7	1007	PROJECT_AE	[null]
8	1008	PROJECT_AF	102
9	1009	PROJECT_AG	[null]
10	1010	PROJECT_AH	104

Total rows: 10 of 10 Query complete 00:00:00.121 Ln 61, Col 1

```
65  ▾ /*
66  Create a view named EmployeeDetails that displays the following columns from the
67  ▾ CREATE VIEW
68      EmployeeDetails AS
69  SELECT
70      EmployeeID,
71      EmployeeFirstName,
72      EmployeeLastName,
73      Salary
74  FROM
75      Employees;
```

Data Output Messages Notifications

CREATE VIEW

```
77  ✓  /*
78      Filtering Data in a View */
79  ✓  CREATE OR REPLACE VIEW
80      EmployeeDetails AS
81  SELECT
82      EmployeeID,
83      EmployeeFirstName,
84      EmployeeLastName,
85      Salary
86  FROM
87      Employees
88  WHERE
89      Salary > 40000;
90
```

Data Output Messages Notifications

CREATE VIEW

Query Query History

```
82      EmployeeID,  
83      EmployeeFirstName,  
84      EmployeeLastName,  
85      Salary  
86  FROM  
87      Employees  
88  WHERE  
89      Salary > 40000;  
90  
91  SELECT * FROM EMPLOYEEDETAILS;  
92
```

Data Output Messages Notifications

	employeeid integer	employeefirstname character varying (50)	employeeelastname character varying (50)	salary integer
1	1	RAMU	SHANKAR	60000
2	2	SITA	KUMARI	89000
3	4	MOU	DAS	150000
4	6	RAJU	SINHA	78000
5	7	MAYA	SHANKAR	75000
6	8	JOY	CHATTERJEE	45000

employeedb/postgres@PostgreSQL 17* X

employeedb/postgres@PostgreSQL 17



Query Query History

```
97 CREATE MATERIALIZED VIEW
98 ActiveProjects AS
99 SELECT
100     ProjectID,
101     ProjectName
102 FROM
103     Projects
104 WHERE
105     DepartmentID IS NOT NULL;
106
107 DROP MATERIALIZED VIEW IF EXISTS ActiveProjects;
```

Data Output Messages Notifications

SELECT 7

Query returned successfully in 43 msec.


```
110  Creating an Index */
111  CREATE INDEX idx_Department ON Employees (DepartmentID);
112
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 54 msec.


```
113  ✓ /*  
114  Refreshing a Materialized View */  
115  REFRESH MATERIALIZED VIEW ActiveProjects;  
116  
117  
118
```

Data Output Messages Notifications

REFRESH MATERIALIZED VIEW

Query returned successfully in 46 msec.

```
117
118 ✓ CREATE VIEW
119 DepartmentEmployeeView AS
120 SELECT
121     E.EmployeeID,
122     CONCAT(E.EmployeeFirstName, ' ', E.EmployeeLastName) AS FullName,
123     D.DepartmentName,
124     E.Salary
125 FROM
126     EMPLOYEES E
127 JOIN
128     DEPARTMENTS D
129     ON
130     E.DepartmentID = D.DepartmentID;
131
132
133
```

Data Output Messages Notifications

CREATE VIEW

Query returned successfully in 61 msec.

```

133  v  /*
134  Write a query using this view to select all employees working in the "Sales" de
135  v  SELECT
136      EmployeeID,
137      FullName,
138      DepartmentName
139  FROM
140      DepartmentEmployeeView
141  WHERE
142      DepartmentName = 'SALES';

```

Data Output Messages Notifications



	employeeid integer	fullname text	departmentname character varying (30)
1	9	RAGHAB KUMAR	SALES
2	6	RAJU SINHA	SALES
3	2	SITA KUMARI	SALES

```
144  */
145  Parameterized Queries Using a View */
146  SELECT * FROM
147      DepartmentEmployeeView
148  WHERE
149      SALARY > 60000;
```

Data Output Messages Notifications

	employeeid integer	fullname text	departmentname character varying (30)	salary integer
1	10	RAGHAVENDRA YADAV	IT	1500000
2	4	MOU DAS	IT	150000
3	6	RAJU SINHA	SALES	78000
4	2	SITA KUMARI	SALES	89000
5	7	MAYA SHANKAR	HR	75000

ANUPURBA-ASSIGNMENT2.sql* X

employeeedb/postgres@PostgreSQL 17

Query History No limit

Query Query History

```
148 WHERE
149     SALARY > 60000;
150
151 /*
152 Materialized View with Aggregate Functions */
153 CREATE MATERIALIZED VIEW
154 TopPaidEmployees AS
155 SELECT * FROM
156     EMPLOYEES
157 WHERE
158     SALARY >=
159     (
160         SELECT PERCENTILE_CONT(0.9) WITHIN GROUP (ORDER BY SALARY) FROM
161     );
162
```

Data Output Messages Notifications

SELECT 1

Query returned successfully in 50 msec.

```
162
163 ✓ SELECT
164     DEPARTMENTID,
165     MAX(SALARY)
166 FROM
167 TOPPAIDEMPLOYEES
168 GROUP BY
169 DEPARTMENTID
170
```

Data Output Messages Notifications



SQL

	departmentid integer	max integer
1	101	1500000

Total rows: 1 of 1 Query complete 00:00:00.146 Ln 170, Col 1

```

171
172 SELECT
173     DEPARTMENTID,
174     MAX(SALARY)
175 FROM
176 EMPLOYEES
177 GROUP BY
178 DEPARTMENTID
179 ORDER BY DEPARTMENTID
180
181

```

Data Output Messages Notifications



	departmentid integer	max integer
1	101	1500000
2	102	45000
3	103	89000
4	104	75000


```
181  ✓ /*  
182  Creating a Composite Index */  
183  CREATE INDEX idx_department_salary ON EMPLOYEES(DEPARTMENTID,SALARY);  
184  
185
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 47 msec.

employeeedb/postgres@PostgreSQL 17

Query Query History

```
178 DEPARTMENTID
179 ORDER BY DEPARTMENTID
180
181 /*
182 Creating a Composite Index */
183 CREATE INDEX idx_department_salary ON EMPLOYEES(DEPARTMENTID,SALARY);
184
185 /*
186 Performance Analysis with Indexes */
187 SELECT COUNT(EMPLOYEEID)
188 FROM EMPLOYEES
189 WHERE SALARY>50000;
```

Data Output Messages Notifications

	count bigint
1	6