

Experiment No. : 3

Implement Horizontal and Vertical Fragmentation and perform operations

Problem Statement

Create a global conceptual schema emp (eno, ename, city, salary) wit eno as a primary key and insert 10 records.

Horizontal Fragmentation:

Divide emp into horizontal fragments using the condition that emph1 contains the tuples with salary≤15000 and emph2 with salary>15000.

Vertical Fragmentation:

Divide emp into vertical fragments using the condition that empv1 contains the attributes (eno, ename) and empv2 contains the attributes (eno, city, salary)

Code:

```
CREATE TABLE emp (  
  eno INT PRIMARY KEY,  
  ename VARCHAR(50),  
  city VARCHAR(50),  
  salary DECIMAL(10, 2)  
);  
INSERT INTO emp (eno, ename, city, salary) VALUES (1, 'Pushkaraj',  
'Pune', 17000);  
INSERT INTO emp (eno, ename, city, salary) VALUES (2, 'Aryan',  
'Mumbai', 18000);  
INSERT INTO emp (eno, ename, city, salary) VALUES (3, 'Rohan',  
'Bangalore', 9000);  
INSERT INTO emp (eno, ename, city, salary) VALUES (4, 'Ankita',  
'Hyderabad', 22000);  
INSERT INTO emp (eno, ename, city, salary) VALUES (5, 'Kiran',
```

```

'Chennai', 15000);
INSERT INTO emp (eno, ename, city, salary) VALUES (6, 'Avdhut',
'Pune', 12000);
INSERT INTO emp (eno, ename, city, salary) VALUES (7, 'Ritesh',
'Kolkata', 13000);
INSERT INTO emp (eno, ename, city, salary) VALUES (8, 'Dhairyasheel',
'Jaipur',
25000);
INSERT INTO emp (eno, ename, city, salary) VALUES (9, 'Ishaan',
'Lucknow', 8000);
INSERT INTO emp (eno, ename, city, salary) VALUES (10, 'Atharv',
'Ahmedabad',
16000);

```

-- Horizontal Fragmentation

```

CREATE TABLE emph1 AS
SELECT * FROM emp
WHERE salary <= 15000;
SELECT * FROM emph1;

```

The screenshot shows the SQL Developer interface with the following components:

- Connections:** A tree view on the left showing the 'BTB22' connection under 'Oracle Connections'.
- Worksheet:** The main area displaying the SQL script:


```

CREATE TABLE emph1 AS
SELECT * FROM emp
WHERE salary <= 15000;
SELECT * FROM emph1;

```
- Script Output:** A tab showing the execution status: 'All Rows Fetched: 5 in 0.25'.
- Query Result:** A table displaying the results of the query:

ENO	ENAME	CITY	SALARY
1	3 Rohan	Bangalore	9000
2	5 Kiran	Chennai	15000
3	6 Avdhut	Pune	12000
4	7 Ritesh	Kolkata	13000
5	9 Ishaan	Lucknow	8000

```
CREATE TABLE emph2 AS
SELECT * FROM emp
WHERE salary > 15000;
SELECT * FROM emph2;
```

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane lists 'BTB22' under 'Oracle Connections'. The main window is titled 'BTB22' and contains a 'Worksheet' tab with the following SQL code:

```
CREATE TABLE emph2 AS
SELECT * FROM emp
WHERE salary > 15000;
SELECT * FROM emph2;
```

Below the worksheet, the 'Query Result' tab displays the results of the second query. It shows a table with 5 rows and 4 columns: ENO, ENAME, CITY, and SALARY. The status bar indicates 'All Rows Fetched: 5 in 0.008 se'.

	ENO	ENAME	CITY	SALARY
1	1	Pushkaraj	Pune	17000
2	2	Aryan	Mumbai	18000
3	4	Ankita	Hyderabad	22000
4	8	Dhairyasheel	Jaipur	25000
5	10	Atharv	Ahmedabad	16000

-- Vertical Fragmentation

```
CREATE TABLE empv1 AS
SELECT eno, ename FROM emp;
SELECT * FROM empv1;
```

The screenshot shows the SQL Developer interface with the 'Connections' pane on the left, displaying 'Oracle Connections' and 'BTB22'. The 'Query Builder' tab is active, showing the following SQL script:

```
CREATE TABLE empv1 AS
SELECT eno, ename FROM emp;
SELECT * FROM empv1;
```

The 'Query Result' pane shows the following data:

ENO	ENAME
1	1 Pushkaraj
2	2 Aryan
3	3 Rohan
4	4 Ankita
5	5 Kiran
6	6 Avdhut
7	7 Ritesh
8	8 Dhairyasheel
9	9 Ishaan
10	10 Atharv

```
CREATE TABLE empv2 AS
SELECT eno, city, salary FROM emp;
SELECT * FROM empv2;
```

The screenshot shows the SQL Developer interface with the 'Connections' pane on the left, displaying 'Oracle Connections' and 'BTB22'. The 'Query Builder' tab is active, showing the following SQL script:

```
CREATE TABLE empv2 AS
SELECT eno, city, salary FROM emp;
SELECT * FROM empv2;
```

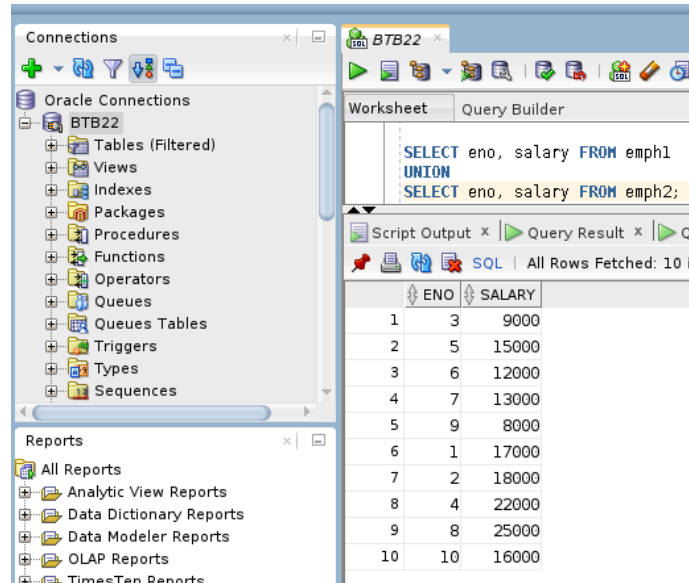
The 'Query Result' pane shows the following data:

ENO	CITY	SALARY
1	1 Pune	17000
2	2 Mumbai	18000
3	3 Bangalore	9000
4	4 Hyderabad	22000
5	5 Chennai	15000
6	6 Pune	12000
7	7 Kolkata	13000
8	8 Jaipur	25000
9	9 Lucknow	8000
10	10 Ahmedabad	16000

Answer below queries.

1. Find the salary of all employees.

```
SELECT eno, salary FROM emp1
UNION
SELECT eno, salary FROM emp2;
```

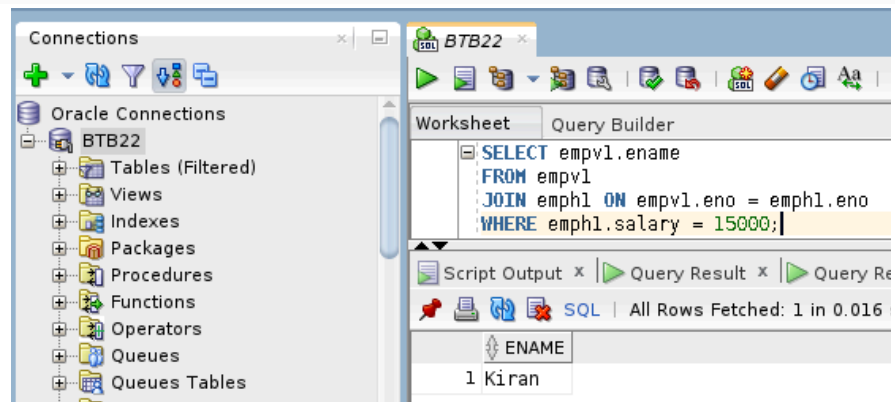


The screenshot shows the SQL Developer interface. On the left, the 'Connections' pane shows 'BTB22' selected under 'Oracle Connections'. The 'Query Builder' tab is active, displaying the SQL query: `SELECT eno, salary FROM emp1 UNION SELECT eno, salary FROM emp2;`. The 'Script Output' pane shows 'All Rows Fetched: 10 i'. The 'Query Result' pane displays a table with two columns: 'ENO' and 'SALARY'.

ENO	SALARY
1	9000
2	15000
3	12000
4	13000
5	8000
6	17000
7	18000
8	22000
9	25000
10	16000

2. Find the name of all employees where salary = 15000.

```
SELECT empv1.ename
FROM empv1
JOIN emp1 ON empv1.eno = emp1.eno
WHERE emp1.salary = 15000;
```



The screenshot shows the SQL Developer interface. On the left, the 'Connections' pane shows 'BTB22' selected under 'Oracle Connections'. The 'Query Builder' tab is active, displaying the SQL query: `SELECT empv1.ename FROM empv1 JOIN emp1 ON empv1.eno = emp1.eno WHERE emp1.salary = 15000;`. The 'Script Output' pane shows 'All Rows Fetched: 1 in 0.016'. The 'Query Result' pane displays a table with one column: 'ENAME'.

ENAME
1 Kiran

3. Find the employee's name and city where employee salary is between 15000 to 25000.

```
SELECT empv1.ename, empv2.city
FROM empv1
JOIN empv2 ON empv1.eno = empv2.eno
```

WHERE empv2.salary BETWEEN 15000 AND 25000;

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane lists 'BTB22' under 'Oracle Connections'. The main window displays a SQL query in the 'Query Builder' tab:

```
SELECT empv1.ename, empv2.city
FROM empv1
JOIN empv2 ON empv1.eno = empv2.eno
WHERE empv2.salary BETWEEN 15000 AND 25000;
```

Below the query, the 'Query Result' tab shows the output of the query:

	ENAME	CITY
1	Pushkaraj	Pune
2	Aryan	Mumbai
3	Ankita	Hyderabad
4	Kiran	Chennai
5	Dhairyasheel	Jaipur
6	Atharv	Ahmedabad

4. Find the employee's name and city where employee number is known.

```
SELECT empv1.ename, empv2.city
FROM empv1
JOIN empv2 ON empv1.eno = empv2.eno
WHERE empv1.eno = 2;
```

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane lists 'BTB22' under 'Oracle Connections'. The main window displays a SQL query in the 'Query Builder' tab:

```
SELECT empv1.ename, empv2.city
FROM empv1
JOIN empv2 ON empv1.eno = empv2.eno
WHERE empv1.eno = 2;
```

Below the query, the 'Query Result' tab shows the output of the query:

	ENAME	CITY
1	Aryan	Mumbai