

Program No:	01
Roll No :	1534
Title of Program :	Partitioning
Objective :	Implementation of Range, Hash and List Partitioning

A. Range Partitioning:

This will be applied to columns.

Partition name-of-partition (criteria)

1. Create table

```
CREATE TABLE sales_range
(salesman_id NUMBER(5),
salesman_name VARCHAR2(30),
sales_amount NUMBER(10),
sales_date DATE)
PARTITION BY RANGE (sales_date)
(
PARTITION p1_sales_jan2000 VALUES LESS
THAN(TO_DATE('01/02/2024','DD/MM/YYYY')),
PARTITION p2_sales_feb2000 VALUES LESS
THAN(TO_DATE('01/03/2024','DD/MM/YYYY')),
PARTITION p3_sales_mar2000 VALUES LESS
THAN(TO_DATE('01/04/2024','DD/MM/YYYY')),
PARTITION p1_sales_apr2000 VALUES LESS
THAN(TO_DATE('01/05/2024','DD/MM/YYYY'))
);
```

```
SQL> CREATE TABLE sales_range
 2     (salesman_id NUMBER(5),
 3       salesman_name VARCHAR2(30),
 4       sales_amount NUMBER(10),
 5       sales_date DATE)
 6     PARTITION BY RANGE (sales_date)
 7     (
 8 PARTITION p1_sales_jan2000 VALUES LESS THAN(TO_DATE('01/02/2024','DD/MM/YYYY')),
 9 PARTITION p2_sales_feb2000 VALUES LESS THAN(TO_DATE('01/03/2024','DD/MM/YYYY')),
10 PARTITION p3_sales_mar2000 VALUES LESS THAN(TO_DATE('01/04/2024','DD/MM/YYYY')),
11 PARTITION p1_sales_apr2000 VALUES LESS THAN(TO_DATE('01/05/2024','DD/MM/YYYY'))
12     );

Table created.
```

2. Insert values into the table

```
INSERT INTO sales_range VALUES(1,'james bond',5000,TO_DATE('23/02/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(2,'Mary kom',6000,TO_DATE('02/03/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(3,'richard hendricks',7000,TO_DATE('22/03/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(4,'kashmira',8000,TO_DATE('31/01/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(5,'ted mosby',9000,TO_DATE('12/02/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(6,'walter white',5500,TO_DATE('15/04/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(7,'bonsoir elliot',10000,TO_DATE('10/04/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(8,'adam',6520,TO_DATE('20/03/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(9,'barney stinson',4875,TO_DATE('29/04/2024','DD/MM/YYYY'));
INSERT INTO sales_range VALUES(10,'Robin
scherbatsky',1000,TO_DATE('16/02/2024','DD/MM/YYYY'));
```

```
SQL> INSERT INTO sales_range VALUES(5,'omprakash',9000,TO_DATE('12/02/2024','DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_range VALUES(6,'Rahul',5500,TO_DATE('15/04/2024','DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_range VALUES(7,'tom cruize',10000,TO_DATE('10/04/2024','DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_range VALUES(8,'Crystal',6520,TO_DATE('20/03/2024','DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_range VALUES(9,'Judy',4875,TO_DATE('29/04/2024','DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_range VALUES(10,'Robin scherbatsky',1000,TO_DATE('16/02/2024','DD/MM/YYYY'));
1 row created.

SQL> SELECT * FROM sales_range PARTITION(p1_sales_apr2000);
```

SALESMAN_ID	SALESMAN_NAME	SALES_AMOUNT	SALES_DAT
6	Rahul	5500	15-APR-24
7	tom cruize	10000	10-APR-24
9	Judy	4875	29-APR-24

```
SQL>
```

3. Create partitions using the range keyword

```
SELECT * FROM sales_range PARTITION(p1_sales_jan2000);
SELECT * FROM sales_range PARTITION(p2_sales_feb2000);
SELECT * FROM sales_range PARTITION(p3_sales_mar2000);
SELECT * FROM sales_range PARTITION(p4_sales_apr2000);
```

```
SQL> SELECT * FROM sales_range PARTITION(p1_sales_jan2000);
```

SALESMAN_ID	SALESMAN_NAME	SALES_AMOUNT	SALES_DAT
4	kashmira	8000	31-JAN-24

```
SQL> SELECT * FROM sales_range PARTITION(p2_sales_feb2000);
```

SALESMAN_ID	SALESMAN_NAME	SALES_AMOUNT	SALES_DAT
1	james bond	5000	23-FEB-24
5	ted mosby	9000	12-FEB-24
10	Robin scherbatsky	1000	16-FEB-24

```
SQL> SELECT * FROM sales_range PARTITION(p3_sales_mar2000);
```

SALESMAN_ID	SALESMAN_NAME	SALES_AMOUNT	SALES_DAT
2	Mary kom	6000	02-MAR-24
3	richard hendricks	7000	22-MAR-24
8	adam	6520	20-MAR-24

```
SQL> SELECT * FROM sales_range PARTITION(p4_sales_apr2000);
```

SALESMAN_ID	SALESMAN_NAME	SALES_AMOUNT	SALES_DAT
6	walter white	5500	15-APR-24
7	bonsoir elliot	10000	10-APR-24
9	barney stinson	4875	29-APR-24

B. Hash Partitioning:

CREATE TABLE

```
CREATE TABLE sales
(
    dept_no number,
    part_no varchar2(20),
    country varchar2(20),
    date1 date,
    amount number
)
PARTITION BY HASH(part_no)
(
    PARTITION p1,
    PARTITION p2,
    PARTITION p3
);
```

```
SQL> CREATE TABLE sales
 2  (
 3  dept_no number,
 4  part_no varchar2(20),
 5  country varchar2(20),
 6  date1 date,
 7  amount number
 8  )
 9  PARTITION BY HASH(part_no)
10  (
11  PARTITION p1,
12  PARTITION p2,
13  PARTITION p3
14  );

Table created.
```

INSERT DATA

```
INSERT INTO sales VALUES (1,'101','India',TO_DATE('23/02/2024','DD/MM/YYYY'),25000);
INSERT INTO sales VALUES (2,'201','India',TO_DATE('10/03/2024','DD/MM/YYYY'),35000);
INSERT INTO sales VALUES (3,'301','US',TO_DATE('05/05/2024','DD/MM/YYYY'),45000);
INSERT INTO sales VALUES (4,'410','UK',TO_DATE('27/04/2024','DD/MM/YYYY'),55000);
INSERT INTO sales VALUES
(5,'551','CANADA',TO_DATE('11/02/2024','DD/MM/YYYY'),65000);
```

```
SQL> select * from sales
2 ;
```

DEPT_NO	PART_NO	COUNTRY	DATE1	AMOUNT
2	201	India	10-MAR-24	35000
3	301	US	05-MAY-24	45000
1	101	India	23-FEB-24	25000
4	410	UK	27-APR-24	55000
5	551	CANADA	11-FEB-24	65000

```
SQL> _
```

```
SELECT * FROM sales PARTITION(p1);
SELECT * FROM sales PARTITION(p2);
SELECT * FROM sales PARTITION(p3);
```

```
SQL> SELECT * FROM sales PARTITION(p2);
```

DEPT_NO	PART_NO	COUNTRY	DATE1	AMOUNT
2	201	India	10-MAR-24	35000
3	301	US	05-MAY-24	45000

```
SQL> SELECT * FROM sales PARTITION(p3);
```

DEPT_NO	PART_NO	COUNTRY	DATE1	AMOUNT
1	101	India	23-FEB-24	25000
4	410	UK	27-APR-24	55000
5	551	CANADA	11-FEB-24	65000

C. List Partitioning:

Enable row movement;

- to move record in case of modification
- ```
CREATE TABLE sales_list (
- salesman_id NUMBER(5),
- salesman_name VARCHAR2(30),
- sales_state VARCHAR2(20),
- sales_amount NUMBER(10),
- sales_date DATE
-)
- PARTITION BY LIST (sales_state) (
- PARTITION sales_west VALUES ('California', 'Hawaii'),
- PARTITION sales_east VALUES ('New York', 'Virginia', 'Florida'),
- PARTITION sales_central VALUES ('Texas', 'Illinois'),
- PARTITION sales_other VALUES (DEFAULT)
-);
```

```
SQL> CREATE TABLE sales_list (
2 salesman_id NUMBER(5),
3 salesman_name VARCHAR2(30),
4 sales_state VARCHAR2(20),
5 sales_amount NUMBER(10),
6 sales_date DATE
7)
8 PARTITION BY LIST (sales_state) (
9 PARTITION sales_west VALUES ('California', 'Hawaii'),
10 PARTITION sales_east VALUES ('New York', 'Virginia', 'Florida'),
11 PARTITION sales_central VALUES ('Texas', 'Illinois'),
12 PARTITION sales_other VALUES (DEFAULT)
13);

Table created.
```

enable row movement;



### **insert into columns:**

-- Insert data, with each record fitting into its respective partition:

```
INSERT INTO sales_list VALUES(1, 'John Smith', 'New York', 6000, TO_DATE('02/03/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(2, 'Mary Jane', 'New York', 6000, TO_DATE('02/03/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(3, 'Neha', 'Virginia', 7000, TO_DATE('22/03/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(4, 'Kush', 'Texas', 8000, TO_DATE('31/01/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(5, 'Luv', 'Florida', 9000, TO_DATE('12/02/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(6, 'Rahul', 'Hawaii', 5500, TO_DATE('15/04/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(7, 'Gwen', 'Illinois', 10000, TO_DATE('10/04/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(8, 'Crystal', 'Virginia', 6520, TO_DATE('20/03/2024', 'DD/MM/YYYY'));
```

-- Insert default case (which will go into 'sales\_other'):

```
INSERT INTO sales_list VALUES(9, 'Judy', 'Unknown', 4875, TO_DATE('29/04/2024', 'DD/MM/YYYY'));
```

```
INSERT INTO sales_list VALUES(10, 'Robin', 'Texas', 1000, TO_DATE('16/02/2024', 'DD/MM/YYYY'));
```



```
SQL> -- Insert data, with each record fitting into its respective partition:
SQL> INSERT INTO sales_list VALUES(1, 'John Smith', 'New York', 6000, TO_DATE('02/03/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(2, 'Mary Jane', 'New York', 6000, TO_DATE('02/03/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(3, 'Neha', 'Virginia', 7000, TO_DATE('22/03/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(4, 'Kush', 'Texas', 8000, TO_DATE('31/01/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(5, 'Luv', 'Florida', 9000, TO_DATE('12/02/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(6, 'Rahul', 'Hawaii', 5500, TO_DATE('15/04/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(7, 'Gwen', 'Illinois', 10000, TO_DATE('10/04/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(8, 'Crystal', 'Virginia', 6520, TO_DATE('20/03/2024', 'DD/MM/YYYY'));
1 row created.

SQL> -- Insert default case (which will go into 'sales_other'):
SQL> INSERT INTO sales_list VALUES(9, 'Judy', 'Unknown', 4875, TO_DATE('29/04/2024', 'DD/MM/YYYY'));
1 row created.

SQL> INSERT INTO sales_list VALUES(10, 'Robin', 'Texas', 1000, TO_DATE('16/02/2024', 'DD/MM/YYYY'));
```

Ensure **row movement** is enabled with:

**ALTER TABLE sales\_list ENABLE ROW MOVEMENT;**

```
SQL> ALTER TABLE sales_list ENABLE ROW MOVEMENT;
Table altered.

SQL> _
```

**UPDATE sales\_list SET sales\_state='Hawaii' WHERE salesman\_id=9;**

```
SQL> UPDATE sales_list SET sales_state='Hawaii' WHERE salesman_id=9;
1 row updated.
```

**SELECT \* FROM sales\_list PARTITION (sales\_west);**

```
SQL> SELECT * FROM sales_list PARTITION (sales_west);
```

| SALESMAN_ID | SALESMAN_NAME | SALES_STATE | SALES_AMOUNT |
|-------------|---------------|-------------|--------------|
| 6           | Rahul         | Hawaii      | 5500         |
| 9           | Judy          | Hawaii      | 4875         |

**SELECT \* FROM sales\_list PARTITION (sales\_east);**

```
SQL> SELECT * FROM sales_list PARTITION (sales_east);
```

| SALESMAN_ID | SALESMAN_NAME | SALES_STATE | SALES_AMOUNT |
|-------------|---------------|-------------|--------------|
| 1           | John Smith    | New York    | 6000         |
| 2           | Mary Jane     | New York    | 6000         |
| 3           | Neha          | Virginia    | 7000         |

  

| SALESMAN_ID | SALESMAN_NAME | SALES_STATE | SALES_AMOUNT |
|-------------|---------------|-------------|--------------|
| 5           | Luv           | Florida     | 9000         |
| 8           | Crystal       | Virginia    | 6520         |

**SELECT \* FROM sales\_list PARTITION (sales\_central);**

```
SQL> SELECT * FROM sales_list PARTITION (sales_central);
```

| SALESMAN_ID | SALESMAN_NAME | SALES_STATE | SALES_AMOUNT |
|-------------|---------------|-------------|--------------|
| 4           | Kush          | Texas       | 8000         |
| 7           | Gwen          | Illinois    | 10000        |
| 10          | Robin         | Texas       | 1000         |