

## Lab report #9

### Sadovskaya Veronika

#### Task 1 - CREATE Example of Range partitioning

##### 1) Range Composite

Create table using Range Partitioning

```
21 CREATE TABLE order_dim
22 (
23   order_id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
24   time_id DATE,
25   client_id NUMBER,
26   CONSTRAINT ORDER_ID_PK PRIMARY KEY ( order_id ) ENABLE
27 )
28 PARTITION BY RANGE (time_id)
29 ( PARTITION order_q1_2022 VALUES LESS THAN (TO_DATE('01.04.2022','DD.MM.YYYY')),
30   PARTITION order_q2_2022 VALUES LESS THAN (TO_DATE('01.07.2022','DD.MM.YYYY')),
31   PARTITION order_q3_2022 VALUES LESS THAN (TO_DATE('01.10.2022','DD.MM.YYYY'))
32   TABLESPACE lab9_01,
33   PARTITION order_q4_2022 VALUES LESS THAN (TO_DATE('01.01.2023','DD.MM.YYYY'))
34   TABLESPACE lab9_01
35 );
```

Script Output x

Task completed in 0,062 seconds

Table ORDER\_DIM created.

Select from table

```
53 SELECT * FROM order_dim
54 ORDER BY order_id;
```

Script Output x Query Result x

SQL | All Rows Fetched: 5 in 0,0

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	03.05.22	1
2	2	24.02.22	1
3	3	03.08.22	1
4	4	07.11.22	1
5	5	15.11.22	1

```

62 SELECT partition_name, tablespace_name
63 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
64

```

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_2022	LAB9_01

### a) Add partition

```

65 ALTER TABLE order_dim
66 ADD PARTITION order_q1_2023 VALUES LESS THAN (TO_DATE('01.04.2023','DD.MM.YYYY'))
67 TABLESPACE lab9_02;

```

Script Output x Query Result x

Task completed in 0,044 seconds

Table ORDER\_DIM altered.

```

61 -----
62 SELECT partition_name, tablespace_name
63 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
64

```

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q1_2023	LAB9_02
3	ORDER_Q2_2022	TS_DATA
4	ORDER_Q3_2022	LAB9_01
5	ORDER_Q4_2022	LAB9_01

### b) Drop partition

```

73 SELECT * FROM order_dim
74 ORDER BY order_id;
75

```

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	03.05.22	1
2	2	24.02.22	1
3	3	03.08.22	1
4	4	07.11.22	1
5	5	15.11.22	1

```

76 DELETE FROM order_dim partition (order_q4_2022);
77 ALTER TABLE order_dim DROP PARTITION order_q4_2022;
78

```

Query Result x Script Output x

Task completed in 0,092 seconds

2 rows deleted.

Table ORDER\_DIM altered.

```

73 SELECT * FROM order_dim
74 ORDER BY order_id;
75

```

Query Result x

SQL | All Rows Fetched: 3 in 0,04

ORDER_ID	TIME_ID	CLIENT_ID
1	03.05.22	1
2	24.02.22	1
3	03.08.22	1

```

71 SELECT partition_name, tablespace_name
72 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
73

```

Query Result x

SQL | All Rows Fetched: 3 in 0,022 seconds

PARTITION_NAME	TABLESPACE_NAME
1 ORDER_Q1_2022	TS_DATA
2 ORDER_Q2_2022	TS_DATA
3 ORDER_Q3_2022	LAB9_01

### c) Merge partitions

```

85 ALTER TABLE order_dim
86 MERGE PARTITIONS order_q3_2022, order_q4_2022
87 INTO PARTITION order_q3_q4_2022;

```

Script Output x

Task completed in 0,083 seconds

Table ORDER\_DIM altered.

```

82 | SELECT partition_name, tablespace_name
83 | FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';

```

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_Q4_2022	TS_DATA

#### d) Move partition

```

94 | ALTER TABLE order_dim
95 | MOVE PARTITION order_q4_2022 TABLESPACE lab9_02;

```

Table ORDER\_DIM altered.

```

91 | SELECT partition_name, tablespace_name
92 | FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
93 |

```

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_2022	LAB9_02

#### e) Split partition

```

104 | SELECT * FROM order_dim PARTITION (order_q4_2022);

```

	ORDER_ID	TIME_ID	CLIENT_ID
1	4	07.11.22	1
2	5	15.11.22	1
3	6	25.11.22	1

```

109 ALTER TABLE order_dim SPLIT PARTITION
110 order_q4_2022 AT (TO_DATE('15.11.2022', 'DD.MM.YYYY')) INTO
111 ( PARTITION order_q4_1_2022, PARTITION order_q4_2_2022);

```

Script Output x

Task completed in 0,114 seconds

Table ORDER\_DIM altered.

```

99 SELECT partition_name, tablespace_name
100 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';

```

Script Output x Query Result x

SQL | All Rows Fetched: 5 in 0,025 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_1_2022	LAB9_01
5	ORDER_Q4_2_2022	LAB9_01

```

113 SELECT * FROM order_dim PARTITION (order_q4_1_2022);

```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0,026 seconds

	ORDER_ID	TIME_ID	CLIENT_ID
1	4	07.11.22	1

```

115 SELECT * FROM order_dim PARTITION (order_q4_2_2022);

```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0,032 seconds

	ORDER_ID	TIME_ID	CLIENT_ID
1	5	15.11.22	1
2	6	25.11.22	1

## f) Truncate partition

```
123 SELECT * FROM order_dim
124 ORDER BY order_id;
```

Query Result x

SQL | All Rows Fetched: 5 in 0,07

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	03.05.22	1
2	2	24.02.22	1
3	3	03.08.22	1
4	4	07.11.22	1
5	5	15.11.22	1

```
126 DELETE FROM order_dim PARTITION (order_q4_2022);
127 ALTER TABLE order_dim TRUNCATE PARTITION order_q4_2022;
```

Script Output x

Task completed in 0,077 seconds

2 rows deleted.

Table ORDER\_DIM truncated.

```
123 SELECT * FROM order_dim
124 ORDER BY order_id;
```

Query Result x

SQL | All Rows Fetched: 3 in 0,03

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	03.05.22	1
2	2	24.02.22	1
3	3	03.08.22	1

```
120 SELECT partition_name, tablespace_name
121 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
```

Query Result x

SQL | All Rows Fetched: 4 in 0,026 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_2022	LAB9_01



## 2) Interval Composite

Create table using Interval partitioning

```
5 CREATE TABLE order_dim
6 (
7   order_id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
8   time_id DATE,
9   client_id NUMBER,
10  CONSTRAINT ORDER_ID_PK PRIMARY KEY ( order_id ) ENABLE
11 )
12 PARTITION BY RANGE (time_id)
13 INTERVAL(NUMTOYMINTERVAL(1, 'MONTH'))
14 ( PARTITION order_m1_2022 VALUES LESS THAN (TO_DATE('01.04.2022','DD.MM.YYYY')),
15   PARTITION order_m2_2022 VALUES LESS THAN (TO_DATE('01.05.2022','DD.MM.YYYY'))
16 ) TABLESPACE lab9_01;
```

Script Output x

Task completed in 0,062 seconds

Table ORDER\_DIM created.

```
37 SELECT partition_name, tablespace_name
38 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
```

Script Output x Query Result x

All Rows Fetched: 3 in 0,028 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_M1_2022	LAB9_01
2	ORDER_M2_2022	LAB9_01
3	SYS_P927	LAB9_01

### a) Add partition

```
40 ALTER TABLE order_dim
41 ADD PARTITION order_m3_2022 VALUES LESS THAN (TO_DATE('01.06.2023','DD.MM.YYYY'))
42 TABLESPACE lab9_02;
```

Script Output x Query Result x

Task completed in 0,063 seconds

Error starting at line : 40 in command -

```
ALTER TABLE order_dim
ADD PARTITION order_m3_2022 VALUES LESS THAN (TO_DATE('01.06.2023','DD.MM.YYYY'))
TABLESPACE lab9_02
```

Error report -

ORA-14760: для объектов, секционированных по интервалам использование ADD PARTITION запрещено

14760. 00000 - "ADD PARTITION is not permitted on Interval partitioned objects"

\*Cause: ALTER TABLE ADD PARTITION was attempted on an Interval partitioned object.

\*Action: Do not perform the operation on an Interval partitioned object.  
Insert a row to create the new partition.

## b) Drop partition

```
50 SELECT * FROM order_dim
51 ORDER BY order_id;
52
```

Query Result x

SQL | All Rows Fetched: 4 in 0,033

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	23.04.22	1
2	2	24.03.22	1
3	3	03.02.22	1
4	4	07.05.22	1

```
53 DELETE FROM order_dim partition (order_m1_2022);
54 ALTER TABLE order_dim DROP PARTITION order_m1_2022;
55
```

Query Result x Script Output x

Task completed in 0,084 seconds

2 rows deleted.

Table ORDER\_DIM altered.

```
50 SELECT * FROM order_dim
51 ORDER BY order_id;
52
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0,034

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	23.04.22	1
2	4	07.05.22	1

```
47 SELECT partition_name, tablespace_name
48 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
49
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0,022 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_M2_2022	LAB9_01
2	SYS_P927	LAB9_01



### c) Merge partitions

```
61 ALTER TABLE order_dim
62 MERGE PARTITIONS order_m1_2022, order_m2_2022 INTO PARTITION order_m1_m2_2022;
63
```

Script Output x

Task completed in 0,119 seconds

Table ORDER\_DIM altered.

```
58 SELECT partition_name, tablespace_name
59 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
60
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0,028 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_M1_M2_2022	LAB9_01
2	SYS_P928	LAB9_01

### d) Move partition

```
69 ALTER TABLE order_dim
70 MOVE PARTITION order_m1_2022 TABLESPACE lab9_02;
71
```

Script Output x

Task completed in 0,06 seconds

Table ORDER\_DIM altered.

```
66 SELECT partition_name, tablespace_name
67 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
68
```

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0,022 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_M1_2022	LAB9_02
2	ORDER_M2_2022	LAB9_01
3	SYS_P929	LAB9_01

### e) Split partition

```
77 | SELECT * FROM order_dim
78 | ORDER BY order_id;
```

Query Result x

SQL | All Rows Fetched: 4 in 0,0:

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	23.04.22	1
2	2	24.03.22	1
3	3	03.02.22	1
4	4	07.05.22	1

```
80 | ALTER TABLE order_dim SPLIT PARTITION
81 | order_ml_2022 AT (TO_DATE('01.03.2022','DD.MM.YYYY')) INTO
82 | ( PARTITION order_ml_1_2022, PARTITION order_ml_2_2022);
```

Script Output x

Task completed in 0,141 seconds

Table ORDER\_DIM altered.

```
84 | SELECT * FROM order_dim PARTITION (order_ml_1_2022);
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0,028 seconds

	ORDER_ID	TIME_ID	CLIENT_ID
1	3	03.02.22	1

```
86 | SELECT * FROM order_dim PARTITION (order_ml_2_2022);
87 |
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0,031 seconds

	ORDER_ID	TIME_ID	CLIENT_ID
1	2	24.03.22	1

## f) Truncate partition

```
93 SELECT * FROM order_dim
94 ORDER BY order_id;
95
```

Query Result x

SQL | All Rows Fetched: 4 in 0,042

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	23.04.22	1
2	2	24.03.22	1
3	3	03.02.22	1
4	4	07.05.22	1

```
96 DELETE FROM order_dim PARTITION (order_m1_2022);
97 ALTER TABLE order_dim TRUNCATE PARTITION order_m1_2022;
98
```

Script Output x

Task completed in 0,081 seconds

2 rows deleted.

Table ORDER\_DIM truncated.

```
93 SELECT * FROM order_dim
94 ORDER BY order_id;
95
```

Query Result x

SQL | All Rows Fetched: 2 in 0,03

	ORDER_ID	TIME_ID	CLIENT_ID
1	1	23.04.22	1
2	4	07.05.22	1

```
90 SELECT partition_name, tablespace_name
91 FROM user_tab_partitions t WHERE t.table_name='ORDER_DIM';
```

Query Result x

SQL | All Rows Fetched: 3 in 0,023 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_M1_2022	LAB9_01
2	ORDER_M2_2022	LAB9_01
3	SYS_P931	LAB9_01

### 3) Hash

Create table using Hash partitioning

```
5 CREATE TABLE orders_by_countries
6 (
7   order_id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
8   country_name VARCHAR(10),
9   order_date DATE,
10  CONSTRAINT ORDER_ID_PK PRIMARY KEY ( order_id ) ENABLE
11 )
12 PARTITION BY HASH (country_name)
13 (
14   PARTITION Belarus,
15   PARTITION Russia,
16   PARTITION Ukraine
17 ) TABLESPACE lab9_01;
```

Script Output x Query Result x

Task completed in 0,212 seconds

Table ORDERS\_BY\_COUNTRIES created.

```
39 SELECT partition_name, tablespace_name
40 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_COUNTRIES';
```

Query Result x

SQL | All Rows Fetched: 3 in 0,071 seconds

PARTITION_NAME	TABLESPACE_NAME
1 BELARUS	LAB9_01
2 RUSSIA	LAB9_01
3 UKRAINE	LAB9_01

#### a) Add partitions

```
41 ALTER TABLE orders_by_countries ADD PARTITION Poland TABLESPACE lab9_01;
```

Query Result x Script Output x

Task completed in 0,073 seconds

Table ORDERS\_BY\_COUNTRIES altered.

```
38
39 SELECT partition_name, tablespace_name
40 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_COUNTRIES';
41
```

Script Output x Query Result x

SQL | All Rows Fetched: 4 in 0,024 seconds

PARTITION_NAME	TABLESPACE_NAME
1 BELARUS	LAB9_01
2 POLAND	LAB9_01
3 RUSSIA	LAB9_01
4 UKRAINE	LAB9_01

## b) Coalesce partition

```
65 ALTER TABLE orders_by_countries
66 COALESCE PARTITION;
```

Script Output x

Task completed in 0,239 seconds

Table ORDERS\_BY\_COUNTRIES altered.

```
62 SELECT partition_name, tablespace_name
63 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_COUNTRIES';
64
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0,03 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	BELARUS	LAB9_01
2	RUSSIA	LAB9_01

## c) Move partition

```
49 ALTER TABLE orders_by_countries
50 MOVE PARTITION Ukraine TABLESPACE lab9_02;
51
```

Script Output x

Task completed in 0,058 seconds

Table ORDERS\_BY\_COUNTRIES altered.

```
45
46 SELECT partition_name, tablespace_name
47 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_COUNTRIES';
48
```

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0,024 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	BELARUS	LAB9_01
2	RUSSIA	LAB9_01
3	UKRAINE	LAB9_02

#### d) Truncate partition

```
57 DELETE FROM orders_by_countries PARTITION (Ukraine);
58 ALTER TABLE orders_by_countries TRUNCATE PARTITION Ukraine;
59
```

Script Output x

Task completed in 0,092 seconds

4 rows deleted.

Table ORDERS\_BY\_COUNTRIES truncated.

```
53
54 SELECT partition_name, tablespace_name
55 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_COUNTRIES';
56
```

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0,023 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	BELARUS	LAB9_01
2	RUSSIA	LAB9_01
3	UKRAINE	LAB9_01

#### 4) List Composite

Create table using list partitioning

```
5 CREATE TABLE orders_by_region (
6   order_id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
7   state_code VARCHAR(2),
8   order_date DATE,
9   CONSTRAINT ORDER_ID_PK PRIMARY KEY ( order_id ) ENABLE
10 )
11 STORAGE(INITIAL 10K NEXT 20K) TABLESPACE lab9_01
12 PARTITION BY LIST (state_code) (
13   PARTITION region_east
14     VALUES ('MA','NY','CT','NH','ME','MD','VA','PA','NJ')
15     STORAGE (INITIAL 8M)
16     TABLESPACE lab9_02,
17   PARTITION region_west
18     VALUES ('CA','AZ','NM','OR','WA','UT','NV','CO')
19     NOLOGGING,
20   PARTITION region_south
21     VALUES ('TX','KY','TN','LA','MS','AR','AL','GA'),
22   PARTITION region_central
23     VALUES ('OH','ND','SD','MO','IL','MI','IA'),
24   PARTITION region_null
25     VALUES (NULL)
26 );
```

Script Output x

Task completed in 0,065 seconds

Table ORDERS BY REGION created.

```

52 SELECT partition_name, tablespace_name
53 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';
54

```

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL	LAB9_01
2	REGION_EAST	LAB9_02
3	REGION_NULL	LAB9_01
4	REGION_SOUTH	LAB9_01
5	REGION_WEST	LAB9_01

### a) Add partition

```

55 ALTER TABLE orders_by_region
56 ADD PARTITION region_central_2
57 VALUES ('RL', 'TG')
58 STORAGE (INITIAL 20K NEXT 20K) TABLESPACE lab9_02

```

Script Output x | Task completed in 0,156 seconds

Table ORDERS\_BY\_REGION altered.

```

52 SELECT partition_name, tablespace_name
53 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';
54

```

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL	LAB9_01
2	REGION_CENTRAL_2	LAB9_02
3	REGION_EAST	LAB9_02
4	REGION_NULL	LAB9_01
5	REGION_SOUTH	LAB9_01
6	REGION_WEST	LAB9_01

### b) Drop partition

```

66 SELECT * FROM orders_by_region
67 ORDER BY order_id;

```

	ORDER_ID	STATE_CODE	ORDER_DATE
1	1	MA	05.12.22
2	2	NY	25.07.22
3	3	WA	13.03.22
4	4	UT	09.02.22
5	5	(null)	15.11.22



```

69 DELETE FROM orders_by_region partition (region_null);
70 ALTER TABLE orders_by_region DROP PARTITION region_null;

```

Script Output x

Task completed in 0,112 seconds

1 row deleted.

Table ORDERS\_BY\_REGION altered.

```

66 SELECT * FROM orders_by_region
67 ORDER BY order_id;

```

Script Output x Query Result x

SQL | All Rows Fetched: 4 in 0,096 sec

	ORDER_ID	STATE_CODE	ORDER_DATE
1	1	MA	05.12.22
2	2	NY	25.07.22
3	3	WA	13.03.22
4	4	UT	09.02.22

```

63 SELECT partition_name, tablespace_name
64 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';

```

Script Output x Query Result x

SQL | All Rows Fetched: 4 in 0,042 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL	LAB9_01
2	REGION_EAST	LAB9_02
3	REGION_SOUTH	LAB9_01
4	REGION_WEST	LAB9_01

### c) Merge partitions

```

77 ALTER TABLE orders_by_region
78 MERGE PARTITIONS region_central, region_null INTO PARTITION region_central_and_null;
79

```

Script Output x

Task completed in 0,073 seconds

Table ORDERS\_BY\_REGION altered.

```

74 | SELECT partition_name, tablespace_name
75 | FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';

```

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL_AND_NULL	LAB9_01
2	REGION_EAST	LAB9_02
3	REGION_SOUTH	LAB9_01
4	REGION_WEST	LAB9_01

#### d) Move partition

```

87 | ALTER TABLE orders_by_region
88 | MOVE PARTITION region_central TABLESPACE lab9_02;

```

Script Output x | Task completed in 0,046 seconds

Table ORDERS\_BY\_REGION altered.

```

84 | SELECT partition_name, tablespace_name
85 | FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';

```

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL	LAB9_02
2	REGION_EAST	LAB9_02
3	REGION_NULL	LAB9_01
4	REGION_SOUTH	LAB9_01
5	REGION_WEST	LAB9_01

#### e) Split partition

```

98 | ALTER TABLE orders_by_region
99 | SPLIT PARTITION region_east VALUES ('CT', 'MA', 'MD')
100 | INTO ( PARTITION region_east_1 TABLESPACE lab9_02,
101 |       PARTITION region_east_2
102 |       STORAGE (NEXT 2M PCTINCREASE 25))
103 | PARALLEL 5;

```

Script Output x | Task completed in 0,138 seconds

Table ORDERS\_BY\_REGION altered.

```

92 | SELECT partition_name, tablespace_name
93 | FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';
94 |

```

Query Result x

SQL | All Rows Fetched: 6 in 0,026 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL	LAB9_01
2	REGION_EAST_1	LAB9_02
3	REGION_EAST_2	LAB9_02
4	REGION_NULL	LAB9_01
5	REGION_SOUTH	LAB9_01
6	REGION_WEST	LAB9_01

```

105 | SELECT * FROM orders_by_region PARTITION (region_east_1);
106 |

```

Query Result x

SQL | All Rows Fetched: 1 in 0,034 seconds

	ORDER_ID	STATE_CODE	ORDER_DATE
1	1	MA	05.12.22

```

107 | SELECT * FROM orders_by_region PARTITION (region_east_2);
108 |

```

Query Result x

SQL | All Rows Fetched: 1 in 0,032 seconds

	ORDER_ID	STATE_CODE	ORDER_DATE
1	2	NY	25.07.22

## f) Truncate partition

```

66 | SELECT * FROM orders_by_region
67 | ORDER BY order_id;

```

Query Result x

SQL | All Rows Fetched: 5 in 0,029 seconds

	ORDER_ID	STATE_CODE	ORDER_DATE
1	1	MA	05.12.22
2	2	NY	25.07.22
3	3	WA	13.03.22
4	4	UT	09.02.22
5	5	(null)	15.11.22

```

117 DELETE FROM orders_by_region PARTITION (region_null);
118 ALTER TABLE orders_by_region TRUNCATE PARTITION region_null;
119

```

Script Output x

Task completed in 0,08 seconds

1 row deleted.

Table ORDERS\_BY\_REGION truncated.

```

114 SELECT * FROM orders_by_region
115 ORDER BY order_id;
116

```

Script Output x Query Result x

All Rows Fetched: 4 in 0,03 seconds

	ORDER_ID	STATE_CODE	ORDER_DATE
1	1	MA	05.12.22
2	2	NY	25.07.22
3	3	WA	13.03.22
4	4	UT	09.02.22

```

111 SELECT partition_name, tablespace_name
112 FROM user_tab_partitions t WHERE t.table_name='ORDERS_BY_REGION';
113

```

Script Output x Query Result x

All Rows Fetched: 5 in 0,027 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	REGION_CENTRAL	LAB9_01
2	REGION_EAST	LAB9_02
3	REGION_NULL	LAB9_01
4	REGION_SOUTH	LAB9_01
5	REGION_WEST	LAB9_01

## 5) Reference

Create table using Reference partitioning

```
7  order_id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
8  time_id DATE,  
9  client_id NUMBER,  
10 CONSTRAINT ORDER_ID_PK PRIMARY KEY ( order_id ) ENABLE)  
11 PARTITION BY RANGE (time_id)  
12 ( PARTITION order_q1_2022 VALUES LESS THAN (TO_DATE('01.04.2022','DD.MM.YYYY')),  
13   PARTITION order_q2_2022 VALUES LESS THAN (TO_DATE('01.07.2022','DD.MM.YYYY')),  
14   PARTITION order_q3_2022 VALUES LESS THAN (TO_DATE('01.10.2022','DD.MM.YYYY'))  
15   TABLESPACE lab9_01,  
16   PARTITION order_q4_2022 VALUES LESS THAN (TO_DATE('01.01.2023','DD.MM.YYYY'))  
17   TABLESPACE lab9_01  
18 );  
19 CREATE TABLE order_items  
20 ( order_id          NUMBER(12) NOT NULL,  
21   line_item_id      NUMBER(3)  NOT NULL,  
22   quantity          NUMBER(8),  
23   CONSTRAINT order_items_fk  
24   FOREIGN KEY(order_id) REFERENCES order_dim(order_id))  
25 PARTITION BY REFERENCE (order_items_fk);
```

Script Output x

Task completed in 0,143 seconds

Table ORDER\_DIM created.

Table ORDER\_ITEMS created.

```
60 SELECT partition_name, tablespace_name  
61 FROM user_tab_partitions t WHERE t.table_name='ORDER_ITEMS';  
62
```

Query Result x

SQL | All Rows Fetched: 4 in 0,064 seconds

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_2022	LAB9_01

### a) Move partition

```
63 ALTER TABLE order_items  
64 MOVE PARTITION order_q2_2022 TABLESPACE lab9_02;  
65
```

Query Result x | Script Output x

Task completed in 0,075 seconds

Table ORDER\_ITEMS altered.

```

60 SELECT partition_name, tablespace_name
61 FROM user_tab_partitions t WHERE t.table_name='ORDER_ITEMS';
62

```

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	LAB9_02
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_2022	LAB9_01

## b) Truncate partition

```

74 DELETE FROM order_items PARTITION (order_q4_2022);
75 ALTER TABLE order_items TRUNCATE PARTITION order_q4_2022;
76

```

Script Output x

Task completed in 0,104 seconds

2 rows deleted.

Table ORDER\_ITEMS truncated.

```

71 SELECT * FROM order_items
72 ORDER BY order_id;
73

```

	ORDER_ID	LINE_ITEM_ID	QUANTITY
1	1	12	10
2	2	3	10
3	3	20	10

```

68 SELECT partition_name, tablespace_name
69 FROM user_tab_partitions t WHERE t.table_name='ORDER_ITEMS';
70

```

	PARTITION_NAME	TABLESPACE_NAME
1	ORDER_Q1_2022	TS_DATA
2	ORDER_Q2_2022	TS_DATA
3	ORDER_Q3_2022	LAB9_01
4	ORDER_Q4_2022	LAB9_01



## Task 2 - Partitioning Facts

The orders are divided by quarters using range partitioning by DATE\_ID. The quarters are divided using hash partitioning by CLIENT\_ID.

Example of a result table of facts with partition taken into account:

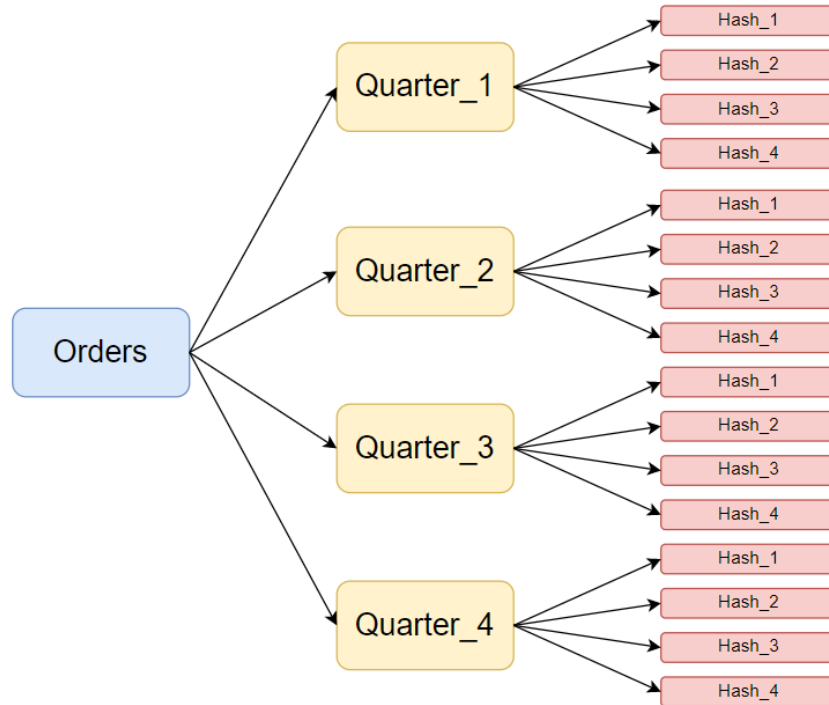


Table with the suggested partition option:

```
3 CREATE TABLE order_fact(  
4   order_id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
5   client_id NUMBER,  
6   employee_id NUMBER,  
7   restaurant_id NUMBER,  
8   date_id DATE,  
9   geo_id NUMBER,  
10  total_cost DECIMAL (11,2) NOT NULL,  
11  delivery CHAR(1) NOT NULL CHECK (delivery IN ('N','Y')),  
12  CONSTRAINT ORDER_ID_PK PRIMARY KEY ( order_id ) ENABLE)  
13  PARTITION BY RANGE (date_id)  
14  subpartition by hash(client_id) subpartitions 4  
15  (  
16  PARTITION quarter_1 VALUES LESS THAN(to_date('01.04.2022','DD.MM.YYYY'))  
17    (  
18      subpartition quarter_1_sub_1,  
19      subpartition quarter_1_sub_2,  
20      subpartition quarter_1_sub_3,  
21      subpartition quarter_1_sub_4  
22    ),  
23  PARTITION quarter_2 VALUES LESS THAN(to_date('01.07.2022','DD.MM.YYYY'))  
24    (  
25      subpartition quarter_2_sub_1,  
26      subpartition quarter_2_sub_2,  
27      subpartition quarter_2_sub_3,  
28      subpartition quarter_2_sub_4  
29    ),  
30  )
```