# Prerequisites

# Oracle Architecture - Partitioning

## Task 01: CREATE Example of Range partitioning

*-----------Range Composite Range-\**

*----------------------Adding Partitions*

DROP TABLE range\_example;

CREATE TABLE range\_example

(

DAY\_ID DATE NOT NULL,

DAY\_NAME VARCHAR2(36)

)

PARTITION BY RANGE (DAY\_ID)

(PARTITION part\_1

VALUES LESS THAN (TO\_DATE ('01/01/2012', 'dd/mm/yyyy')),

PARTITION part\_2

VALUES LESS THAN (TO\_DATE ('01/01/2013', 'dd/mm/yyyy')),

PARTITION part\_3

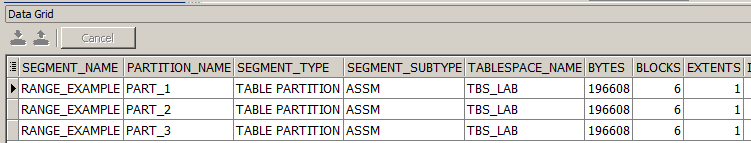
VALUES LESS THAN (TO\_DATE ('01/01/2014', 'dd/mm/yyyy')));

INSERT INTO range\_example SELECT DAY\_ID, DAY\_NAME

FROM TIMES WHERE rownum < 1000;

SELECT \* FROM range\_example;

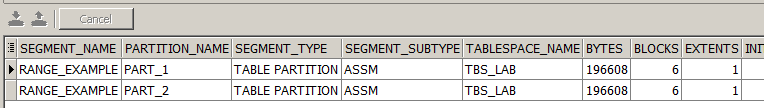
SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example')



*----------------------Dropping Partitions*

ALTER TABLE range\_example DROP PARTITION PART\_3

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example');

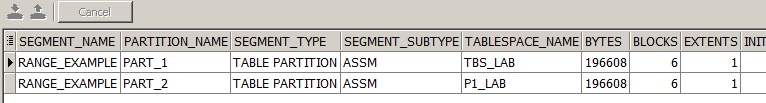


*----------------------Moving Partitions*

ALTER TABLE range\_example MOVE PARTITION PART\_2

TABLESPACE P1\_LAB NOLOGGING;

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example');



*----------------------Splitting Partitions*

ALTER TABLE range\_example SPLIT PARTITION

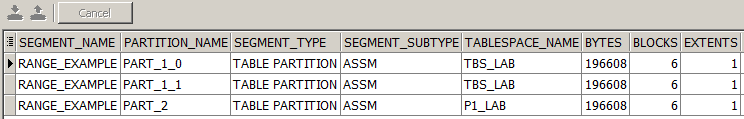
part\_1 at (TO\_DATE ('01/07/2011', 'dd/mm/yyyy')) INTO

( PARTITION part\_1\_0,

PARTITION part\_1\_1

);

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example')



*----------------------Merging Partitions*

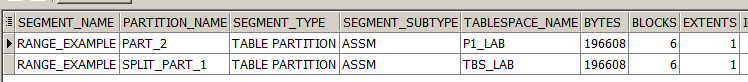
ALTER TABLE range\_example MERGE PARTITIONS

part\_1\_0, part\_1\_1

INTO PARTITION split\_part\_1

UPDATE INDEXES;

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example');



*----------------------Truncating Partitions*

alter table range\_example truncate partition split\_part\_1

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example')

*All data contained in split\_part\_1 have been truncated*

*-----------Interval Composite Interval-\**

*----------------------Adding Partitions*

*----------------------Dropping Partitions*

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('range\_example')

*----------------------Merging Partitions*

*----------------------Moving Partitions*

*----------------------Splitting Partitions*

*----------------------Truncating Partitions*

*-----------Hash*

*----------------------Adding Partitions*

CREATE TABLE hash\_example

(Date\_hash\_key date,

day\_name varchar2(36)

)

PARTITION BY HASH (Date\_hash\_key)

(

partition part\_1 tablespace p1\_lab,

partition part\_2 tablespace p2\_lab

);

INSERT INTO hash\_example

SELECT

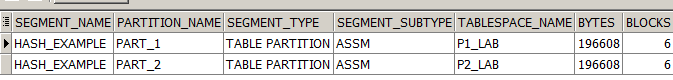
TO\_DATE(day\_id) hash\_key\_column, day\_name

FROM times

WHERE rownum < 1000;

select \* from hash\_example;

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('hash\_example')



*----------------------Coalescing Partitions*

ALTER TABLE hash\_example COALESCE PARTITION;

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('hash\_example')



*----------------------Moving Partitions*

*----------------------Truncating Partitions*

*-----------List Composite List-\**

*----------------------Adding Partitions*

CREATE TABLE list\_example (day\_id DATE, day\_name VARCHAR2 (36))

PARTITION BY LIST (day\_name)

(PARTITION part\_1

VALUES ('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'),

PARTITION part\_2

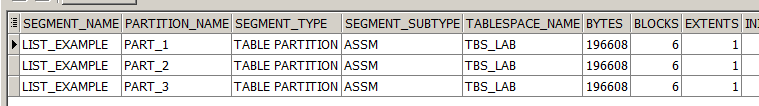
VALUES ('Saturday', 'Sunday'),

PARTITION part\_3

VALUES (DEFAULT));

INSERT INTO list\_example SELECT day\_id, day\_name FROM times;

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('list\_example')



*----------------------Dropping Partitions*

*----------------------Moving Partitions*

*----------------------Splitting Partitions*

ALTER TABLE list\_example SPLIT PARTITION part\_1 VALUES ( 'Friday', 'Thursday') INTO ( PARTITION part\_1\_0

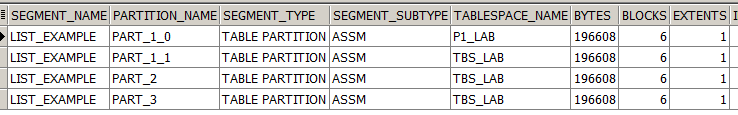
TABLESPACE p1\_lab,

PARTITION part\_1\_1

STORAGE (NEXT 5M PCTINCREASE 25))

PARALLEL 5;

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('list\_example')



*----------------------Merging Partitions*

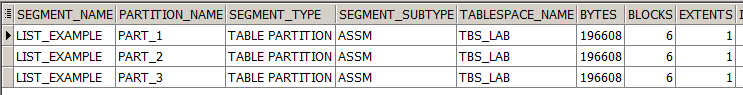
ALTER TABLE list\_example MERGE PARTITIONS

part\_1\_0, part\_1\_1

INTO PARTITION part\_1

STORAGE(MAXEXTENTS 20);

SELECT \* FROM user\_segments WHERE segment\_name = UPPER ('list\_example')



*----------------------Truncating Partitions*

*-----------Reference*

CREATE TABLE reference\_example

(

day\_id date,

day\_name VARCHAR2(26),

PRIMARY KEY (day\_id)

)

enable row movement

PARTITION BY RANGE (day\_id)

(

PARTITION part\_1 VALUES LESS THAN (TO\_DATE ('01/01/2009', 'dd/mm/yyyy')),

PARTITION part\_2 VALUES LESS THAN (TO\_DATE ('01/01/2012', 'dd/mm/yyyy')),

PARTITION part\_3 VALUES LESS THAN (TO\_DATE ('01/01/2015', 'dd/mm/yyyy'))

);

INSERT INTO reference\_example select day\_id, day\_name from times;

select \* from reference\_example;



CREATE TABLE reference\_example2

(

DAY\_NUMBER\_IN\_YEAR number,

day\_id date NOT NULL,

constraint c1\_pkey primary key(DAY\_NUMBER\_IN\_YEAR),

constraint c1\_fkey\_p foreign key(day\_id) references reference\_example (day\_id)

)

enable row movement

partition by reference(c1\_fkey\_p)

;

INSERT INTO reference\_example2

SELECT

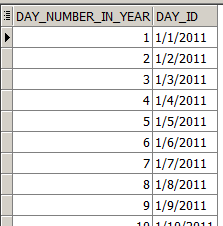
DAY\_NUMBER\_IN\_YEAR,

day\_id

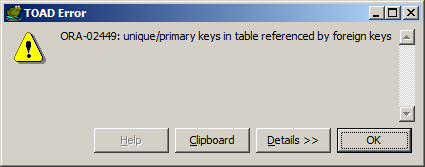
FROM times

WHERE rownum < 1000;

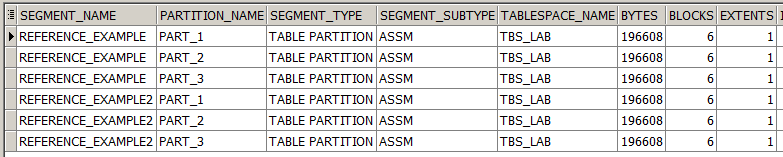
select \* from reference\_example2;



DROP TABLE reference\_example;



SELECT \* FROM user\_segments WHERE segment\_name in (UPPER ('reference\_example'), UPPER('reference\_example2'))



*----------------------Moving Partitions*

*----------------------Truncating Partitions*