

# **Teradata Bacis**

Lesson 07: Teradata Popular  
OLAP Examples

## Module Object

- To be familiar with popular OLAP functions.
- To be familiar with the PARTITION By concept.
- To be familiar with RANK() . ROW\_NUMBER(), QULIFY functions.

## Example Table "OLAP\_EXAMPLE"

```
SELECT * FROM DIM_ER_TRNG_DB.OLAP_EXAMPLE ;
```

TID: B I

	TransID	ItemID	YearNum	MonthNum	ProdID	Sales
1	1	E001	2011	1	P001	330
2	2	E001	2011	1	P002	150
3	3	E001	2011	1	P003	270
4	4	E001	2011	1	P004	210
5	5	V001	2011	1	P001	10
6	6	V001	2011	1	P002	1
7	7	V001	2011	1	P003	50
8	8	V001	2011	1	P004	50
9	9	N001	2011	1	P001	120
10	10	N001	2011	1	P002	130
11	11	N001	2011	1	P003	100
12	12	N001	2011	1	P004	90
13	13	S001	2011	1	P001	250
14	14	S001	2011	1	P002	250
15	15	S001	2011	1	P003	280
16	16	S001	2011	1	P004	300
17	17	E001	2011	2	P001	0
18	18	E001	2011	2	P002	0
19	19	E001	2011	2	P003	0
20	20	E001	2011	2	P004	0
21	21	V001	2011	2	P001	0
22	22	V001	2011	2	P002	100
23	23	V001	2011	2	P003	150
24	24	V001	2011	2	P004	100
25	25	N001	2011	2	P001	0
26	26	N001	2011	2	P002	90
27	27	N001	2011	2	P003	80
28	28	N001	2011	2	P004	50
29	29	S001	2011	2	P001	0
30	30	S001	2011	2	P002	30
31	31	S001	2011	2	P003	20
32	32	S001	2011	2	P004	30
33	33	E001	2011	3	P001	120
34	34	E001	2011	3	P002	1
35	35	E001	2011	3	P003	150
36	36	E001	2011	3	P004	180
37	37	V001	2011	3	P001	100
38	38	V001	2011	3	P002	0
39	39	V001	2011	3	P003	90
40	40	V001	2011	3	P004	120
41	41	N001	2011	3	P001	0
42	42	N001	2011	3	P002	0
43	43	N001	2011	3	P003	0
44	44	N001	2011	3	P004	0
45	45	S001	2011	3	P001	180
46	46	S001	2011	3	P002	0
47	47	S001	2011	3	P003	220
48	48	S001	2011	3	P004	250

## Usage of PARTITION by to derive RANK

How to find the rank for each product based on the sum sales value within each StoreID.

```
SEL STOREID, PRODID, SUM(SALES) SSales,
RANK() OVER (PARTITION BY STOREID ORDER BY SSales DESC) AS RNK
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE
GROUP BY STOREID, PRODID;
```

	StoreID	ProdID	SSALES	RNK
1	E001	P001	450	1
2	E001	P003	420	2
3	E001	P004	390	3
4	E001	P002	150	4
5	N001	P002		
6	N001	P003		
7	N001	P004		
8	N001	P001		
9	S001	P004		
10	S001	P003		
11	S001	P001		
12	S001	P002		
13	W001	P004		
14	W001	P003		
15	W001	P002		
16	W001	P001		

## Usage of RANK & QUALIFY

How to find the Rank#1 product based upon the sum sales value (desc) within each store.

	StoreID	Prod	SSALES	
1	E001	P001		1
2	N001	P002		1
3	S001	P004		
4	W001	P004		

```
SEL STOREID,PRODID, SUM(SALES) SSALES,  
RANK() OVER (PARTITION BY STOREID ORDER BY SSALES DESC) AS RNK  
FROM TD_BIM_FRL_TRNG_DB.OLAP_EXAMPLE  
GROUP BY STOREID, PRODID  
QUALIFY RNK = 1;
```

## Usage of ROWS BETWEEN (PRECEDING-FOLLOWING CURRENT ROW)

How to find the moving sum for each product including current row and 3 rows preceding.

```

-- find moving sum for each product including current row and 3 rows preceding.
SUM(SALES) OVER (PARTITION BY PRODID ORDER BY TRANID
ROWS BETWEEN 3 PRECEDING AND CURRENT ROW)
FROM TID_131.1.11_OLAP_EXAMPLE;
B.O.LA

```

RowID	ProdID	Sales	Moving Sum (Sales)
1	P001	330	330
2	P001	10	340
3	P001	120	460
4	P001		710
5	P001		360
6	P001	0	370
7	P001		250
8	P001		
9	P001		
10	P001		
11	P001		
12	P001		
13	P002		1500
14	P002		175
15	P002		1000
16	P002		1555
17	P002	0	
18	P002		
19	P002		
20	P002		
21	P002		2248
22	P002		238
23	P002	0	110
24	P002	0	35
25	P003	270	270
26	P003		320
27	P003	100	420
28	P003	280	700
29	P003	0	430
30	P003		530
31	P003		810
32	P003		250
33	P003		400
34	P003	50	340
35	P003	90	260
36	P003	230	460
37	P003	210	210
38	P004	30	290
39	P004	90	380
40	P004	300	680
41	P004	0	470
42	P004	100	490
43	P004	50	450
44	P004	30	180
45	P004	180	360
46	P004	120	380
47	P004	0	330
48	P004	250	

## Example Table "OLAP\_EXAMPLE\_CLASS"

```
SEL * FROM TD_BIM_FR_TRNG_DB.O _EXAMPLE_C S
```

	STUDID	STUDNAME	SUBID	MARKS
1	1	A	C	85
2	1	A	M	80
3	1	A	P	90
4	2	B	C	80
5	2	B	M	95
6	2	B	P	92
7	3	C	C	79
8	3	C	M	93
9	3	C	P	83
10	4	D	C	82
11	4	D	M	67
12	4	D	P	78
13	5	E	C	80
14	5	E	M	75
15	5	E	P	78
16	6	F	C	74
17	6	F	M	71
18	6	F	P	70
19	7	G	C	69
20	7	G	M	56
21	7	G	P	50
22	8	H	C	49
23	8	H	M	56
24	8	H	P	61
25	9	I	C	66
26	9	I	M	73
27	9	I	P	78
28	10	J	C	52
29	10	J	M	80
30	10	J	P	56

## Usage of RANK & PARTITION BY

How to find the RANK of each student on the subject based on the marks obtained.

	STUDID	STUDENT	MARKS	SUBJECT	Rank (MARKS)
1	1	A	85	C	1
2	4	D	82	C	2
3	2	B	80	C	3
4	5	E	80	C	3
5	3	C	79	C	5
6	6	F	74	C	
7	7	G	69	C	
8	9	I	66	C	
9	10	J	52	C	9
10	8	H	49	C	10
11	2	B	95	M	
12	3	C	93	M	2
13	1	A	80	M	
14	10	J	80	M	3
15	5	E	75	M	5
16	9	I	73	M	6
17	6	F	71	M	7
18	4	D	67	M	
19	8	H	56	M	9
20	7	G	56	M	9
21	2	B	92	P	
22	1	A	90	P	
23	3	C	88	P	3
24	4	D	78	P	4
25	5	E	78	P	4
26	9	I	78	P	4
27	6	F	70	P	7
28	8	H	61	P	8
29	10	J	56	P	9
30	7	G	50	P	10



## Usage of RANK & PARTITION BY

How to find the RANK of  
each student on each  
subject based on their  
marks.

```
SEL STUDID, STUDNAME, MARKS, SUBID,  
RANK() OVER ( PARTITION BY SUBID ORDER BY MARKS DESC)  
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE_CLASS ;
```

	STUDID	STUDNAME	MARKS	SUBID	Rank (MARKS)
1	1	A	85	C	1
2	4	D	82	C	2
3	2	B	80	C	3
4	5	E	80	C	3
5	3	C	79	C	5
6	6	F	74	C	6
7	7	G	69	C	7
8	9	I	66	C	8
9	10	J	52	C	9
10	8	H	49	C	10
11	2	B	95	M	1
12	3	C	93	M	2
13	1	A	80	M	3
14	10	J	80	M	3
15	5	E	75	M	5
16	9	I	73	M	6
17	6	F	71	M	7
18	4	D	67	M	8
19	8	H	56	M	9
20	7	G	56	M	9
21	2	B	92	P	1
22	1	A	90	P	2
23	3	C	83	P	3
24	4	D	78	P	4
25	5	E	78	P	4
26	9	I	78	P	4
27	6	F	70	P	7
28	8	H	61	P	8
29	10	J	56	P	9
30	7	G	50	P	10

## Usage of RANK, QUALIFY & PARTITION BY

How to find the students within Rank 3 for SUBID 'M'.

	STUDID	STUDNAME	MARKS	SUBID	Rank (MARKS)
1	2	B	95	M	1
2	3	C	93	M	2
3	1	A	80	M	3
4	10	J	80	M	3

## Usage of RANK, QUALIFY & PARTITION BY

How to find the students within Rank 3 for SUBID 'M'.

	STUDID	STUDNAME	MARKS	SUBID	Rank (MARKS)
1	2	B	95	M	1
2	3	C	93	M	2
3	1	A	80	M	3
4	10	J	80	M	3

```
SEL STUDID,STUDNAME,MARKS,SUBID,  
RANK() OVER ( PARTITION BY SUBID ORDER BY MARKS DESC)  
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE_CLASS  
QUALIFY RANK() OVER ( PARTITION BY SUBID ORDER BY MARKS DESC) <= 3  
AND SUBID = 'M' ;
```

## Usage of RANK, QUALIFY & PARTITION BY

How to find the student who has scored minimum in all 3 subjects.

	STUDID	STUDNAME	marks	RNK
1	8	H	166	10

## Usage of RANK, QUALIFY & PARTITION BY

How to find the student who has scored minimum in all 3 subjects.

	STUDID	STUDNAME	smarks	RNK
1	8	H	166	10

```
SEL STUDID, STUDNAME, SUM(MARKS) AS SMARKS,  
RANK () OVER ( ORDER BY SMARKS DESC) AS RNK  
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE_CLASS  
GROUP BY STUDID, STUDNAME  
QUALIFY RNK = 10 ;
```

## Usage of RANK, QUALIFY & PARTITION BY

How to find the change in RANKs for each student.

	STUDID	STUDNAME	SMARKS	RNK	CHANGE
1	2	B	267	1	1
2	1	A	255	2	-1
3	3	C	255	2	1
4	5	E	233	4	1
5	4	D	227	5	-1
6	9	I	217	6	3
7	6	F	215	7	-1
8	10	J	188	8	2
9	7	G	175	9	-2
10	8	H	166	10	-2

## Usage of RANK, QUALIFY & PARTITION BY

How to find the change in RANKs for each student.

	STUDID	STUDNAME	SMARKS	RNK	CHANGE
1	2	B	267	1	1
2	1	A	255	2	-1
3	3	C	255	2	1
4	5	E	233	4	1
5	4	D	227	5	-1
6	9	I	217	6	3
7	6	F	215	7	-1
8	10	J	188	8	2
9	7	G	175	9	-2
10	8	H	166	10	-2

```
SEL STUDID, STUDNAME, SUM(MARKS) AS SMARKS,
RANK () OVER ( ORDER BY SMARKS DESC) AS RNK,
(STUDID - RNK) AS CHANGE
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE_CLASS
GROUP BY STUDID, STUDNAME ;
```

## Usage of RANK & PARTITION BY

Find the Subtotal marks for students with STUDID 1 and 2.

```
SEL STUDID, STUDNAME, MARKS
FROM ID_BIM_FR_TANG_DB.OLAP_EXAMPLE_CLASS
WITH SUM(MARKS) <TITLE 'INDIVIDUAL_MARKS' > BY STUDID, STUDNAME
WHERE STUDID in ( 1, 2 ) ;
```

```
*** Query completed. 8 rows found. 3 columns returned.
*** Total elapsed time was 1 second.
```

STUDID	STUDNAME	MARKS
1	A	80
1	A	85
1	A	90
INDIVIDUAL_MARKS		255
2	B	95
2	B	80
2	B	92
INDIVIDUAL_MARKS		267

BTEQ -- Enter your SQL request or BTEQ command:



## Usage of RANK & PARTITION BY

Difference between RANK    ROW\_NUMBER.

```
SEL STUDID, STUDNAME, SUM(MARKS) AS SMARKS
RANK () OVER ( ORDER BY SMARKS DESC)
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE_CLASS
GROUP BY STUDID, STUDNAME ;
```

	STUDID	STUDNAME	SMARKS	Rank (SMARKS)
1	2	B	267	1
2	3	C	255	2
3	1	A	255	2
4	5	E	233	4
5	4	D	227	5
6	9	I	217	6
7	6	F	215	7
8	10	J	188	8
9	7	G	175	9
10	8	H	166	10

```
SEL STUDID, STUDNAME, SUM(MARKS) AS SMARKS,
ROW_NUMBER () OVER ( ORDER BY SMARKS DESC)
FROM TD_BIM_FR_TRNG_DB.OLAP_EXAMPLE_CLASS
GROUP BY STUDID, STUDNAME ;
```

	STUDID	STUDNAME	SMARKS	Row_Number ()
1	2	B	267	1
2	1	A	255	2
3	3	C	255	3
4	5	E	233	4
5	4	D	227	5
6	9	I	217	6
7	6	F	215	7
8	10	J	188	8
9	7	G	175	9
10	8	H	166	10

## Q&A

- 1. Which two partitioning expressions are available to both single-level and multi-level partitioned tables? (Choose two.)
  - A. MODULO\_N partitioning
  - B. CASE\_N partitioning
  - C. RANGE\_N partitioning
  - D. Direct partitioning on a numeric column
- 2. What are two reasons a customer would choose table partitioning? (Choose two.)
  - A. to improve performance of full table scans
  - B. to reduce the I/O for range constrained queries
  - C. for better distribution of data between the AMPs
  - D. for the ability to archive specific partitions in a table



## Q&A

- 3. On which two table types can a multi-level partitioned primary index (MLPPI) be created? (Choose two.)
  - A. Volatile tables
  - B. Derived tables
  - C. Global temporary tables
  - D. Compressed join indexes
  
- 4. Which two table types support multi-level partitioned primary indexes (MLPPI)? (Choose two.)
  - A. Base tables
  - B. Compressed Join Indexes
  - C. Global temporary Trace tables
  - D. Non-compressed join indexes

