Mittwoch, 26. Mai 2021 16:56

Aggregation Functions

Dienstag, 1. Juni 2021 13:53

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
	101	11 Patil	2000-05-01 00:00:00	180000	(null)
	102	12 Durmaz	2005-07-01 00:00:00	120000	(null)
	103	13 Blaschke	2002-11-01 00:00:00	93000	(null)
	104	13 Stone	2006-06-01 00:00:00	42000	(null)
	105	13 Dalal	2018-02-02 00:00:00	38000	1000
	106	14 Li	2002-12-01 00:00:00	89000	(null)
	107	14 Nguyen	2006-07-01 00:00:00	41000	(null)
	108	14 Sanchez	2014-04-01 00:00:00	39000	1500
	109	15 Umarani	2006-07-01 00:00:00	142000	(null)
	110	16 Ortega	2005-09-02 00:00:00	90000	(null)
	111	16 Doshi	2010-01-02 00:00:00	42000	(null)
	112	16 Singh	2012-03-01 00:00:00	43000	2100
	113	16 Jadhav	2001-08-01 00:00:00	91000	(null)
	114	17 Popov	2009-03-02 00:00:00	34000	(null)
	115	17 Kumar	2013-05-01 00:00:00	32000	(null)
	116	17 Krause	2011-08-01 00:00:00	31000	(null)
	117	17 Oezdem	2014-08-01 00:00:00	33000	1900
	118	17 Okeke	2013-11-01 00:00:00	32000	1900

select

sum(salary),
round(avg(salary)),
min(salary),
max(salary),
count(*),
count(bonus)
from employee;

*	SUM(SALARY)	ROUND(AVG(SALARY))	MIN(SALARY)	MAX(SALARY)	COUNT(*)	COUNT(BONUS)	
1	1212000	67333	31000	180000	1	8	5

Grouping

Dienstag, 1. Juni 2021 13:53

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
	101	11 Patil	2000-05-01 00:00:00	180000	(null)
	102	12 Durmaz	2005-07-01 00:00:00	120000	(null)
	103	13 Blaschke	2002-11-01 00:00:00	93000	(null)
	104	13 Stone	2006-06-01 00:00:00	42000	(null)
	105	13 Dalal	2018-02-02 00:00:00	38000	1000
	106	14 Li	2002-12-01 00:00:00	89000	(null)
	107	14 Nguyen	2006-07-01 00:00:00	41000	(null)
	108	14 Sanchez	2014-04-01 00:00:00	39000	1500
	109	15 Umarani	2006-07-01 00:00:00	142000	(null)
	110	16 Ortega	2005-09-02 00:00:00	90000	(null)
	111	16 Doshi	2010-01-02 00:00:00	42000	(null)
	112	16 Singh	2012-03-01 00:00:00	43000	2100
	113	16 Jadhav	2001-08-01 00:00:00	91000	(null)
	114	17 Popov	2009-03-02 00:00:00	34000	(null)
	115	17 Kumar	2013-05-01 00:00:00	32000	(null)
	116	17 Krause	2011-08-01 00:00:00	31000	(null)
	117	17 Oezdem	2014-08-01 00:00:00	33000	1900
	118	17 Okeke	2013-11-01 00:00:00	32000	1900

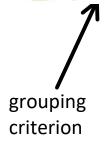
select ouid,
sum(salary),
round(avg(salary)),
min(salary),
max(salary),
count(*),
count(bonus)
from employee
group by ouid
order by ouid;

* 0	DUID	SUM(SALARY)	ROUND(AVG(SALARY))	MIN(SALARY)	MAX(SALARY)	COUNT(*)	COUNT(BONUS)
1	1	1 180000	180000	180000	180000	1	0
2	12	2 120000	120000	120000	120000	1	0
3	13	3 173000	57667	38000	93000	3	1
4	14	4 169000	56333	39000	89000	3	1
5	1:	5 142000	142000	142000	142000	1	0
6	10	266000	66500	42000	91000	4	1
7	1	7 162000	32400	31000	34000	5	2

Grouping - Conceptual Level

Dienstag, 1. Juni 2021 14:24

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
	101	11 Patil	2000-05-01 00:00:00	180000) (nul
	102	12 Durmaz	2005-07-01 00:00:00	120000) (nul
	103	13 Blaschke	2002-11-01 00:00:00	93000) (nul
	104	13 Stone	2006-06-01 00:00:00	42000) (nul
	105	13 Dalal	2018-02-02 00:00:00	38000) 100
	106	14 Li	2002-12-01:00:00:00	89000	(nul
	107	14 Nguyen	2006-07-01-00:00:00	41000	mel
	108	14 Sanchez	2014-04-01 00:00:00	39000) 150
	109	15 Umarani	2006-07-01 00:00:00	142000) (nul
	110	16 Ortega	2005-09-02-00:00:00	90000	(nul
	111	16 Doshi	2010-01-02 00:00:00	42000)— (nej
	112	16 Singh	2012-03-01 00:00:00	43000	210
	113	16 Jadhav	2001-08-01 00:00:00	91000) (nul
	114	17 Popov	2009-03-02 00:00:00	34000) (nui
	115	17 Kumar	2013-05-01-00:00:00	32000)(nul
	116	17 Krause	2011-08-01 00:00:00	31000	(nul
	117	17 Oezdem	2014-08-01 00:00:00	33000	190
	118	17 Okeke	2013-11-01 00:00:00	32000	190



- 1. partitioning of detail data into different groups according to grouping criterion
- 2. application of aggregation function on each individual group
- 3. one record per group in the output

Aggregation Levels

Dienstag, 1. Juni 2021 14

select ouid,
 sum(salary)
from employee
group by ouid
order by ouid;

all output columns on same aggregation level

ouid	part of group by
salar	aggregation function applied on column

* 01	JID SUM(S	ALARY)
1	11	180000
2	12	120000
3	13	173000
4	14	169000
5	15	142000
6	16	266000
7	17	162000

select ouid, lastname, sum(salary) from employee group by ouid order by ouid;

output columns on different aggregation levels

ouid	aggregated
lastname	on detail level (not possible)
salary	aggregated

error message: not a GROUP BY expression

Grouping and Join

Dienstag, 1. Juni 2021

EID OUID	LASTNAME	HIREDATE	SALARY	BONUS
101	11 Patil	2000-05-01 00:00:00	180000	(null)
102	12 Durmaz	2005-07-01 00:00:00	120000	(null)
103	13 Blaschke	2002-11-01 00:00:00	93000	(null)
104	13 Stone	2006-06-01 00:00:00	42000	(null)
105	13 Dalal	2018-02-02 00:00:00	38000	1000
106	14 Li	2002-12-01 00:00:00	89000	(null)
107	14 Nguyen	2006-07-01 00:00:00	41000	(null)
108	14 Sanchez	2014-04-01 00:00:00	39000	1500
109	15 Umarani	2006-07-01 00:00:00	142000	(null)
110	16 Ortega	2005-09-02 00:00:00	90000	(null)
111	16 Doshi	2010-01-02 00:00:00	42000	(null)
112	16 Singh	2012-03-01 00:00:00	43000	2100
113	16 Jadhav	2001-08-01 00:00:00	91000	(null)
114	17 Popov	2009-03-02 00:00:00	34000	(null)
115	17 Kumar	2013-05-01 00:00:00	32000	(null)
116	17 Krause	2011-08-01 00:00:00	31000	(null)
117	17 Oezdem	2014-08-01 00:00:00	33000	1900
118	17 Okeke	2013-11-01 00:00:00	32000	1900

OUID	HEAD	SUPERUNIT	NAME
1	101	(null)	Company
12	2 102	2 11	Administration
13	3 103	3 12	HR
14	1 106	5 12	Accounting
15	109	11	Production
16	110) 15	Plant
17	7 109	15	Warehouse

select ou.name,
 sum(salary),
 count(*)
from employee e
 join orgunit ou on ou.ouid=e.ouid
group by ou.name
order by ou.name;

*	NAME	SUM(SALARY)	COUNT(*)
1	Accounting	169000	3
2	Administration	120000	1
3	Company	180000	1
4	HR	173000	3
5	Plant	266000	4
6	Production	142000	1
7	Warehouse	162000	5

Grouping with Expressions 1

Dienstag, 1. Juni 2021 14:5

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
10	1	11 Patil	2000-05-01 00:00:00	180000	(null)
10	2	12 Durmaz	2005-07-01 00:00:00	120000	(null)
10	3	13 Blaschke	2002-11-01 00:00:00	93000	(null)
10	4	13 Stone	2006-06-01 00:00:00	42000	(null)
10	5	13 Dalal	2018-02-02 00:00:00	38000	1000
10	6	14 Li	2002-12-01 00:00:00	89000	(null)
10	7	14 Nguyen	2006-07-01 00:00:00	41000	(null)
10	8	14 Sanchez	2014-04-01 00:00:00	39000	1500
10	9	15 Umarani	2006-07-01 00:00:00	142000	(null)
110	0	16 Ortega	2005-09-02 00:00:00	90000	(null)
11	1	16 Doshi	2010-01-02 00:00:00	42000	(null)
113	2	16 Singh	2012-03-01 00:00:00	43000	2100
11.	3	16 Jadhav	2001-08-01 00:00:00	91000	(null)
114	4	17 Popov	2009-03-02 00:00:00	34000	(null)
11.	5	17 Kumar	2013-05-01 00:00:00	32000	(null)
110	6	17 Krause	2011-08-01 00:00:00	31000	(null)
11	7	17 Oezdem	2014-08-01 00:00:00	33000	1900
118	8	17 Okeke	2013-11-01 00:00:00	32000	1900

select

extract(year from hiredate) as hireyear, sum(salary), count(*)
from employee
group by extract(year from hiredate)
order by extract(year from hiredate);

* HIRE	EYEAR SUM(S	SALARY) COUNT	Γ(*)
1	2000	180000	1
2	2001	91000	1
3	2002	182000	2
4	2005	210000	2
5	2006	225000	3
6	2009	34000	1
7	2010	42000	1
8	2011	31000	1
9	2012	43000	1
10	2013	64000	2
11	2014	72000	2
12	2018	38000	1

Grouping with Expressions 2

Dienstag, 1. Juni 2021 14:57

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
	101	11 Patil	2000-05-01 00:00:00	180000	(null)
	102	12 Durmaz	2005-07-01 00:00:00	120000	(null)
	103	13 Blaschke	2002-11-01 00:00:00	93000	(null)
	104	13 Stone	2006-06-01 00:00:00	42000	(null)
	105	13 Dalal	2018-02-02 00:00:00	38000	1000
	106	14 Li	2002-12-01 00:00:00	89000	(null)
	107	14 Nguyen	2006-07-01 00:00:00	41000	(null)
	108	14 Sanchez	2014-04-01 00:00:00	39000	1500
	109	15 Umarani	2006-07-01 00:00:00	142000	(null)
	110	16 Ortega	2005-09-02 00:00:00	90000	(null)
	111	16 Doshi	2010-01-02 00:00:00	42000	(null)
	112	16 Singh	2012-03-01 00:00:00	43000	2100
	113	16 Jadhav	2001-08-01 00:00:00	91000	(null)
	114	17 Popov	2009-03-02 00:00:00	34000	(null)
	115	17 Kumar	2013-05-01 00:00:00	32000	(null)
	116	17 Krause	2011-08-01 00:00:00	31000	(null)
	117	17 Oezdem	2014-08-01 00:00:00	33000	1900
	118	17 Okeke	2013-11-01 00:00:00	32000	1900

```
select
case
  when hiredate >= to_date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to_date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
end as entry_year_range,
sum(salary),
count(*)
from employee
group by
case
  when hiredate >= to_date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to_date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
end;
```

*	ENTRY_YEAR_RANGE	SUM(SALARY)	COUNT(*)
1	From2010	290000	8
2	Until2009	922000	10

Grouping with several Columns

Dienstag, 1. Juni 2021 1

- 4	Λ		$\overline{}$	\neg	
- 1	Д	۰	ה	/	

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
	101	11 Patil	2000-05-01 00:00:00	180000	(null)
	102	12 Durmaz	2005-07-01 00:00:00	120000	(null)
	103	13 Blaschke	2002-11-01 00:00:00	93000	(null)
	104	13 Stone	2006-06-01 00:00:00	42000	(null
	105	13 Dalal	2018-02-02 00:00:00	38000	1000
	106	14 Li	2002-12-01 00:00:00	89000	(null
	107	14 Nguyen	2006-07-01 00:00:00	41000	(null)
	108	14 Sanchez	2014-04-01 00:00:00	39000	1500
	109	15 Umarani	2006-07-01 00:00:00	142000	(null
	110	16 Ortega	2005-09-02 00:00:00	90000	(null
	111	16 Doshi	2010-01-02 00:00:00	42000	(null
	112	16 Singh	2012-03-01 00:00:00	43000	2100
	113	16 Jadhav	2001-08-01 00:00:00	91000	(null
	114	17 Popov	2009-03-02 00:00:00	34000	(null)
	115	17 Kumar	2013-05-01 00:00:00	32000	(null
	116	17 Krause	2011-08-01 00:00:00	31000	(null
	117	17 Oezdem	2014-08-01 00:00:00	33000	1900
	118	17 Okeke	2013-11-01 00:00:00	32000	1900

* OUID		SUM(SALARY)	COUNT(*)
1	11 Until2009	180000	1
2	12 Until2009	120000	1
3	13 From2010	38000	1
4	13 Until2009	135000	2
5	14 From2010	39000	1
6	14 Until2009	130000	2
7	15 Until2009	142000	1
8	16 From2010	85000	2
9	16 Until2009	181000	2
10	17 From2010	128000	4
11	17 Until2009	34000	1

```
select ouid,
 case
  when hiredate >= to_date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
 end as entry year range,
 sum(salary),
 count(*)
from employee
group by
 ouid,
 case
  when hiredate >= to_date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to_date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
 end
order by ouid;
```

```
Mittwoch, 25. August 2021 09:26
```

```
select
 case
  when hiredate >= to_date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to_date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
end as entry year range,
sum(salary),
count(*)
from employee
group by
 case
  when hiredate >= to date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to_date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
end;
```

*	ENTRY_YEAR_RANGE	SUM(SALARY)	COUNT(*)
1	From2010	290000	8
2	Until2009	922000	10

adding a column in **group by** leads to new groups, due to value combinations

```
select ouid,
 case
  when hiredate >= to date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
 end as entry_year_range,
 sum(salary),
count(*)
from employee
group by
 ouid,
 case
  when hiredate >= to_date('2010/01/01', 'YYYY/MM/DD')
    then 'From2010'
  when hiredate >= to date('2000/01/01', 'YYYY/MM/DD')
    then 'Until2009'
 end
order by ouid;
```

* OU	ID 📝	ENTRY_YEAR_RANGE	SUM(SALARY)	COUNT(*)
1	11	Until2009	180000	1
2	12	2 Until2009	120000	1
3	13	From2010	38000	1
4	13	Until2009	135000	2
5	14	From2010	39000	1
6	14	1 Until2009	130000	2
7	15	Until2009	142000	1
8	16	From2010	85000	2
9	16	Until2009	181000	2
10	17	7 From2010	128000	4
11	17	7 Until2009	34000	1

Grouping with dependent Columns

Dienstag, 1. Juni 2021

L4:58

EID	OUID	LASTNAME	HIREDATE	SALARY	BONUS
	101	11 Patil	2000-05-01 00:00:00	180000	(null)
	102	12 Durmaz	2005-07-01 00:00:00	120000	(null
	103	13 Blaschke	2002-11-01 00:00:00	93000	(null)
	104	13 Stone	2006-06-01 00:00:00	42000	(null)
	105	13 Dalal	2018-02-02 00:00:00	38000	1000
	106	14 Li	2002-12-01 00:00:00	89000	(null
	107	14 Nguyen	2006-07-01 00:00:00	41000	(null)
	108	14 Sanchez	2014-04-01 00:00:00	39000	1500
	109	15 Umarani	2006-07-01 00:00:00	142000	(null)
	110	16 Ortega	2005-09-02 00:00:00	90000	(null
	111	16 Doshi	2010-01-02 00:00:00	42000	(null
	112	16 Singh	2012-03-01 00:00:00	43000	2100
	113	16 Jadhav	2001-08-01 00:00:00	91000	(null)
	114	17 Popov	2009-03-02 00:00:00	34000	(null
	115	17 Kumar	2013-05-01 00:00:00	32000	(null)
	116	17 Krause	2011-08-01 00:00:00	31000	(null
	117	17 Oezdem	2014-08-01 00:00:00	33000	1900
	118	17 Okeke	2013-11-01 00:00:00	32000	1900

OUID	HEAD	SUPERUNIT	NAME
11	101	(null)	Company
12	102	11	Administration
13	103	12	HR
14	106	12	Accounting
15	109	11	Production
16	110	15	Plant
17	109	15	Warehouse

not possible, name is on detail level error message: no GROUP BY expression

select ou.ouid, ou.name, sum(salary)
from employee e
 join orgunit ou on ou.ouid=e.ouid
group by ou.ouid
order by ou.ouid;

add ou.name to **group by** doesn't lead to new combinations because ou.name is dependent on ou.ouid

select ou.ouid, ou.name, sum(salary)
from employee e
 join orgunit ou on ou.ouid=e.ouid
group by ou.ouid, ou.name
order by ou.ouid;

*	OUID	NAME	SUM(SALARY)
1		11 Company	180000
2		12 Administration	120000
3		13 HR	173000
4		14 Accounting	169000
5		15 Production	142000
6		16 Plant	266000
7		17 Warehouse	162000

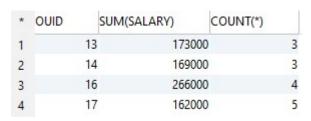
Conditions on Group Level

Mittwoch, 2. Juni 2021 09:28

select ouid,
 sum(salary),
 count(*)
from employee
group by ouid
order by ouid;

* OUI	D SUM(S	ALARY) COUNT(*)
	11	180000	1
2	12	120000	1
3	13	173000	3
1	14	169000	3
5	15	142000	1
5	16	266000	4
,	17	162000	5

select ouid,
 sum(salary),
 count(*)
from employee
group by ouid
having count(*) > 1
order by ouid;



Conditions on Detail Level and Group Level

Mittwoch, 2. Juni 2021 09:28

S	SALARY BON	HIREDATE	LASTNAME	OUID
(null	180000	2000-05-01 00:00:00	11 Patil	101
(null	120000	2005-07-01 00:00:00	12 Durmaz	102
(null	93000	2002-11-01 00:00:00	13 Blaschke	103
(null	42000	2006-06-01 00:00:00	13 Stone	104
100	38000	2018-02-02 00:00:00	13 Dalal	105
(null	89000	2002-12-01 00:00:00	14 Li	106
(null	41000	2006-07-01 00:00:00	14 Nguyen	107
150	39000	2014-04-01 00:00:00	14 Sanchez	108
(null	142000	2006-07-01 00:00:00	15 Umarani	109
(null	90000	2005-09-02 00:00:00	16 Ortega	110
(null	42000	2010-01-02 00:00:00	16 Doshi	111
210	43000	2012-03-01 00:00:00	16 Singh	112
(null	91000	2001-08-01 00:00:00	16 Jadhav	113
(null	34000	2009-03-02 00:00:00	17 Popov	114
(null	32000	2013-05-01 00:00:00	17 Kumar	115
(null	31000	2011-08-01 00:00:00	17 Krause	116
190	33000	2014-08-01 00:00:00	17 Oezdem	117
190	32000	2013-11-01 00:00:00	17 Okeke	118

select ouid,
 sum(salary),
 count(*)
from employee
group by ouid
having count(*) > 1
order by ouid;

*	OUID	SUM(SALARY)	COUNT(*)
1	13	173000	3
2	14	169000	3
3	16	266000	4
4	17	162000	5

select ouid,
 sum(salary),
 count(*)
from employee
where bonus is null
group by ouid
having count(*) > 1
order by ouid;

*	OUID	SUM(SALARY)	COUNT(*)
1	13	135000	2
2	14	130000	2
3	16	223000) 3
4	17	97000	3

Structure of a Query with Grouping

Dienstag, 3. Dezember 2019

09:55

select from where group by having order by