

Date :

No:

SESSION 6 (1) 02/10/1998 (C)

II GROUP BY

بعض الـ Dept ID و Salary لـ Instructor

INSTRUCTORS

Department ID + Instructor فتحت في
بعض Department و Instructor

Select DeptId, Min(Salary)
From INSTRUCTOR where DeptId IS
Group By DeptId IS notnull
result of Group By II على الأقل

Result → DeptId Min
DeptId Salary (10) → 1000
و DeptId سطوة كل

Condition to choose column to
Grouping Based It

Groups تكون بذكر معانٍ أكثر Column يلزم
*, PR و تكون مختلفة في كل من 03

Ex

Select Item, Count (*)

From Table

Group By Item

الوارد الموجودة في Items

Group II

الوارد

Having // Condition on Grouping
Can be using aggregate

Date : _____
Q → Get # of students IS study
in each Department

Select Dept-Id, Count(*) as name
of Column

From Student
where Dept-Id is not null
Grouping By Dept-Id

* Aggregate Fun not used in
Clause where \rightarrow not Valid
row by row Table \rightarrow where is
Per Table exists 3 solution aggregate &
having

① used to check on Groups out
From Grouping BY.

② When you ~~don't~~ want used
aggregate Fun without GroupBY

↓ example for calculate sum of
Salary when # of instructor
is less than 10

Select Sum(Salary)

From INSTRUCTOR

having Count(*) $<=$ 10

Table \rightarrow if it has one
ID as one group

Date:

Q1 If you select item with aggregate function and it's ~~not~~ not result
Put in Group BY

Select Dept_id, st_address,
Count(*) as
From Student

Where Dept_id is not null,
st_address is not null

Group BY Dept_id, st_address

فقط في جدول واحد في Group BY

لأن Two Columns will be grouped (Dept_id + st_address)

① Dept_id Dept_id + st_address Group
وكان
وستكون Sub Group because of Groups elements

Groups

① Dept_id N 10 , Address = 'Cairo' ← address

② " " 10 , " = 'Alex'

③ " N-Null , " = 'Cairo'

Dept number

st_address

10 → Cairo

20 → Alex

30 → Null

Null

Date : _____
Note : _____

Then check if Group have data
because Count() return Group
have data

* If you have Two Solution
one → using join, sec → by Grouping
using second → join(\rightarrow) is joined 1st &
Table is data \rightarrow join P is joined to
the all is (1)

Ex1 اكتب اجابة كل ما في السؤال واعذر المطلاع
Select Std. St-Fname Count (*)
From Student Std, Student Super
Where Std. St-Super = Super. St-Id
Group By Super. St-Fname

* Sub Query Slow بطيء جداً
is a query which runs inside another query to fetch data from the database.
Main query is called Outer query and subquery is called Inner query.
Resultset of subquery is used in outer query using IN operator.

Date :

Ex For Subquery

- Outer query

Select St-Id, St-Fname, St-Age

From Student

Where St-Age > ~~Select~~ ⁱⁿ Result

(Select Avg(St-Age)

Subquery] From Student)

another example

Select St-Id, Count

(Select Count(*) From)
Student

From Student

* Subquery with DML

① Subquery with delete

Delete

From Student-Course

where St-Id in (Select St-Id

From Student
Where

St-Address
= 'Cairo')

This example can execute
by join

Date :

*TOP (Keyword)

Select TOP(2) *
 From Student

→ all data
 باعث اول اثنين

Select TOP(2) st-ID, st-Fname
 From Student

لهم اخراج اربع اجيبي اخر اثنين
 *last Two Students

Select TOP(2)
 From Student
 Order By st-ID DESC

الاول هويتهم بالعكس ↓

* IF You want Select Instructor which
have Max Salary

TOP(1)

Select ins-name, ins-ID, salary
 From Instructor
 Order By salary DESC

Select Second Max Salary

Select Max(Salary) as secondSalary
 From Instructor
 Where Salary != (Select Max(Salary)
 From Instructor)

Date :

* TOP [US] TOP with Ties

TOP

Age

| |
|----|
| 30 |
| 29 |
| 28 |
| 27 |
| 26 |
| 26 |
| 24 |

Select TOP(5) St-Age
From Student
Order By St-Age Desc

return

TOP with Ties

Select TOP(5) Withties
St-Age

From Student

Order By St-Age Desc

order By TOPwith ties

ASC, DESC Ties

Return 5 ملحوظات
أكبر 5 ملحوظات
values الـ 5 elements
list checked list

* Random Selection

Global universal ID

Select NewId() -- GUID

This runs ← run 3000s → unique

ASPX

* Select S+ Fname, NewId()
From Student

* Select TOP(2) *

From Student

Order BY NewId()

Get 2 rows
1st row J5
2nd row J6

1st row Random

2nd row Random

Ex For second Max by TOP result sets

Select TOP(1) *

From(Select TOP(2) ins_id, ins_name
From INSTRUCTOR Order By desc)
Order By Salary

as #

* Offset

using to offset

↳ Offset ~~it specifies # of rows to skip before starting to return from result set.~~

Ex Select Column 1, Column 2

From Your Table

Order By specific columns

Offset 20

You will get 20 rows after skipping 20 rows

Ex

Create Table Orders

(
Id Varchar(32) Default NewId()
Primary Key

Total decimal(9,2))

Date:

No:

* Ranking Function

- ① Row-Number() Take No Parameter
Deal with Row by Row
(Sequence)

Select ins_id, ins_name, salary,

Row-Number() over (order By salary)

From INSTRUCTOR

Desc)

as RN

- ② Dense-Rank() No Take Parameter
Deal with Row by Row
(sequence)

Select ins_id, ins_name, salary,

Dense-Rank() over (order By salary Des

From INSTRUCTOR

as DR

- ③ Rank() No Take Parameter
Deal with Row by Row
(sequence)

Select ins_id, ins_name, salary,

Rank() over (order By salary Desc)

as R_R

From INSTRUCTOR

| EId | Ename | salary | Address | D-Id | RN | DR | Rank(%) |
|-----|---------|--------|---------|------|----|----|---------|
| 1 | ahmed | 10,000 | Cairo | 10 | 1 | 1 | 1 |
| 2 | ali | 10,000 | Cairo | 10 | 2 | 1 | 1 |
| 3 | emeen | 9000 | Cairo | 10 | 3 | 2 | 3 |
| 4 | khald | 9000 | alex | 10 | 4 | 2 | 3 |
| 5 | Seamn | 8000 | alex | 10 | 5 | 3 | 5 |
| 6 | Youssif | 8000 | alex | 10 | 6 | 3 | 5 |
| 7 | alaia | 7000 | alex | 20 | 7 | 4 | 7 |
| 8 | Mohame | 7000 | alex | 20 | 8 | 4 | 7 |
| 9 | Naeem | 6000 | Cairo | 20 | 9 | 5 | 9 |
| 10 | ola | 6000 | Cairo | 20 | 10 | 5 | 9 |

Date : _____

① Get 2 older students at Students Table

-- Using TOP (Best Practise in This Case)

```
Select Top(2) St_Id, Fname, Age  
From Student  
orderBy Age Desc
```

-- Using Ranking

```
Select * From  
(Select St_Id, Fname, Age,  
Row_Number() over(Order By Age Desc)  
From Student ) # ASRN  
Where RN in (1,2)
```

② Get The 5th younger student

-- Using TOP (Best : result of subquery) ↑ 5 rows

```
Select Top(1) *  
From (Select Top(5) From Student  
orderBy Age) as #  
orderBy Age Desc
```

-- Using Ranking (- Row_Number())

```
Select * From (result of subquery)  
(Select St_Id, St_name, Tvalue)  
Row_Number() over(Order By Age)  
From Student Where Age is not null  
and age  
Where RN = 5
```

① Get Student # 5 امتحانات
--- using TOP (Best Practice) (جواب افضل)

Select TOP(1)* with ties

from (select TOP(5)* with ties

From Student

order BY Age) AS T

order BY Age Desc

↳ Can solve by using Dense_Rank()
or Rank() مُؤكدة بـ الـ TIES (في الـ US)

Select * From

(Select St_Id, St_name, St_age,

Row_Number() over (Partition By DeptId
order By St_Age)

From Student

Where (DeptId is not null and
age is not null) AS T

Where Rank = 1

* Get The younger student at each
Department

مُؤكدة بـ DeptId rank ليس بـ DeptId assumed
پس DeptId لـ insert

Select Minimum Age in Department

Select Dept_Id, Min(Age)

From Student

Where Dept_Id is not null

Group By Dept_Id

Dates
Rank Type

④ NTile (int) Take Parameter \rightarrow one \uparrow int
Deal with Groups \rightarrow ~~Partitions~~
int # \rightarrow ~~total number of rows~~ \rightarrow ~~number of levels~~ \rightarrow ~~number of groups~~

Select ins_id, ins_name, salary
NTile(3) over (order by salary Desc)
From Instructor

If you have Select

Select * From

(Select ins_id, ins_name, salary,
NTile(3) over (order by salary Desc)
as LevelNumber

From Instructor) as t
Where LevelNumber = 1