

```

create table emp
(
    eid int Primary key,
    ename varchar(20) not null,
    eage int,
    eadd varchar(20) default 'cairo',
    hiredate date default getdate(),
    Dnum int
)

```

`alter table emp add sal int` لو عايزة ازود column

`alter table emp alter column sal bigint` لو عايزة اعدل ال data type بقى column

`alter table emp drop column sal` لو عايزة امسح column

`drop table emp` لو عايزة اشيل ال table بال structure

```

--DML
--insert update delete
insert into emp
values(1,'ali',NULL,'alex','1/1/2010',NULL)

insert into emp(ename,eid)
values('eman',2)

```

لو عايزة ت insert values في كل ال columns

لو عايزة ت insert values في columns معينة

	eid	ename	eage	eadd	hiredate	Dnum
1	1	ali	NULL	alex	2010-01-01	NULL
2	2	eman	NULL	cairo	2020-11-02	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL

طيب ينفع أ ؟ Query في نفس ال Row لكان insert values - اه ينفع

```
--insert constructor
insert into emp(ename,eid)
values('eman',8),('ali',12),('nada',7)
```

(3 row(s) affected)

```
update emp
    set ename='omar'
where eid=1
```

هنا هو بـ update اسم ال employee اللي ال ID بتاعة 1

```
update emp
    set ename='omar',eage=22
where eid=1
```

هنا هو بـ update اسم و سن ال employee اللي ال ID بتاعة 1

```
update emp
    set eage+=1
```

هنا هو عايز يزود سنة علي ال age بتاع كل موظف

```
delete from emp
```

هنا بن keep لكن بن ال delete employee table

```
delete from emp
where eid=1
```

هنا بن delete بس ال Row اللي ال ID بتاعة 1

```
alter table emp drop column sal
```

لو عايز امسح column

```
update emp
    set ename=NULL
```

هنا بيخللي كل أسماء الموظفين ب Null

```
drop table emp
```

هنا بن delete وكمان بن delete table بتاع ال structure

```
--DQL  
select * from Student
```

```
select st_id,st_fname from Student
```

```
select st_id,st_fname from Student  
where st_age>=25
```

```
select * from Student  
order by st_age desc
```

```
select st_fname+' '+st_lname as fullname  
from student
```

```
select *  
from Student  
where st_fname is NULL
```

```
select *  
from Student  
where st_fname is not NULL
```

```
select *  
from Student  
where st_fname is not NULL and st_lname is not null
```

```
select distinct st_fname  
from Student
```

```
select *  
from student  
where st_address='mansoura' and st_Address='alex'
```

```
select *  
from student  
where st_address='mansoura' or st_Address='alex'
```

==

```
select *  
from student  
where st_address in('cairo','alex','mansoura')
```

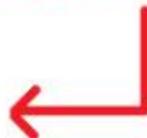
```
select *  
from student  
where st_address='mansoura' and st_age>=25
```

```
select *\nfrom student\nwhere st_age>=23 and st_age<=25
```

=

كده ال رقمين كمان معانا مش اللي بينهم بس

```
select *\nfrom Student\nwhere st_age between 23 and 25
```



Types of joins

--Cross join

-----Cartesian Product

--Inner Join

-----Equi Join

--Outer Join

-----Left Outer join

-----Right Outer Join

-----Full Outer Join

--Self Join

(Unary relationship)

EX:-

Student

<u>Sid</u>	Sname	did
1	ahmed	10
2	khalid	10
3	eman	20
3	omar	NULL

Dept

<u>Did</u>	Dname
10	SD
20	HR
30	IS
40	Admin

1) First Type

--Cross join

-----Cartesian Product

Select Sname , Dname

From Student , Dept

=

Select Sname , Dname

From Student Cross join Dept

Sname	Dname
ahmed	SD
khalid	SD
eman	SD
Omar	SD
ahmed	HR
khalid	HR
Eman	HR
Omar	HR
.....

هنا كل قيمة من الجدول الأول مع كل قيمة من الجدول الثاني
يعني في المثال ده هيجيب ١٦ قيمة (16 Rows)

2) Second Type

--Inner Join

-----Equi Join

Select Sname , Dname

From Student , Dept

Where Dept.did = Student.did

=

Select Sname , Dname

From Student S , Dept D

Where D.Did = S.Did

=

Select Sname , Dname

From Student S inner join Dept D

On D.Did = S.Did

Sname	Dname
ahmed	SD
khalid	SD
eman	HR

هنا هو عايز الطلاب اللي ليهم أقسام

Hint :-

Select Sname , Dname

left table

From Student S inner join

right table

Dept D

On D.Did = S.Did



3) Third Type

-----Left Outer join

Select Sname , Dname

From Student S left outer join Dept D

On $D.Did = S.Did$

Sname	Dname
ahmed	SD
khalid	SD
eman	HR
omar	NULL

هنا هو عايز كل الطلاب سواء ليهم أقسام أو لا

4) Fourth Type

-----Right Outer Join

Select Sname , Dname

From Student $S^{\text{right outer join}}$ Dept D

On D.Did = S.Did

Sname	Dname
ahmed	SD
Kahlid	SD
eman	HR
NULL	IS
NULL	Admin

هنا هو عايز كل الأقسام سواء فيها طلاب أو لا

5)Fifth Type

-----Full Outer Join

Left Outer join + Right Outer Join

Select Sname , Dname

left right

From Student S full outer join Dept D

On D.Did = S.Did

Sname	Dname
ahmed	SD
khalid	SD
eman	HR
Omar	NULL
NULL	IS
NULL	Admin

Sname	Dname
ahmed	SD
khalid	SD
eman	HR
omar	NULL

+

Sname	Dname
ahmed	SD
Kahlid	SD
eman	HR
NULL	IS
NULL	Admin

=

Sname	Dname
ahmed	SD
khalid	SD
eman	HR
Omar	NULL
NULL	IS
NULL	Admin

6) Sixth Type

--Self Join

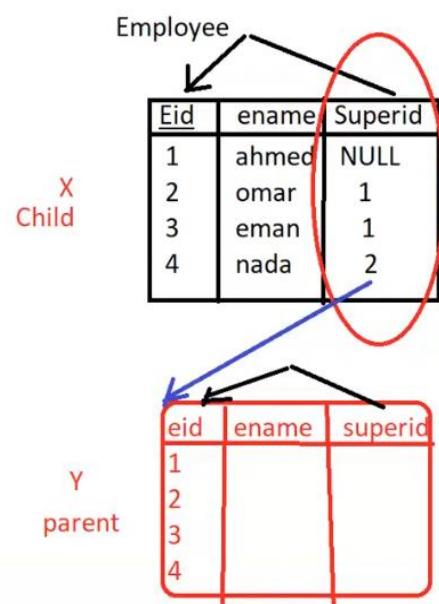
(Unary relationship)

Select X.Ename as EmpName , Y.Ename as SuperName

From Employee X , Employee Y

where Y.eid = X.superid

Empname	SuperName
omar	Ahmed
eman	Ahmed
nada	omar



هنا ف ال unary relationship انا بتعامل مع self join
ف انا ازاي هعمل join بين table ونفسه
ف كده احنا محتاجين نأخذ نسخة من ال table ف ال memory بحيث يكون
عندنا 2 tables

طيب انهي child table وال انهي parent
أو انهي foreign table اللي فيه primary والنهي اللي فيه
ال parent table الأساسي بيكون child والنسخة بتكون ال
طيب ليه ؟

لأن ببساطة في المثال بتاعنا ال foreign هو superid في ال الأساسي
(child table)

ف الطبيعي كده ان ال primary table عباره عن Eid الثاني (table

يعني من الآخر انا عندي ال primary supervisor ف ال child foreign منه

طيب لو انا عايز اسم الموظف وكل بيانات ال supervisor بتاعه ؟

Select X. Ename , Y.*

From employee X , employee Y

where y.eid=x.superid

د ه مشهور اسمه error (ambiguous column) بسبب اني ب 2 tables نفس الأسم في column

The screenshot shows a SQL query window with the following code:

```
select st_fname,dept_name,dept_id  
from Student S , Department D  
where D.Dept_Id=S.Dept_Id
```

The column name `dept_id` is underlined with a red wavy line, indicating it is ambiguous. Below the query window, a messages pane displays:

Msg 209, Level 16, State 1, Line 16
Ambiguous column name 'dept_id'.

ف الحل هنا اتنا نحدد ال table اللي احنا عايزين ال column يظهر فيه

```
select st_fname,dept_name,d.dept_id  
from Student S , Department D  
where D.Dept_Id=S.Dept_Id
```

طيب لو انا عايز كل معلومات ال student table وعايزة كل معلومات ال Department table

```
select st_fname,*
from Student S , Department D
where D.Dept_Id=S.Dept_Id
```

	st_fname	St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super	Dept_Id	Dept_Name	Dept_Desc	Dept_Location	Dept_Manager	Manager_hiredate
1	Ahmed	1	Ahmed	Hasan	Cairo	20	10	NULL	10	SD	System Development	Cairo	NULL	2000-01-01
2	Amr	2	Amr	Magdy	Cairo	21	10	1	10	SD	System Development	Cairo	NULL	2000-01-01
3	Mona	3	Mona	Saleh	Cairo	22	10	1	10	SD	System Development	Cairo	NULL	2000-01-01
4	Ahmed	4	Ahmed	Mohamed	Alex	23	10	1	10	SD	System Development	Cairo	NULL	2000-01-01
5	Khalid	5	Khalid	Mohamed	Alex	24	10	1	10	SD	System Development	Cairo	NULL	2000-01-01
6	Heba	6	Heba	Farouk	Mansoura	25	20	NULL	20	EL	E-Learning	Mansoura	NULL	2002-10-02
7	Ali	7	Ali	Hussien	Cairo	25	20	6	20	EL	E-Learning	Mansoura	NULL	2002-10-02
8	Mohamed	8	Mohamed	Fans	Alex	28	20	6	20	EL	E-Learning	Mansoura	NULL	2002-10-02
9	Saly	9	Saly	Ahmed	Mansoura	24	30	NULL	30	Java	Java	Cairo	NULL	2008-05-04
10	Fady	10	Fady	Ali	Alex	24	30	9	30	Java	Java	Cairo	NULL	2008-05-04
11	Marwa	11	Marwa	Ahmed	Cairo	24	30	9	30	Java	Java	Cairo	NULL	2008-05-04
12	Noha	12	Noha	Omar	Cairo	21	40	NULL	40	MM	Multimedia	Alex	NULL	2009-01-01
13	Said	13	Said	NULL	NULL	NULL	40	12	40	MM	Multimedia	Alex	NULL	2009-01-01

بس ال query دى طلعت غلط لأنها بتجيب ال * لـ 2 tables ف ايه الحل ؟

```
select st_fname,D.*
from Student S , Department D
where D.Dept_Id=S.Dept_Id
```

	st_fname	Dept_Id	Dept_Name	Dept_Desc	Dept_Location	Dept_Manager	Manager_hiredate
1	Ahmed	10	SD	System Development	Cairo	NULL	2000-01-01
2	Amr	10	SD	System Development	Cairo	NULL	2000-01-01
3	Mona	10	SD	System Development	Cairo	NULL	2000-01-01
4	Ahmed	10	SD	System Development	Cairo	NULL	2000-01-01
5	Khalid	10	SD	System Development	Cairo	NULL	2000-01-01
6	Heba	20	EL	E-Learning	Mansoura	NULL	2002-10-02
7	Ali	20	EL	E-Learning	Mansoura	NULL	2002-10-02
8	Mohamed	20	EL	E-Learning	Mansoura	NULL	2002-10-02
9	Saly	30	Java	Java	Cairo	NULL	2008-05-04
10	Fady	30	Java	Java	Cairo	NULL	2008-05-04
11	Marwa	30	Java	Java	Cairo	NULL	2008-05-04
12	Noha	40	MM	Multimedia	Alex	NULL	2009-01-01
13	Said	40	MM	Multimedia	Alex	NULL	2009-01-01

```
|select st_fname,dept_name  
from Student S inner join Department D  
on D.Dept_Id=S.Dept_Id  
where St_Address='alex'
```



```
|select st_fname,dept_name  
from Student S inner join Department D  
on D.Dept_Id=S.Dept_Id and St_Address='alex'
```

لأن ال **or** هنا **كأنها**

---join Multi tables

```
select st_fname,crs_name,grade  
from Student S , Stud_Course SC , Course C  
where S.St_Id=Sc.St_Id and  
c.Crs_Id=sc.Crs_Id
```

عندی **2 joins between 3 tables**

طيب لو عايز اعملها ب inner join

```
select st_fname,crs_name,grade  
from Student S inner join Stud_Course SC  
on S.St_Id=Sc.St_Id  
inner join  
Course C  
on c.Crs_Id=Sc.Crs_Id
```

لازم اقسمها لنصين لأن ال inner join بين 2 tables ب تكون بين 2 tables بس مش أكثر من كده

غف انا كده عملت inner join بين 2 tables والناتج عملته ال الثالث table

```
select st_fname,crs_name,grade,dept_name  
from Student S inner join Stud_Course SC  
on S.St_Id=Sc.St_Id  
inner join  
Course C  
on c.Crs_Id=Sc.Crs_Id  
inner join  
Department D  
on d.Dept_Id=s.Dept_Id
```

هذا برضو نفس الفكرة - ف لما زودت table رابع عملت inner join زيادة

١ - **tables = عدد ال inner joins** -

4 tables & 3 inner joins - يعني هنا انا عندي

```
--join DML
```

```
-- join update
```

```
update Stud_Course  
set grade+=10
```

هنا انا عايز اعمل ازود ١٠ درجات لكل الطلبة

```
select grade  
from Student S , Stud_Course sc  
where s.St_Id=sc.St_Id and St_Address='cairo'
```

وكمان انا بعرض درجات الطلاب اللي من cairo

طيب لو انا عايز ازود ١٠ درجات للطلاب اللي من cairo
اعملها ازاي ؟

```
update Stud_Course  
set grade+=10  
|  
from Student S , Stud_Course sc  
where s.St_Id=sc.St_Id and St_Address='cairo'
```

من الاخر بحط ال update مكان ال select

وبرضو ممكن أعملها كده

```
update SC
    set grade+=10
from Student S , Stud_Course SC
where s.St_Id=sc.St_Id and St_Address='cairo'
```

Stud_Course = SC لأن

```
select st_fname  
from Student
```

Results Messages

	st_fname
8	Mohamed
9	Saly
10	Fady
11	Marwa
12	Noha
13	Said
14	NULL
15	ali
16	ali

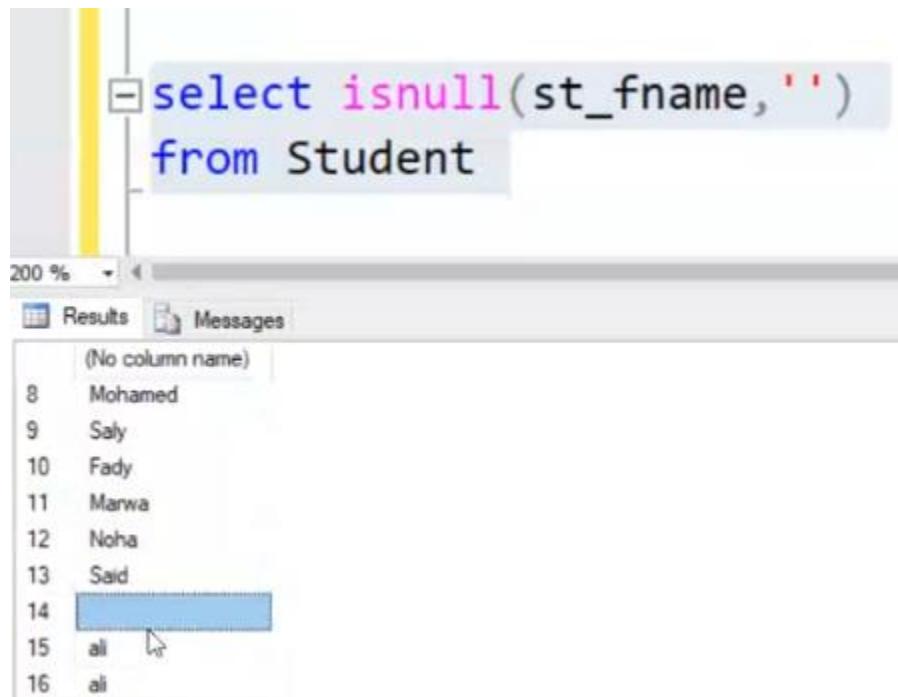
هنا احنا عملنا select لـ first name
بس لاقينا عندنا Null
طيب لو عايز اجيب الطلاب الموجودين بس من غير Null

```
select st_fname  
from Student  
where st_Fname is not null
```

Results Messages

	st_fname
1	Ahmed
2	Amr
3	Mona
4	Ahmed
5	Khalid
6	Heba
7	Ali
8	Mohamed
9	Saly
10	Eman

طيب لو عايز أ Replace ال Null بقيمة
isnull عن طريق اسمها built in fun.
Null value دي بتخليني ادي قيمة لأي isnull



The screenshot shows a SQL query window with the following code:

```
select isnull(st_fname, '')  
from Student
```

The results pane shows the following data:

	(No column name)
8	Mohamed
9	Saly
10	Fady
11	Marwa
12	Noha
13	Said
14	ali
15	ali
16	ali

بتأخذ ال column : لو في قيمة بتتعرض عادي لكن لو في قيمة ب Null
بتأخذ القيمة اللي انت مدتها اللي هي هنا string فاضية ''

```
select isnull(st_fname, 'has No Name')  
from Student
```

Results Messages

	(No column name)
1	Ahmed
2	Amr
3	Mona
4	Ahmed
5	Khalid
6	has No Name
7	Ali
8	has No Name
9	Saly

Null بتاخد ال column : لو في قيمة بتتعرض عادي لكن لو في قيمة ب Null
بتاخد القيمة اللي انت مدعيها اللي هي هنا ' has No Name '

طيب ممكن أ **Replace** ال **column** ده ب **column** تاني ؟

- اه ينفع

The screenshot shows a SQL query window with the following code:

```
select isnull(st_fname,st_lname)
from Student
```

The results pane shows the output of the query:

	(No column name)
8	Fam
9	Saly
10	Fady
11	Marwa
12	Noha
13	Said
14	Saleh
15	ali
16	ali

طيب ممكن أ **Replace** ال **column** **st_name** ب **value** **Null** ؟
بتاخد ال **value** اللي انت مديهالها اللي هي هنا

طيب افرض ان انا ب **replace** **last name** بال **Null** وطلع برضو **Null** زي ال
؟ **first name**

- هنا مش هقدر استخدم **isnull** لأنها بت ال **value** ب **column** واحدة بس
ف لو لقت ال **Null** ب **last name** وال **first name** ف هت
ف الآخر **Null**

لكن اقدر استخدم **built in fun.** **replace** تانية بت **value** اكتر من
(multiple replacement)

coalesce اسمها

Array دی بتاخد Coalesce

The screenshot shows a SQL query window with the following code:

```
select Coalesce(st_fname,st_lname,st_address,'No Data')
from Student
```

The results pane displays the following data:

	(No column name)
1	Ahmed
2	Amr
3	Mona
4	Ahmed
5	Khalid
6	Farouk
7	Ali
8	Alex
9	Saly
10	Ema

هنا عندي كذا Replacement

كده المفروض يعرض ال first name

طيب لو ال first name ب Null

يعرض ال last name

طيب لو ال last name ب Null

يعرض ال address

طيب لو ال address ب Null

يعرض ال No Data <-- ده string

طيب هل ممكن اجمع 2 Columns مع بعض
- اه بس يكونوا من نفس ال Data Type

زي المثال ده

```
select st_fname+' '+st_lname  
from student
```

لكن لو ال Data Types مختلفة
Error - هيعمل

A screenshot of the SQL Server Management Studio interface. In the query editor window, there is a code block:

```
select st_fname+' '+st_age  
from student
```

Below the code, the results pane shows an error message:

Msg 245, Level 16, State 1, Line 118
Conversion failed when converting the nvarchar value 'Ahmed ' to data type int.

طيب ايه الحل ؟

- الحل ان انا اخلي ال **Columns 2** من نفس ال **Data Type**

The screenshot shows a SQL query window with the following code:

```
select st_fname+' '+convert(varchar(2),st_age)
from student
```

The results pane shows the output of the query:

	(No column name)
1	Ahmed 20
2	Amr 21
3	Mona 22
4	Ahmed 23
5	Khalid 24
6	NULL
7	Ali 25
8	NULL
9	Saly 24
10	NULL

هنا انا حولت ال **Char** ل **int** age بقى بـتاتع ال **Data Type** من

وبالمناسبة ال **Select** انا اقدر اكتب فيها اي حاجة

The screenshot shows a modified SQL query window with the following code:

```
select 'studName= '+st_fname+' &age= '+convert(varchar(2),st_age)
from student
```

The results pane shows the output of the query:

	(No column name)
1	studName= Ahmed &age= 20
2	studName= Amr &age= 21
3	studName= Mona &age= 22
4	studName= Ahmed &age= 23
5	studName= Khalid &age= 24
6	NULL
7	studName= Ali &age= 25
8	NULL
9	studName= Saly &age= 24
10	NULL

يعني هنا انا كتب جملتين String مع ال **2 Columns** اللي انا عايز
اجمعهم

طيب في المثال اللي فوق ده

- افرض ان اي Column من الآتینين ب Null سواء ال studname او ال age

- للأسف ال Null لما بتدخل ف اي operation لما بتدخل

- نتيجة ال operation دي بتطلع كلها ب Null

عشان كده طلعي القيم دي ب Null

```
select 'studName= '+st_fname+' &age= '+convert(varchar(2),st_age)
from student
```

	studName=	age=
1	Ahmed	20
2	Amr	21
3	Mona	22
4	Ahmed	23
5	Khalid	24
6	NULL	
7	Ali	25
8	NULL	
9	Saly	24
10	Ema	24

طيب ايه حل الموضوع ده ؟

- اني استخدم isnull

```
select isnull(st_fname, '')+' '+convert(varchar(2),isnull(st_age,0))
from student
```

1	Ahmed	20
2	Amr	21
3	Mona	22
4	Ahmed	23
5	Khalid	24
6		25
7	Ali	25
8		28
9	Saly	24
10	Ema	24

ف هنا انا بعوض عن ال Null ف ال st_fname

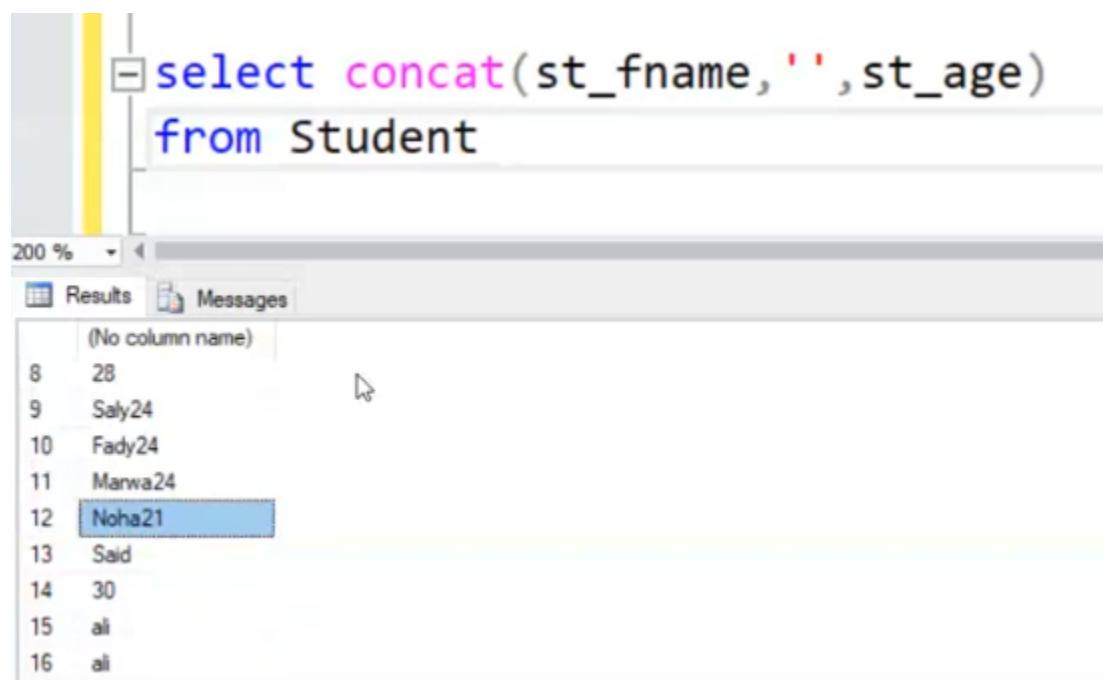
و بعوض عن ال Null ف ال st_age

بس كده الموضوع بقى صعب جدا لأنني بستخدم كذا Fun. جوا بعض
وده بيأثر على ال Performance

عشان كده Microsoft عملت
concat اسمها

? concat طيب ايه ميزة

- انها بتحول كل اللي جواها ل String
- بتشيل أي Null وتحط مكانها Empty String



The screenshot shows a SQL query window with the following code:

```
select concat(st_fname, ' ', st_age)
from Student
```

The results grid below shows the output:

	(No column name)
8	28
9	Saly24
10	Fady24
11	Marwa24
12	Noha21
13	Said
14	30
15	ali
16	ali

Wild Cards

- | _ one char
- | % zero or More char

```
select *
from Student
where st_fname like 'a%'
```

Results

St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super
1	Ahmed	Hassan	Cairo	20	10	NULL
2	Amr	Magdy	Cairo	21	10	1
3	Ahmed	Mohamed	Alex	23	10	1
4	Ali	Hussien	Cairo	25	20	6
5	ali	NULL	NULL	NULL	NULL	NULL
6	59	ali	NULL	NULL	NULL	NULL

St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super
3	Ahmed	Mohamed	Alex	23	10	1
4	Ali	Hussien	Cairo	25	20	6
5	ali	NULL	NULL	NULL	NULL	NULL
6	59	ali	NULL	NULL	NULL	NULL

(a) هنا انا بجيب كل ال first name اللي بتبدأ بحرف ال a

```

select *
from Student
where st_fname like '%a'

```

Results

	St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super
1	3	Mona	Saleh	Cairo	22	10	1
2	11	Marwa	Ahmed	Cairo	24	30	9
3	12	Noha	Omar	Cairo	21	40	NULL

هنا انا بجيب كل ال first name اللي بتنتهي بحرف ال (a)

```

select *
from Student
where st_fname like '%a%'

```

Results

	St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super
1	1	Ahmed	Hassan	Cairo	20	10	NULL
2	2	Amr	Magdy	Cairo	21	10	1
3	3	Mona	Saleh	Cairo	22	10	1
4	4	Ahmed	Mohamed	Alex	23	10	1
5	5	Khalid	Moahmed	Alex	24	10	1
6	7	Ali	Hussien	Cairo	25	20	6
7	9	Saly	Ahmed	Mansoura	24	30	NULL
8	10	Fady	Ali	Alex	24	30	9
9	11	Marwa	Ahmed	Cairo	24	30	9

هنا انا بجيب كل ال first name اللي بتحتوي على حرف ال (a) سواء ف الأول أو في الآخر أو في النص

```

select *
from Student
where st_fname like '_a%'

```

Results

St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super
9	Saly	Ahmed	Mansoura	24	30	NULL
10	Fady	Ali	Alex	24	30	9
11	Marwa	Ahmed	Cairo	24	30	9
13	Said	NULL	NULL	NULL	40	12

هنا انا بجيب كل ال first name اللي الحرف الثاني منهم حرف (a)

'a%h'	أي حاجة أولها (a) وأخرها (h)
'%a_'	الحرف قبل الأخير (a)
'ahm%'	بتبدأب (ahm)
'[ahm]%'	بتبدأب (a) أو (h) أو (m)
'[^ahm]%'	مش بتبدأب (a) أو (h) أو (m)
'[a-h]%'	أي حاجة بتبدأ ف ال range ده (a:h)
'[^a-h]%'	أي حاجة بتبدأ بحرف غير ال range ده (a:h)
'[346]%'	أي حاجة بتبدأب (3) أو (4) أو (6)
'%[%]'	أي حاجة آخرها علامة (%)
'%[_]%'	أي حاجة في نصها علامة (_)
'[_]%[_]'	أي حاجة أولها وأخرها علامة (_)

```
select st_fname,dept_id,st_age  
from Student  
order by st_address
```

هل ال Run دی تنفع ت ؟

اه ينفع / لأن ال **st_address Column** بـتاتع
Student table

```
select st_fname,dept_id,st_age  
from Student  
order by 1
```

دي معناها أعمل **order by** بأول **column** في ال **Select**
(order by st_fname) يعني أعمل

```
|select st_fname,dept_id,st_age  
|from Student  
|order by 3
```

دي معناها أعمل **order by** بـثالث **column** في ال **Select**
(order by st_age) يعني أعمل

طيب هل أقدر أعمل **order by** بعدها
- اه أقدر

```
select st_fname,dept_id,st_age
from Student
order by dept_id,st_age
```

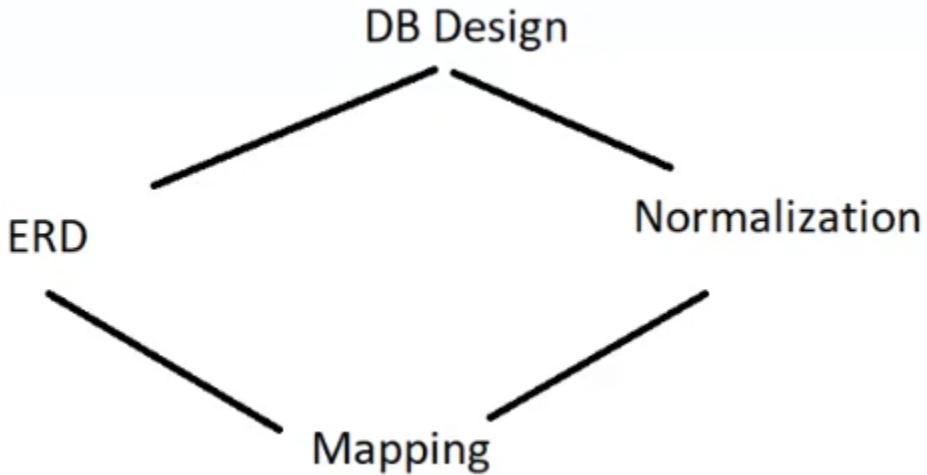
هنا احنا عملنا **order by st_age** ثم **order by dept_id**

طيب أمتى هنستخدم الـ **order by st_age**
- لما اعمل **order by dept_id** ويطلع كذا قيمة متساوية
- ف ساعتها بس أقدر أستخدم الـ **order by st_age**

	st_fname	dept_id	st_age
3	NULL	NULL	30
4	Ahmed	10	20
5	Amr	10	21
6	Mona	10	22
7	Ahmed	10	23
8	Khalid	10	24
9	NULL	20	25
10	Ali	20	25
11	NULL	20	28
12	Saly	30	24
13	Fady	30	24
14	Manwa	30	24
15	Said	40	NULL
16	Noha	40	21

ف من الواضح هنا انا بنرتب بالـ **dept_id**
ولما لقينا كذا **dept_id** ليهم نفس القيمة
رتبا اللي ليهم نفس القيمة دول بالـ **st_age**

Normalization



احنا ايه هدفا ف ال ? DB Design

- اننا نعمل Mapping كويس

ف انا بعمل ERD لما ببدأ ف System جديد
أو لو عندي System موجود بالفعل بس ال client عايز تعديل حاجات كتير
أو هيحط requirements كتير جديدة

طيب لو عندي ال System موجود بالفعل وفيه Tables أو Excel Sheets
بس في شوية حاجات محتاجه تتعدل
ف هل احنا نقدر نعدل في الحاجات دي عشان نوصل لـ Schema السليمة ؟
- اه نقدر عن طريق اننا نعمل Normalization

طيب ايه المشاكل اللي بنقى عايزين نعمل **Normalization** بسببها لـ
؟ **System**

- **Duplication**
- ال **Normal Behaviour** شغالة بس مش شغالة بال **DML Queries**
- بتاعها
- ال **System** بطيء

يبقى كده ال **Normalization** يعني : إعادة هيكلة ال **Tables** أو ال **Excel Sheets** دى عشان أوصل لشكل ال **Database** السليم

Hint

من الأفضل إنك تطبق قواعد ال **Normalization** بعد م تخلص ال **ERD** الطبيعي اللي بعد ال **Mapping**

عشان تتأكد فعلا إن ال **Database** في شكلها السليم ومفيش أي مشاكل ولو في أي مشاكل أعالجها قبل م أبدأ أشتغل فعليا على ال **Database** دى ف أي **application** بقى أو أي حاجة معتمدة عليها

- ▶ **Normalization:** The process of structuring data to minimize duplication and inconsistencies.
- ▶ The process usually involves breaking down a

- ▶ The process usually involves breaking down a **single Table** into **two or more** tables and defining relationships between those tables.

؟ طيب إزاي بعمل ال **Normalization**

عن طريق إني بقسم ال **Table** لكتذا **Table** وفي نفس الوقت أحافظ على ترابط ال **Tables** دي ببعض

طيب ايه المشاكل اللي بنبقى عايزين نعمل **Normalization** على ال **Tables** عشان أعيد هيكلتها

طيب ايه المشاكل اللي بنبقى عايزين نعمل **Normalization** بسببها لل
؟ **System**

Duplication -
ال **Normal Behaviour** شغالة بس مش شغالة بال **DML Queries** -
باتبعها
ال **System** بطيء -

ي يعني ايه ال DML Queries شغاله بس مش شغاله بال Normal Behaviour **بتاعها**

- Goal is to avoid anomalies
 - **Insertion Anomaly** – adding new rows forces user to create duplicate data
 - **Deletion Anomaly** – deleting rows may cause a loss of data that would be needed for other future rows
 - **Modification Anomaly** – changing data in a row forces changes to other rows because of duplication

يعني انا عندي مشكلة مثلا

ف هلاقيه مثلا طالب مني ان انا اضيف بيانات الطلاب لـ insert table : **Inertion Anomaly**

ف هلاقيه مثلا طالب مني ان انا اضيفها بعدين

برضو في حين ان انا ممكن اضيفها بعدين

ف هلاقيه مثلا Delete project معين : **Deletion Anomaly**

ف هلاقيه ي Delete بيانات الموظفين برضو

(1 row) : **Update Anomaly** ف هلاقيه قالي مثلا Update حاجة لموظف معين (1 row)

ف هلاقيه قالي مثلا 3 rows affected

ده معناه ان بيانات الموظف ده متكرره 3 مرات

ف ال ده Table

طيب ليه بيحصل ده ؟

لأنه مجمع كذا Entity مع بعض في نفس ال Table

a table should not have more than one entity type

طيب نشوف مثال

SID	Sname	Bdate	City	ZipCode	Subject	Grade	Teacher
1	Ahmed	1/1/1980	Cairo	1010	DB	A	Hany
1	Ahmed	1/1/1980	Cairo	1010	Math	B	Eman
1	Ahmed	1/1/1980	Cairo	1010	WinXP	A	khalid
2	Ali	1/1/1983	Alex	1111	DB	B	Hany
2	Ali	1/1/1983	Alex	1111	SWE	B	Heba
3	Mohamed	1/1/1990	Mansoura	1210	NC	C	Mona

What's the primary key? Answer – Composite: SID, Subject

هنا انا عايز اعدل ال city للطالب ahmed < --

هنا لو عايز امسح مادة ال NC هيمسح بيانات الطالب Mohamed
وهكذا

طيب ليه كل ده ؟

Why do these anomalies exist?

Because we've combined two themes (entity types) into one relation. This duplication, and an unnecessary dependency between the entities

طيب انا عشان افهم ال **Normalization** لازم اعرف حاجة اسمها

Functional Dependency

ف ال **Database** العادي

- في علاقات بين ال **Tables** وبعض
- في علاقات بين ال **Rows** وبعض (زي ال **self relationship** مثلا)

لكن عرمنا م سمعنا عن علاقات بين ال **Columns** وبعض

دائماً يتحقق العلاقات بين الـ **Columns** على شكل (a -----→ b)

Functional Dependency

طيب يعني ايه (a -----→ b)

- يعني **b** بتعتمد في وجودها على **a**

يعني أنا لو عرفت **a** اعرف أجيب **b**

بس لو عرفت **b** مش لازم اعرف أجيب **a**

- مع كل قيمة جديدة ل **a** بيظهر قيمة جديدة ل **b**

Functional dependency

- a constraint between two attributes (columns) or two sets of columns
- A → B if “for every valid instance of A, that value of A uniquely determines the value of B”
- or ...A → B if “existing of B depending on a value of A”

مثال

A -----> B

SSN ----> Ename

Pnum ---> ProLocation

لو عرفت a هعرف اجيب b

لو عرفت رقم البطاقة هعرف اجيب الأسم

لو عرفت رقم المشروع هعرف اجيب مكان المشروع

طيب امتى اقول ال Table ده سليم ؟

لو (a -----> b,c,d)

يعني عن طريق ال Primary key أقدر اجيب باقي ال Columns اللي في ال Table (Pk-----> Columns)

يعني هنا

Sid ---->Name,Age,address

عن طريق ال Sid أقدر اجيب ال Name,Age,address

لكن مقدرش عن طريق ال Age مثلا اجيب ال Name,address,Sid

لأن ال Data دي بتتكرر في ال Name,Age,address

عكس ال Sid مبيتكررش لأنه Unique

some examples

- social security number determines employee name
 $SSN \rightarrow ENAME$

لو عرفت ال **Ename** هعرف ال SSN

- project number determines project name and location
 $PNUMBER \rightarrow \{PNAME, PLOCATION\}$

لو عرفت ال **PLocation** هعرف ال Pname **وال PNumber**

- employee ssn and project number determines the hours per week that the employee works on the project
 $\{SSN, PNUMBER\} \rightarrow HOURS$

لو عرفت ال **Hours** هعرف ال PNumber **وال SSN**

Types of functional dependency

- **Full Functional Dependency**

Attribute is fully Functional Dependency on a PK if its value is determined by the **whole PK**

- **Partial Functional Dependency**

Attribute if has a Partially Functional Dependency on a PK if its value is determined by **part of the PK**(Composite Key)

- **Transitive Functional Dependency**

Attribute is Transitively Functional Dependency on a table if its value is determined by another **non-key attribute** which it self determined by PK

واحدة واحدة

➤ Full Functional Dependency

Attribute is fully Functional Dependency on a PK if its value is determined by the **whole PK**

PK ال **Attribute** معتمدة كلية على ال **Full Function Dependency** (**Single or composite**) سواء ال PK ده

SSN -> ENAME زی دی **Single**

{SSN, PNUMBER} -> HOURS زی دی **composite** و

➤ Partial Functional Dependency

Attribute if has a Partially Functional Dependency on a PK if its value is determined by **part of the PK**(Composite Key)

PK ال **Attribute** معتمدة جزئيا على **Partial Function Dependency** ال PK وال PK ده بيكون (**composite**) اكيد

➤ Transitive Functional Dependency

Attribute is Transitively Functional Dependency on a table if its value is determined by another **non-key attribute** which it self determined by PK

يعني ال Attribute مش معتمد
على ال PK

يعني بيكون عندي 2 Attributes non-keys وبيكونوا
وبيكونوا بيعتمدوا على بعض

ملخص للكلام اللي فوق ده :-

لو عندي Table فيه Simple PK

يبقى ايه أنواع العلاقات اللي ممكن ألاقيها جواه ؟

SSN → ENAME

زي ده

Full Function Dependency -

Transitive Function Dependency -

طيب لو عندي Table فيه composite PK

يبقى ايه أنواع العلاقات اللي ممكن ألاقيها جواه ؟

{SSN, PNUMBER} → HOURS

زي ده Full Function Dependency -

Partial Function Dependency -

Transitive Function Dependency -

Example

SID	SName	Birthdate	City	Zip Code	Subject	Grade	Teacher
1	Ahmed	1/1/1980	Cairo	1010	DB	A	Hany
1	Ahmed	1/1/1980	Cairo	1010	Math	B	Eman
1	Ahmed	1/1/1980	Cairo	1010	WinXP	A	khalid
2	Ali	1/1/1983	Alex	1111	DB	B	Hany
2	Ali	1/1/1983	Alex	1111	SWE	B	Heba
3	Mohamed	1/1/1990	Cairo	1010	NC	C	Mona

Full Functional Dependency

Sid,Subject → Grade

Partial Functional Dependency

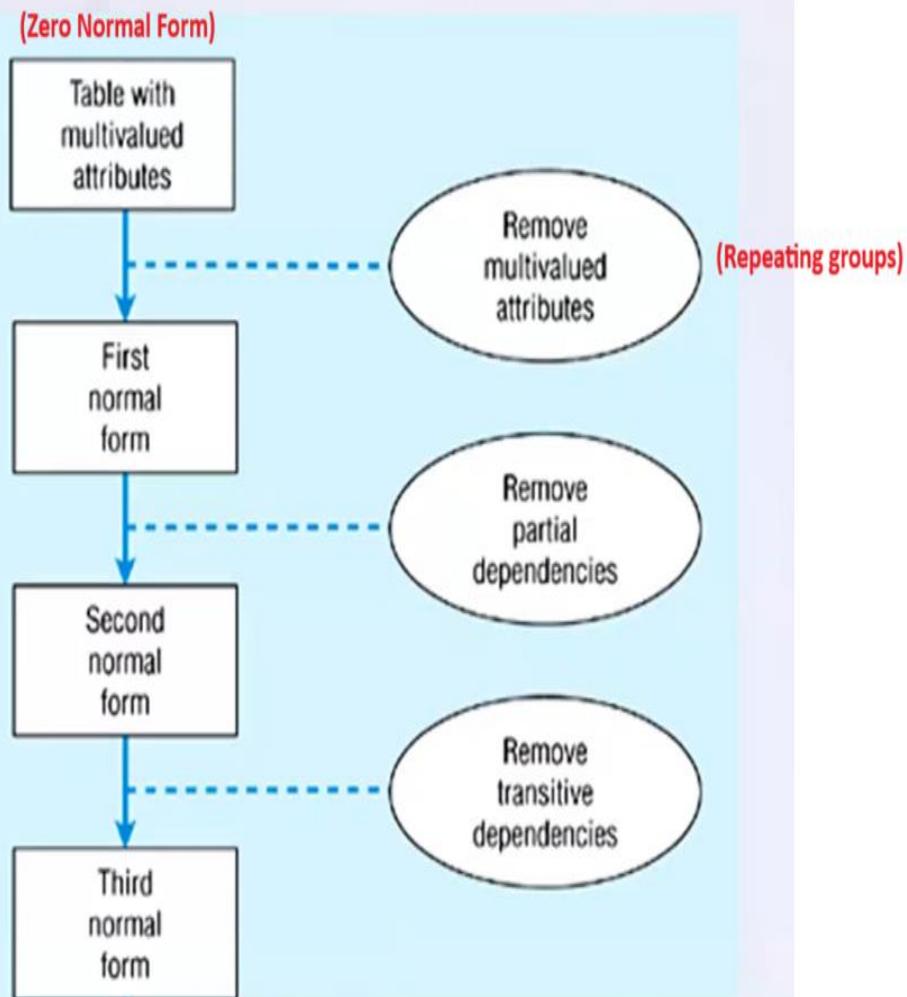
Sid → SName
Subect → Teacher

Transitive Functional Dependency

ZipCode → City

Activat
Go to Se

Steps in normalization



1NF

- relation is in **first normal form** if it contains no multivalued or composite attributes
- remove repeating groups to a new table as already demonstrated, “carrying” the PK as a FK
- All columns (fields) must be atomic
 - Means : no repeating items in columns

Example

SID	SName	Birthdate	City	Zip Code	Subject	Grade	Teacher
1	Ahmed	1/1/1980	Cairo	1010	DB	A	Hany
1	Ahmed	1/1/1980	Cairo	1010	Math	B	Eman
1	Ahmed	1/1/1980	Cairo	1010	WinXP	A	khalid
2	Ali	1/1/1983	Alex	1111	DB	B	Hany
2	Ali	1/1/1983	Alex	1111	SWE	B	Heba
3	Mohamed	1/1/1990	Cairo	1010	NC	C	Mona

SID	SName	Birthdate	City	Zip Code	Subject	Grade	Teacher
1	Ahmed	1/1/1980	Cairo	1010	DB Math WinXP	A	Hany
						B	Eman
						A	khalid
2	Ali	1/1/1983	Alex	1111	DB SWE	B	Hany
						B	Heba
3	Mohamed	1/1/1990	Cairo	1010	NC	C	Mona

Repeating Group:
Or multivalued



Activate W
Go to Settings

1NF

Student(SID, Sname, Birthdate, City, Zip Code)

SID	SName	Birthdate	City	Zip Code
1	Ahmed	1/1/1980	Cairo	1010
2	Ali	1/1/1983	Alex	1111
3	Mohamed	1/1/1990	Cairo	1010

Stud_Subject (SID, Subject, Grade, Teacher)

SID	Subject	Grade	Teacher
1	DB	A	Hany
1	Math	B	Eman
1	WinXP	A	khalid
2	DB	B	Hany
2	SWE	B	Heba
3	NC	C	Mona

2NF

- a relation is in **second normal form** if it is in first normal form AND every nonkey attribute is fully functionally dependant on the primary key
- i.e. remove partial functional dependencies, so no nonkey attribute depends on just part of the key

2NF

Student(SID, Sname, Birthdate, City, Zip Code)

<u>SID</u>	<u>SName</u>	<u>Birthdate</u>	<u>City</u>	<u>Zip Code</u>
1	Ahmed	1/1/1980	Cairo	1010
2	Ali	1/1/1983	Alex	1111
3	Mohamed	1/1/1990	Cairo	1010

2NF

Because there is no
Composite PK

Stud_Subject (SID, Subject, Grade, Teacher)

<u>SID</u>	<u>Subject</u>	<u>Grade</u>	<u>Teacher</u>
1	DB	A	Hany
1	Math	B	Eman
1	WinXP	A	khalid
2	DB	B	Hany
2	SWE	B	Heba
3	NC	C	Mona

SID, Subject → Grade.....FFD

Subject → Teacher.....PFD

Activate W
Go to Settings

2020

2NF

Student(SID, Sname, Birthdate, City, Zip Code)

<u>SID</u>	<u>SName</u>	<u>Birthdate</u>	<u>City</u>	<u>Zip Code</u>
1	Ahmed	1/1/1980	Cairo	1010
2	Ali	1/1/1983	Alex	1111
3	Mohamed	1/1/1990	Mansoura	1210

Stud_Subject (SID, Subject, Grade)

<u>SID</u>	<u>Subject</u>	<u>Grade</u>
1	DB	A
1	Math	B
1	WinXP	A
2	DB	B
2	SWE	B
3	NC	C

Subject (Subject,Teacher)

<u>Subject</u>	<u>Teacher</u>
DB	Hany
Math	Eman
WinXP	khalid
SWE	Heba
NC	Mona

Activate
Go to Sett

Third Normal Form

- 2NF PLUS ***no transitive dependencies***
(one attribute functionally determines a second, which functionally determines a third)

2NF

Student(SID, Sname, Birthdate, City, Zip Code)

SID	SName	Birthdate	City	Zip Code
1	Ahmed	1/1/1980	Cairo	1010
2	Ali	1/1/1983	Alex	1111
3	Mohamed	1/1/1990	Cairo	1010

Zip Code ->City TFD

Stud_Subject (SID, Subject, Grade)

SID	Subject	Grade
1	DB	A
1	Math	B
1	WinXP	A
2	DB	B
2	SWE	B
3	NC	C

Subject (Subject, Teacher)

Subject	Teacher
DB	Hany
Math	Eman
WinXP	khalid
SWE	Heba
NC	Mona

3NF

Because there is no Transitive Functional Dependency

Activate W
Go to Settings

3/2020

3NF

Student(SID, Sname, Birthdate,)

SID	SName	Birthdate	ZipCode
1	Ahmed	1/1/1980	1010
2	Ali	1/1/1983	1111
3	Mohamed	1/1/1990	1010

Stud_City(City, Zip Code)

City	Zip Code
Cairo	1010
Alex	1111

Stud_Subject (SID, Subject, Grade)

SID	Subject	Grade
1	DB	A
1	Math	B
1	WinXP	A
2	DB	B
2	SWE	B
3	NC	C

Subject (Subject, Teacher)

Subject	Teacher
DB1	Hany
Math	Eman
WinXP	khalid
DB2	Hany
SWE	Heba
NC	Mona

مثال تانى :-

ITI Example								
ITI Students Sheet								
Platform Name : SWE			Platform Description: Software Engineering					
Graduate Manager: Dr.Baha								
Appno	Name	F-code	Faculty	Address	Telno	Grade	Att. Hrs	Sdate
123	Ahmed	SC-phy	Science	Haram	3386842	A	600	14 Sep
124	Mona	Eng-cs	Engineering	Dokki	3389745, 3389744, 5123445	B	591	15 Sep
127	Ali	Com-ac	Commerce	Nasr City	2241593, 2222345	A	550	21 Sep
223	Karim	Med-bio	Medicine	Sheraton	2286845	C	600	14 Sep

طبعا واضح هنا إن

Platform Name : SWE Platform Description: Software Engineering
Graduate Manager: Dr.Baha

ال 3 columns دوں هما لے multivalued ہے ال 9 columns کی تھیں

وفرضی عینی ال Telno column بفرضی multivalued ہے

طیب کدھ من المتوقع یبھی عینی کام Table ف ال normal form کا ملے

3 Tables

1NF :

- **Platform** :pfname , pfdesc , pfManager
- **Students**: pfname, appno, name , faculty , F-Code, address, grade, attd , start_date
- **Std_Tel**: appno, telno

2NF

- **Students:** appno, name , faculty , FCode, address
- **Students_pf:** pfname,appno, grade, attd , start_date

Unchanged Tables

- **Platform :**pfname , pfdesc , pfManager
- **Std_Tel:** appno, telno

Activate
Go to Settings

3NF

- **Students:** appno, name , FCode, address
- **Fac_majors:**faculty , FCode

Unchanged Tables

- **Platform :**pfname , pfdesc , pfManager
- **Std_Tel:** appno, telno
- **Students_pf:** pfname,appno, grade, attd , start_date

Activ
Go to

مثال تانى :-

Repeated Cols	Repeating Groups
Eid,Ename,Eadd	Pro_ID,Pro_Name,PlocID,PlocCity,hours,Bouns

هنا منفعش أن أنا أخلي ال Eid يبقى PK
لأن الموظف لو بيشتغل في كذا مشروع ، ال Eid هيتكرر معايا
ف كان لازم أفصلهم في 2 Tables

1St

Employee(Eid,ename,eadd)
Emp_Pro(Eid,Pro_id,Pro_name,PlocID,PlocCity,hours,Bonus)

2nd

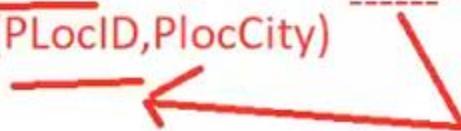
Employee (Eid,ename,eadd)
Project(Pro_id,Pro_name,PlocID,PLocCity)
Emp_pro(Eid,Pro_id,hours,bonus)

3rd

employee OK

Emp_Pro ok

project(Pro_id,Pro_Name,Plocid)
locations(PLocID,PlocCity)



ويبقى ده الشكل النهائي

Repeated Cols

Repeating Groups

Eid,Ename,Eadd,Pro_ID,Pro_Name,PlocID,PlocCity,hours,Bouns

1st

Employee(Eid,ename,eadd)
Emp_Pro(Eid,Pro_id,Pro_name,PlocID,PlocCity,hours,Bonus)

2nd

Employee (Eid,ename,eadd)
Project(Pro_id,Pro_name,PlocID,PLocCity)
Emp_pro(Eid,Pro_id,hours,bonus)

3rd

employee OK
Emp_Pro ok
project(Pro_id,Pro_Name,Plocid)
locations(PLocID,PlocCity)

انا عملت ال 3rd normal form

علي أساس إن ال hours , bonus ملهمش علاقة بعض

طيب لو ليهم علاقة بعض ؟

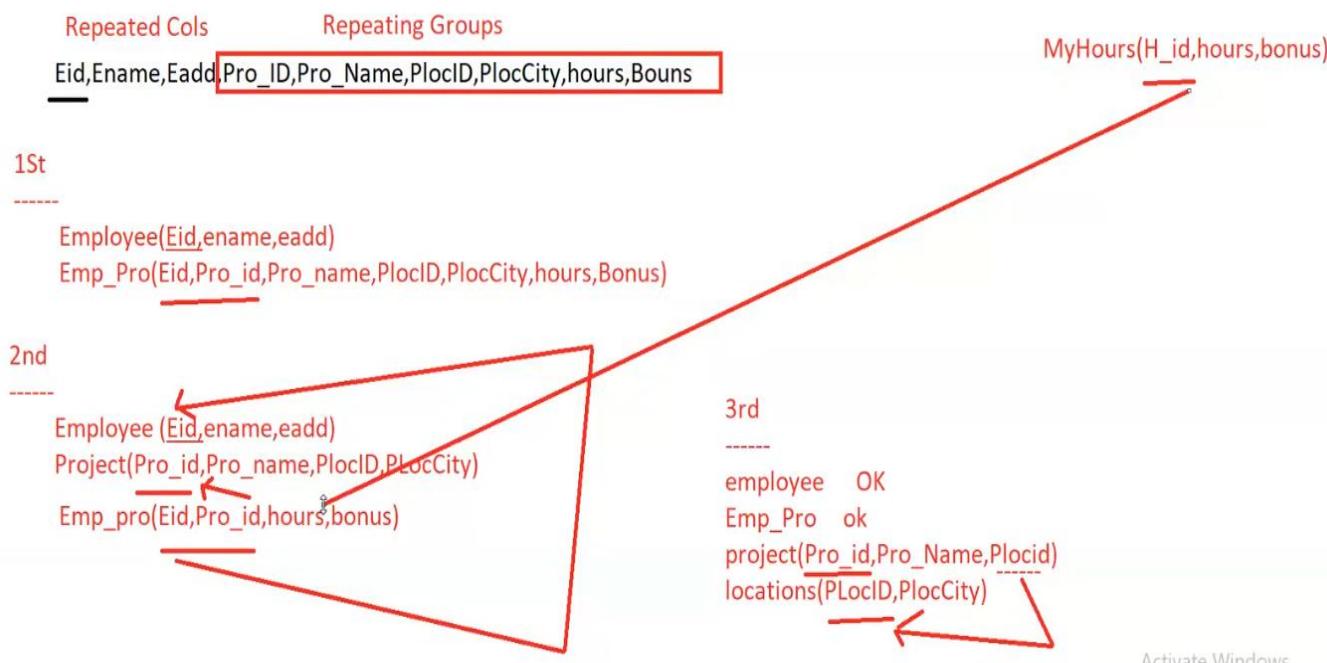
عمل hours , bonus Table جديد لـ

وأعملهم primary key جديد وهسميه مثلا H_id

وأرجع اعدل الجدول ف اللي ف ال 2nd normal form

واشيل ال hours , bonus واحظ ال Foreign key H_id

ف هيقي ده الشكل النهائي



Aggregate Function

Eid	Ename	Salary	Address	did
1	ahmed	3000	cairo	10
2	ali	5000	cairo	10
3	eman	2000	cairo	10
4	khalid	1000	alex	10
5	yousef	4000	alex	10
6	sameh	5000	alex	10
7	mohamed	6000	alex	20
8	alaa	7000	alex	20
9	ola	4000	cairo	20
10	reem	2000	cairo	20
11	nada	9000	cairo	20
12	sayed	8000	mansoura	30
13	reham	1500	mansoura	30
14	sally	2000	mansoura	30
15	omar	3000	mansoura	30

NULL

ال (aggregate fns.) مش بتحسب ال Null

Aggregate Function

Count , Max ,Min , Avg ,Sum

Select Sum(Salary)

From employee

62500

Select Min(Salary),Max(salary)

From Employee

1000 | 9000

Select Count(eid)

from employee

15

Count(*) -----> 15

Count(eid) -----> 15

count(ename) ----> 14

Select Avg(salary)

from employee

عشان كده Count(ename) = 14

لكن ال Count(eid) = 15

لما يستخدم group by لوحدها عادي

زي -- Select min(salary) < --

From employee ;

Select Min(salary) did
from employee
group by did

1000	10
2000	20
1500	30

Select Count(eid),Address
from employee
group by Address

6	Cairo
5	Alex
4	Mansoura

لكن لما با aggregate fun. مع ال retrieve column

لازم استخدم Group by لـ column ده

Select Count(eid),address
from employee
where did in(10,30)
group by address

3	Cairo
3	alex
4	Mansoura

Select Min(salary),did
from employee
where address like '_a%'
group by did

2000	10
2000	20
1500	30

ويمكن استخدام كمان where مع ال aggregate fun.

Aggregate Function

Count , Max ,Min , Avg ,Sum

Select Sum(Salary) From employee 62500

Select Min(Salary),Max(salary) From Employee 1000 9000

Select Count(eid) from employee 15

Count(*) -----> 15

Count(eid) -----> 15

count(ename) ----> 14

Select Avg(salary) from employee

Select Min(salary),did
from employee
group by did

1000	10
2000	20
1500	30

Select Count(eid),Address
from employee
group by Address

6	Cairo
5	Alex
4	Mansoura

Select Count(eid),address
from employee
where did in(10,30)
group by address

3	Cairo
3	alex
4	Mansoura

Select Min(salary),did
from employee
where address like '_a%'
group by did

2000	10
2000	20
1500	30

Eid	Ename	Salary	Address	did
1	ahmed	3000	cairo	10
2	ali	5000	cairo	10
3	eman	2000	cairo	10
4	khalid	1000	alex	10
5	yousef	4000	alex	10
6	sameh	5000	alex	10
7	mohamed	6000	alex	20
8	ala	7000	alex	20
9	ola	4000	cairo	20
10	reem	2000	cairo	20
11	nada	9000	cairo	20
12	sayed	8000	mansoura	30
13	reham	1500	mansoura	30
14	sally	2000	mansoura	30
15	omar	3000	mansoura	30

NULL

I

Activate Window

Ctrl + Shift + Esc

طيب ايه ترتيب تنفيذ ال Query ؟

4 Select Count(eid),address

1 from employee

2 where did in(10,30)

3 group by address

3	Cairo
3	alex
4	Mansoura

```
Select Sum(salary),did  
from employee  
group by did  
where Sum(salary)>=22000
```

ال Query دي انا عندي فيها غلطتين

١ - ال group by بتنكتب قبل Where

٢ - مينفعش ال aggregate fun. تكتب مع Where

```
where Sum(salary)>=22000
```

لكن ممكن استخدم حاجة تانية غير Where مع ال aggregate fun.

وهي **Having**

Select Sum(salary),did
from employee
group by did
having Sum(salary)>=22000

28000 | 20

Select Min(salary),did
from employee
group by did

1000	10
2000	20
1500	30

Select Count(eid),Address
from employee
group by Address

6	Cairo
5	Alex
4	Mansoura

4 Select Count(eid),address
1 from employee
2 where did in(10,30)
3 group by address

Select Min(salary),did
from employee
where address like '_a%'
group by did

2000	10
2000	20
1500	30

Eid	Ename	Salary	Address	did
1	ahmed	3000	cairo	10
2	ali	5000	cairo	10
3	eman	2000	cairo	10
4	khalid	1000	alex	10
5	yousef	4000	alex	10
6	sameh	5000	alex	10
7	mohamed	6000	alex	20
8	ala	7000	alex	20
9	ola	4000	cairo	20
10	reem	2000	cairo	20
11	nada	9000	cairo	20
12	sayed	8000	mansoura	30
13	reham	1500	mansoura	30
14	sally	2000	mansoura	30
15	omar	3000	mansoura	30

NULL

Activate Window

هنا انا بقوله عايز ال sum(salary) , did
لالأقسام اللي مجموع ال salary فيها أكبر من 22000

Select Sum(salary),did
from employee
group by did
having Sum(salary)>=22000

28000 | 20

Select Count(eid),address
from employee
group by address
having Count(eid)>=5

6	cairo
5	alex

Select Sum(salary),did
from employee
group by did
having Count(eid)>5

20000 | 10

Select Min(salary),did
from employee
group by did

1000	10
2000	20
1500	30

Select Count(eid),Address
from employee
group by Address

6	Cairo
5	Alex
4	Mansoura

4 Select Count(eid),address
1 from employee
2 where did in(10,30)
3 group by address

Select Min(salary),did
from employee
where address like '_a%'
group by did

2000	10
2000	20
1500	30

Eid	Ename	Salary	Address	did
1	ahmed	3000	cairo	10
2	ali	5000	cairo	10
3	eman	2000	cairo	10
4	khalid	1000	alex	10
5	yousef	4000	alex	10
6	sameh	5000	alex	10
7	mohamed	6000	alex	20
8	ala	7000	alex	20
9	ola	4000	cairo	20
10	reem	2000	cairo	20
11	nada	9000	cairo	20
12	sayed	8000	mansoura	30
13	reham	1500	mansoura	30
14	sally	2000	mansoura	30
15	omar	3000	mansoura	30

NULL

Activate Window:
Go to Settings to activate

طيب ينفع نستخدم Where و Having مع بعض ؟

- اه ينفع لأن Where : بتسخدم عشان اعمل condition على ال Rows
وال Having : بتسخدم عشان اعمل condition على Group

بتيجي مع ال columns الأساسية : Where
لكن aggregate fun. : Having

Select Sum(salary),did
from employee
where address like '_a%'
group by did
having sum(salary)>12000

15000	20
14500	30

Select Max(salary),address
from employee
where did in(10,30)
group by address
having Count(eid)>3

Eid	Ename	Salary	Address	did
1	ahmed	3000	cairo	10
2	ali	5000	cairo	10
3	eman	2000	cairo	10
4	khalid	1000	alex	10
5	yousef	4000	alex	10
6	sameh	5000	alex	10
7	mohamed	6000	alex	20
8	alaa	7000	alex	20
9	ola	4000	cairo	20
10	reem	2000	cairo	20
11	nada	9000	cairo	20
12	sayed	8000	mansoura	30
13	reham	1500	mansoura	30
14	sally	2000	mansoura	30
15	omar	3000	mansoura	30

NULL

Select Sum(salary),did
from employee
where address like '_a%'
group by did
having sum(salary)>12000

15000	20
14500	30

Eid	Ename	Salary	Address	did
1	ahmed	3000	cairo	10
2	ali	5000	cairo	10
3	eman	2000	cairo	10
4	khalid	1000	alex	10
5	yousef	4000	alex	10
6	sameh	5000	alex	10

Select Max(salary),address
from employee
where did in(10,30)
group by address
having Count(eid)>3

8000	mansoura
------	----------

12	sayed	8000	mansoura	30
13	reham	1500	mansoura	30
14	sally	2000	mansoura	30
15	omar	3000	mansoura	30

NULL

```

select count(*),count(st_id),count(st_lname),count(st_age)
from Student

```

Results

	(No column name)	(No column name)	(No column name)	(No column name)
1	14	14	12	13

لیه نتایج ال Count مختلفه في ال Columns

لأن ال aggregate funs. مبتاخدش ال Null

وتأكدنا على كده إن نتيجة ال 2 دول مختلفة

```

select avg(st_age)
from Student

```

=

Select sum(st_age) / count (st_age)
From Student


```

select sum(st_age)/count(*)
from Student

```

Results

	(No column name)
1	23

	(No column name)
1	22

لکن کده

The screenshot shows the SQL Server Management Studio interface. At the top, there are two queries:

```
select avg(isnull(st_age,0))  
from Student
```



```
select sum(st_age)/count(*)  
from Student
```

Below the queries, the results pane displays two rows of data:

	(No column name)
1	22

	(No column name)
1	22

انا خلیت کل ال 0 = Null ف کده بالنسباله مفیش Nulls خالص

ف نتیجه ال 2 Queries 2 طلعوا زی بعض

لکن ال Query دی ه تكون أبطأ

```
select sum(st_age)/count(*)  
from Student
```

لأن فيها 2 aggregate funs.

طيب انا بعمل aggregate fun. مع ال column group by لكل dept_id واحدة محتاج group by

```
select sum(salary),dept_id  
from Instructor  
group by dept_id
```

لكن هنا انا محتاج اعمل 2 group by

```
select sum(salary),dept_id,dept_name  
from Instructor  
group by dept_id
```

لأن انا معايا 2 columns مع ال aggregate fun.

ف المفروض تبقى كده

```
select sum(salary),dept_id,dept_name  
from Instructor  
group by dept_id  
group by dept_name
```

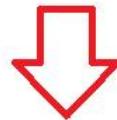
طيب لو ال 2 Tables دول في 2 Columns مختلفين

هحتاج اعمل Join -

```
select sum(salary),d.dept_id,dept_name  
from Instructor i inner join Department d  
on d.Dept_Id=i.Dept_Id  
group by d.dept_id,dept_name
```

وهنا انا استخدمت ال alias مع ال dept_id لأن ال dept_id بنفس الأسم في ال 2 Tables

```
select sum(salary),d.dept_id,dept_name  
from Instructor i inner join Department d  
on d.Dept_Id=i.Dept_Id  
group by d.dept_id,dept_name
```



معونة :-

وهنا هو بيعمل dept_name ثم لـ dept_id group by

```
select avg(st_age), *  
from Student  
group by *
```

```
-----  
select avg(st_age), st_id  
from Student  
group by st_id
```

وطبعا ال Queries 2 دول غلط أو ملهمش فايدة

لأن الآلتين معناهم إن أنا بعمل group by بال PK

وطبعا ده مش منطقى

لأن أنا لما بعمل group by بيكون لحاجات بتتكرر

لكن ال PK مبتكرش

```

select sum(salary),dept_id
from Instructor
group by dept_id

select sum(salary),dept_id
from Instructor
where salary>1000
group by dept_id

```

The screenshot shows the SQL Editor with two queries and their results in the Results pane.

	(No column name)	dept_id
1	20900.00	10
2	129000.00	20
3	21200.00	30

	(No column name)	dept_id
1	19000.00	10
2	129000.00	20
3	21000.00	30

طيب ايه الفرق بين ال 2 Queries دول ؟

الأولي : عمل ال 3 groups و جاب ال sum (salary)

الثانية : عمل ال 3 groups و جاب ال sum (salary) بس للي ال salary بتاعهم أكبر من 1000

بس للي ال salary بتاعهم أكبر من 1000

ف كده ال where بتتأثر على ال values اللي راجعة مع ال groups

لكن مش بتتأثر على تواجد ال group من عدمه

إلا في حالة واحدة لو ال where مسحت كل ال Rows اللي في ال group

في الحالة دي ال group هش هيظهر لأن كل ال Rows اللي جواه مش موجودة

طيب نو عاييز اشيل group كامل مش rows فقط

استخدم **Having**

```
select sum(salary),dept_id
from Instructor
group by dept_id

select sum(salary),dept_id
from Instructor
group by dept_id
having sum(salary)>100000
```

The screenshot shows the SQL Editor with two queries. The first query groups the salary by department ID and calculates the sum. The second query does the same but includes a HAVING clause to filter only those groups where the sum of salary is greater than 100,000. Below the editor is a Results pane displaying the output of the second query.

(No column name)	dept_id
1 209000.00	10
2 129000.00	20
3 21200.00	30

(No column name)	dept_id
1 129000.00	20

هنا هو عمل ايه ؟

الأولي : عمل ال 3 groups و جاب ال (sum (salary))

الثانية : عمل ال 3 groups و جاب ال (sum (salary))

ورجعلي بس اللي ال (sum (salary)) بتاعهم أكبر من 100000

طبعاً احنا عارفين إن احنا بنعمل column group by لـ select اللي في ال aggregate fun. مع ال اللي مع ال

لـ Having
هل لازم ال aggregate الموجودة في ال select تكون هي هي ال aggregate الموجودة في ال Having ؟
- لا

```
select sum(salary),dept_id
from Instructor
group by dept_id
having Count(ins_id)<6
```

(No column name)	dept_id
129000.00	20
21200.00	30

تأكد للمعلومة :-

Rows condition على ال Where - بتعمل aggregate funs. مش بيجي معاهها -

groups condition على ال Having - بتعمل aggregate funs. مش بيجي معاهها غير ال group by غير ف وجود rows conditions على ال group مش ال لأنها بتعمل

--Subqueries

```
select *
from Student
where st_age < (select avg(st_age) from student)
```

St_Id	St_Fname	St_Lname	St_Address	St_Age	Dept_Id	St_super
1	Ahmed	Hassan	Cairo	20	10	NULL
2	Amr	Magdy	Cairo	21	10	1
3	Mona	Saleh	Cairo	22	10	1
12	Noha	Omar	Cairo	21	40	NULL

معلومات :-

- ممكن تتكتب ف أي مكان Sub Query

سواء ف ال (select , from , where , Having)

- هل ال (inner join) وال (outer join) لازم يكونوا من نفس ال ? Table

- لا مش لازم (ممكن يكونوا من 2 Tables مختلفين)

- هل ال aggregate دى بيستخدم عشان تحل مشكلة ال sub Query بس ؟ funs.

- لا مش لازم

طيب ايه الدليل ؟

The screenshot shows a SQL query in the query editor:

```
select dept_name
from Department
where dept_id in (select distinct(dept_id)
                    from Student
                    where dept_id is not null)
```

The results pane shows the following data:

dept_name
SD
EL
Java
MM

هنا مثلا هو عايز اسماء الأقسام اللي فيها طلاب
وممكن برضو يعمل العكس ويجيب اسماء الأقسام اللي مفيهاش طلاب

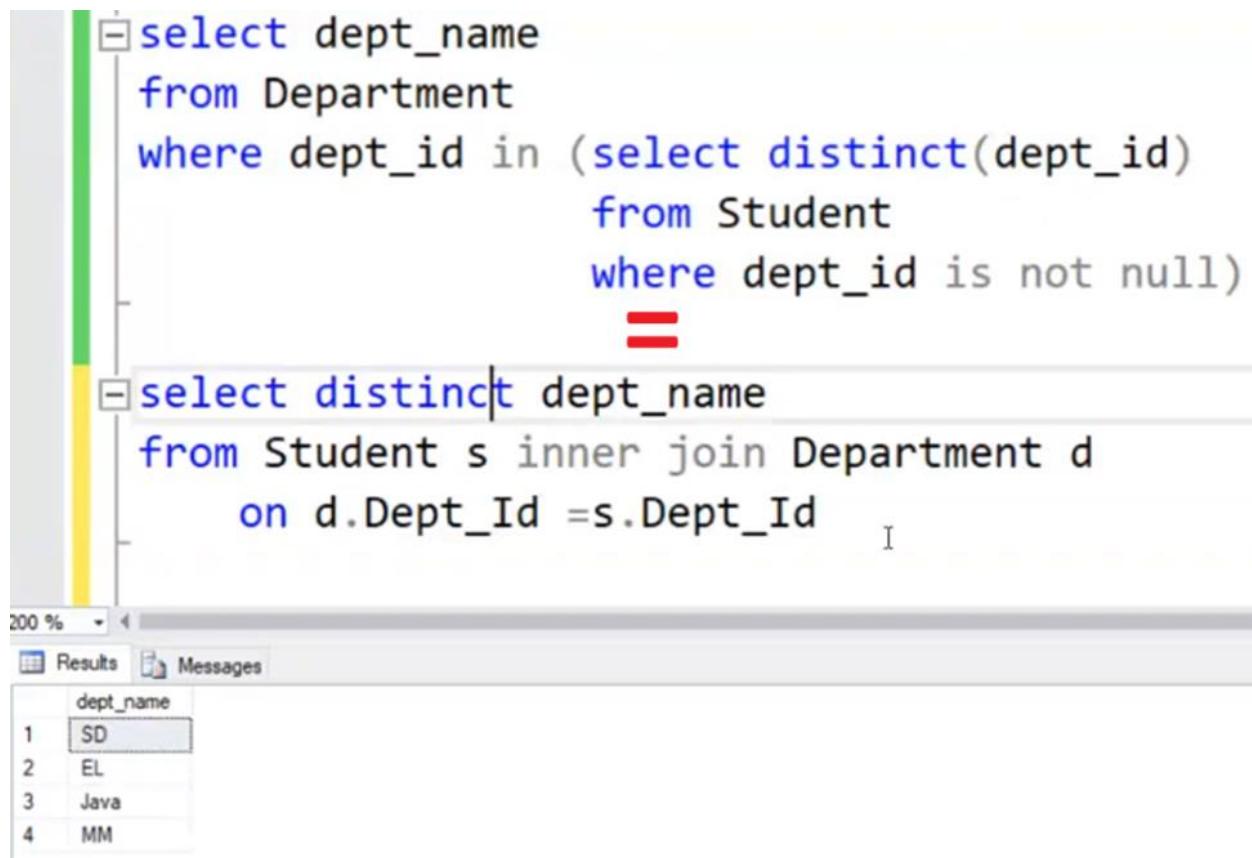
The screenshot shows a SQL query in the query editor:

```
select dept_name
from Department
where dept_id not in (select distinct(dept_id)
                        from Student
                        where dept_id is not null)
```

The results pane shows the following data:

dept_name
Unix
NC
EB

وممكن برضو أعمل اخر صورة بال join لأنها في الأصل عبارة عن Join
وف الأساس استخدام ال Join أسرع وال Performance بتاعها أفضل



```
select dept_name
from Department
where dept_id in (select distinct(dept_id)
                   from Student
                   where dept_id is not null)
=
```

```
select distinct dept_name
from Student s inner join Department d
on d.Dept_Id =s.Dept_Id
```

dept_name
SD
EL
Java
MM

ف كده طبعاً لو هعمل علاقة بين ال **Join** مش **Sub-Query** بين 2 Tables

طيب لو العلاقة بين ال Tables دول محتاجة إني أعمل **join** لأكتر من Table
ف هنا ف الغالب هيكون مفضل ليها ال **Joins** لأن **Sub-Query** لآن ه تكون كتير أوي

```
delete from Stud_Course  
where st_id=1
```

هل ممكن استخدام Sub-Query مع DML

ا

--Subquery + DML

```
delete from Stud_Course  
where st_id in (select st_id from Student where St_Address='cairo')
```

```
--union family  
union all      union      intersect    except
```

union intersect except |
ال 3 دول

بيجيبوا ال distinct
لكن

union all

بيجيب كله حتى لو في Rows متكررة

معلومات :-

عادي اعمل علاقة من دول بين 2 Tables مختلفين
ومش لازم يكون بينهم علاقة أصلا
لكن لازم يكون نفس ال Data Type
ولازم يكون عدد ال Select Columns في ال الأولى
يكون نفس عدد ال Select Columns في ال الثانية

```
    |- select st_fname  
    |   from Student  
    |   union all  
    |- select ins_name  
    |   from Instructor
```

```
select st_fname, st_id  
from Student  
union all  
select ins_name, ins_id  
from Instructor
```

```
select st_fname  
from Student  
union  
select ins_name  
from Instructor
```

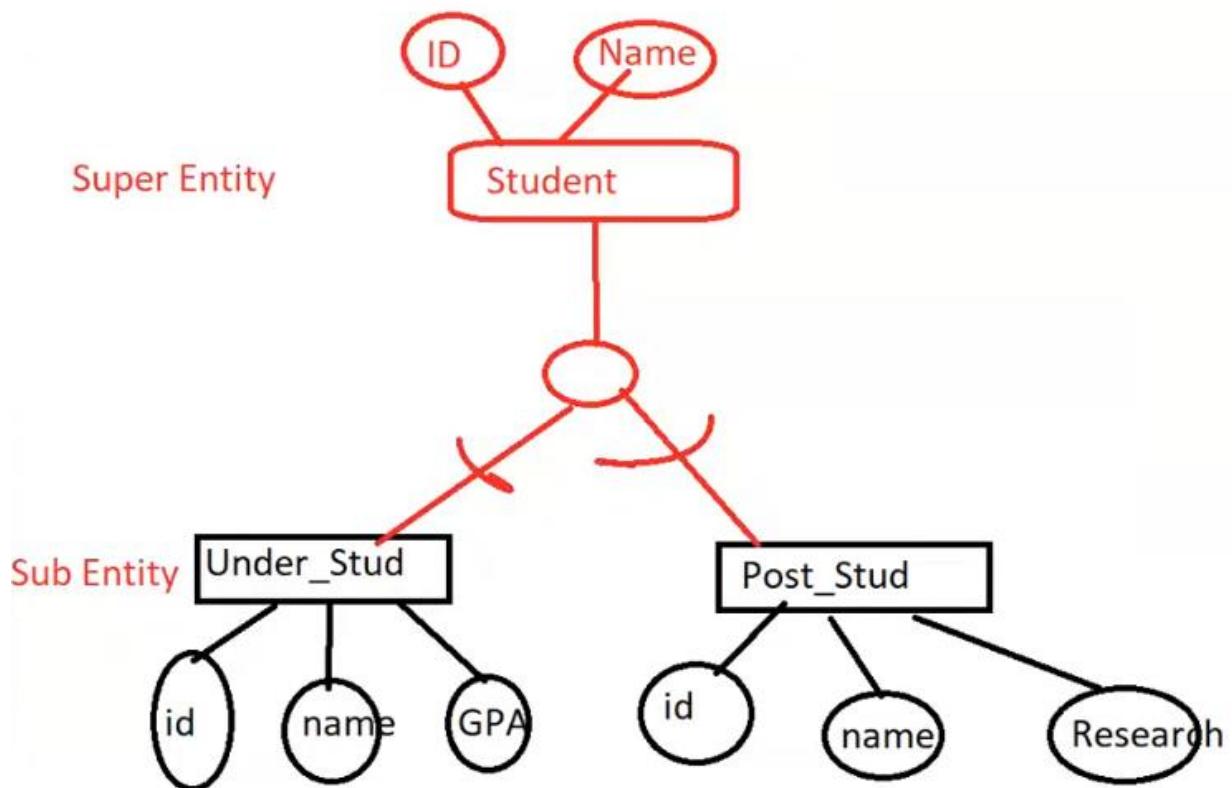
```
select st_fname  
from Student  
intersect  
select ins_name  
from Instructor
```

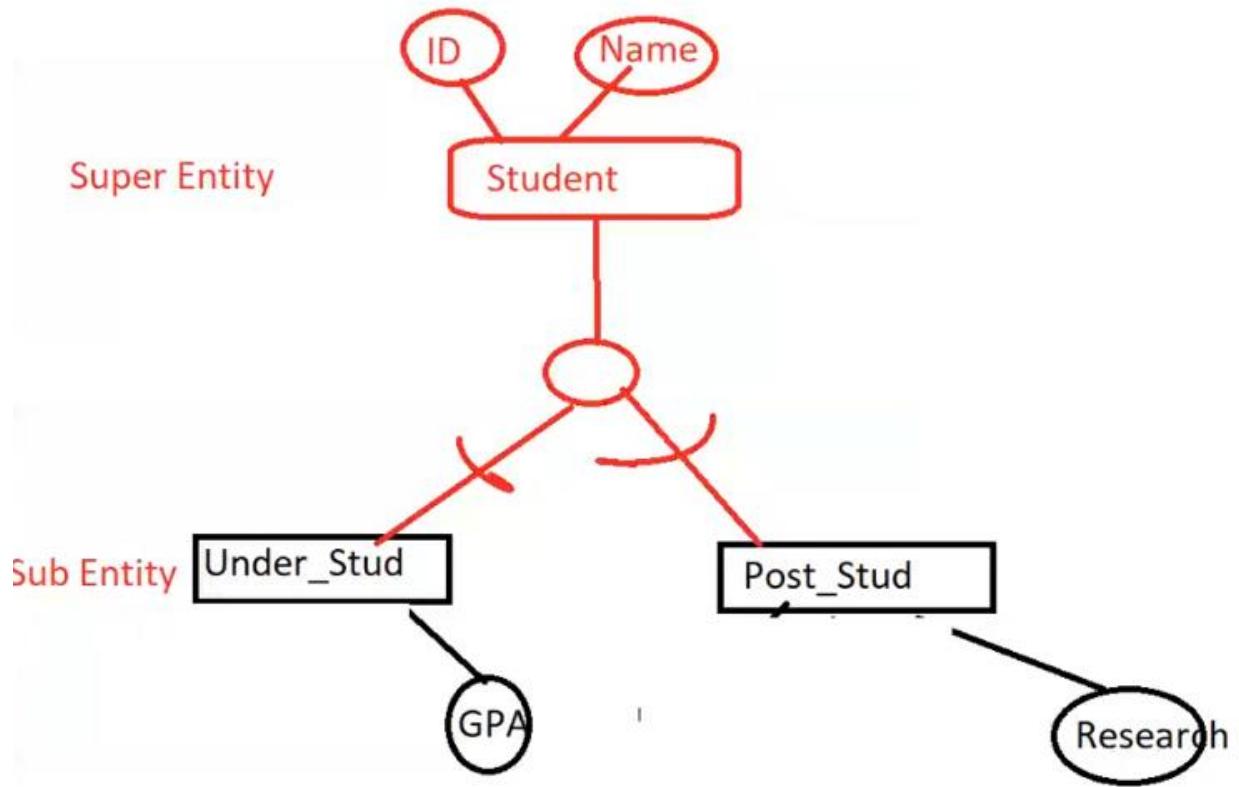
```
select st_fname, st_id  
from Student  
intersect  
select ins_name, ins_id  
from Instructor
```

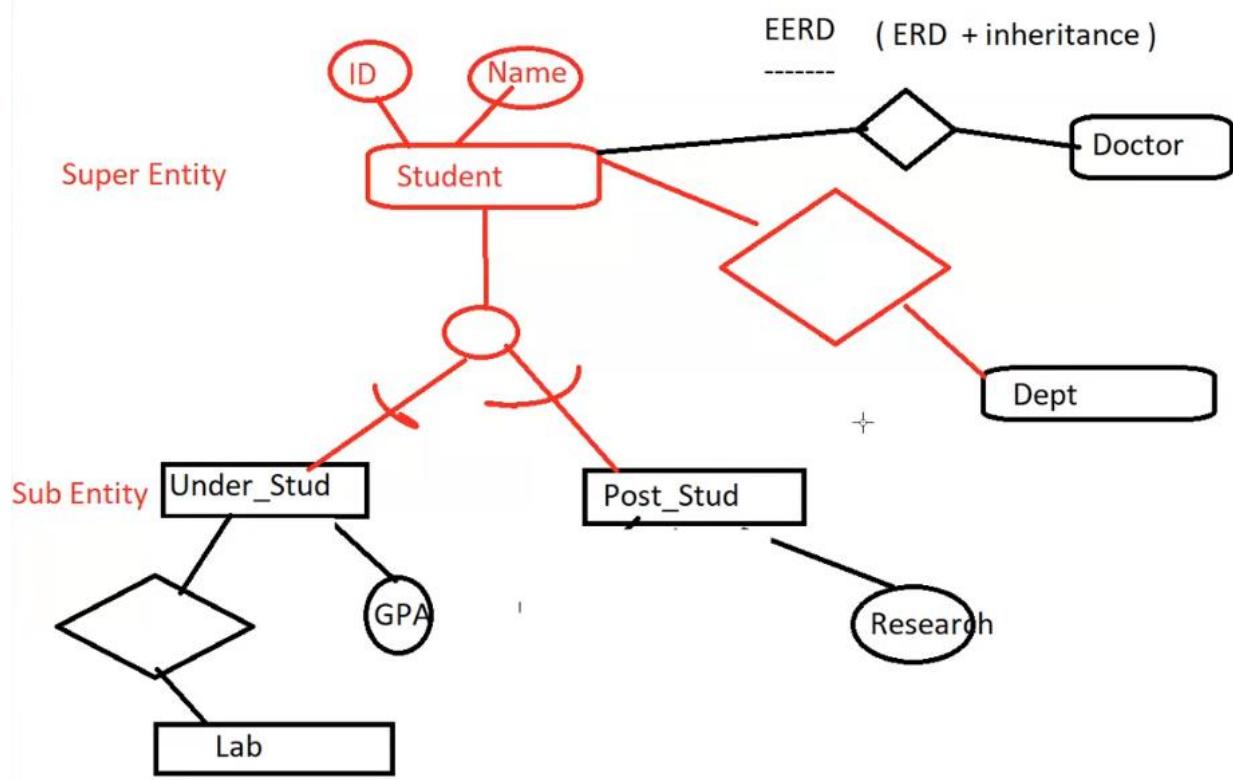
```
select st_fname  
from Student  
except  
select ins_name  
from Instructor
```

(هات اللي موجود حاجة ومش موجود في الثانية) Except = Minus

EERD (ERD + inheritance)







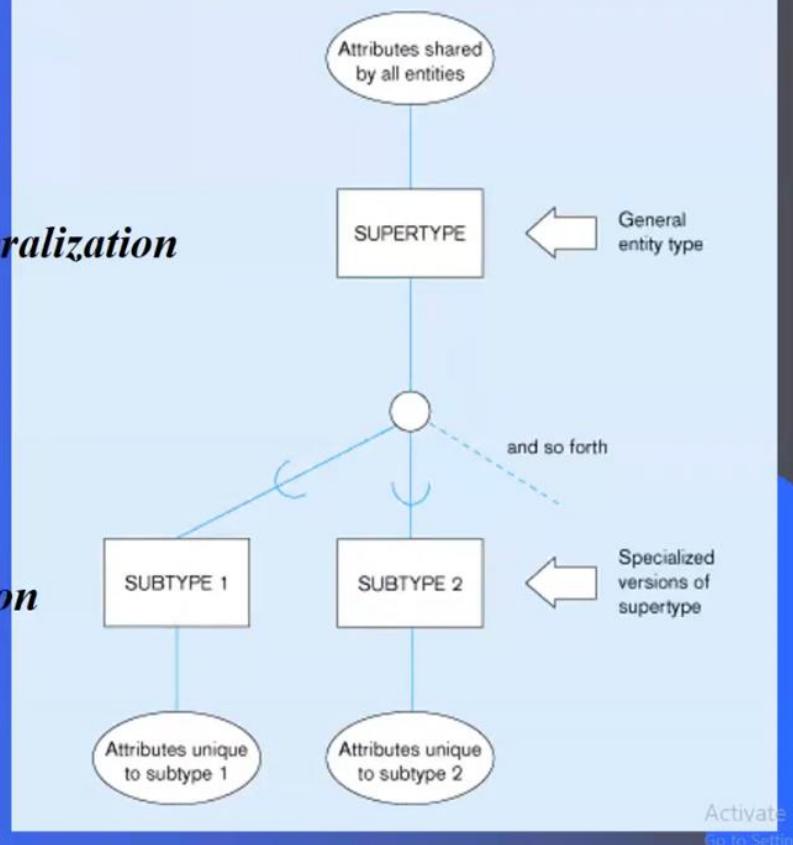
Supertypes and Subtypes

- ↪ **Subtype:** A subgrouping of the entities in an entity type which has attributes that are distinct from those in other subgroupings
- ↪ **Supertype:** An generic entity type that has a relationship with one or more subtypes
- ↪ **Inheritance:**
 - Subtype entities inherit values of all attributes of the supertype
 - An instance of a subtype is also an instance of the supertype

Basic notation for supertype/subtype relationships

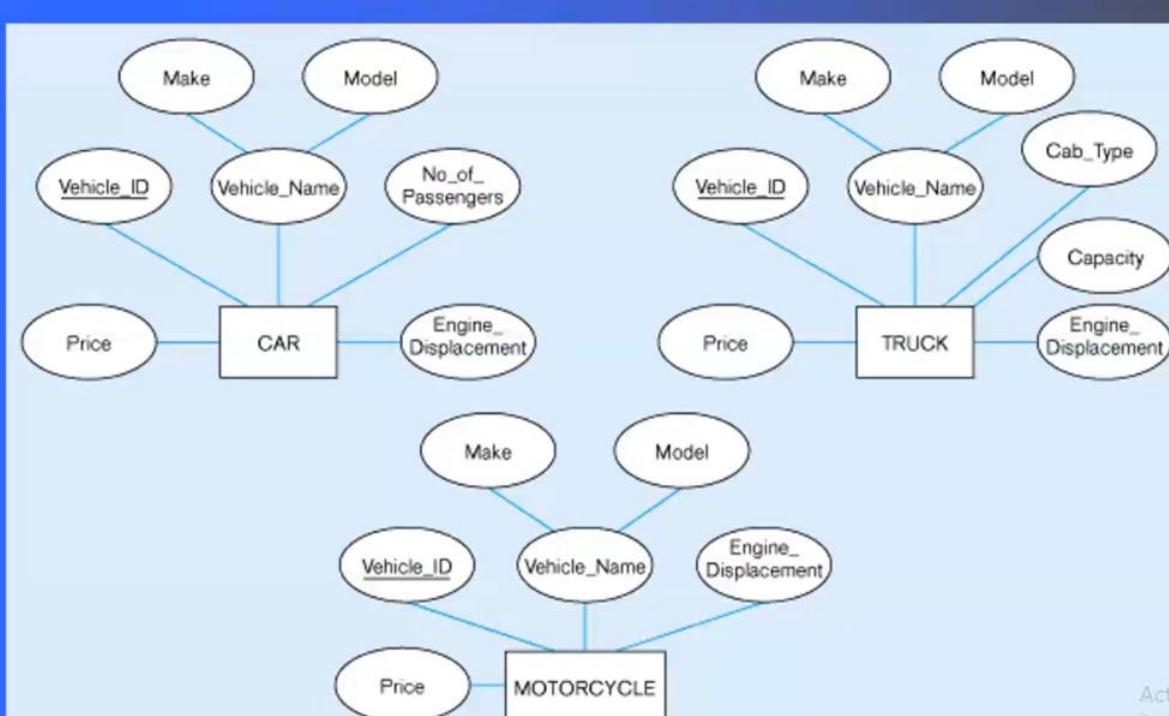
Generalization

Specialization



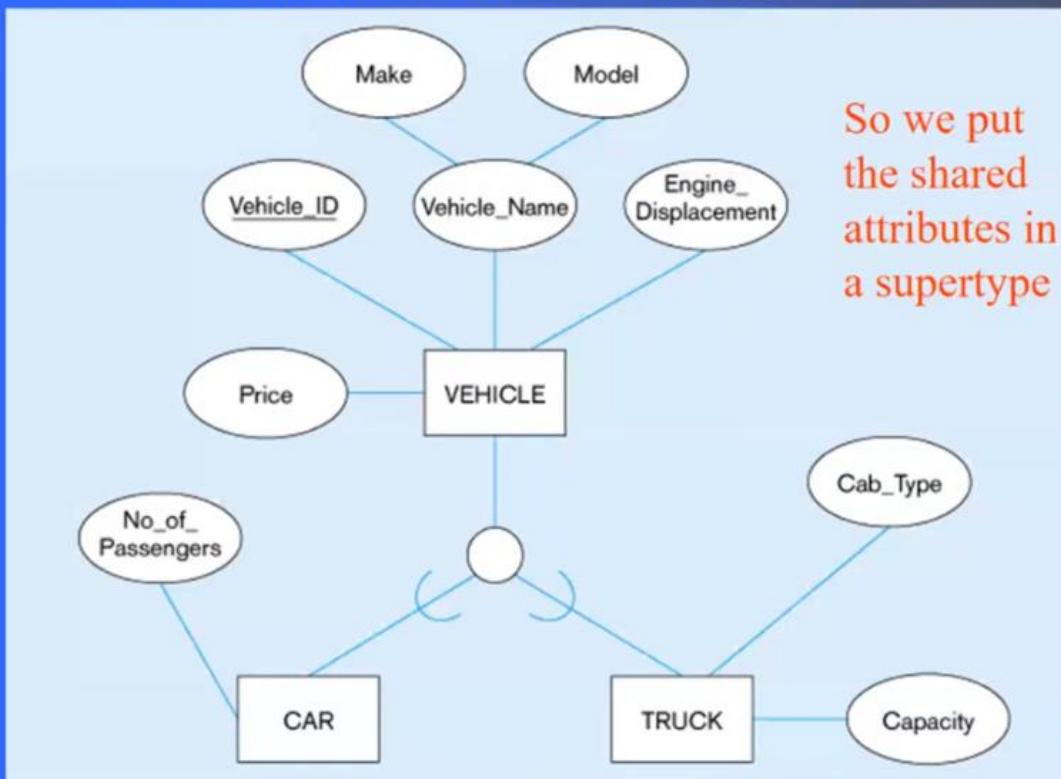
Example of generalization

(a) Three entity types: CAR, TRUCK, and MOTORCYCLE



Activate
Go to setti...
5

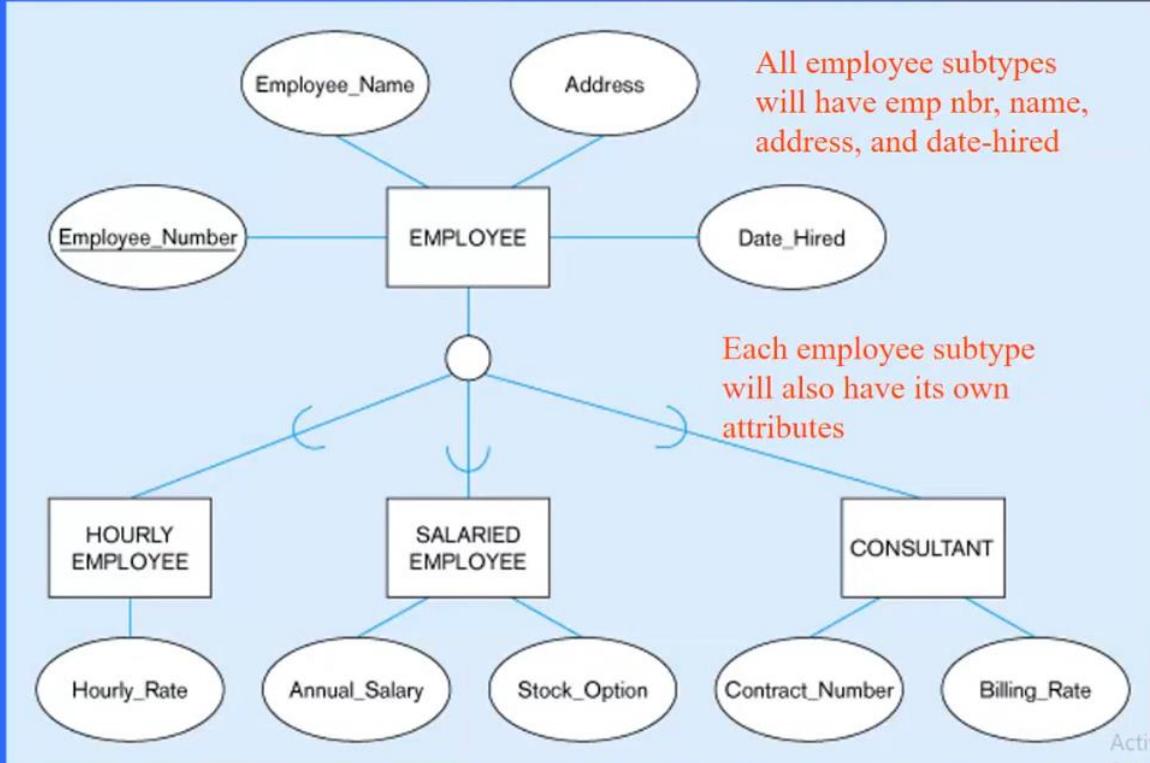
Generalization to VEHICLE supertype



Note: no subtype for motorcycle, since it has no unique attributes

Employee supertype with three subtypes

Employee supertype with three subtypes

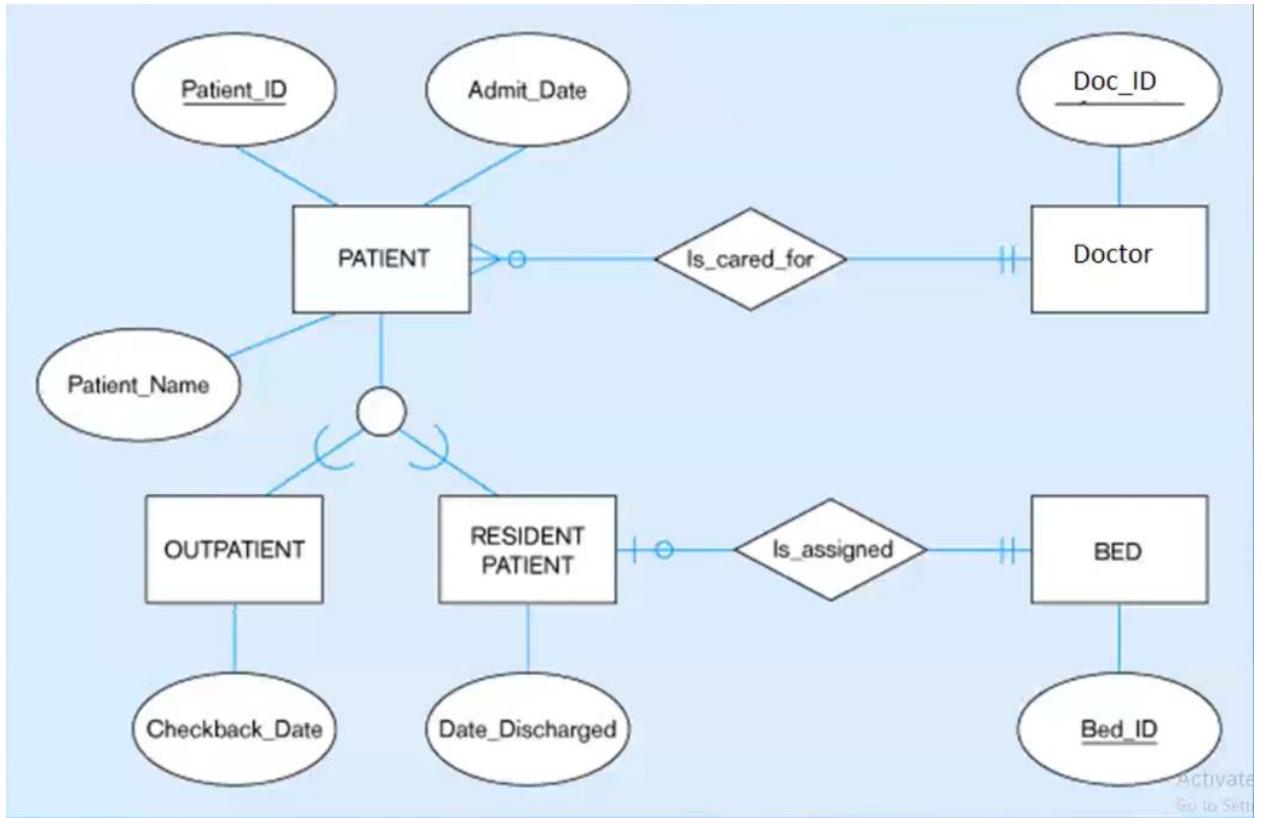


Relationships and Subtypes

Relationships at the *supertype* level indicate that all subtypes will participate in the relationship

The instances of a *subtype* may participate in a relationship unique to that subtype. In this situation, the relationship is shown at the subtype level

Supertype/subtype relationships in a hospital



Constraints in Supertype

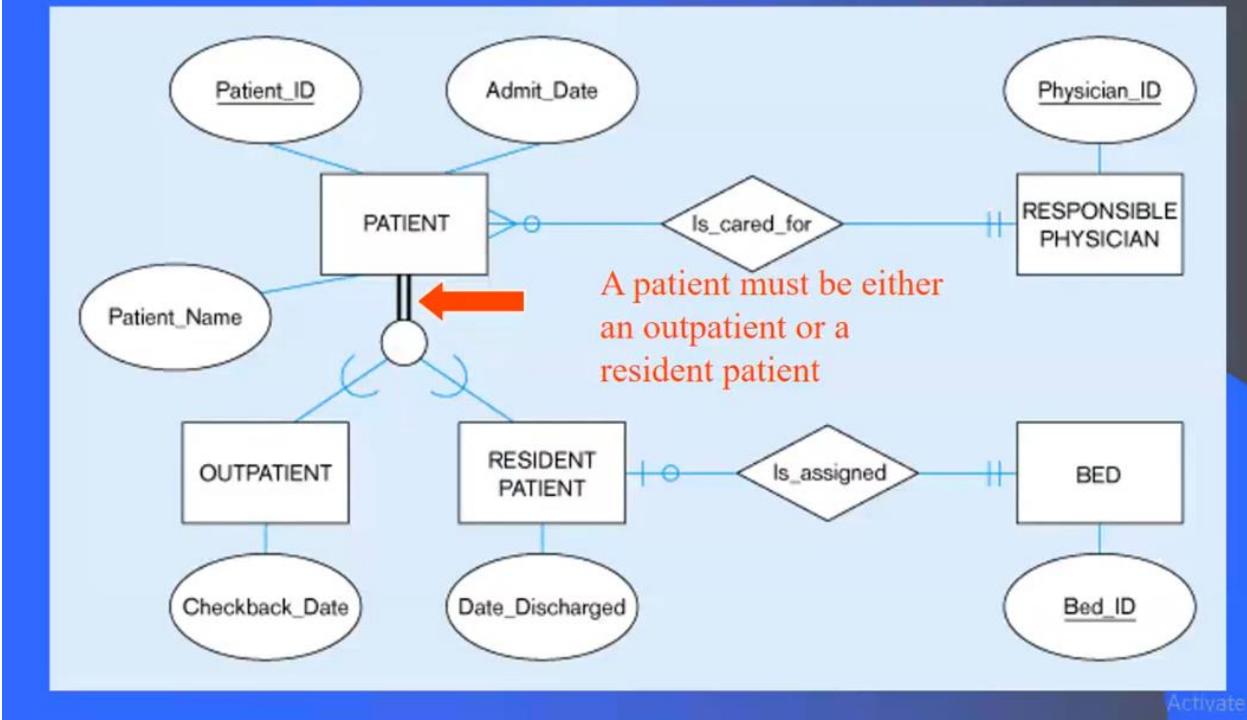
Completeness Constraints:

- Total Specialization Rule: Yes (double line)
- Partial Specialization Rule: No (single line)

Disjointness Constraints:

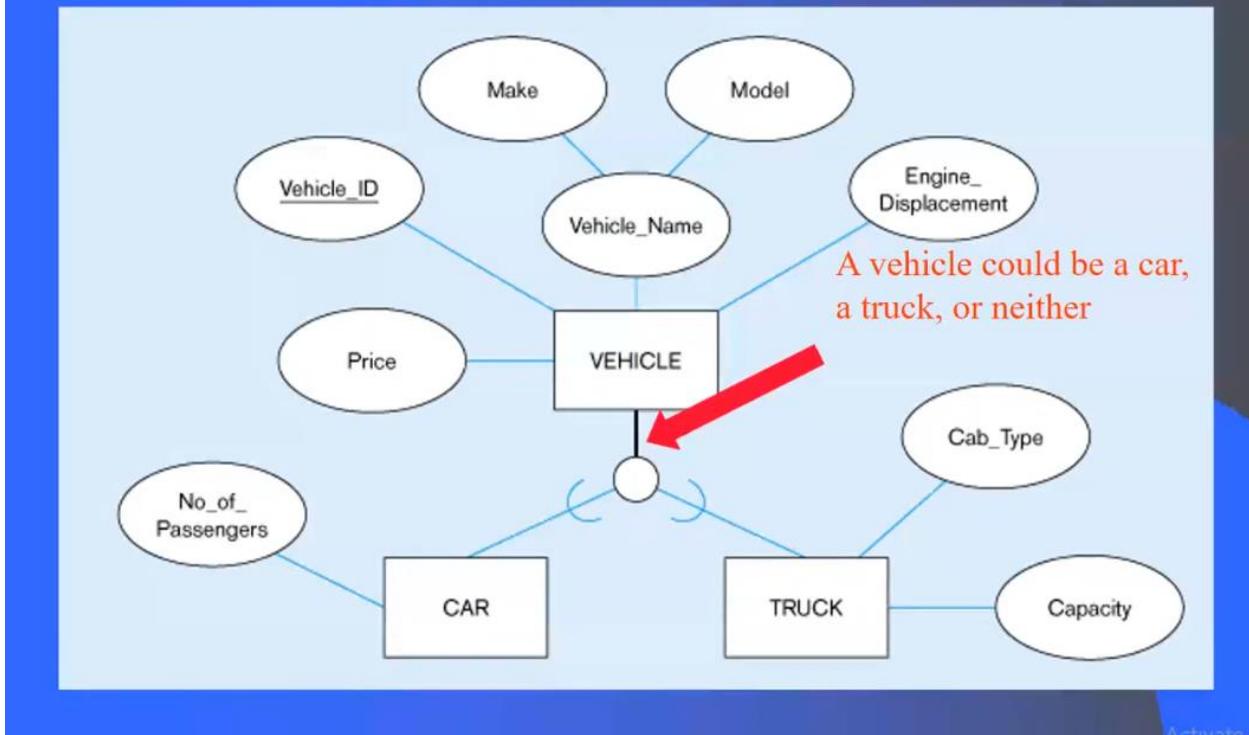
- Total Disjoint (d)
- Overlap Rule (o)

Total specialization rule



يعني كل ال **Columns** الموجودة في **Patient** ي إما نوعها **Resident Patient** ، ي إما **Outpatient**

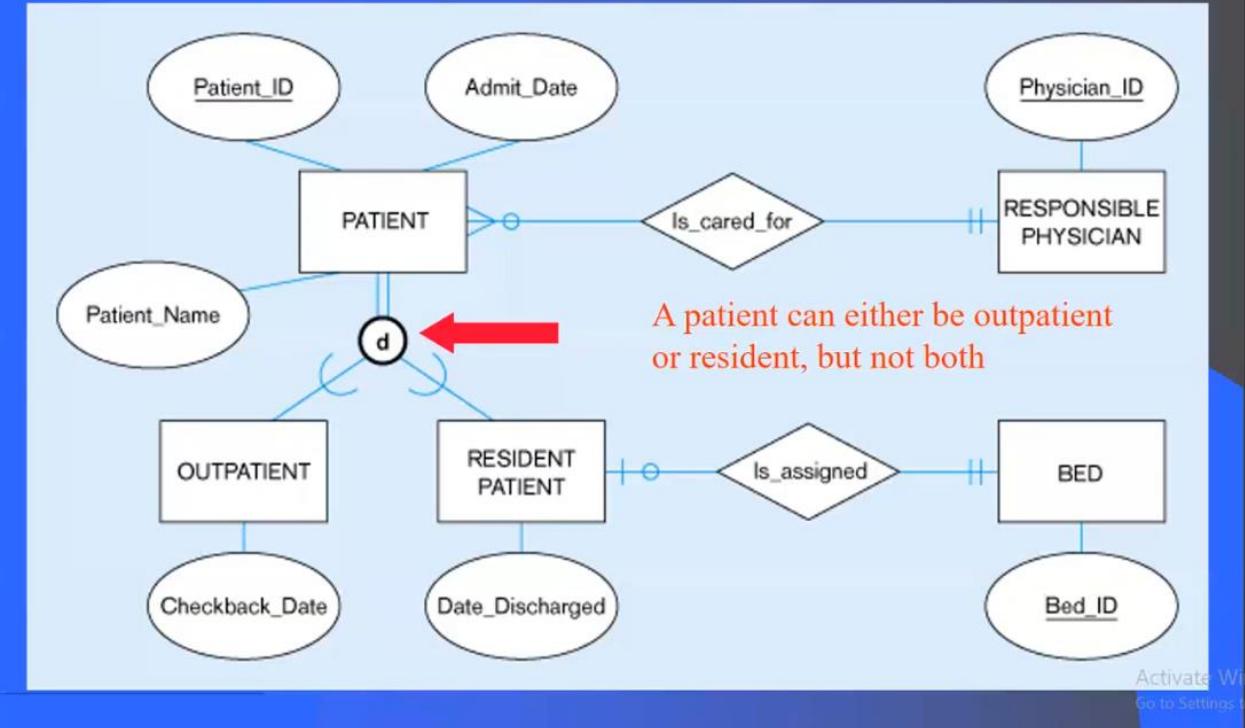
Partial specialization rule



لكن دي Partial عشان مش كل ال Columns اللي ف ال Vehicle
ي إما نوعها Car ، ي إما Truck بس
لكن في نوع تالت مخفي جوا ال Vehicle
اللي هو ال Motor Cycle

Examples of disjointness constraints

Disjoint rule

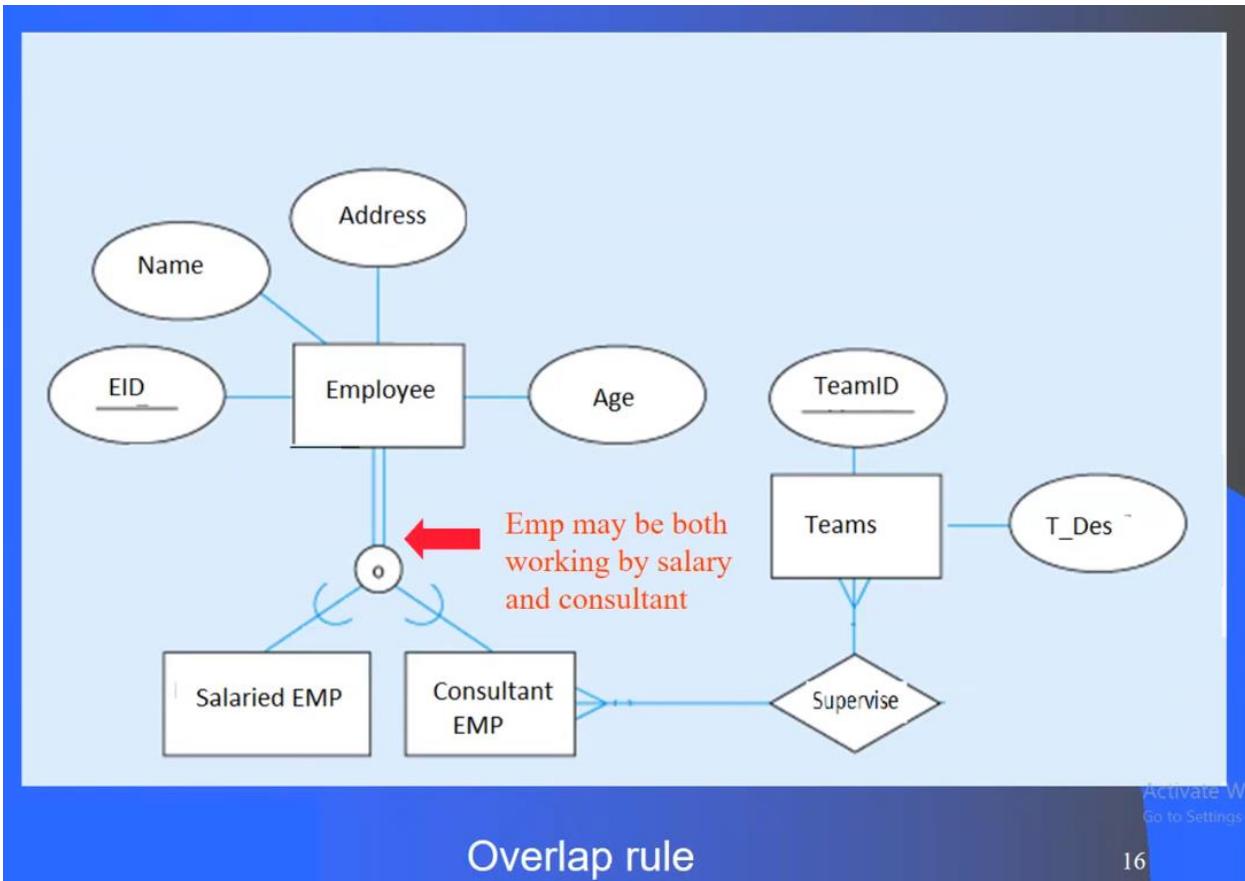


هل ال Patient ينفع يبقى Resident Patient و Outpatient في نفس الوقت ؟

أكيد لا

يعني كده أي Row من اللي فوق ف ال Patient Super مبيتكررش تحت

يعني ال 1 Child موجود في 1 Parent



لكن هنا هل ال **Employee** ينفع يبقى **Salaried EMP** و **Consultant EMP** في نفس الوقت ؟

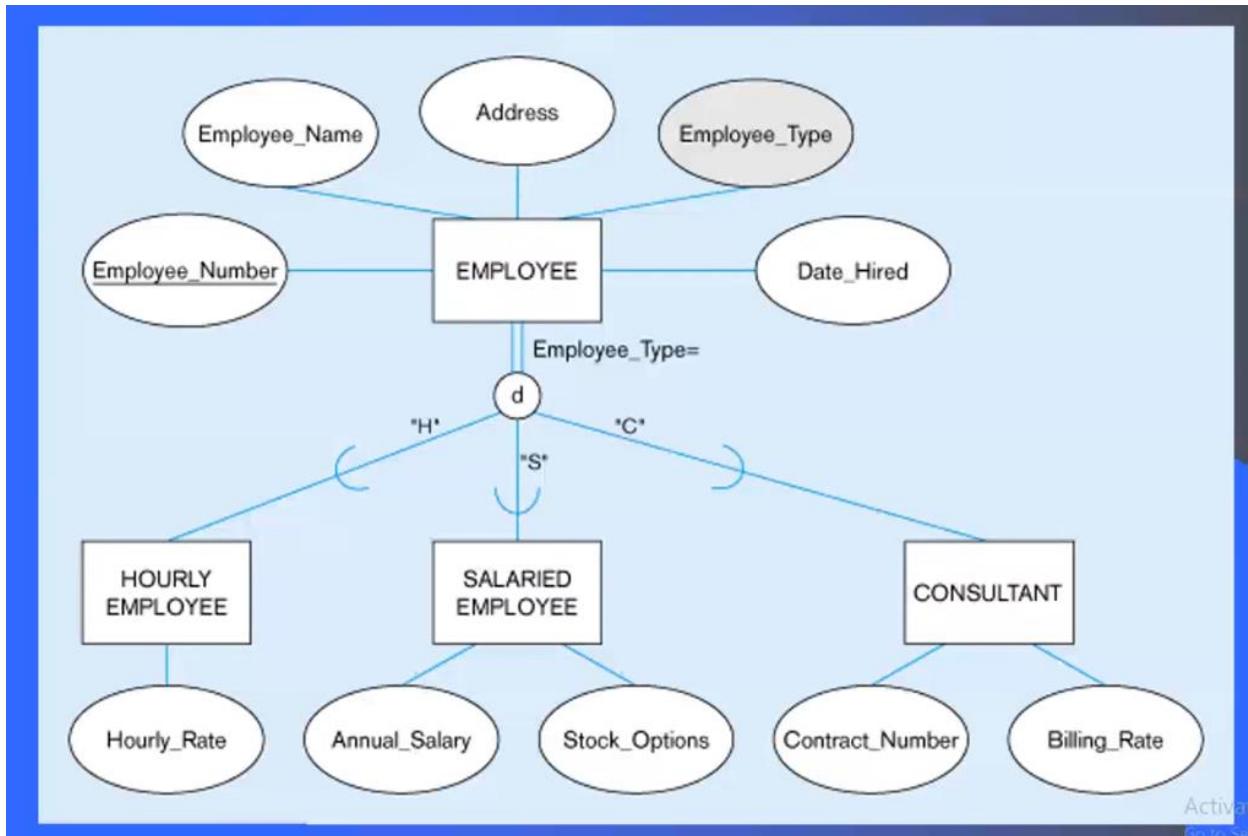
ا

يعني كده ممكن Row من اللي فوق ف ال **(Employee)** Super ف ال **Salaried EMP** و **Consultant EMP** يتكرر تحت عادي (لو موظف بيقبض بالشهر وف نفس الوقت)

يعني ال **1 Parent** ممكن يتكرر في أكثر من **1 Child**
لأن بينهم Overlap

discriminator

Introducing a subtype discriminator (**disjoint** rule)



هنا لو حبيت أعرض كل أسامي الموظفين (Employee _Name)

? Table من كام Select هـ

(Employee) ١ -

طیب لو حبیت اعرض مجموع ال (Annual_Salary)
ھے من کام Select ؟
(Salaried Employee) ۱ -

طیب لو حبیت اعرض کل أسامي الموظفين (Employee_Name)
وال (Annual_Salary)
ھے من کام Select ؟
(Annual_Salary) و (Employee) ۲ -

طیب لو حبیت اعرض کل أسامي الموظفين (Employee_Name)
وال (Contract_Number) وال (Annual_Salary)
ھے من کام Select ؟
(Consultant) و (Annual_Salary) و (Employee) ۳ -

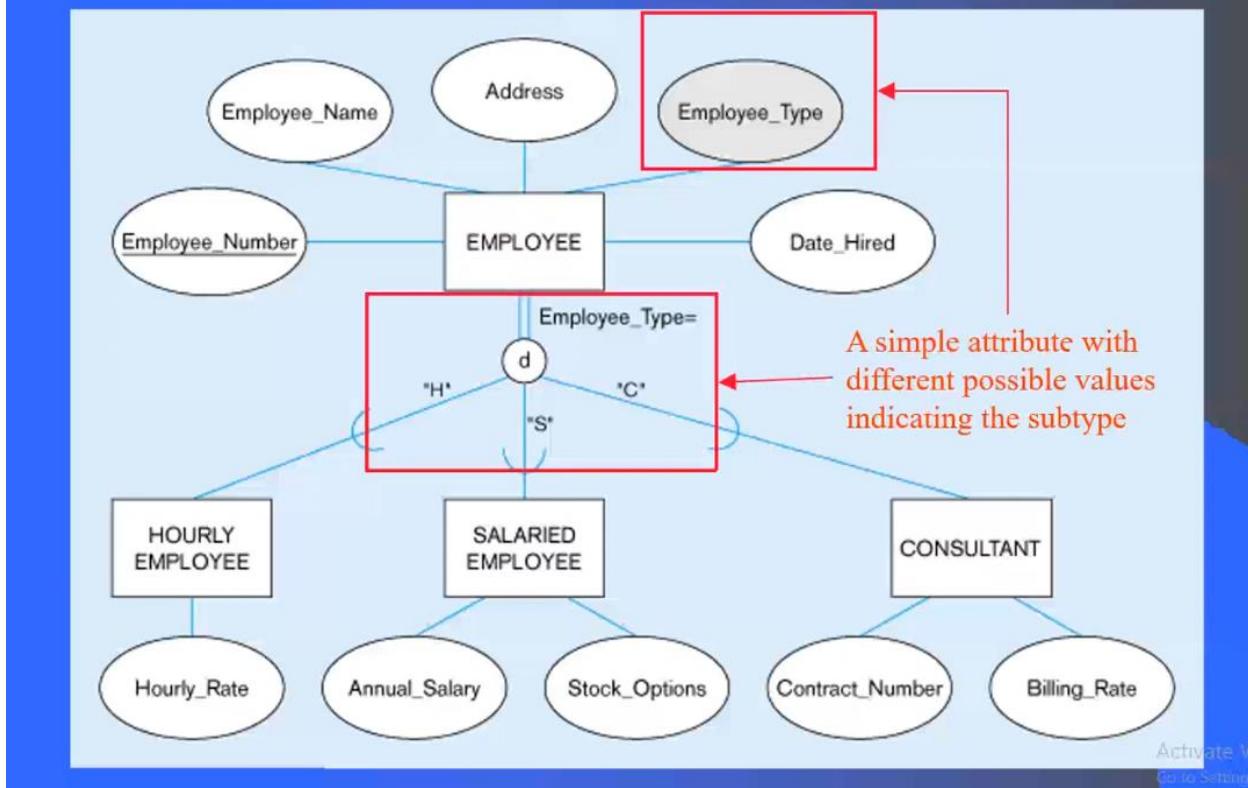
طبعاً كل اللي استخدمته ف 2 Tables أو اکتر دي **Joins**

طيب لو حبيت اعرض كل أسامي الموظفين (Employee _Name)
اللي بيقبضوا (Annual Salary)
لكن مش عايز اعرف ال (Annual Salary) كام ،
هـ من كام Select ؟

- هنا المفروض برضو اعمل Join
- لكن انا هعمل حاجة اسمها Type
- هنعمل Column اسمه Employee_Type ونحطه ف ال Super
- واحظ جواها H , S , C
- بحيث تكون ال Employee_Type متقسمة لـ ٣ أنواع

(disjoint rule)

Introducing a subtype discriminator (disjoint rule)



ف كده لما اقول 'H' Where Employee_Type = 'H'

يجيبلي ال Hourly Employee

ولما اقول 'S' Where Employee_Type = 'S'

يجيبلي ال Salaried Employee

ولما اقول 'C' Where Employee_Type = 'C'

يجيبلي ال Consultant

ف كده انا حطيت معلومة في ال Super Entity

تدل على نوع ال Super Entity بالنسبة لـ Sub_Entities اللي موجودين عندي

طيب انا استفدت ايه من ال discriminator ؟

إن انا مش مضطر اعمل join

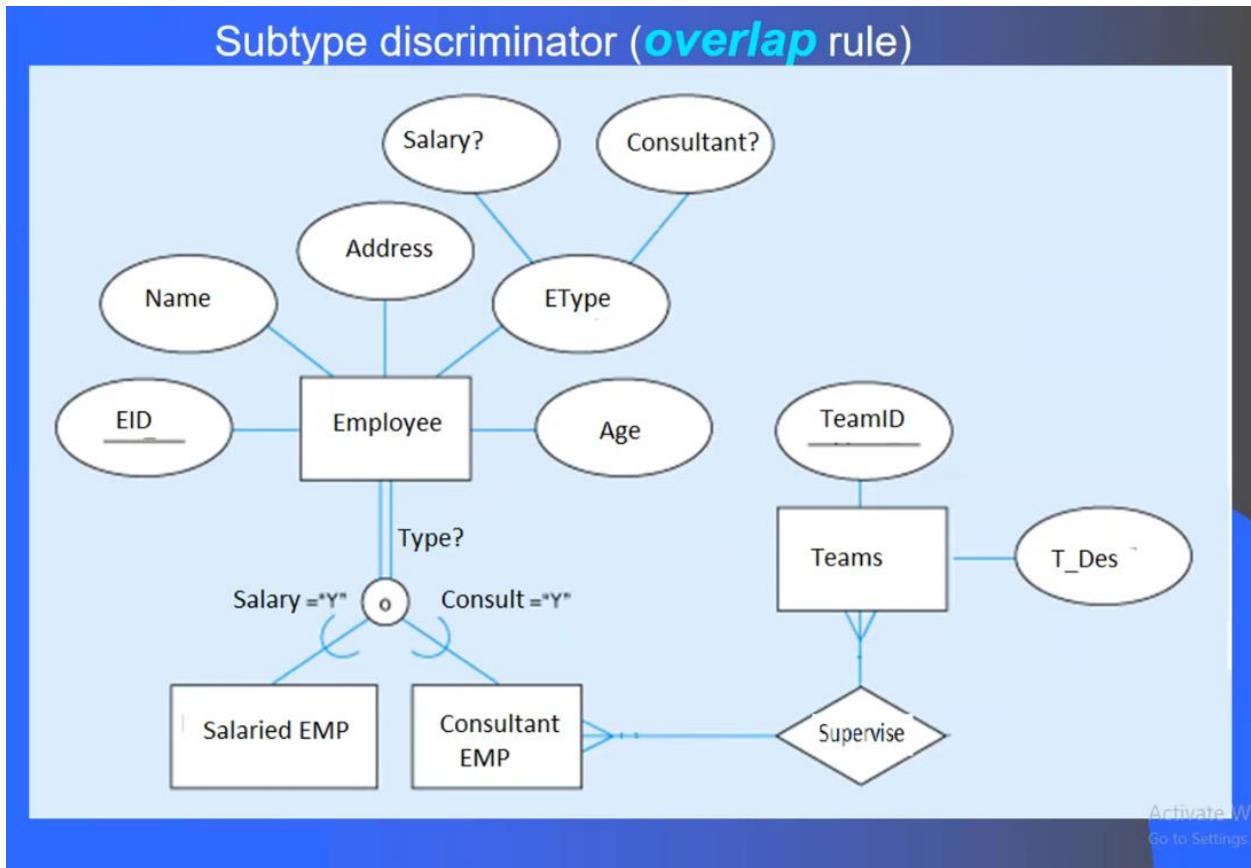
ف ال Query بالطريقة دي ه تكون أسرع

طیب نو (overlap)

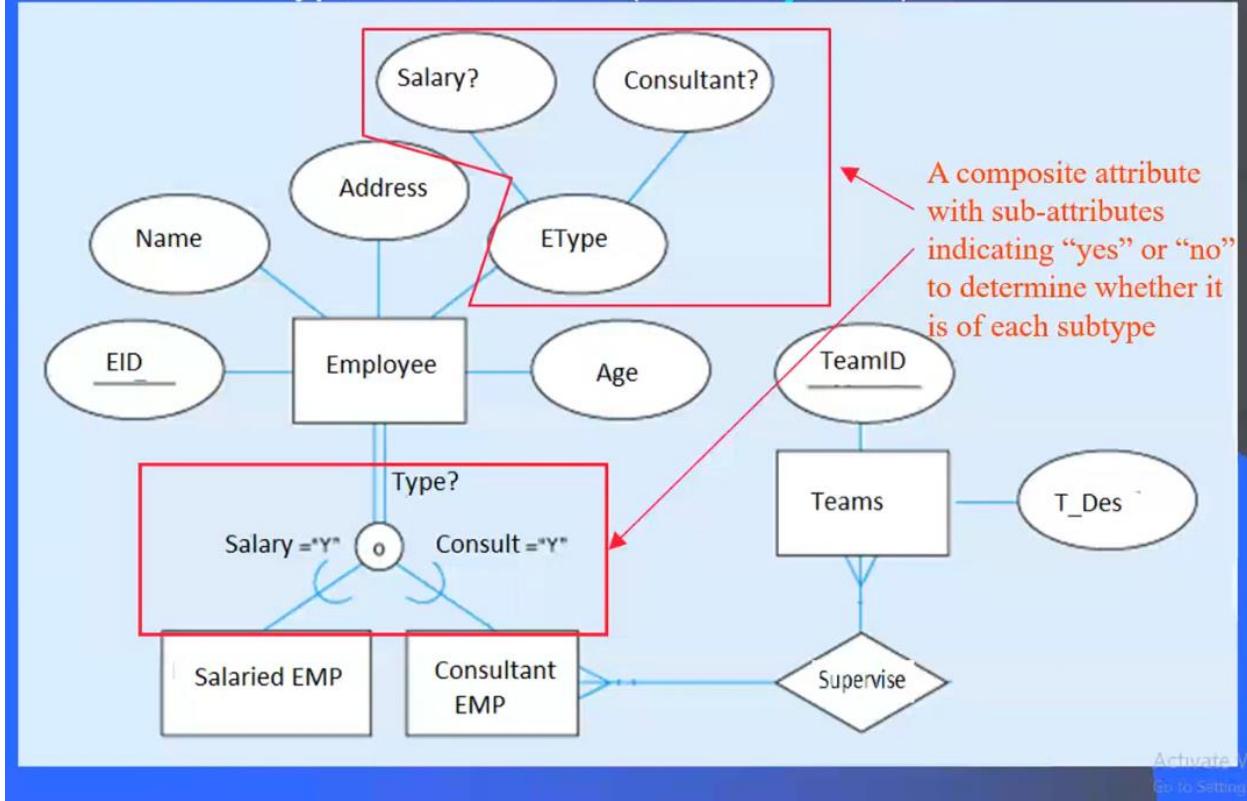
(**overlap** rule)

بدل م هعمل (Attribute) **Type**

هضرر اعمل (Composite Attribute) **Composite Type**



Subtype discriminator (**overlap** rule)



ف هنا لو (Y , Y) يبقى انا بقبض وشغال Consultant

ولو (N , Y) يبقى انا بقبض بس

ولو (Y , N) يبقى انا شغال بس Consultant

ولو (N , N) يبقى انا مش بقبض ولا شغال Consultant

ف كده من الآخر

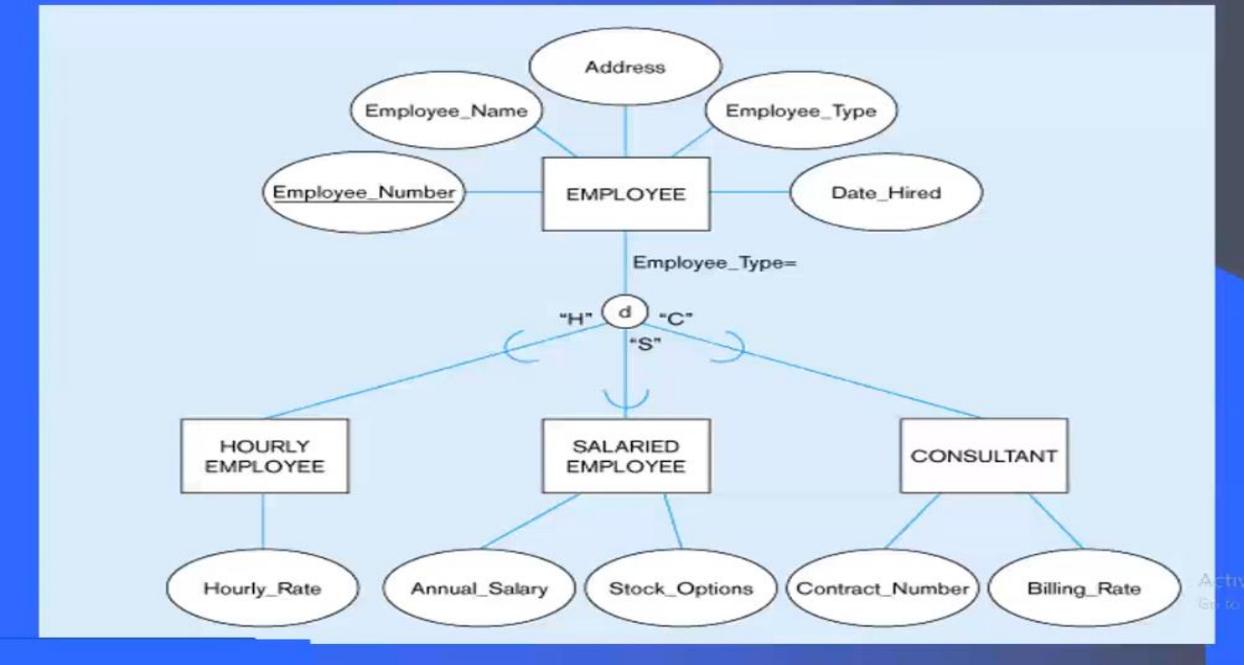
عمل ال disjoint (Single Attribute) discriminator

وبعمل ال Overlap (Composite Attribute) discriminator

احنا کده عملنا ال Design

ازاي بقی نعمل ؟ Mapping

Transforming EER Diagrams into Relations

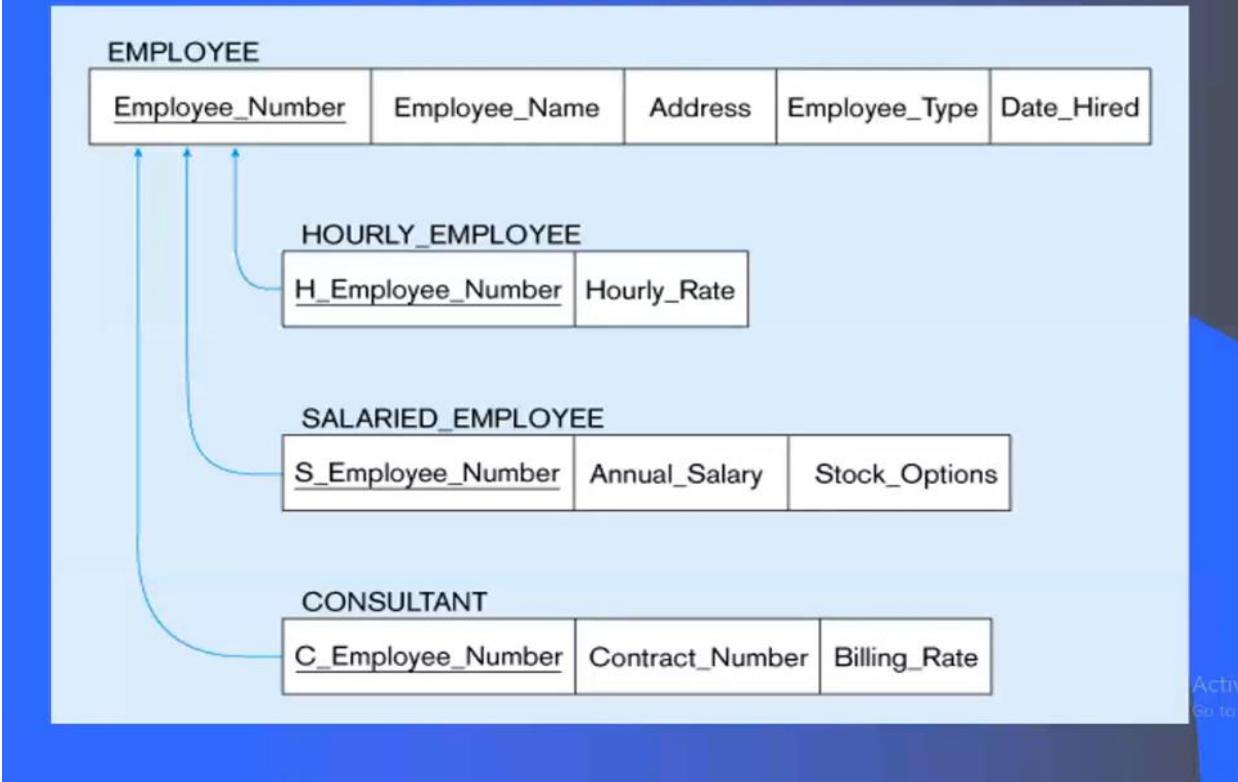


عدد ال Sub Tables يكون نفس عدد ال Entites ف ال Super Tables وال
يعني ف المثال ده عندي 4 Tables

ال Primary Key ف ال (Employee_Number) هيكون 4 Tables

ف ال (Employee) هيبي PK ف جدول ال (Employee_Number)
وال (Employee) هيبي PK و FK في نفس الوقت
في الجداول (Salaried Employee) و (Hourly Employee) و
(Consultant)

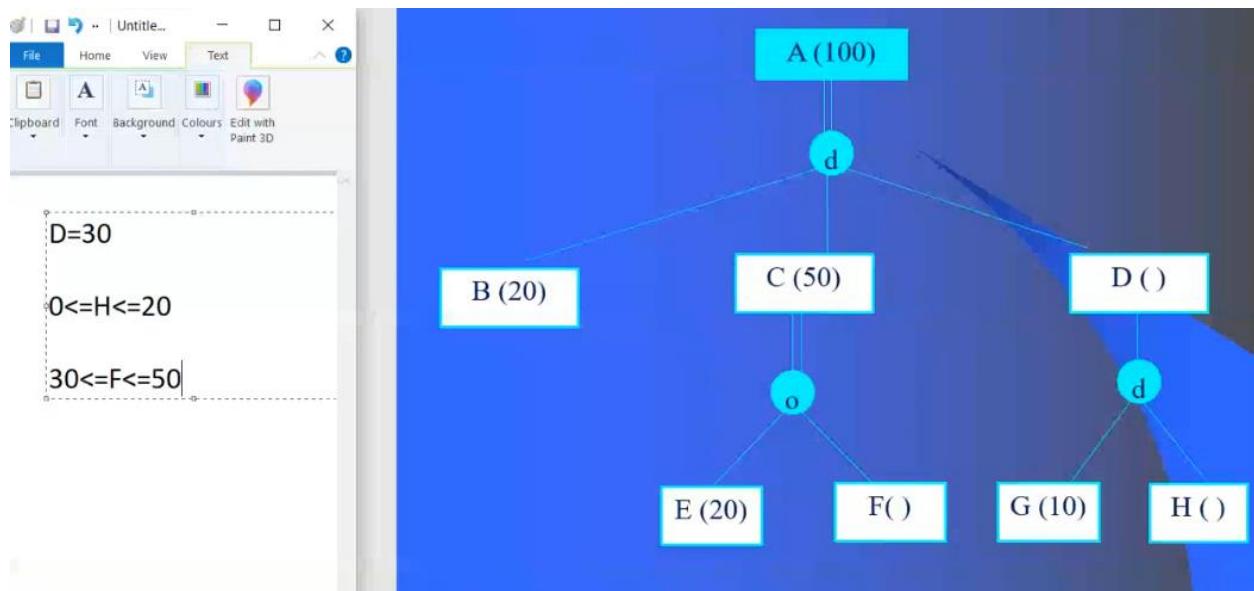
Mapping Supertype/subtype relationships to relations



ال (Employee_Number) هي PK و FK في نفس الوقت
في الجداول (Salaried Employee) و (Hourly Employee) و (Consultant)

PK : لأنه مينفعش يتكرر
FK : لأنه لازم يكون بي match قيمة من ال Parent Table (Employee)
(يعني مثلاً مدرس احط ال Annual Salary لموظف مش موجود)
(ف اكيد لازم ي match قيمة من ال [Super Entity] Parent Table)

مثال



D = 30

عشان هي disjoint و Total

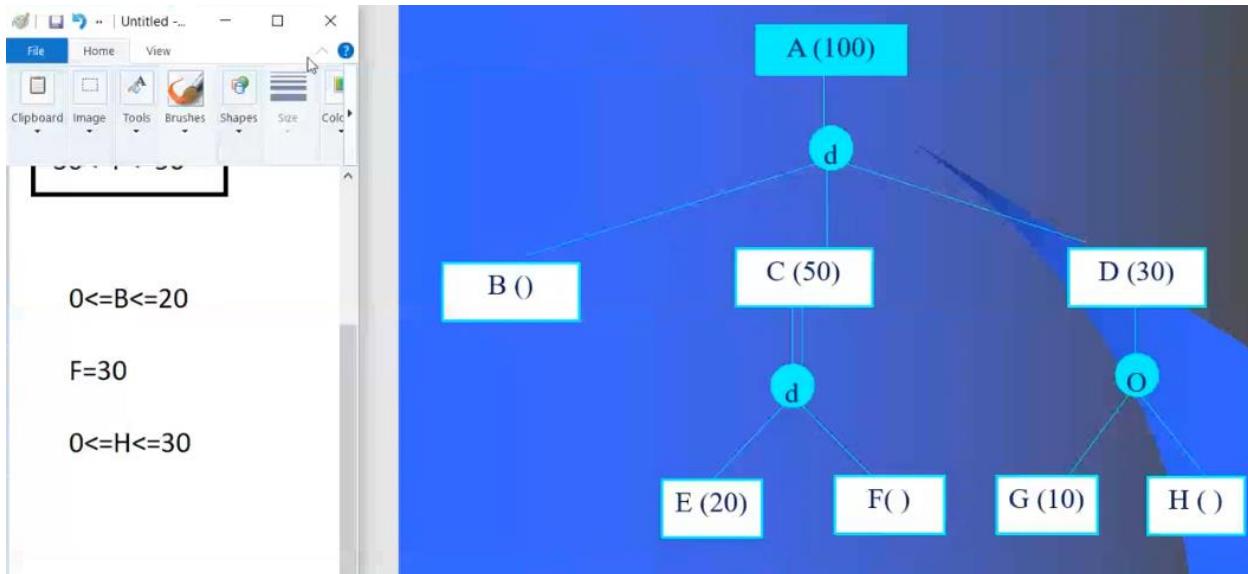
0 <= H <= 20

عشان هي disjoint و Partial

30 <= F <= 50

عشان هي overlap و Total

مثال :-



0 <= B <= 20

عشان هي disjoint و Partial

F = 30

عشان هي disjoint و Total

0 <= H <= 30

عشان هي overlap و Partial

هل ممكن اعمل Order by بحاجة مش موجودة في ال Select ؟

اه ينفع

The screenshot shows a SQL query window with the following code:

```
select st_fname, st_age, dept_id  
from Student  
order by st_address
```

The results grid displays 12 rows of data from the Student table:

	st_fname	st_age	dept_id
1	Said	NULL	40
2	Ahmed	23	10
3	NULL	24	10
4	Mohamed	28	20
5	NULL	24	30
6	Marwa	24	30
7	Noha	21	40
8	Ahmed	20	10
9	Amr	21	10
10	Mona	22	10
11	Ali	25	20
12	Heba	25	20

طيب كده ؟

The screenshot shows a SQL query window with the following code:

```
select st_fname, st_age, dept_id  
from Student  
order by 1
```

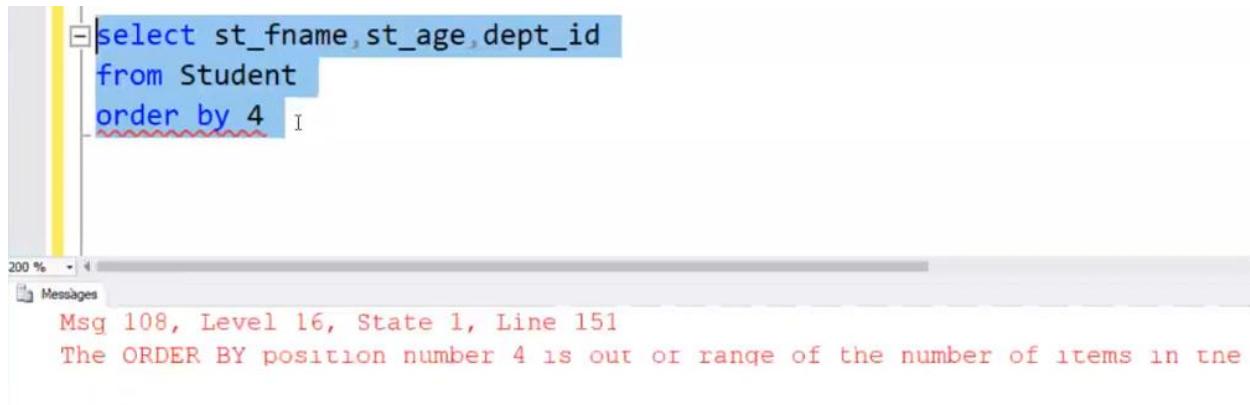
The results pane displays the following data:

	st_fname	st_age	dept_id
1	NULL	24	10
2	NULL	24	30
3	NULL	30	NULL
4	Ahmed	23	10
5	Ahmed	20	10
6	Ali	25	20
7	Amr	21	10
8	Heba	25	20
9	Marwa	24	30
10	Mohamed	28	20
11	Mona	22	10
12	Noha	21	40

هنا انا قولته اعمل Order by أول Column ف ال

اللي هو (st_fname)

لکن کدہ ؟



A screenshot of a SQL query editor window. The query entered is:

```
select st_fname, st_age, dept_id  
from Student  
order by 4
```

The line `order by 4` is highlighted in red, indicating an error. Below the editor, the message area shows:

Msg 108, Level 16, State 1, Line 151
The ORDER BY position number 4 is out of range of the number of items in the

طبيعي جايبلی Error

لأن مفيش Column رابع اصلا

هـما 3 Columns بـس

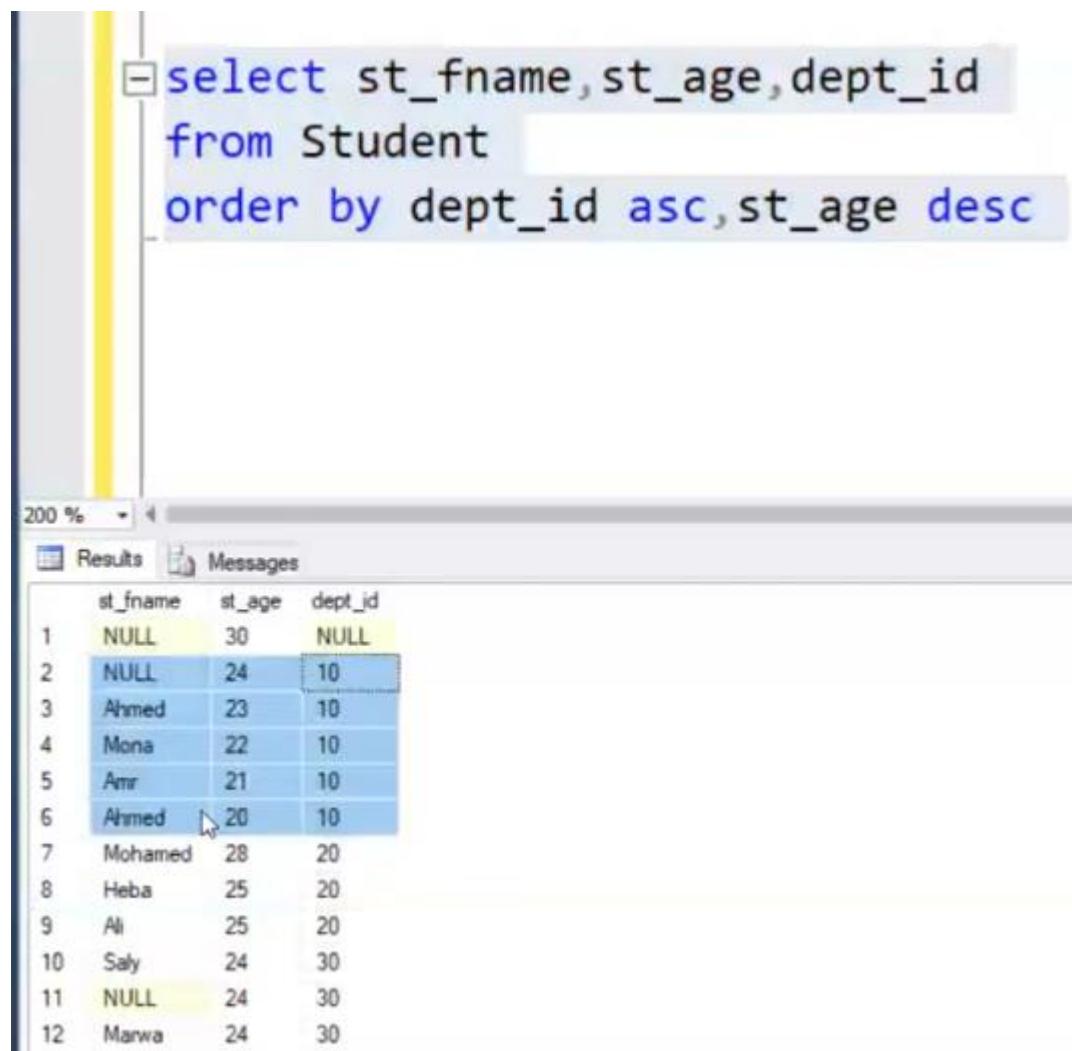
```
select st_fname,st_age,dept_id  
from Student  
order by dept_id,st_age
```

Results Messages

	st_fname	st_age	dept_id
1	NULL	30	NULL
2	Ahmed	20	10
3	Amr	21	10
4	Mona	22	10
5	Ahmed	23	10
6	NULL	24	10
7	Heba	25	20
8	Ali	25	20
9	Mohamed	26	20
10	Saly	24	30
11	NULL	24	30
12	Marwa	24	30

هذا هو عمل Order by على (dept_id) ثم (st_age)

ويمكن برضو اعملها كده



The screenshot shows a SQL query being run in SQL Server Management Studio. The query is:

```
select st_fname, st_age, dept_id  
from Student  
order by dept_id asc, st_age desc
```

The results grid displays 12 rows of data with columns: st_fname, st_age, and dept_id. The data is ordered by dept_id (asc) and st_age (desc). The first two rows have NULL values for st_fname and dept_id. Rows 3 through 12 have st_fname values like Ahmed, Mona, Amr, Mohamed, Heba, Ali, Saly, Marwa, and Marwa, with st_age values ranging from 20 to 30, and dept_id values mostly 10 or 20.

	st_fname	st_age	dept_id
1	NULL	30	NULL
2	NULL	24	10
3	Ahmed	23	10
4	Mona	22	10
5	Amr	21	10
6	Ahmed	20	10
7	Mohamed	28	20
8	Heba	25	20
9	Ali	25	20
10	Saly	24	30
11	NULL	24	30
12	Marwa	24	30

معلومات :-

مینفعش نـ (Child لیه Parent (Edit or Delete)

عشان كده ال 2 دول بيطاعولي Error Queries

```
delete from Department where dept_id=20
```

```
update Department set dept_id=4000 where dept_id=20
```

لكن لو ملهمش Child هيشتغلوا عادي جدا

طيب ايه حل الموضوع ده ؟

إني اعدل او امسح ال (Child)
بحيث اقدر اعدل او امسح ال (Parent) (dept_id) ده

وطبعا ليها طرق Automatic بس هنقولها قدام

--builtin functions

--Agg Functions

getdate()

isnull

coalesce

concat

Convert

year

```
select year(getdate())
```

The screenshot shows a SQL query window with the following content:

```
select year(getdate())
```

Below the query window, there is a results pane showing the output:

(No column name)
2020

month

```
select month(getdate())
```

The screenshot shows a SQL query window with the following content:

```
select month(getdate())
```

Below the query window, there is a results pane showing the output:

(No column name)
11

substring

```
select substring(st_fname,1,3)
from Student
```

Results Messages

(No column name)
1 Ahm
2 Amr
3 Mon
4 Ahm
5 NULL
6 Heb
7 Ali
8 Moh
9 Sel
10 NULL
11 Mar
12 Noh

db_name

```
select db_name()
```

Results Messages

(No column name)
1 master

suser_name

```
select suser_name()
```

Results Messages
(No column name)
DESKTOP-VF50P25\Ram

معلومات :-

في حاجة اسمها View Template Explorer

ب يكون فيه كل ال Syntax اللي هكون محتاجها

