10/22 (Guest Lecture, Prof. TJ Green)

- SQL nested queries! (correlated vs uncorrelated)
  - [not] exists
- NULL values

⇒ Midterm
⇒ HW #3
⇒ conduct

ta-166@googlegroups.com

- *List all customers who have an account greater than all customers from Davis.*

```
select * from CUSTOMERS
where Account > all
    (select Account from CUSTOMERS
     where CAddress like '%Davis%');
```

⇒ returns 0 tuples

... where Account >= all (...)

- *Give all pairs of suppliers that offer exactly the same products.*

```
select distinct O1.SName, O2.SName
from offers O1, offers O2
where O1.SName < O2.SName
  and not exists
        (( (select Prodname
            from offers
            where SName = O1.SName)
          Except minus
           (select Prodname
            from offers
            where SName = O2.SName)
         )
         union
         ( (select Prodname
            from offers
            where SName = O2.SName)
           minus
           (select Prodname
            from offers
            where SName = O1.SName)
         ))
order by O1.SName, O2.SName;
```

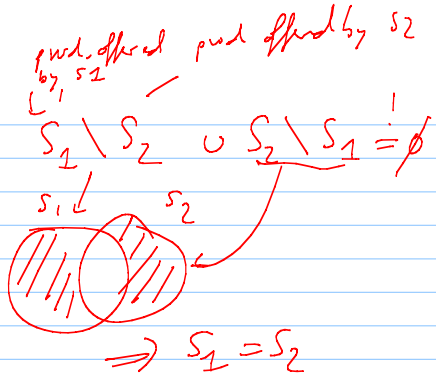
(i) eliminate (ii) O1.SN <> O2.SN

pwd. offered by S1

pwd by S2

pwd. by S2

pwd by S1



- *Find the name and address of customers who have ordered a product from Hibbert Lumber.*

uncorrelated subquery

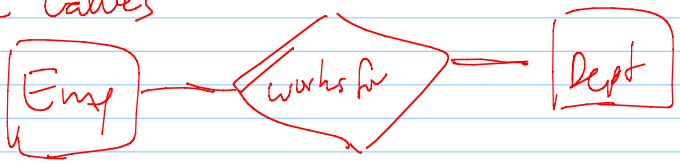
```
select * from CUSTOMERS
where (FName, LName) in (select FName, LName
                         from orders
                         where SName = 'Hibbert Lumber');
```

- *List all customers who have ordered a product from a supplier in Davis.*

correlated subquery

```
select * from CUSTOMERS C
where exists (select *
              from orders O, SUPPLIERS S
              where O.SName = S.SName
                and O.FName = C.FName
                and O.LName = C.LName
                and SAddress like '%Davis%');
```

NULL values



EMP

Eid	Name	...
e1	..	..

works for

E	D
e1	NULL

and	true	false	null
true	true	false	null
null	null	false	null
false	false	false	false

or	true	false	null
true	true	true	true
null	true	null	null
false	true	false	null

not	
true	false
null	null
false	true

offers

P1	S1
P1	S2
P1	S3
P2	S2
P3	S2
Null	S4
Null	Null

$\Pi_{PName}(offers) = \{P_1, P_2, P_3\}$

select count(PName) from offers;
⇒ 5

select count(distinct PName) from offers
⇒ 3

select count(*) from offers
⇒ 6

What is the minimum and maximum price for products offered by Davis Lumber?

select min(Price), **max**(Price) **from** offers
where SName = 'Davis Lumber';

offers

SName	Prod	Price
'DL'		$0
'DL'		$1000

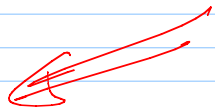
min(Price)	max(Price)
$0	$1000

single result

For each supplier, list the name of the supplier and the total number of products the supplier offers.

```
select SName, count(Prodname) , avg (Price)
from offers
group by SName;
```

SN	count(PN)
S1	3
S2	2
S3	2



offers

SN	PN	Pr
S_1	P_1	..
S_1	P_2	..
S_1	P_3	..
S_2	P_1	..
S_2	P_4	..
S_3	P_1	..
S_3	P_2	..

S1 → 3 ; S2 → 2 ; S3 → 2

For each customer, list the total quantity of orders.

```
select FName, LName, sum(Quantity)
from orders
group by FName, LName;
```

orders

FN	LN	SN	PN	Q
J	D			
J	D			
J	A			

(J, d)

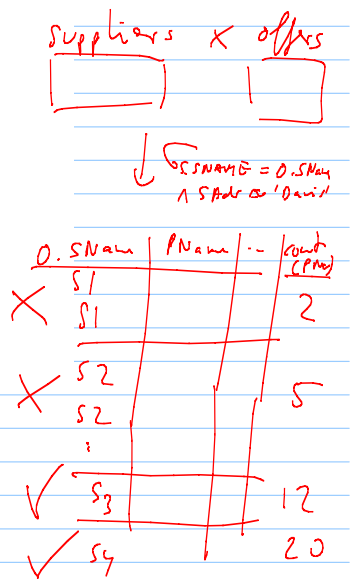
(J, a)

select <attribute(s) [with aggregate function]>
from $R_1, R_2, \ldots, R_m$
[**where** P]
group by <grouping attribute(s)>
[**having** <condition on group>];

List all suppliers from Davis that offer more than 10 products.

4. **select** O.SName, **count**(Prodname)
0. **from** SUPPLIERS S, offers O
1. **where** S.SName = O.SName **and** SAddress **like** '%Davis%'
2. **group by** O.SName
3. **having count**(Prodname) > 10;

O.SName	count(ProdName)
S_3	12
S_4	20



What is the minimum total quantity of all orders for a product?

```
select min(sum(Quantity))
from orders
group by Prodname;
```

```
select Prodname, sum(Quantity)
from orders
group by Prodname;
```

⇒ P1 6 min(sum Quantity)
P2 4 ⇒ 4
P3 7

orders

PN	Qty	
P1	1	} 6
P1	2	
P1	3	
P2	2	} 4
P2	2	
P3		} 7
1		
P3		