

Fordypning database

November 2023

Dag 2

Null

= ingen verdi / ikke oppgitt

er IKKE det samme som 0!

`select kjede_nr, gjelder_fra, ifnull(gjelder_til, '2099-12-31') gjelder_til, bonussats_pct`
`from kjedebonus order by 1,2` (MS SQL Server: isnull())

kjede_nr	gjelder_fra	gjelder_til	bonussats_pct
100	01.01.2023 00:00:00	2099-12-31	1
200	01.01.2023 00:00:00	2023-10-23 23:59:59	1
200	24.10.2023 00:00:00	2023-10-24 23:59:59	3
200	25.10.2023 00:00:00	2099-12-31	1

now()

Nå:

MySQL: now()

MS SQL Server: getdate()

C(r)UD

insert into kunde **values**(1001, 'Ellen', 3907);

update kunde **set** kunde_nv = 'Ellen Fakset' **where** kunde_nr = 1001;

delete from kunde where kunde_nr = 1001;

insert into kunde (**kunde_nr, kunde_nv, kommune_nr**) **values**(1001, 'Ellen', 3907);

C(r)UD med Subselect

Oppdater gjelder_til for utvalgte varegrupper:

```
update kjedevarepris  
set gjelder_til = '2023-10-31 23:59:59'  
where vare_nr in  
    (select vare_nr from vare  
     where varegruppe_nr in (4, 16, 19, 12, 45)  
    );
```

Opprett en ny rad for de samme varegruppene:

```
insert into kjedevarepris  
select kjede_nr, vare_nr, '2023-11-01', null, enhetpris_bel from kjedevarepris  
where vare_nr in ...
```

Oppdaterer nye rader

Setter ned prisen på grønnsaker:

```
update kjedevarepris  
set enhetpris_bel = enhetpris_bel * 0.9 where gjelder_til is null and vare_nr in (select vare_nr from vare  
where varegruppe_nr in (4));
```

Setter opp prisen på “usunne” varer, mest hos Admiral:

```
update kjedevarepris  
set enhetpris_bel = enhetpris_bel * 1.1 where gjelder_til is null and kjede_nr = 100 and vare_nr in (select  
vare_nr from vare where varegruppe_nr in (16, 19, 12, 45));
```

```
update kjedevarepris  
set enhetpris_bel = enhetpris_bel * 1.2 where gjelder_til is null and kjede_nr = 200 and vare_nr in ...
```

NodeJS

```
app.get('/varegrupper', function(req, res) {  
  const sql = "select varegruppe_nr as id, varegruppe_nv as name from varegruppe";  
  connection.query(sql, function (err, rows)  
  {  
    if (err)  
      throw err;  
  
    res.json({ message: rows });  
  });  
});
```


C#

```
private int  
updateCustomer(int custId, TextBox customerTb)  
{  
    string sql = "update kunde set kunde_nv = '" + customerTb.Text + '"  
                where kunde_nr = " + custId;  
    var cmd = new MySqlCommand(sql, _sqlConn);  
    int rowCount = cmd.ExecuteNonQuery();  
  
    return rowCount;  
}
```

Java

```
public HashMap<Integer, County>
getCounties()
{
    String sql = "select fylke_nr, fylke_nv from fylke where fylke_nr > 0";
    HashMap<Integer, County> counties = new HashMap<>();

    try
    {
        PreparedStatement stmt = this.connection.prepareStatement(sql);
        ResultSet result = stmt.executeQuery();
        while (result.next())
        {
            counties.put(result.getInt(1), new County(result.getInt(1), result.getString(2)));
        }
    }
    catch (SQLException e)
    {
        System.err.println("Feil i getCounties: " + sql + " " + e);
    }
    return counties;
}
```

- Oppgave 2.1 - 2.5
- Lunsj
- Oppstart kl 12.15

Outer join

Vis butikker uavhengig av handler

```
select butikk_nv, count(handel_nr)
from handel
```

```
right outer join butikk on butikk.butikk_nr = handel.butikk_nr
group by butikk_nv
```

```
select butikk_nv, count(handel_nr)
from butikk
```

```
left outer join handel on butikk.butikk_nr = handel.butikk_nr
group by butikk_nv
```

Butikk	Handler
Hårfagre mat	394
Pommern super	425
Statholderen storkiosk	266
Nilsen på hjørnet	233
Nyby Mat og sånn	446
Kolonialen bak parken	389
Ellens kiosk	0

(Not) Exists

Vis kasser hvor det ikke har vært handler

```
select * from kasse
where not exists (
    select 1 from handel
    where kasse.butikk_nr = handel.butikk_nr and kasse.kasse_nr = handel.kasse_nr)
```

Kartesisk produkt

```
select * from fylke, kommune
```

vs.

```
select * from fylke f, kommune k  
      where f.fylke_nr = k.fylke_nr
```

fylke_nr	fylke_nv	kommune_nr	kommune_nv
40	Telemark	3901	Horten
39	Vestfold	3901	Horten
40	Telemark	3903	Holmestrand
39	Vestfold	3903	Holmestrand
40	Telemark	3905	Tønsberg
39	Vestfold	3905	Tønsberg
40	Telemark	3907	Sandefjord
39	Vestfold	3907	Sandefjord
40	Telemark	3909	Larvik

Ikke så lett å se:

```
select vg.varegruppe_nv, sum(vl.varelinje_bel)
```

```
from vare v, varelinje vl, varegruppe vg, handel h
```

```
where v.varegruppe_nr = vg.varegruppe_nr and h.handel_nr = vl.handel_nr
```

```
group by vg.varegruppe_nv
```

Kartesisk produkt - som nyttevekst / Self Join

```
insert into fotball_lag values(1, 'Sandefjord');
```

```
insert into fotball_lag values(2, 'Odd');
```

```
insert into fotball_lag values(3, 'Vålerenga');
```

```
insert into fotball_lag values(4, 'Ham-kam');
```

```
insert into fotball_lag values(5, 'Lillestrøm');
```

```
insert into fotball_lag values(6, 'Start');
```

```
insert into fotball_lag values(7, 'Bodø/Glimt');
```

```
insert into fotball_lag values(8, 'Rosenborg');
```

```
insert into fotball_lag values(9, 'Molde');
```

```
insert into fotball_lag values(10, 'Stabæk');
```

```
insert into fotball_lag values(11, 'Aalesund');
```

```
insert into fotball_lag values(12, 'Lyn');
```

Hjemme	Borte
Lyn	Sandefjord
Aalesund	Sandefjord
Stabæk	Sandefjord
Molde	Sandefjord
Rosenborg	Sandefjord
Bodø/Glimt	Sandefjord
Start	Sandefjord
Lillestrøm	Sandefjord
Ham-kam	Sandefjord
Vålerenga	Sandefjord
Odd	Sandefjord
Lyn	Odd
Aalesund	Odd

```
insert into fotball_lag values(2, 'Odd');
```

```
insert into fotball_lag values(3, 'Vålerenga');
```

```
insert into fotball_lag values(4, 'Ham-kam');
```

```
insert into fotball_lag values(5, 'Lillestrøm');
```

```
insert into fotball_lag values(6, 'Start');
```

```
insert into fotball_lag values(7, 'Bodø/Glimt');
```

```
insert into fotball_lag values(8, 'Rosenborg');
```

```
insert into fotball_lag values(9, 'Molde');
```

```
insert into fotball_lag values(10, 'Stabæk');
```

```
insert into fotball_lag values(11, 'Aalesund');
```

```
insert into fotball_lag values(12, 'Lyn');
```

```
select
```

```
    fl1.fotball_lag_nv Hjemme,
```

```
    fl2.fotball_lag_nv Borte
```

```
from fotball.fotball_lag fl1, fotball.fotball_lag fl2
```

```
where fl1.idfotball_lag != fl2.idfotball_lag
```

Between forts.

Hva er samlet bonus for grønnsaker hos Admiral over tid?

```
select date_format(vb.gjelder_fra, '%d.%m.%Y') vgFra, date_format(b.gjelder_fra, '%d.%m.%Y') BFra,  
date_format(b.gjelder_til, '%d.%m.%Y') BTil, vb.bonussats_pct + b.bonussats_pct 'S%'  
from kjedevaregruppebonus vb, kjedebonus b  
where b.kjede_nr = vb.kjede_nr  
and  
(  
    (vb.gjelder_fra between b.gjelder_fra and ifnull(b.gjelder_til, '2099-12-31')  
    and ifnull(vb.gjelder_til, '2099-12-31') between b.gjelder_fra and ifnull(b.gjelder_til, '2099-12-31')  
    )  
    or  
    (b.gjelder_fra between vb.gjelder_fra and ifnull(vb.gjelder_til, '2099-12-31')  
    and ifnull(b.gjelder_til, '2099-12-31') between vb.gjelder_fra and ifnull(vb.gjelder_til, '2099-12-31')  
    )  
)  
and vb.kjede_nr = 200 and varegruppe_nr = 4
```

VgFra	BFra	BTil	S%
01.01.2023	01.01.2023	23.10.2023	2
01.01.2023	24.10.2023	24.10.2023	4
01.01.2023	25.10.2023	null	2

Mer Union - Don't Panic!

```
select 'Total' Tekst, kunde_nv navn, handel_bel beløp from handel, kunde where handel.kunde_nr  
= kunde.kunde_nr and handel_nr = 42
```

union

```
select 'Vare', vare_nv, varelinje_bel from varelinje, vare where varelinje.vare_nr = vare.vare_nr  
and handel_nr = 42
```

union

```
select 'Bonus handel', '', bonus_bel from bonus where varegruppe_nr = 0  
and handel_nr = 42
```

union

```
select concat('Bonus ', varegruppe_nv), varegruppe_nv, bonus_bel  
from bonus, varegruppe  
where varegruppe.varegruppe_nr = bonus.varegruppe_nr  
and varegruppe.varegruppe_nr != 0 and handel_nr = 42
```

With

```
with utvalg as (select 200 as queryid from dual),  
kjeder as (select kjede_nr, kjede_nv from kjede)  
select kjede_nr, kjede_nv from utvalg, kjeder  
where queryid = kjede_nr
```

```
with utvalg as  
(select 200 as queryid from dual),  
kjedebonus as  
(select kjede_nr, varegruppe_nr, gjelder_fra, bonussats_pct from kjedevaregruppebonus  
union  
select kjede_nr, 0, gjelder_fra, bonussats_pct from kjedebonus)  
select kjede_nr Kjede, varegruppe_nr Vgr, gjelder_fra, bonussats_pct '%'  
from utvalg, kjedebonus where queryid = kjede_nr
```

Kjede	Vgr	gjelder_fra	%
200	3	01.01.2023 00:00:00	1
200	4	01.01.2023 00:00:00	1
200	0	01.01.2023 00:00:00	1
200	0	24.10.2023 00:00:00	3
200	0	25.10.2023 00:00:00	1

With

```
with utvalg as (select 42 as id from dual),
rader as (
select 'Total' Tekst, handel_nr as id, kunde_nv navn, handel_bel beløp, kunde.kunde_nr as rowId from handel, kunde
where handel.kunde_nr = kunde.kunde_nr
union
select 'Vare', handel_nr, vare_nv, varelinje_bel, vare.vare_nr from varelinje, vare where varelinje.vare_nr =
vare.vare_nr
union
select 'Bonus handel', handel_nr, '', bonus_bel, 0 from bonus where varegruppe_nr = 0
union
select concat('Bonus ', varegruppe_nv), handel_nr, varegruppe_nv, bonus_bel, varegruppe.varegruppe_nr
      from bonus, varegruppe
      where varegruppe.varegruppe_nr = bonus.varegruppe_nr and varegruppe.varegruppe_nr != 0
)
select rowid as id, rader.id as Nr, Tekst, Navn, Beløp from utvalg, rader
where utvalg.id = rader.id
```

Konkatenering

```
select concat(  
    'Varegruppe ',  
    varegruppe_nv,  
    '(', varegruppe_nr,  
    ') Mva %: ', mvasats_pct) Vgr  
from varegruppe
```

Vgr
Varegruppe Ikke kategorisert(0) Mva %: 0
Varegruppe Pålegg(1) Mva %: 15
Varegruppe Ferdigmat(2) Mva %: 15
Varegruppe Fukt og bær(3) Mva %: 15
Varegruppe Grønnsaker(4) Mva %: 15
Varegruppe Kjøtt(5) Mva %: 15
Varegruppe Sjømat(6) Mva %: 15
Varegruppe Hermetiske og tørkede grønnsaker(7) Mva %: 15

Datoformat

MySQL:

```
select kjede_nr Kjede, varegruppe_nr Vgr,  
date_format(gjelder_fra,'%d.%m.%Y %H:%i:%s') Fra,  
bonussats_pct '%'  
from kjedevaregruppebonus
```

SQL Server:

```
select kjede_nr Kjede, varegruppe_nr Vgr,  
convert(varchar(45), gjelder_fra, 104) + ' ' +  
convert(varchar(45), gjelder_fra, 108) Fra,  
bonussats_pct '%' from kjedevaregruppebonus
```

Kjede	Vgr	Fra	%
100	3	01.01.2023 00:00:00	2
100	4	01.01.2023 00:00:00	2
200	3	01.01.2023 00:00:00	1
200	4	01.01.2023 00:00:00	1

Finne riktig rad

Finn dagens pris for vare 42 hos kjede 200

```
select enhetpris_bel  
from kjedevarepris  
where kjede_nr = 200 and vare_nr = 42 and gjelder_til is null
```

enhetpris_bel

28.26

```
select enhetpris_bel  
from kjedevarepris kvpy  
where kjede_nr = 200  
and vare_nr = 42  
and gjelder_fra =  
(select max(gjelder_fra)  
  from kjedevarepris kvpi  
  where kvpi.kjede_nr = kvpy.kjede_nr and kvpi.vare_nr = kvpy.vare_nr)
```

Function

```
CREATE DEFINER=`root`@`localhost` FUNCTION `get_age`(ibirthdate date) RETURNS int
NO SQL
BEGIN
    declare oage int;
    declare tyear1 int;
    declare tyear2 int;
    declare tyeardiff int;
    declare tdate date;

    set tyeardiff = year(CURRENT_DATE()) - year(ibirthdate);

    set tdate = date_add(ibirthdate, interval tyeardiff YEAR);
    if tdate > CURRENT_DATE()
    then
        set tyeardiff = tyeardiff - 1;
    end if;
    set oage = tyeardiff;

    return oage;

END
```

select get_age('2015-12-16') Alder
from dual

Alder

7

select get_age('2015-11-16') Alder
from dual

Alder

8

Stored Procedure

```
CREATE DEFINER='root'@'localhost' PROCEDURE `calculate_bonus`(in ihandelnr int, in iinsert boolean, out
obonusbel decimal(7,2))
BEGIN
/* 2 parametere IN, 1 OUT
   Prosedyren kan kjøres på to måter:
insert = false: Bonus beregnes, men blir ikke lagt til i bonus-tabellen
iinsert = true: Rader legges til i 'bonus'
*/
/* Deklarerer alle variabler som skal brukes */
declare tbonusbel decimal(7,2) default 0;
declare tbonusbel decimal(7,2);
declare tknr int;
declare tvgr int;
declare finished int default 0;

/* Deklarerer Cursor for data for handelen,
dvs. innhold til bonusrader for utvalgte varegrupper
*/
declare cvgr cursor for
select handel.kunde_nr, vare.varegruppe_nr, sum(round(varelinje_bel * bonussats_pct / 100.0, 2)) as
bonus_bel
from handel, varelinje, butikk, vare, kjedevaregruppebonus
where handel.handel_nr = ihandelnr
and handel.butikk_nr = butikk.butikk_nr
and butikk.kjede_nr = kjedevaregruppebonus.kjede_nr
and handel.handel_nr = varelinje.handel_nr
and varelinje.vare_nr = vare.vare_nr and vare.varegruppe_nr = kjedevaregruppebonus.varegruppe_nr
group by handel.handel_nr, vare.varegruppe_nr;

/* finished settes til 1 når det ikke er flere rader å behandle i cursoren */
DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;

/* Hvis iinsert = true, slettes alle bonus for handelen */
if iinsert
then
delete from bonus where handel_nr = ihandelnr;
end if;
```

```
open cvgr; /* Åpner cursor */
getbonus: loop /* og løper gjennom */
fetch cvgr into tknr, tvgr, tbonusbel; /* verdier fra resultatsettet legges inn i variabler */
if finished = 1
then
leave getbonus; /* Ingenting mer å behandle */
end if;
if tknr > 0
then
if iinsert
then
insert into bonus (handel_nr, varegruppe_nr, bonus_bel) values (ihandelnr, tvgr, tbonusbel);
end if;
set tbonusbel = tbonusbel + tbonusbel;
end if;
end loop getbonus;

close cvgr;

if tknr > 0
then
/* Til slutt: Bonus for hele handelen */
select round(handel_bel * bonussats_pct / 100.0, 2) into tbonusbel
from handel, butikk, kjedebonus
where handel.handel_nr = ihandelnr
and handel.butikk_nr = butikk.butikk_nr
and butikk.kjede_nr = kjedebonus.kjede_nr
and handel.tidspunkt between gjelder_fra and ifnull(gjelder_til, '2099-12-31');

if iinsert
then
insert into bonus (handel_nr, bonus_bel) values (ihandelnr, tbonusbel);
end if;
set tbonusbel = tbonusbel + tbonusbel;
end if;

set obonusbel = tbonusbel;
END
```


MS SQL Server

```
alter PROCEDURE calculate_bonus @ihandelnr int, @iinsert tinyint, @obonusbel decimal(7,2) OUTPUT
as
/* 2 parametere IN, 1 OUT
   Prosedyren kan kjøres på to måter:
       iinsert = 0: Bonus beregnes, men blir ikke lagt til i bonus-tabellen
       iinsert = 1: Rader legges til i 'bonus'
*/
/* Deklarerer alle variabler som skal brukes */
declare @tbonusbelt decimal(7,2);
declare @tbonusbel decimal(7,2);
declare @tvgr int;
declare @tknr int;

/* Deklarerer Cursor for data for handelen,
dvs. innhold til bonusrader for utvalgte varegrupper
*/
set @tbonusbelt = 0;
declare cvgr cursor for
select handel.kunde_nr, vare.varegruppe_nr, sum(round(varelinje_bel * bonussats_pct / 100.0, 2)) as
bonus_bel
from handel, varelinje, butikk, vare, kjedevaregruppebonus
where handel.handel_nr = @ihandelnr
  and handel.butikk_nr = butikk.butikk_nr
  and butikk.kjede_nr = kjedevaregruppebonus.kjede_nr
  and handel.handel_nr = varelinje.handel_nr
  and varelinje.vare_nr = vare.vare_nr and vare.varegruppe_nr =
kjedevaregruppebonus.varegruppe_nr
group by handel.kunde_nr, vare.varegruppe_nr;

/* Hvis iinsert = 1, slettes alle bonus for handelen */
if @iinsert = 1
  delete from bonus where handel_nr = @ihandelnr;
```

```
open cvgr; /* Åpner cursor */
fetch cvgr into @tknr, @tvgr, @tbonusbel; /* verdier fra resultatsettet legges inn i variabler */

WHILE @@FETCH_STATUS = 0
BEGIN
  if @tknr > 0
  begin
    if @iinsert = 1
      insert into bonus (handel_nr, varegruppe_nr, bonus_bel) values (@ihandelnr, @tvgr, @tbonusbel);
    set @tbonusbelt = @tbonusbelt + @tbonusbel;
  end
  fetch cvgr into @tknr, @tvgr, @tbonusbel;
end;

close cvgr;
deallocate cvgr;

if @tknr > 0
begin
  /* Til slutt: Bonus for hele handelen */
  select @tbonusbel = round(handel_bel * bonussats_pct / 100.0, 2)
  from handel, butikk, kjedebonus
  where handel.handel_nr = @ihandelnr
  and handel.butikk_nr = butikk.butikk_nr
  and butikk.kjede_nr = kjedebonus.kjede_nr
  and handel.tidspunkt between gjelder_fra and isnull(gjelder_til, '2099-12-31');

  if @iinsert = 1
    insert into bonus (handel_nr, varegruppe_nr, bonus_bel) values (@ihandelnr, 0, @tbonusbel);
  set @tbonusbelt = @tbonusbelt + @tbonusbel;
end
set @obonusbel = @tbonusbelt;
Go
```

Oppgave 2.6 - 2.13