

## Databashantering YH00541, Vecka 46

### *Introduktion, Praktiska frågor*

Information på <https://studentportal.nackademin.se/course/view.php?id=4158> uppdateras efter hand.

Vecka	Dag	Område
46	Mån 14 nov	Introduktion, Använda SQL
46	Tis 15 nov	Fortsättning SQL
46	Tors 17 nov	Installera SQL Server (eller motsvarande)
47	Mån 21 nov	Modellera data, Skapa databaser
47	Tis 22 nov	Hantera säkerhet, Optimera databaser
47	Tors 24 nov	Olika typer av databaser
48	Mån 28 nov	Koppla applikation till databas
48	Tis 29 nov	Lagra data, Olika driftmiljöer
48	Fre 2 dec	Inlämningsuppgift, Summering

Ordningsfrågor, närvaro mm.

Önskemål om lunch 11.30 – 12.30 (istället för 12.00 – 13.00)?

Förmiddagar mest teoretiska genomgångar. Eftermiddagar mest praktiska övningar.

Avsluta veckan med någon form av quiz, för egen kontroll av områden som kanske behöver repeteras. För eller emot?

Inga särskilda förberedelser krävs för de första två dagarna.

Inför dag tre, när SQL Server ska installeras, kan vara bra att läsa på om systemkrav mm.

### ***Kort presentationsrunda***

Lärare: Johan Faxér, [johan.faxer@nackademin.se](mailto:johan.faxer@nackademin.se) (tillgänglig måndag-torsdag).

Vilka är ni, och vad har ni gjort på föregående kurs?

Övergripande mål och förväntningar.

## Måndag 14 nov – Introduktion till databaser

Vad är en databas, vad är en tabell, vad är kolumner, rader etc.

En samling data, organiserad och strukturerad.

Tabeller som motsvarar olika objekt.

Kolumner som motsvarar egenskaper eller fält.

Rader som motsvarar unika poster eller instanser av objekt.

Några egenskaper eller funktioner:

- Lagring, åtkomst, uppdatering
- Kontroll av regler, beroenden mellan data, osv
- Kontroll av behörigheter
- Hantering av samtidiga användare, Hantering av transaktioner
- Säkerhetskopiering

En relationsdatabas, med kopplingar mellan tabeller.

Två nyckelbegrepp (!) är primära och främmande nycklar (mer om detta senare).

SQL = Structured Query Language, för att ...

- definiera schema
- fråga efter data (select)
- manipulera data (insert, update, delete)
- ändra behörigheter
- hantera transaktioner

Ett deklarativt språk, där du anger vad som ska göras, inte exakt hur.

ISO-standard, men olika “dialekter” mellan olika tillverkare.

## ***Datatyper***

Precis som inom programmering, olika slags behållare för olika slags data.

- Numeriska värden: integer, float/real, decimal, currency/money
- Text: char, varchar, text/memo
- Logiska värden: boolean
- Tidsangivelser: datetime, timestamp
- Binära data: image, blob

## ***SQL-frågor för att lägga till data, göra urval, uppdatera eller radera***

Det vill säga ...

- insert
- select
- update
- delete

Grundläggande syntax:

- insert into tabell1(kolumn1, kolumn2) values('värde 1', 'värde 2');
- select kolumn1, kolumn2 from tabell1;
- update tabell1 set kolumn1 = 'ett nytt värde';
- delete from tabell1;

Några praktiska exempel ...

[...]

Diskussion: Utifrån de nyss visade praktiska exemplen, vad verkar det finnas för brister i den databasen?

## ***Urvalsvillkor, logiska operatorer***

Nyckelordet “where”, eftersom det är väldigt sällan man vill läsa, uppdatera eller radera alla poster i en tabell.

Vi kan hänga på ...

- where kolumn1 = ‘värde’
- where kolumn1 like ‘vär%’
- where kolumn1 > ‘värde’
- where kolumn1 between ‘värde1’ and ‘värde2’
- och så vidare

Går att kombinera flera villkor med “and” och “or”. (Tänk på reglerna för i vilken ordning uttryck utvärderas.)

Negationer med “not”, eller “<>” eller “!=”.

Flera “or” efter varandra går att ersätta med operatoren “in( ... )”.

För att jämföra med det speciella värdet “null” måste “is null” användas.

Använd “select distinct” för att få enbart unika poster.

## ***Lite översikt***

Använd “coalesce( ... )” för att returnera det första värdet i en lista som inte är “null”.

Utvärdera ljudlikhet med “soundex( ... )” eller “sounds like”.

Resultatet från ett urval med “select” går att använda som inparameter till infoga data med “insert”.

Två eller flera urvalsmängder går att kombinera med “union” och “intersect”. (Men om man inte tänker efter kan resultatet bli helt meningslöst.)

## ***Några resurser online***

Testa att göra urval, manipulera data, även skapa och ändra tabeller, mm.

Vissa av tjänsterna har även tester eller uppgifter att lösa.

- <https://data.stackexchange.com/>
- <https://sqlbolt.com/>
- <http://sqlfiddle.com/>
- <https://sqltest.net/>
- <https://www.w3schools.com/sql/default.asp>

## ***Enklare databas för test på egna datorn***

Sqlite som går att hämta från <https://www.sqlite.org/>

Grafiskt gränssnitt DB Browser for Sqlite går att hämta från <https://sqlitebrowser.org/>

## ***Övningar***

Lite lättare:

Exercises på <https://www.w3schools.com/sql/exercise.asp> (hoppa över “order by”, “functions”, “join”, “group by” och “database”)

Lite svårare:

Laborera på egen hand med frågor till <https://data.stackexchange.com/stackoverflow/query/new>

Ett tips: Använd “select top 10 ...” vid test med större tabeller. Kan annars ta lång tid att hämta data.

## Tisdag 15 nov – Fortsättning SQL

Liten summering av gårdagen.

Vad lärde vi oss om ...

- definitioner av tabeller och kolumner
- datatyper
- urvalsvillkor
- mm

Frågor?

Förbättringsförslag?

## ***SQL-frågor för att skapa, ändra och radera databaser, tabeller, kolumner***

Definiera och ändra “schema” på olika sätt.

- create / drop database
- create / drop table
- alter table
  - add / alter / drop column

Speciellt för Sqlite: Går inte att skapa ny databas med “create database ...”, använd istället kommandot “sqlite3 *Databasnamn*”.

Sätta olika slags regler:

- Id-värden (“primary key”).
- Default-värden (“default”).
- Giltiga värden (“check”).
- Obligatoriska fält (“not null”).
- Unika värden (“unique”).

Begrepp vi ska kolla mer på senare:

- Nycklar, primära och främmande.
- Skapa index, begränsningar (constraints), mm.



### ***Sortera och gruppera urval***

Sortera med “order by”.

Stigande (“asc”) respektive fallande (“desc”).

Gruppera med “group by”. Med eller utan “having” efter.

Måste den kolumn man sorterar / grupperar efter finnas med i urval?

Går att sortera / gruppera efter två eller flera kolumner.

För- och nackdelar med att ange namn på kolumner vs att ange deras turordning i urval.

## ***Funktioner för att beräkna eller bearbeta***

Ett flertal inbyggda funktioner:

- `count()`, `sum()`, `avg()`, `min()`, `max()`
- `square()`, `sqrt()`, `exp()`, `log()`, `pi()`, `random()`, `sin()`, `cos()`
- `abs()`, `ceiling()`, `floor()`, `round()`
- `getdate()`, `datediff()`, `dateadd()`
- `format()`
- osv

Apropå datumfunktioner, några ord om nationella inställningar, tidszoner mm. Som även påverkar sorteringsordning (aka “Varför kommer Ö efter O?”).

Går även att skapa helt egna funktioner med “create function”.

Använda alias för kolumner, beräkningar mm.

Bearbeta värden med ...

- `length()`
- `left()`, `right()`, `trim()`
- `upper()`, `lower()`
- `replace()`
- `ascii()`, `char()`
- osv

Konvertera data med ...

- `str()`
- `cast()`
- `convert()`

## ***Hämta data från flera olika tabeller***

Nu blir det mängdlära igen!

- inner join
- outer join
- left join
- right join

Något av det vanligaste: Slå upp värden från en “stöd-tabell” vid sidan av (lookup table).

Exempel: `select t1.x, t1.y, t1.z, t2.x from t1 left join t2 on t1.annat_id = t2.id`

Den andra tabellen kan vara tex kategorinamn, begrepp en nivå upp i hierarki, eller annan sorts “föräldertyp” till den första tabellen.

Apropå “föräldertyp”, går det att göra en join med samma tabell?

Ledtråd: Tänk på exemplet med släktforskning där Person pekar på två andra Person som är föräldrar.

## Övningar

Lite lättare:

Fortsätt med Exercises på <https://www.w3schools.com/sql/exercise.asp> (nu även “order by”, “functions”, “join”, “group by” och “database”)

Lite svårare:

Laborera på egen hand med frågor till <https://data.stackexchange.com/stackoverflow/query/new>

Förslag på frågor att ställa:

- De topp tio Users som har högst Reputation, vad har de för DisplayName, och hur många dagar har var och en varit registrerad användare?
- De topp tio Posts som har högst Score, vad har de för Title och vilka användare har skrivit dem?
- De topp tio Users som har gjort flest Comments, vad har de för DisplayName, och vad har deras Comments för genomsnittlig Score?

Bonusuppgift:

Undersök hur matematiska funktioner som tex `sin()`, `round()`, `random()` beter sig.

Varför kan man behöva kontrollera resultaten?