



**Campus Nyköping**

# NET 2 - 04

SQL2: CRUD och Skapa Db

# Förra gången

- Utvecklingsmiljön
- SQL 1
  - SELECT

# Idag

- Ny gruppindelning
- Lösningförslag på övningarna
- Mer om SSMS
- SQL 2
  - CRUD
    - INSERT
    - SELECT
    - ALTER
    - DELETE
  - CREATE
  - DROP
  - M m...

# Grupper

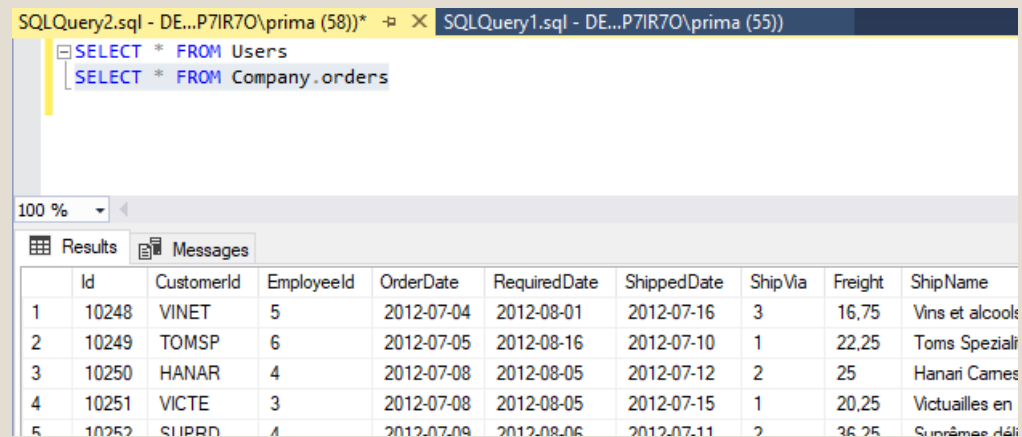
A	Pirdaoud, Didar
A	Alexander, Archie
A	Mahamed, Abdimajid Mumin
A	Victor, Vincent
B	Larsson, Dwina
B	Johansson, Rasmus
B	Bubicic, Anton
B	Åkesson, Mathias
C	Enehöjd, Martin
C	Shoble, Abdallatif Ali
C	Spector, Annika
C	Duchene, Emmanuel
D	Abbassi, Ramin
D	Lindberg, Simon
D	Karjalainen, Mira
D	Nguyen, Kiet
E	Tallberg, Tim
E	Berg, Gustaf
E	Sand, Elin
E	Silfver, Robin
E	Ballan, Antwanet
F	Lidman, Cassandra
F	Falk, Tintin
F	Blomqvist, Alicia
F	Öhrström, Emil
G	Engström, Anton
G	Tran, Quoc Tung
G	Wigren, Mattias
G	Gejskog, Kevin
H	Agubi, Michael
H	Runnberg Jansson, Elvira
H	Jonsson, Robin
H	Klasson, Eric

# SSMS

- Köra del av SQL-satsen
- Flera SQL-satser i samma query
- Kopia på db-tabell

# Köra delar av SQL-satsen

- Ett enkelt sätt att köra bara visa rader i en sql-sats är att markera raden, och klicka på Execute.

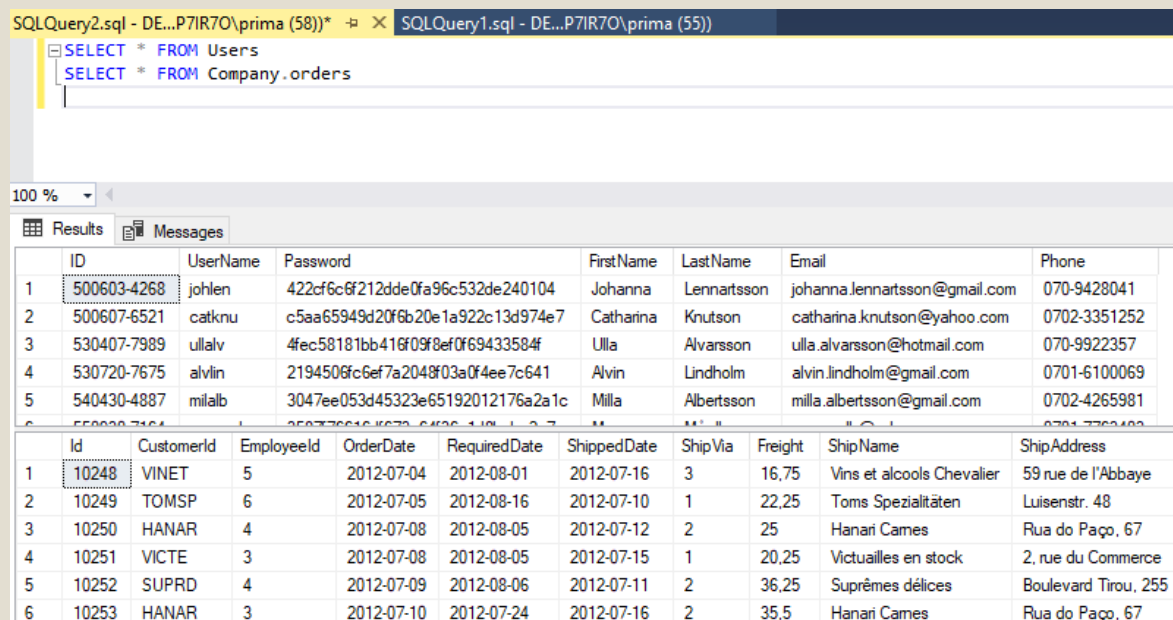


The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery2.sql - DE...P7IR7O\prima (58))' and 'SQLQuery1.sql - DE...P7IR7O\prima (55))'. The active tab contains two SQL statements: 'SELECT \* FROM Users' and 'SELECT \* FROM Company.orders'. Below the query window is a 'Results' tab showing a table with 10 columns: Id, CustomerId, EmployeeId, OrderDate, RequiredDate, ShippedDate, ShipVia, Freight, and ShipName. The table contains 5 rows of data.

	Id	CustomerId	EmployeeId	OrderDate	RequiredDate	ShippedDate	ShipVia	Freight	ShipName
1	10248	VINET	5	2012-07-04	2012-08-01	2012-07-16	3	16,75	Vins et alcools
2	10249	TOMSP	6	2012-07-05	2012-08-16	2012-07-10	1	22,25	Toms Spezial
3	10250	HANAR	4	2012-07-08	2012-08-05	2012-07-12	2	25	Hanari Camer
4	10251	VICTE	3	2012-07-08	2012-08-05	2012-07-15	1	20,25	Victuailles en
5	10252	SUPRD	4	2012-07-09	2012-08-06	2012-07-11	2	36,25	Suprêmes déli

# Köra flera SQL-satser

- Det går bra att köra flera SQL-satser efter varandra. Flera SELECT-satser öppnar flera tabellvyer.



SQLQuery2.sql - DE...P7IR7O\prima (58)) \* SQLQuery1.sql - DE...P7IR7O\prima (55))

```
SELECT * FROM Users
SELECT * FROM Company.orders
```

100 %

Results Messages

	ID	UserName	Password	FirstName	LastName	Email	Phone
1	500603-4268	johlen	422cf6c6f212dde0fa96c532de240104	Johanna	Lennartsson	johanna.lennartsson@gmail.com	070-9428041
2	500607-6521	catknu	c5aa65949d20f6b20e1a922c13d974e7	Catharina	Knutson	catharina.knutson@yahoo.com	0702-3351252
3	530407-7989	ullalv	4fec58181bb416f09f8ef0f69433584f	Ulla	Alvarsson	ulla.alvarsson@hotmail.com	070-9922357
4	530720-7675	alvin	2194508fc6ef7a2048f03a0f4ee7c641	Alvin	Lindholm	alvin.lindholm@gmail.com	0701-6100069
5	540430-4887	milalb	3047ee053d45323e65192012176a2a1c	Milla	Albertsson	milla.albertsson@gmail.com	0702-4265981

	Id	CustomerId	EmployeeId	OrderDate	RequiredDate	ShippedDate	ShipVia	Freight	ShipName	ShipAddress
1	10248	VINET	5	2012-07-04	2012-08-01	2012-07-16	3	16,75	Vins et alcools Chevalier	59 rue de l'Abbaye
2	10249	TOMSP	6	2012-07-05	2012-08-16	2012-07-10	1	22,25	Toms Spezialitäten	Luisenstr. 48
3	10250	HANAR	4	2012-07-08	2012-08-05	2012-07-12	2	25	Hanari Cames	Rua do Paço, 67
4	10251	VICTE	3	2012-07-08	2012-08-05	2012-07-15	1	20,25	Victuailles en stock	2, rue du Commerce
5	10252	SUPRD	4	2012-07-09	2012-08-06	2012-07-11	2	36,25	Suprêmes délices	Boulevard Tirou, 255
6	10253	HANAR	3	2012-07-10	2012-07-24	2012-07-16	2	35,5	Hanari Cames	Rua do Paço, 67

# Snabb SELECT

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'company.regions' table is selected in the 'Tables' folder, and a context menu is open with 'Select Top 1000 Rows' highlighted. The right pane shows the generated SQL query:

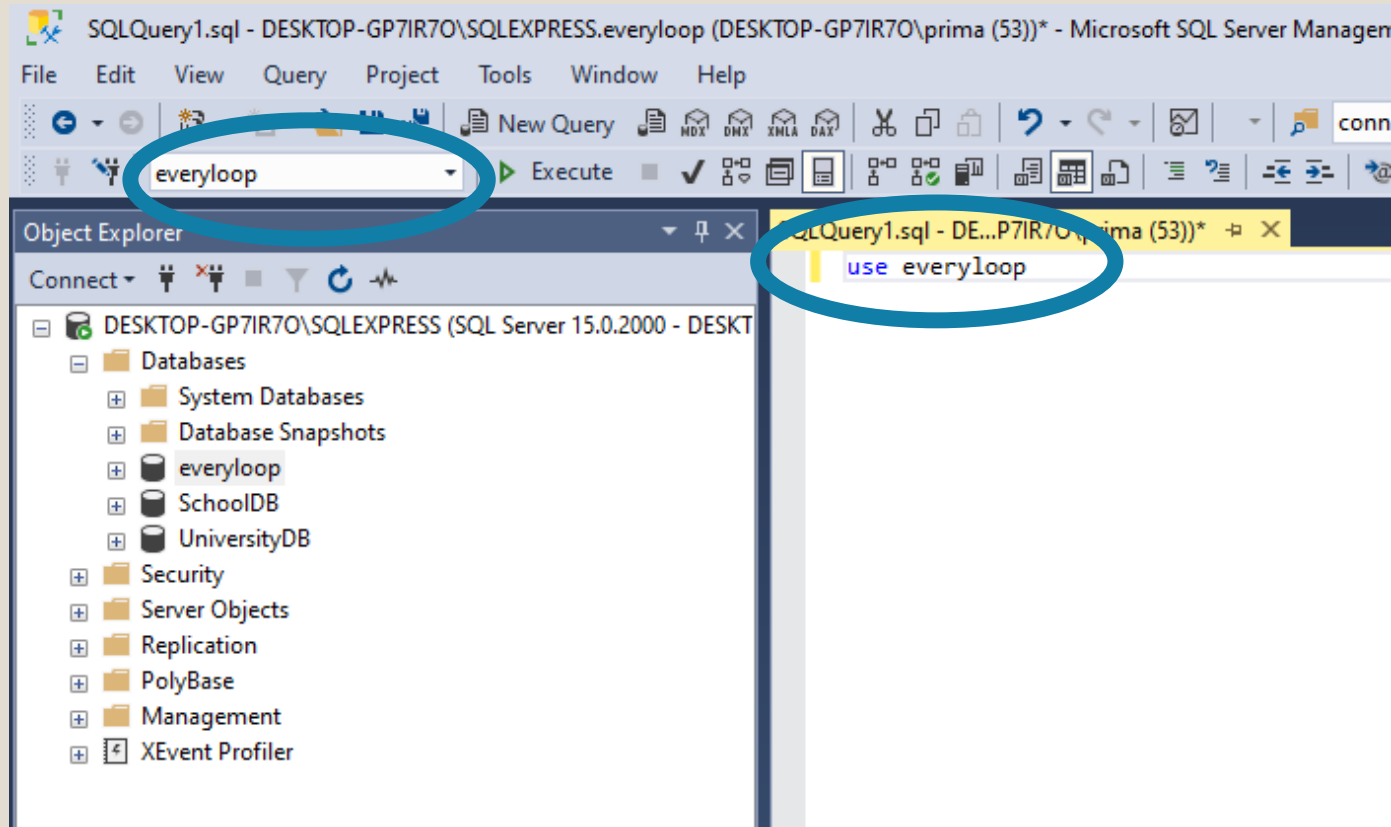
```
/****** Script for SelectTopNRows command *****  
SELECT TOP (1000) [Id]  
      ,[RegionDescription]  
FROM [everyloop].[company].[regions]
```

Below the query, the 'Results' tab shows the following data:

	Id	RegionDescription
1	1	Eastern
2	2	Western
3	3	Northern
4	4	Southern



# Växla mellan databaser



# Mer SQL

- SQL består av:
  - Data Manipulation Language (DML)
    - **SELECT**, INSERT, UPDATE, DELETE
  - Data Definition Language (DDL)
    - CREATE, DROP, ALTER
    - GRANT, REVOKE

# Kopiera databastabell

- Ibland vill man göra en kopia av en tabell, allra helst då vi övar på SQL
- Välj rätt databas
- `SELECT * INTO <NyDatabas> FROM <GamlaDatabasen>`
- Exempel:
  - `SELECT * INTO company.categoriesTest FROM company.categories;`
  - `SELECT CategoryName, Id INTO company.categoriesTest2 FROM company.categories;`
  - `SELECT CategoryName as Cat, Id as IdNumber INTO company.categoriesTest3 FROM company.categories;`

# CRUD

- CRUD är en förkortning som används i de allra flesta programspråk, som en allmän beskrivning av då man hanterar data.
- C - Create
  - Skapa en databaspost, eller ett object i C#
- R – Read
  - Läs från en eller flera databasposter, eller t ex loopa igenom en List i C#
- U – Update
  - Ändra i en eller flera befintliga poster i en databas, eller ändra properties i ett C#-object
- D – Delete
  - Ta bort en eller flera poster i en databas, eller ta bort ett object ur en lista i C#

# Create => INSERT med SQL

- Kommandot **INSERT**

- INSERT INTO <tabellen> VALUES (<värden>)
- INSERT INTO <tabellen> (<kolumner>) VALUES (<värden>)
- INSERT INTO <tabellen> (<kolumner>) SELECT <kolumner> FROM <annantabell>
- `INSERT INTO company.categoriesTest VALUES(13, 'Candy', 'Candy is the best')`
- `INSERT INTO company.categoriesTest(CategoryName, Id) VALUES('Candy2', 16)`
- `INSERT INTO company.categoriesTest(Id, CategoryName, Description) SELECT 25, CategoryName, 'Gamlakategorier: ' + Description FROM company.categoriesTestB`

# Bulk INSERT

```
INSERT INTO company.categoriesTest(CategoryName, Id, Description) VALUES  
( 'Pizza', 16, 'Description' ),  
( 'Cookies', 17, 'Description' ),  
( 'Veggies', 18, 'Description' ),  
( 'Water', 19, 'Description' )
```

# Read => SELECT med SQL

- SELECT \* FROM <tabellen>
- Förra lektionen...

# Update => UPDATE med SQL

- Kommandot UPDATE
  - UPDATE <tabell> SET <kolumn=uttryck> WHERE <villkor>
    - OBS! Glöm inte WHERE-klausulen, den talar om VAD som ska ändras.
  - Exempel
    - UPDATE Employees SET LastName = 'Brown' WHERE EmployeeID = 1
    - UPDATE Employees SET Salary = Salary \* 1.5, JobTitle = 'Senior ' + JobTitle WHERE DepartmentID = 3



# Jämförelse med NULL

- Jämförelser med NULL
- • Kontrollera om något är NULL:
- SELECT LastName, ManagerId FROM Employees
- WHERE ManagerId **IS** NULL
- SELECT LastName, ManagerId FROM Employees
- WHERE ManagerId **IS NOT** NULL

# DELETE = Delete med SQL

- Ta bort rader från en tabell
  - DELETE FROM <tabell> WHERE <villkor>
- Exempel:
  - DELETE FROM Employees WHERE EmployeeID = 1
  - DELETE FROM Employees WHERE LastName LIKE 'S%'
- Ta bort alla rader från en tabell på en gång
  - TRUNCATE TABLE <tabell>

# Skapa en helt ny databas

- `CREATE DATABASE` MinEgenDatabas2
- `/* Utför ovanstående */`
- `GO`
- `/* Växla till nya databasen */`
- `USE` MinEgenDatabas2

# Radera en databas

- `DROP DATABASE MinEgenDatabas`
- `GO`
- `WARNING!!!`
- Radera innehållet i en tabell, men bevara tabellstrukturen
  - `TRUNCATE TABLE company.categoriesTestB`

# Skapa en ny tabell i en databas

- `CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
)`

- Exempel

- `CREATE TABLE Persons (  
    Id int,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255),  
  
    )`

# Datatyper

- Det är viktigt att välja rätt datatyp för att:
  - Den data man vill lagra KAN lagras
  - Att man kan skapa queries som fungerar som det är tänkt
  - Att spara på prestanda och lagringsutrymme
- [SQL Data Types for MySQL, SQL Server, and MS Access \(w3schools.com\)](https://www.w3schools.com/sql/sql_datatypes.asp)
  - Scrolla ner till avdelningen för "SQL Server Data Types"

# Sträng- datatyper

Data type	Description	Max size	Storage
char(n)	Fixed width character string	8,000 characters	Defined width
<b>varchar(n)</b>	Variable width character string	8,000 characters	2 bytes + number of chars
varchar(max)	Variable width character string	1,073,741,824 characters	2 bytes + number of chars
text	Variable width character string	2GB of text data	4 bytes + number of chars
nchar	Fixed width Unicode string	4,000 characters	Defined width x 2
<b>nvarchar</b>	Variable width Unicode string	4,000 characters	
nvarchar(max)	Variable width Unicode string	536,870,912 characters	
ntext	Variable width Unicode string	2GB of text data	
binary(n)	Fixed width binary string	8,000 bytes	
varbinary	Variable width binary string	8,000 bytes	
varbinary(max)	Variable width binary string	2GB	
image	Variable width binary string	2GB	

# Numeriska datatyper

Data type	Description	Storage
<b>bit</b>	Integer that can be 0, 1, or NULL	
tinyint	Allows whole numbers from 0 to 255	1 byte
smallint	Allows whole numbers between -32,768 and 32,767	2 bytes
<b>int</b>	Allows whole numbers between -2,147,483,648 and 2,147,483,647	4 bytes
bigint	Allows whole numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807	8 bytes
decimal(p,s)	Fixed precision and scale numbers. Allows numbers from $-10^{38} + 1$ to $10^{38} - 1$ . The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0	5-17 bytes
numeric(p,s)	Fixed precision and scale numbers. Allows numbers from $-10^{38} + 1$ to $10^{38} - 1$ . The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0	5-17 bytes
smallmoney	Monetary data from -214,748.3648 to 214,748.3647	4 bytes
money	Monetary data from -922,337,203,685,477.5808 to 922,337,203,685,477.5807	8 bytes
<b>float(n)</b>	Floating precision number data from $-1.79E + 308$ to $1.79E + 308$ . The n parameter indicates whether the field should hold 4 or 8 bytes. float(24) holds a 4-byte field and float(53) holds an 8-byte field. Default value of n is 53.	4 or 8 bytes
real	Floating precision number data from $-3.40E + 38$ to $3.40E + 38$	4 bytes



# Datum och tid

datetime	From January 1, 1753 to December 31, 9999 with an accuracy of 3.33 milliseconds	8 bytes
<b>datetime2</b>	From January 1, 0001 to December 31, 9999 with an accuracy of 100 nanoseconds	6-8 bytes
smalldatetime	From January 1, 1900 to June 6, 2079 with an accuracy of 1 minute	4 bytes
date	Store a date only. From January 1, 0001 to December 31, 9999	3 bytes
time	Store a time only to an accuracy of 100 nanoseconds	3-5 bytes
datetimeoffset	The same as datetime2 with the addition of a time zone offset	8-10 bytes
timestamp	Stores a unique number that gets updated every time a row gets created or modified. The timestamp value is based upon an internal clock and does not correspond to real time. Each table may have only one timestamp variable	

# Funktioner

- SQL-Server har en massa inbyggda funktioner, som kan vara användbara.
- Ett exempel
  - **FLOOR(RAND()\*(b-a))+a**;
  - FLOOR(RAND()\*(100-30))+30;
    - FLOOR = avrunda
    - RAND = Slumtalt mellan 0 och 1
    - Exemplet ovan ger ett slumtalt mellan 31-99
- [SQL Server: Functions - Listed Alphabetically \(technet.net\)](http://technet.net)
  - [SQL Server: RAND Function \(technet.net\)](http://technet.net)

# Övningar

## Övningsuppgifter 2 – CRUD

Låt dessa övningar ta lite tid, och lägg tid på att resonera kring bästa sättet att lösa dem.

Se först till att kopiera varje originaltabell i everyloop som ni jobbar med, till en ny tabell, som ni sedan kan modifiera. På så vis har ni alltid originalet kvar oförändrat.

Exempel: **select \* into Users2 from Users**

Om ni råkat göra fel i övningen och vill ha tillbaks det så kan ni återställa tabellen från originaltabellen.

# Övning 7 - Enterprise

- Gör en kopia av tabellen MoonMissions
- Lägg till en ny post med rymdskeppet Enterprise(Star Trek), med hjälp av INSERT
  - Ange alla värden som behövs (hitta på), och låt Outcome vara 'Successful'
- Ändra alla mån-uppdrag som opererats av NASA till 'Successful' med UPDATE
- Radera alla poster som inte är Successful med DELETE

# Övning 8- Från Kelvin till Celsius

- Gör kopia på tabellen Elements
- Uppdatera MeltingPoint och Boilingpoint-värdena i tabellen till Celsius (från Kelvin)
- Bygg sedan om SQL-queryt i Övning 4 (där man visar vilka grundämnen som är fast-flytande-gas) så det stämmer med de nya värdena.

# Övning 9 – Ändra användarnamn

- **Uppdatera** (kopia på) tabellen **user** och sätt username för alla användare så den blir de 2 första bokstäverna i förnamnet, och de 2 första i efternamnet (istället för 3+3 som det är i originalet). Hela användarnamnet ska vara i små bokstäver.
- Ledtråd: `LOWER(SUBSTRING(FirstName, 0, 3))`

# Övning 10 – Bort med NULL

- Gör kopia på tabellen airports
- **Uppdatera** tabellen airports så att alla null-värden i kolumnerna Time och DST byts ut mot '-'

# Övning 12 – Ta bort grundämnen

- Ta bort de rader från (kopia) tabellen Elements där "Name" är någon av följande: 'Erbium', 'Helium', 'Nitrogen', 'Platinum', 'Selenium', samt alla rader där "Name" börjar på någon av bokstäverna d, k, m, o, eller u.



# Övning 13 – Mitt CV

- Bygg en databas (CV), med en tabell (Meriter)
- Följande kolumner ska finnas med
  - Id
  - Arbetsgivare
  - Branch
  - Startdatum
  - Slutdatum
    - Bool för “pågående”
  - Arbetsbeskrivning (Längre text)
- Fyll på med en handfull egna meriter.

# Grupppuppgift – Egen databas

- Bygg en egen databas, i Gruppen.
- Välj en “produkt” av saker ni själva är intresserade av
  - Mina favoritspel
  - Min musiksamling
  - Mina släktingar
  - Favoritdrinkar
  - Mina prylar
  - Osv...
- Fundera på vilka kolumner ni behöver
  - Exempel: Favoritspel
    - Spelnamn, Tillverkare, Kategori, Pris, Betyg, Beskrivning, Länk till webbsida, level du är på osv
  - Fundera därefter på vilka datatyper som behövs
  - Skapa databas, tabell/er och fyll databasen med minst 10 poster
  - Skapa också två queries som gör ett urval av data
    - Textspel med betyg minst 4
    - Alla Actionspel
    - Alla Fantasyspel som är “jättedåliga”
- Kort redovisning på onsdag, där ni visar hur ni skapat databasen, tabellen samt hur ni kan ta bort/ändra/lägga till poster samt era egna SELECTs

# Ändra befintliga databaser

- [SQL ALTER TABLE Statement \(w3schools.com\)](https://www.w3schools.com/sql/sql_alter.asp)

# Demo – CRUD och skapa DB

- Dokument: sqldemo2.sql och sqldemo3.sql

# Länkar

- [SQL INSERT INTO Statement \(w3schools.com\)](https://www.w3schools.com/sql/default.asp)
- [SQL UPDATE Statement \(w3schools.com\)](https://www.w3schools.com/sql/default.asp)
- [SQL DELETE Statement \(w3schools.com\)](https://www.w3schools.com/sql/default.asp)
- [Learn SQL in 1 Hour - SQL Basics for Beginners – YouTube](https://www.youtube.com/watch?v=U01iLBRhk18)
- [SQL Data Types for MySQL, SQL Server, and MS Access \(w3schools.com\)](https://www.w3schools.com/sql/default.asp)
- [Date Functions in SQL Server and MySQL \(w3schools.com\)](https://www.w3schools.com/sql/default.asp)
- [SQL CREATE DATABASE Statement \(w3schools.com\)](https://www.w3schools.com/sql/default.asp)
- [SQL Tutorial \(w3schools.com\)](https://www.w3schools.com/sql/default.asp) (Igen)
- Quiz: [W3Schools SQL Quiz](https://www.w3schools.com/sql/default.asp)