

Introduktion till SQL

© Copyright

Mahmud Al Hakim

mahmud@webacademy.se

www.webacademy.se

Vad är SQL?

- Structured Query Language (SQL) är ett standardiserat frågespråk för att hämta och modifiera data i en **relationsdatabas**.
- Under 1970 utvecklade IBM "Structured English Query Language" (SEQUEL).
- *SEQUEL* byttes senare ut mot SQL eftersom 'SEQUEL' var ett varumärke.
- 1986 SQL är ANSI-standard
- 1987 SQL är ISO-standard
- SQL:2011 är den senaste versionen (standard ANSI)

SQL Tutorial

www.w3schools.com/sql

SQL Tutorial

« W3Schools Home [Next Chapter »](#)

SQL is a standard language for accessing databases. Our SQL tutorial will teach you how to use SQL to access and manipulate data in: MySQL, SQL Server, Access, Oracle, Sybase, DB2, and other database systems.

Examples in Each Chapter

With our online SQL editor, you can edit the SQL statements, and click on a button to view the result.

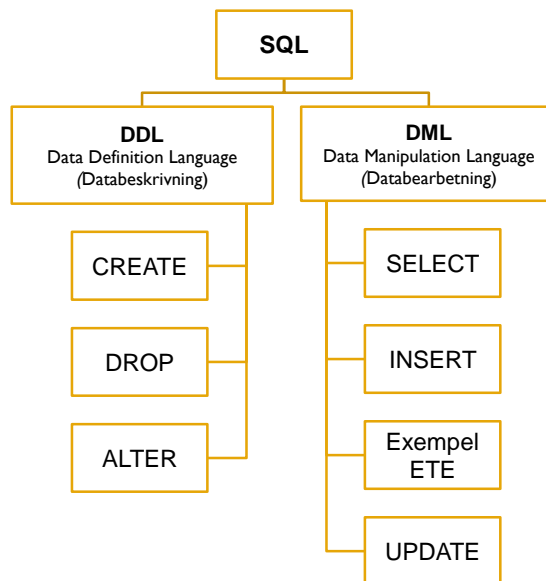
Example

```
SELECT * FROM Customers;
```

[Try it yourself >](#)

3

SQL (Structured Query Language)



SQL Syntax

- SQL-frågor består av ett antal satser (statements)
- SQL är INTE “Case Sensitive”
- SQL Nyckelord skrivs normalt i versaler

5

SELECT-satsen

- SELECT är den viktigaste satsen i SQL.
- Används för att hämta rader från en eller flera tabeller.

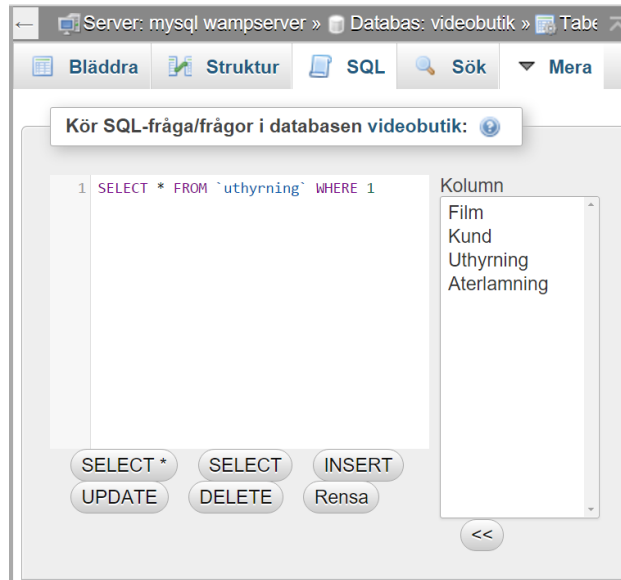
SELECT * FROM tabell;

Eller

**SELECT kolumn1, kolumn2
FROM tabell;**

6

Kör SQL-frågor via phpMyAdmin



7

SQL – Exempel I

-- Enkla frågor mot enstaka tabeller

```
SELECT * FROM Kunder
```

```
SELECT * FROM Filmer
```

```
SELECT * FROM Uthyrning
```

8

SQL – Exempel 2

-- Visa ej återlämnade filmer.

```
SELECT * FROM Uthyrning  
WHERE Aterlamning IS NULL
```

9

SQL – Exempel 3

-- Vilka olika filmer med "Mel Gibson" i huvudrollen finns i databasen?

```
SELECT * FROM Filmer  
WHERE Huvudroll = 'Mel Gibson'
```

10

SQL – Exempel 4

-- Visa antal filmer som inte lämnats tillbaka per kund (Visa kundnummer).

```
SELECT Kund, Count(Film) AS 'Antal filmer'  
FROM Uthyrning  
WHERE Aterlamning IS NULL  
GROUP BY Kund
```

11

SQL – Exempel 5

-- Visa en lista på alla kunder (kundnummer) som har mer än 10 filmer.

```
SELECT U.Kund, Count(U.film)  
FROM Uthyrning U  
WHERE U.Aterlamning IS NULL  
GROUP BY U.Kund  
HAVING Count(U.film) >= 10
```

12

SQL – Exempel 6

-- Cross Join (Kartesisk produkt)

```
SELECT * FROM kunder, Uthyrning;
```

```
SELECT * FROM Filmer, Uthyrning;
```

13

SQL – Exempel 7

-- Join (Inner Join) (koppla flera tabeller)

```
SELECT * FROM Kunder, Uthyrning
```

```
WHERE kunder.Kundnummer = Uthyrning.Kund
```

14

SQL – Exempel 8

-- Visa några fält från olika tabeller

```
SELECT Kunder.Namn, Uthyrning.Film,  
Uthyrning.Uthyrning  
FROM Kunder, Uthyrning  
WHERE kunder.Kundnummer = Uthyrning.Kund
```

15

SQL – Exempel 9

-- Använd Alias

```
SELECT K.Namn, U.Film, U.Uthyrning  
FROM Kunder K, Uthyrning U  
WHERE K.Kundnummer = U.Kund
```

16

SQL – Exempel 10

-- Använd nyckelorden INNER JOIN ON istället för WHERE

```
SELECT K.Namn, U.Film, U.Uthyrning
FROM Kunder AS K INNER JOIN Uthyrning AS U
ON K.Kundnummer = U.Kund
```

17

SQL – Exempel 11

-- INNER är inte obligatoriskt (Bara JOIN = INNER JOIN)

```
SELECT K.Namn, U.Film, U.Uthyrning
FROM Kunder AS K JOIN Uthyrning AS U
ON K.Kundnummer = U.Kund
```

18

SQL – Exempel 12

```
-- Visa ej återlämnade filmer. Visa kundnamnet och filmID (FK).
```

```
SELECT K.Namn, U.Film, U.Uthyrning  
FROM Kunder AS K, Uthyrning AS U  
WHERE K.Kundnummer = U.Kund  
AND U.Aterlamning IS NULL
```

19

SQL – Exempel 13

```
-- Visa ej återlämnade filmer.  
-- Visa Filmtitel och uthyrningsdatumet.  
-- (Alltså, vilka filmer finns ute)
```

```
SELECT Filmer.Titel, Uthyrning.Uthyrning  
FROM Uthyrning, Filmer  
WHERE Aterlamning IS NULL  
AND Filmer.FilmID = Uthyrning.Film  
ORDER BY Uthyrning DESC
```

20

SQL – Exempel 14

```
-- Visa ej återlämnade filmer.  
-- Visa Kundnamn, Filmtitel och uthyrningsdatumet.
```

```
SELECT K.Namn, F.Titel, U.Uthyrning  
FROM Filmer F, Kunder K, Uthyrning U  
WHERE K.Kundnummer = U.Kund  
AND F.FilmID = U.Film  
AND Aterlamning IS NULL
```

21

SQL – Exempel 15

```
-- Visa kundnamn och filmtitel som finns i tabellen uthyrning!
```

```
SELECT K.Namn, F.Titel, U.Uthyrning, U.Aterlamning  
FROM Kunder AS K, Filmer AS F, Uthyrning AS U  
WHERE K.Kundnummer = U.Kund  
AND F.FilmID = U.Film  
ORDER BY F.Titel
```

22

SQL – Exempel 16

-- Hur många gånger har "Dödligt vapen 1" hyrts ut?

```
SELECT Count(u.uthyrning)
FROM Filmer AS F , Uthyrning U
WHERE F.FilmID = U.Film
AND F.Titel = 'Dödligt vapen 1'
```

23

SQL – Exempel 17

-- Visa antal filmer som inte lämnats tillbaka per kund
-- (Visa Kundnamnet).

```
SELECT K.Namn, Count(U.Film) AS 'Antal filmer'
FROM Uthyrning U, Kunder K
WHERE Aterlamning IS NULL
AND K.Kundnummer = U.Kund
GROUP BY K.Namn
```

24

SQL – Exempel 18

```
-- Visa en lista på alla kunder (Kundnamn) som har mer än 10 filmer.
```

```
SELECT K.Namn, Count(U.film) AS Antal  
FROM Uthyrning U, Kunder K  
WHERE U.Aterlamning IS NULL  
AND K.Kundnummer = U.Kund  
GROUP BY K.Namn  
HAVING Count(U.film) >= 10
```

25

SQL – Exempel 19

```
-- Visa en lista på alla filmer som är ute  
-- Visa kundnamn och filmtitel
```

```
SELECT K.Namn, F.Titel  
FROM Uthyrning U, Kunder K, Filmer F  
WHERE U.Aterlamning IS NULL  
AND K.Kundnummer = U.Kund  
AND F.FilmID = U.Film  
ORDER BY K.Namn
```

26

SQL – Exempel 20

```
-- Vilka dramor har "Vera Lindberg" hyrt? (Kategori=Drama)

SELECT F.Titel, K.Namn, U.Uthyrning, F.Kategori
FROM Kunder K, Filmer F, Uthyrning U
WHERE K.Kundnummer = U.Kund
AND F.FilmID = U.Film
AND F.Kategori = 'Drama'
AND K.Namn = 'Vera Lindberg'
```

27

SQL – Exempel 21

```
-- Använd JOIN istället för WHERE

SELECT F.Titel, K.Namn, U.Uthyrning, F.Kategori
FROM Uthyrning U
JOIN Kunder K ON K.Kundnummer = U.Kund
JOIN Filmer F ON F.FilmID = U.Film
WHERE F.Kategori = 'Drama'
AND K.Namn = 'Vera Lindberg'
```

28

SQL – Exempel 22

```
-- Vilka filmer som kostar 30 kronor att hyra har Natalie  
Karlsmark hyrt?
```

```
SELECT F.Titel, F.Pris, K.Namn  
FROM Kunder K, Filmer F, Uthyrning U  
WHERE F.Pris = 30  
AND K.Namn = 'Natalie Karlsmark'  
AND K.Kundnummer = U.Kund  
AND F.FilmID = U.Film
```

29

SQL – Exempel 23

```
-- Använd JOIN istället för WHERE
```

```
SELECT F.Titel, F.Pris, K.Namn  
FROM Uthyrning U  
JOIN Kunder K ON K.Kundnummer = U.Kund  
JOIN Filmer F ON F.FilmID = U.Film  
WHERE F.Pris = 30  
AND K.Namn = 'Natalie Karlsmark'
```

30