

# Grundläggande CRUD-operationer

Create, Read, Update, Delete

v45



IT-HÖGSKOLAN

Här startar din IT-karriär.

# CRUD

Förkortningen CRUD nämns ibland i databas-sammanhang och bokstäverna står för de grundläggande funktionerna i en databas:

- **CREATE** (skapa nya rader)
- **READ** (läs ut data / SELECT)
- **UPDATE** (uppdatera befintliga rader)
- **DELETE** (ta bort rader)

Vi har hittills använt oss av olika select-satser (READ).

Låt oss nu kolla på de andra CRUD-operationerna ...



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Insert into

- När man vill lägga till rader i en tabell använder man sig av "Insert".
- Syntax: **INSERT INTO** tabellnamn (k1 , k2 , ... , kn ) values(v1 , v2 , ..., vn ) (där kn är kolumn-namn och vn är värden/data).
- Exempel: **INSERT INTO** people (name, age) values('Fredrik', 38);
- Man skriver alltså en INSERT-sats för vardera rad man vill lägga till.
- Tabellen man sätter in rader i måste redan finnas.



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Insert into select from

- Man kan även använda "Insert into" tillsammans med "select from", för att sätta in hela resultatet i en redan existerande tabell.
- Syntax: INSERT INTO tabellnamnmål (k1 , k2 , ... , kn ) SELECT v1 , v2 , ..., vn from tabellnamnkälla där kn är kolumn-namn och vn är värden/kolumner från annan tabell).
- Syntax: INSERT INTO tabellnamnmål (k1 , k2 , ... , kn ) SELECT v1 , v2 , ..., vn from tabellnamnkälla där kn är kolumn-namn och vn är värden/kolumner från annan tabell).



# Update

- Om vi vill ändra värden på en eller flera rader använder vi "Update".
- Syntax: UPDATE tabellnamn SET k1 = v1 , k2 = v2 , ..., kn = vn (där kn är kolumn-namn och vn är värden/data).
- Exempel: UPDATE people SET age = 39 WHERE name = 'Fredrik';
- OBS! Om man inte begränsar vilken/vilka rader som ska uppdateras med hjälp av "Where" så kommer samtliga rader i tabellen att uppdateras.



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Delete

- "Delete" används för att ta bort en eller flera rader ur en tabell.
- Syntax: DELETE FROM tabellnamn
- Exempel: DELETE FROM users WHERE id > 100;
- OBS! Om man inte begränsar vilken/vilka rader som ska tas bort med hjälp av "Where" så kommer samtliga rader i tabellen att raderas.



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Upsert

Utöver de vanliga CRUD-operationerna pratar man ibland om "upsert" som en mix av "update" och "insert". Det man då avser är att en rad ska uppdateras om den existerar, men skapas om den inte existerar.

I många fall får man lösa detta själv genom att först skicka en selectquery för att avgöra om raden existerar för att därefter besluta om den ska uppdateras eller om en ny ska skapas.

I senare versioner av SQL-server finns dock ett kommando "Merge" som gör detta automatisk.



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Varning!

- Man bör vara väldigt försiktig när man manuellt skriver SQLkommandon mot en databas. Alla queries man skickar till servern utförs direkt, oåterkalleligt, och utan varningar. ("Är du säker?").
- Sitter man mot en skarp databas som innehåller viktig information bör tänka sig för, och dubbelkolla sin query både en och två gånger innan man skickar den om det är något annat än en ren "select"-sats.
- Se till att alltid ha en backup som kan återställa all data om det skulle behövas, men tänk på att även med backup så är det krångligt och tar tid att återställa data.



IT-HÖGSKOLAN

Här startar din IT-karriär.



# Försiktighetsåtgärder

Några saker som kan hjälpa om man ändå behöver ändra manuellt:

- Kolla din query noga så du inte missat något villkor eller skrivit fel kolumn någonstans, då det kan vara förödande för din data.
- Kör en "select" med samma villkor som den delete (update) du tänkt köra och se hur många rader du får ut. Sedan när du är säker på att de bara är de rader du ska radera så byt "select" mot "delete".
- Kör en "Begin Transaction" innan din delete / update. Sen när du ser hur din delete / update gick så kan du välja commit eller rollback.



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Create table

- SQL-kommandot "Create table" används för att skapa en ny tabell:
- Syntax:

```
Create table tabellnamn (  
    kolumn1 datatyp, ...  
    kolumnn datatyp,  
)
```

- I sin enklaste form ser det ut som ovan, men man kan även lägga till andra definitioner när man skapar en tabell. T.ex att en kolumn är primary key, eller att en kolumn inte kan innehålla null-värden.



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Truncate table

- Om man vill ta bort alla rader ur en tabell är det mycket bättre att använda "truncate table" än "delete from".
- Syntax: TRUNCATE TABLE tabellnamn
- Exempel: TRUNCATE TABLE users;



IT-HÖGSKOLAN

Här startar din IT-karriär.

# Drop table

- Om man vill ta bort hela tabellen, inte bara innehållet så använder man kommandot "Drop table".
- Syntax: DROP TABLE tabellnamn
- Exempel: DROP TABLE users;



IT-HÖGSKOLAN

Här startar din IT-karriär.