# Database 02

|  |  |
| --- | --- |
| **Læringsmål** | Du kan:   * **1Pf4:** anvende centrale metoder og teknikker til at realisere modeller i et databasesystem […] |
| **Forventet læringsudbytte [SOLO]** | Programmeringssporet:   * [Unistrukturel] Du kan nævne enkelte kendetegn ved **database design** * [Unistrukturel] Du kan beskrive enkelte kendetegn ved en tabel (**database implementering**) |
| **Din forberedelse** | Programmeringssporet:   * Læs den praktiske guide til design og implementering af relationelle SQL-databaser (Kom i gang med databaser.docx på ItsLearning)   + Til og med ”Fase B: Database-design” i Del 1: Grundlæggende databaseudvikling * Udfør installationen af SSMS og Cisco AnyConnect med udgangspunkt i *SSMS og Cisco AnyConnect.docx* (kan findes under planer på ItsLearning) * Inspicer *Database oversigt - 2023.docx* (kan findes under planer på ItsLearning) * Læs om [SQL SELECT](https://www.w3schools.com/sql/sql_select.asp) * Læs om [SQL WHERE](https://www.w3schools.com/sql/sql_where.asp) * Læs om [SQL CREATE TABLE](https://www.w3schools.com/sql/sql_create_table.asp) * Læs om [SQL Server Data Type Mappings](https://docs.microsoft.com/en-us/sql/connect/ado-net/sql-server-data-type-mappings)   + Med fokus på, hvordan man går fra Int32 (*int),* String *(string),* Double *(double),* DateTime *og* Boolean (*bool)* i .NET Framework til SqlDbType   + Bemærk, at vi benytter os af .NET og ikke .NET Framework * Læs om [SQL AUTO INCREMENT](https://www.w3schools.com/sql/sql_autoincrement.asp)   + Med fokus på den syntaks, der benyttes for SQL Server (MS SQL Server) * Læs om [SQL INSERT INTO](https://www.w3schools.com/sql/sql_insert.asp) * Læs om [SQL DELETE](https://www.w3schools.com/sql/sql_delete.asp) * Læs om [SQL DROP TABLE](https://www.w3schools.com/sql/sql_drop_table.asp) * Læs om [SQL Comments](https://www.w3schools.com/sql/sql_comments.asp)   + Med fokus på notationen for Single Line og Multi-line * **Note:** Vi opretter tabeller i ental ikke flertal |

# Quote of the day

Sherlock Holmes: “*It is a capital mistake to theorize before one has data*.”

# Øvelse 1: Terminologi

Del teamet op i to mindre grupper (3-4 pers. i hver), og brug **Ordet rundt** til at reflektere over begreberne ”R-DBMS”, ”SQL”, ”SQL-statement”, ”Auto-increment” og ”SSMS ”. Sørg for, at alle får mulighed for at tale.

*Tidsramme: 20 minutter*

# Øvelse 2: Præsentation af udarbejdede artefakter

Skimlæs opgaven, og aftal med minimum to andre teams, hvornår I videndeler, samt hvordan I gør dette.

# Øvelse 3: UML-databasemodel, The Movies

Du skal i denne øvelse gå fra dit teams tidligere udarbejdede domænemodel til et relationelt databaseskema, for dernæst at gå fra dette relationelle databaseskema til en UML-databasemodel. Herefter skal du benytte denne UML-databasemodel til at udarbejde en række af SQL-tabeller for casen The Movies. Undervejs skal du inspicere mappingen mellem datatyperne i .Net og SQL Server Database Engine.

## Øvelse 3.1: Fra domænemodel til et relationelt databaseskema

I sidste opgave (Ex39), gik du fra en enkelt konceptuel klasse til et relationsskema. I denne opgave skal du og dit team tage udgangspunkt i *hele* jeres domænemodel (for The Movies casen) og udarbejde et relationelt databaseskema.

Udfør følgende:

* Beskriv alle tabeller, der repræsenterer jeres model
* For hver tabel beskriv alle dets attributter
* Angiv primærnøgler og fremmednøgler til at repræsentere alle relationer mellem tabeller

Brug tekstnotationen, som angivet i forberedelsen (Kom i gang med databaser.docx på ItsLearning), til at repræsentere hver enkelt tabel.

Benyt den CL-struktur, du finder bedst egnet.

## Øvelse 3.2: Inspektion af mapping

Som du også har læst dig til i forberedelsen, så er det vigtigt, at man har styr på, hvordan man mapper mellem datatyperne i .Net (C#) og SqlDbType. Derfor skal du nu genopfriske hukommelsen, ved at inspicere nedenstående tabel:

|  |  |
| --- | --- |
| .Net (C#) datatype | SQL datatype |
| int | Int |
| char | Nchar |
| string | NVarChar(\*) |
| double | Float |
| DateTime | DateTime2 |
| bool | Bit |

Figur: Mapping mellem datatyperne i .Net (C#) og SqlDbType

## Øvelse 3.3: Fra relationelt databaseskema til en UML-databasemodel

Nu, hvor du har fået styr på det relationelle databaseskema, samt hvordan mappingen foregår mellem datatyperne i .Net (C#) og SQL Server Database Engine, så er du og dit team klar til at udarbejde en databasemodel.

Fremgangsmåde:

* Start med at modellere et relationsskema i det relationelle databaseskema ad gangen (i datamodellen via en UML-klasse)
  + *Det er tilladt at blive inspireret af datatyperne i ens DCD, samt at benytte ens viden om mappingen mellem .Net (C#) og SQL Server Database Engine i dette punkt*
* Tilføj herefter de relevante oplysninger i henhold til at der er tale om en tabel (relation) i en database, samt bl.a. hvilken attribut der er primærnøgle ved hjælp af UML-stereotyper
  + **Hint**: <<table>>, <<PK>>, <<surrogate>>
* Fortsæt til at alle relationsskemaer er modelleret som UML-klasser i databasemodellen – altså at du og dit team har en samlet UML-databasemodel

**Note:** Vi kommer til at se på emnet ”normalisering” (herunder normalformerne; 1NF, 2NF, 3NF og 3,5NF/BCNF) i en fremtidig opgave, hvilket kommer til at have en direkte effekt på vores relationsskemaer/relationelle databaseskema.

# Øvelse 4: Fra databasemodel til realisering i R-DBMS

I denne del af øvelsen, skal du gå fra databasemodellen til en række af SQL ”CREATE TABLE”-sætninger, samt en række af SQL ”INSERT INTO”-sætninger.

## Øvelse 4.1: Fra relationel databasemodel til SQL ”CREATE TABLE”-sætning

Du skal i denne øvelse gå fra databasemodellen for ”The Movies”-casen til en række af SQL-tabels. Benyt din viden fra forberedelse til udformningen.

Inspiration fra forberedelsen til at komme i gang:

* [Create table](https://www.w3schools.com/sql/sql_create_table.asp)
  + [auto increment](https://www.w3schools.com/sql/sql_autoincrement.asp)
  + [korrekt mapping](https://docs.microsoft.com/en-us/sql/connect/ado-net/sql-server-data-type-mappings), se øvelse 3.2

## Øvelse 4.2: INSERT INTO

Du skal nu inspicere den tidligere udleverede Ex38-TheMovies.xlsx-fil, og med udgangspunkt i denne fil og jeres netop udarbejde SQL ”CREATE TABLE”-sætninger, skal du og din sidemakker oprette minimum 10 SQL ”INSERT INTO”-sætninger.

Inspiration fra forberedelsen til at komme i gang:

* [INSERT INTO](https://www.w3schools.com/sql/sql_insert.asp)

# Øvelse 5: Få forbindelse til DB via SQL Server Management Studio (SSMS)

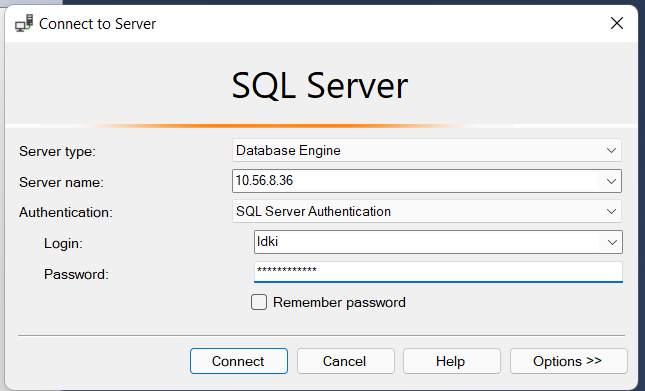
I denne øvelse skal du benytte SQL Management Studio til at forbinde til din udleverede database. For at se, hvilket database du har fået udleveret, skal du se på ItsLearning: Database oversigt - 2023.docx

**Bemærk:** Inden du går videre, skal du åbne ***Cisco VPN*** og logge ind med dine studielegitimationsoplysninger, som beskrevet i forberedelsen til i dag.

## Øvelse 5.1: Login i SSMS

Du skulle gerne kunne finde dig selv på listen i ”Database oversigt - 2023.docx”-dokumentet. Informationerne skal du benytte til at logge ind på serveren.

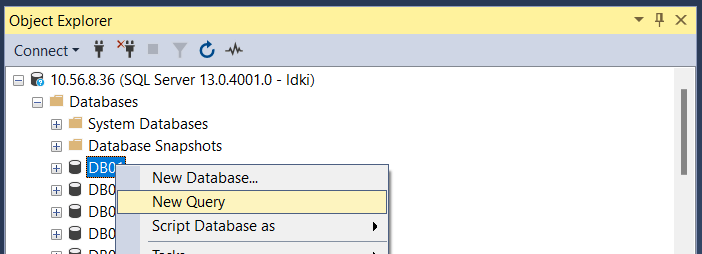
Åbn SQL Server Management Studio, og sikr dig, at der er valgt ”Database Engine” som ”Server type”. Bekræft også, at ”Authentication” er sat til ”SQL Server Authentication”.

Benyt nu dine logininformationer (fra Database oversigt - 2023.docx) Du burde se noget på din skærm, som ligner nedenstående billede. 

Tryk på ”Remember password” og på ”Connect”-knappen. Herefter burde du være i stand til at se serveren i din ”Object Explorer”.

## Øvelse 5.2: Se din Query

Fold ”Databases” ud ved at tykke på det lille kryds, og find din database. Højreklik på din database, og vælg ”New Query”, som på billedet nedenunder:



Nu er du klar til at indsætte SQL statements.

# Øvelse 6: Realisering af SQL-sætninger via SSMS

Benyt parprogrammering.

## Øvelse 6.1: Benytte Query, Oprette tabel

Du skal i denne øvelse gå sammen med din sidemakker. I skal benytte (udføre) jeres SQL ”CREATE TABLE”-sætninger, som du/I udarbejdede i øvelse 4.1. Åben jeres ”Query” (se øvelse 5.2, hvis du/I ikke kan finde din Query), og indsæt jeres SQL ”CREATE TABLE”-sætninger.

**Bemærk:** Du skal genopfriske din Object Explorer (inde i SSMS), før du kan se jeres oprettede tabel. Og du skal genopfriske din Local Cache for at få IntelliSense i din Query: Edit 🡪 IntelliSense 🡪 Refresh Local Cache

## Øvelse 6.2: Indsætte data i tabel

Du skal i denne øvelse gå sammen med din sidemakker og benytte jeres SQL ”INSERT INTO”-sætninger, som du/I udarbejdede i øvelse 4.2 (The Movies).

**Bemærk:** Igen; Du skal genopfriske din Object Explorer (inde i SSMS), før du kan se jeres oprettede tabel.