**Tips and tricks**

*Her er nogle tips, det er fortalt i undervisningen, men bare så I husker det.*

*Også baseret på de spørgsmål jeg har modtaget for nylig*

**Må man lave projekter/database på forhånd ?**

JA, Det er altid en god ide at lave så meget på forhånd som muligt og de må gerne lægges op i Azure på forhånd.

Lav tomme projekter f.eks

LibraryEksamen, RestEksamenService og Consumer med de nædvanlige NuGet pakker

Man kan endda lade sin Rest refere til library på forhånd☺

Gør også en Azure DB-tabel eller en lokal DB klar

Det samme gælder Test og TCP og UDP.

**Så spilder men ikke tid ved eksamen. Men har derimod forberedt sig godt.**

**Test generelt**

For et Test-projekt: Husk også MicroSoft.AspnetCore.Cors pakken

**Michael hvordan tester man exception-indhold/typer?**

Exceptions kan kastes både fra get/set-metoder for properties, constructor og metoder

Antag du har en throw der kaster en exception af typen MyException med indholdet ”Fejl”

Og du i en test nu vil verificere at det er denne type der er kastet

Try

{her foregår et eller andet som kunne kaste en exception}

catch (Exception e)

{//test af type

AreEquals(e.GetType(), TypeOf(MyException);

//test af korrekt besked

Assert.AreEqual(e.Message, "Fejl")

}

<https://social.msdn.microsoft.com/Forums/vstudio/en-US/659a9e1b-898e-4c96-bc0a-f25bb6eeee27/how-to-compare-exceptions-exceptionstypes?forum=csharpgeneral>

**Properties**

Kan autogeneres med *prop* og *propfull* med dobbeltanslag på TAB

<https://www.dotnetperls.com/property>

**XML-kommentarer, hvordan ?**

Det sker vha 3x/ og nøgleord som summary og return f.eks. ///summary

Se mere på

<https://docs.microsoft.com/en-us/dotnet/csharp/codedoc>

**Json**

Såfremt projektet kræver det, så husk at installere Newtonsoft.Json fra Project -> Manage NuGet packages

Eksempel på Json format: {"Id":2,"Name":"Niels","Age":12"}

Fiddler og Json Ved POST angiv:

Content-type: Application/json

**SQL –DB**
Husk når man sammenligner strings (varchars) in en database skal der anvendes gnyffer

f.eks. dyr = ‘giraf’

Det er en god ide at have et standard script i word notepad liggende, f.eks.

CREATE TABLE [dbo].[KundeTable]

(

 [Id] INT IDENTITY (1, 1) NOT NULL,

 [Navn] VARCHAR(50) NULL,

 [Adresse] VARCHAR(50) NULL,

 [Height] INT NULL,

PRIMARY KEY CLUSTERED ([Id] ASC)

)

**AzureDB**

I stien til databasen

Server=tcp:dinserver.database.windows.net,1433;Initial Catalog=dinDB;Persist Security Info=False;UserID=**{your\_username};**Password=**{your\_password};**MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;

skal {your\_password} og {your\_username} erstattes med dit password og brugernavn

Husk at configurere firewall settings til din IP-adresse når der skal forbindes til databasen vha. Visual Studio

**TCP og UDP**

Undgå port 9000 8000, 8888 og 9999, da de også bruges til andre netværksformål.

**SOAP – services**
Det kan være nødvendigt at refere til endpointConfigurationName

String endpointConfigurationName = "BasicHttpBinding\_IService1";

Hvis man publicerer en ny udgave af sit SOAP service program, provider, i Azure,

Så skal man huske at Update sin servicerefence i sit consumer program.

Ellers er ændringer ikke tilgængelige via wsdl-filen.