

## Øvelser - Dag 6 - WPF

### 6.1

Lav et basalt WPF program. Der skal blot være en stor knap med stor grøn text samt en CheckBox. Der kan f.eks. stå "Vis knap" som tekst til CheckBox'en

### 6.2

Lav et nyt WPF program, som indeholder et Grid. Tilføj 2 rækker og 3 søjler – dvs. det bliver et 2x3 grid. Tilføj en knap til hver af cellerne i den midterste søjle. Tilføj labels (dvs component typen Label) til alle de andre celler, som alle skal indeholde forskellige typer af text – F.eks. fire forskellige navne på superhelte eller hvad du nu har lyst til 😊

Du kan arbejde direkte i xaml filen eller bruge designeren til at dragge/droppe eller en kombination af de to ting.

Det kunne f.eks. se sådan her ud :



### 6.3

Gå tilbage til opgave 6.1

Lav en eventhandler til knappen (dens Click event). Eventhandleren skal ændre teksten på knappen (dens content-attribut). Husk knappen skal have et navn.

### 6.4

Fortsæt med opgave 6.3-projektet

Lav, så når checkboxen er checked er knappen synlig, ellers usynlig.  
Man skal fange Checked og Unchecked events. Man kan enten lave en handler for hver eller lade samme metode fange begge events.

Man kan også bruge Click-event, men det kører ikke hvis IsChecked ændres programmatisk.

Sammenhængen skal også gælde fra start, så enten eksplicit sæt synlighed fra start, eller sørg for, at CheckBox'en har rigtig værdi fra starten.

## 6.5

Gå tilbage til opgave 6.2

Tilføj en clickhandler til hver af de to knapper fra opgave 6.2. Click handleren skal bytte om på de to labels som står i cellen til højre og til venstre for knappen. Dvs. den øverste knap bytter om på teksten i de to øverste labels, mens den nederste knap skal bytte om på de to nederste labels.

**HINT:** For at ændre teksten på en label, så skal du ændre på Propertien "Content". Og du bliver nok nødt til at have en temp variabel for at bytte om på de to ting.

## 6.6

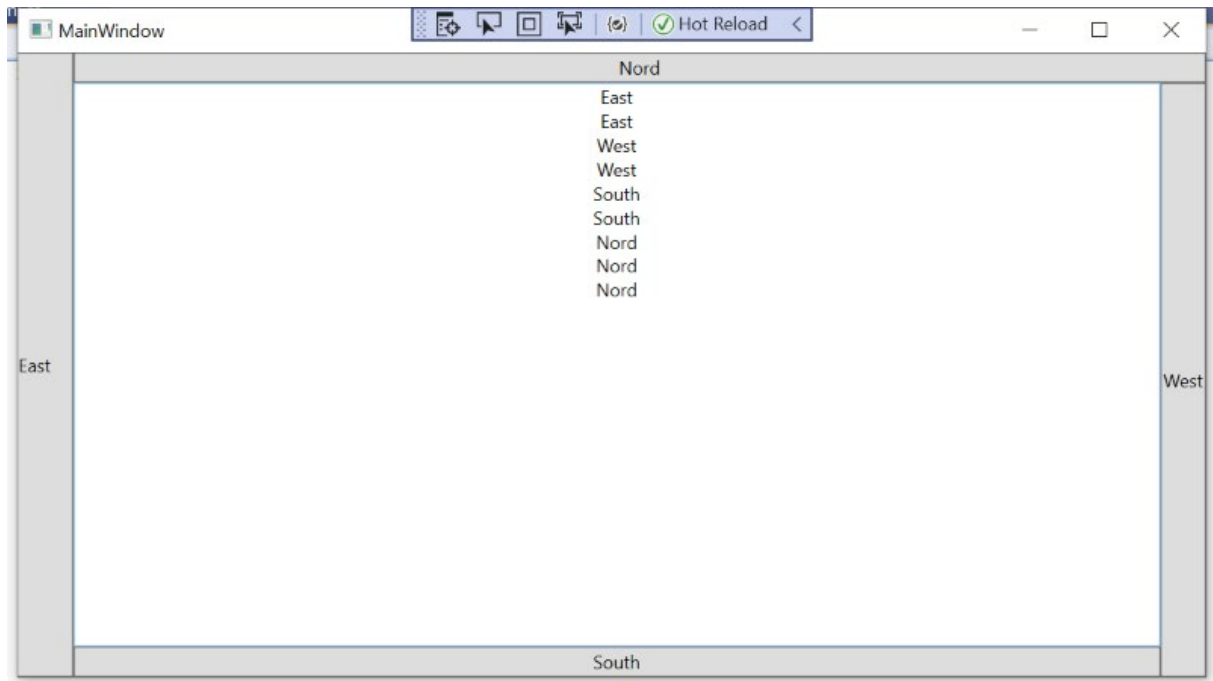
Lav en ny WPF applikation som bruger et DockPanel.

Du kan evt. hente hjælp i denne tutorial : <https://wpf-tutorial.com/panels/dockpanel/>

Tilføj en TextBox til center området og knapper til alle 4 border områder. Alle knapper skal have en forskellig label – f.eks. Nord, Syd, Øst, Vest eller noget i den stil. Lav clickhandlers, sådan at når en knap trykkes så tilføjes navnet på knappen til en ny linje i den TextBox i midten. Dvs. at hver gang man trykker kommer der en linje text mere i den midterste TextBox.

Hint: For at få det midterste element til at fylde resten af skærmen, så sæt denne Property i DockPanel erklæringen i xaml filen : `LastChildFill="True"`

Prøv også at få teksten til at stå centreret i textboxen.  
Det kunne f.eks. se sådan her ud når du er færdig:



## 6.7

Lav en ny WPF applikation med et Grid som indeholder en status bar og en menu bar. Menuen skal implementere disse muligheder:

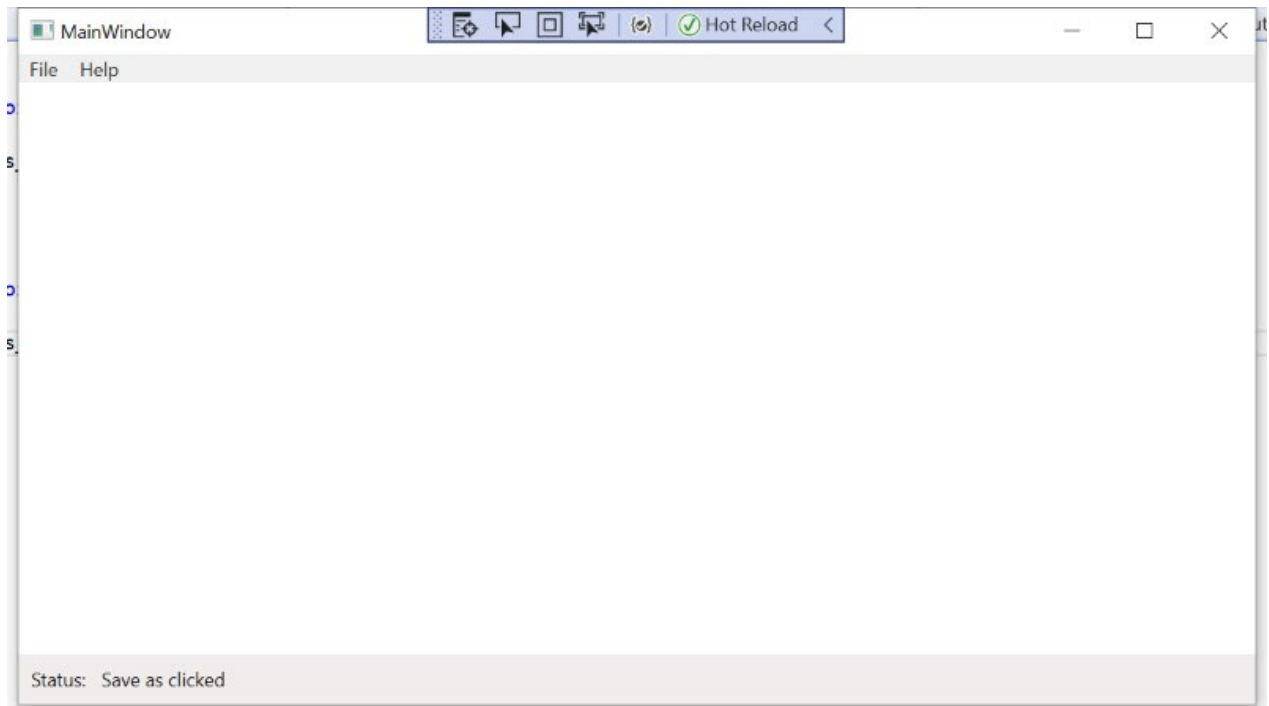
File->Open

File->Save as ...

File->Exit

Help->About

Det kunne evt. Se sådan her ud (under menuer ikke vist her):



Alle items skal have en clickhandler og StatusBar skal altid vise (navnet på) den sidste menu man har valgt – altså klikket på.

Hint: Du kan evt. se her hvordan menuer kan laves:

<https://wpf-tutorial.com/common-interface-controls/menu-control/>

men lad være med at bruge det, der står om commands.

For at lave eventhændlere kan det være nødvendigt at editere XAML-filen direkte

og du kan læse om status baren her:

<https://wpf-tutorial.com/common-interface-controls/statusbar-control/>

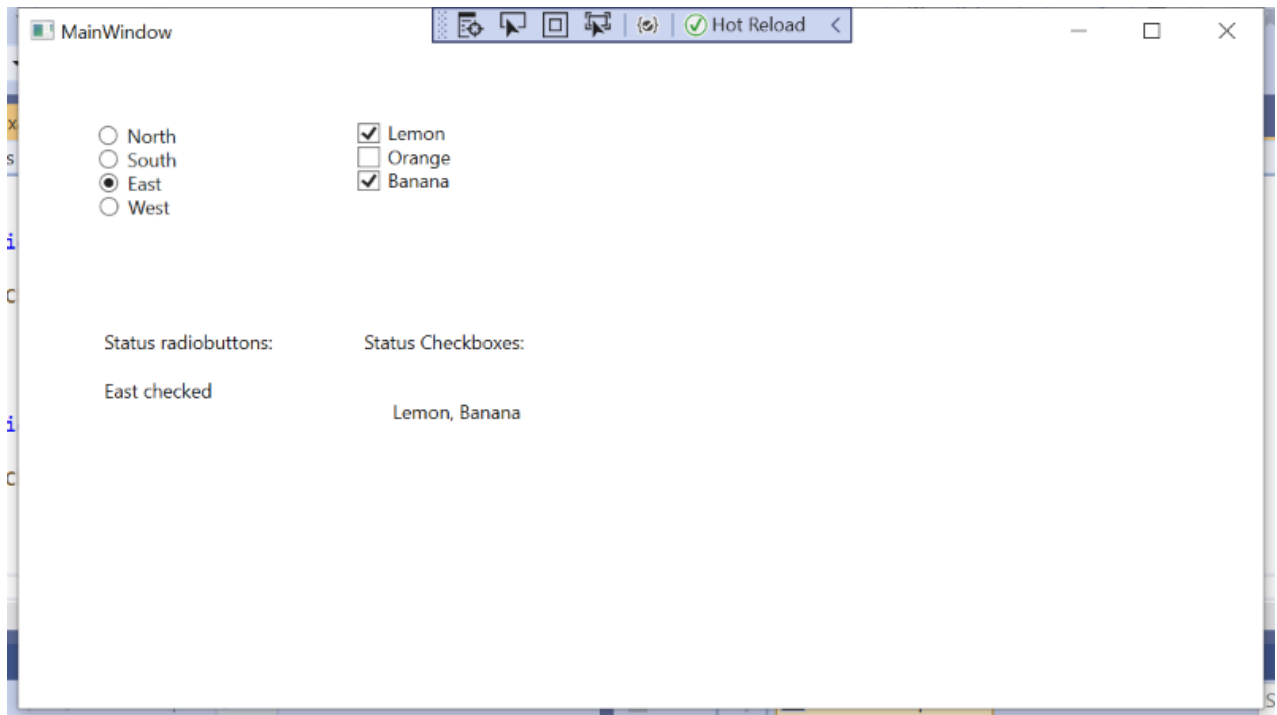
Hint : Det kan være fornuftigt at bruge et DockPanel og så sætte menuen i toppen og så sætte statusbar i bottom, men det kan også løses via Grid direkte.

## 6.8

Lav en ny WPF applikation, som indeholder et StackPanel med 4 radiobuttons (dvs. de er i samme "gruppe" – så kun en kan være valgt) og 4 checkboxes.

En statusbar i applikationen skal indikere den nuværende checked status på radiobuttons og checkbuttons. I stedet for en status bar, kan du også have to status labels som holder styr på hvad der er valgt af henholdsvis radio og checkbuttons.

Et eksempel på hvordan det kunne se ud er følgende:



Her kan du læse mere om Checkboxes:

<https://wpf-tutorial.com/basic-controls/the-checkbox-control/>

og RadioButtons:

<https://wpf-tutorial.com/basic-controls/the-radiobutton-control/>

or <https://www.c-sharpcorner.com/UploadFile/mahesh/radiobutton-in-wpf/>

**NOTE: Husk at teste at status OGSÅ opdateres korrekt, når du unchecker en CheckBox....**

## 6.9

Prøv at lave en WPF applikation som ligner dette: (eller lidt pænere 😊 )



Lad de to store felter være listboxes.

Det er en god træning i de forskellige UI components i WPF.

Der kræves ikke nogen funktionalitet. Det er blot UI'en i denne omgang.....

Ekstra opgave: implementer knapperne. Add skal tilføje det, brugeren har skrevet, til listboxen. ==> pilene skal flytte mellem listboxene. Ryd skal, ja gæt selv.

Flyt er ikke trivielt hvis man laver multiselect.