



METHODS

C# OPGAVEHÆFTE

Opgaver med metoder

Opgave 1:**Program: Returnering af en streng**

Lav et program, der kalder en metode, som returnerer en streng. Denne streng skal indeholde teksten **"Hello World!"**. Metoden skal kaldes fra Main-metoden og bruges som input til en **Console.WriteLine()**, så beskeden udskrives i konsollen.

Opgave 2:**Program: Udskrivning af brugerinput**

Lav et program, hvor brugeren kan indtaste en tekststreng som input. Derefter skal du oprette en metode, der modtager denne streng som parameter og udskriver den i konsollen. Metoden skal kaldes fra Main-metoden.

Opgave 3:**Program: Summen af tre tal**

Lav et program, hvor brugeren kan indtaste tre tal. Opret en metode, der tager imod de tre tal som parametre, lægger dem sammen, og returnerer summen. Summen skal derefter udskrives i konsollen via **Console.WriteLine()**.

Opgave 3a:**Program: Subtraktion af tre tal**

Dette program skal være en udvidelse af Opgave 3. I stedet for at lægge de tre tal sammen, skal metoden trække de tre tal fra hinanden og returnere resultatet, som derefter skal udskrives i konsollen.

Opgave 3b:**Program: Multiplikation af tre tal**

Udvid Opgave 3 ved at oprette en metode, der modtager tre tal som input, ganger dem sammen, og returnerer resultatet. Resultatet skal derefter udskrives i konsollen.

Opgave 3c:**Program: Summen af to tal divideret med et tredje**

Lav et program, der modtager tre tal som input fra brugeren. Opret en metode, der lægger de to første tal sammen og derefter dividerer summen med det tredje tal. Resultatet skal returneres og udskrives i konsollen.

Opgave 4:**Program: Navn og alder med gruppeinddeling**

Lav et program, hvor brugeren kan indtaste et navn og en alder. Opret en metode, der modtager disse input, og tjekker alderen for at bestemme, hvilken gruppe brugeren tilhører, baseret på følgende aldersintervaller:

- 0-1 år: Du er nyfødt
- 2-3 år: Du er i dagpleje eller vuggestue
- 4-5 år: Du er i børnehave
- 6-18 år: Du går i skole

- 19 år og opefter: Nu begynder livet at blive alvor

Metoden skal returnere en streng, som kombinerer navnet og en passende besked baseret på aldersgruppen, og denne besked skal derefter udskrives i konsollen.

Opgave 5:

Program: Opdeling af komma-separeret streng

Lav et program, hvor brugeren indtaster en enkelt streng med værdier adskilt af kommaer. Opret en metode, der modtager denne streng, splitter den op i et array af værdier, og returnerer arrayet. I Main-metoden skal du udskrive størrelsen på arrayet og bruge en **for-løkke** til at udskrive hver værdi i arrayet.

Opgave 6:

Program: Gæt et tal-spil

Lav et program, der simulerer et klassisk "gæt et tal"-spil. Programmet skal generere et tilfældigt tal mellem 1 og 25. Brugeren skal have op til 5 forsøg på at gætte tallet. Opret flere metoder til spillet:

- En metode til at generere det tilfældige tal.
- En metode til at tjekke, om brugerens gæt er korrekt.
- En metode, der informerer brugeren, om gættet er for højt eller for lavt.

Hvis brugeren gætter rigtigt, skal spillet afsluttes med en besked, der fortæller, at de har vundet. Hvis de løber tør for forsøg, skal spillet afslutte med en taberbesked.

Opgave 7.

Du skal lave en hovedmenu i et C#-konsolprogram, hvor brugeren kan navigere mellem flere forskellige opgaver. Menuen skal give brugeren mulighed for at vælge en opgave, køre den, og derefter give mulighed for at vende tilbage til hovedmenuen eller afslutte programmet. Hovedmenuen skal være dynamisk og kunne tilføje flere opgaver, hvis nødvendigt.

Krav:

1. Main Menu:

- Menuen skal præsenteres for brugeren, når programmet startes.
- Menuen skal liste numrene på forskellige opgaver med tilhørende beskrivelse.
- Menuen skal have et **valg** for at afslutte programmet (f.eks. "Tryk 0 for at afslutte").
- Når brugeren vælger et nummer fra menuen, skal det køre den tilsvarende opgave.
- Efter at have kørt opgaven, skal brugeren have mulighed for at vende tilbage til hovedmenuen eller afslutte programmet.

2. Brugerinteraktion:

- Brugeren skal kunne indtaste et nummer (f.eks. "1" for opgave 1) for at vælge hvilken opgave, der skal køres.
- Hvis brugeren vælger et ugyldigt nummer, skal der vises en fejlmeddelelse, og brugeren skal kunne prøve igen.

3. Menu Struktur:

- Implementer en **MainMenu()** metode, der indeholder menuen og håndterer valg af opgaver.
- Når en opgave er valgt, skal menuen kalde en metode, der kører den specifikke opgave (f.eks. Opgave1(), Opgave2(), osv.).

4. Tilbage til menu:

- Efter hver opgaveløsning skal brugeren kunne vælge, om de vil tilbage til hovedmenuen eller afslutte programmet.
- Hvis brugeren ønsker at afslutte, skal programmet lukke ordentligt.

Krav til Implementering:

- **Metoder:** Implementer en metode for hver opgave (f.eks. Opgave1(), Opgave2(), osv.), som håndterer opgavebeskrivelsen. Disse metoder kan være baseret på de tidligere opgaver.
- **Fejlhåndtering:** Hvis brugeren indtaster et ugyldigt valg, skal menuen præsentere en passende fejlmeddelelse og give brugeren mulighed for at vælge igen.
- **Luk programmet:** Når brugeren vælger at afslutte programmet, skal det afsluttes korrekt.

Ekstra Funktioner (valgfrit):

- **Udvidelse:** Du kan give brugeren mulighed for at se en kort beskrivelse af opgaverne, inden de vælger, om de vil køre dem.
- **Under-menuer:** Hvis du har mange opgaver, kan du overveje at opdele opgaverne i kategorier (f.eks. Matematikopgaver, String-manipulationer osv.) og lave under-menuer.

Trin for trin implementering:

1. Lav en metode, der viser hovedmenuen og håndterer brugerens valg.
2. Implementer separate metoder for hver opgave, som kaldes fra hovedmenuen baseret på brugerens valg.
3. Sørg for, at programmet returnerer til hovedmenuen efter hver opgaveløsning.
4. Implementer en metode til at afslutte programmet, når brugeren vælger at afslutte.

Opgave 7a.

I denne opgave skal du lave et program, der kan kaldes fra jeres hovedmenu (som er oprettet i Opgave 7). Programmet skal kunne omregne temperaturer mellem fire forskellige temperaturtyper: **Celsius**, **Fahrenheit**, **Kelvin** og **Réaumur**. Brugeren skal kunne indtaste en temperatur, der enten er i Celsius eller Fahrenheit, og herefter vælge, hvilken temperaturtype de har indtastet. Programmet skal derefter konvertere den indtastede temperatur til de andre tre typer og præsentere resultaterne på skærmen.

Krav:

1. Brugerinput:

- Programmet skal modtage en temperaturværdi fra brugeren. Temperaturen kan være et heltal eller et decimaltal (med komma eller punktum som decimalseparator).
- Brugeren skal efterfølgende angive, om den indtastede temperatur er i **Celsius** eller **Fahrenheit**.

2. Temperaturkonvertering:

- Programmet skal derefter omregne temperaturen til **Celsius**, **Fahrenheit**, **Kelvin** og **Réaumur** baseret på det brugervalgte input.
- Konverteringerne skal udføres ved hjælp af de korrekte matematiske formler (se nedenfor).

3. Udskrivning af resultater:

- Efter omregning skal alle fire temperaturtyper (Celsius, Fahrenheit, Kelvin, Réaumur) vises pænt og tydeligt på skærmen med passende enheder.

4. Formler til temperaturkonvertering:

- **Fra Celsius:**
 - $\text{Fahrenheit} = (\text{Celsius} \times 9/5) + 32$
 - $\text{Kelvin} = \text{Celsius} + 273.15$
 - $\text{Réaumur} = \text{Celsius} \times 4/5$
- **Fra Fahrenheit:**
 - $\text{Celsius} = (\text{Fahrenheit} - 32) \times 5/9$
 - $\text{Kelvin} = ((\text{Fahrenheit} - 32) \times 5/9) + 273.15$
 - $\text{Réaumur} = (\text{Fahrenheit} - 32) \times 4/9$

Programflow:

1. Programmet beder brugeren om at indtaste en temperatur.
2. Brugeren vælger, om den indtastede temperatur er i **Celsius** eller **Fahrenheit**.
3. Programmet konverterer den indtastede temperatur til de tre andre temperaturtyper ved hjælp af de nævnte formler.
4. Resultaterne for alle fire temperaturtyper (Celsius, Fahrenheit, Kelvin, Réaumur) udskrives pænt på skærmen.
5. Efter resultaterne har brugeren mulighed for at vende tilbage til hovedmenuen eller afslutte programmet.

Information til opgave 7a:

Celsius	$(\text{Fahrenheit} - 32) * 5 / 9$
Fahrenheit	$(\text{Celsius} * 9 / 5) + 32$
Kelvin	$\text{Celsius} + 273.15$
Réaumer	$\text{Celsius} * 0.8$

Opgave 7b.

I denne opgave skal du oprette et program, der kan omregne et tal, som brugeren indtaster, til tre forskellige repræsentationer: **decimal**, **hexadecimal** og **binær**. Hver omregning skal udføres i separate metoder, der modtager tallet og returnerer resultatet. Programmet skal kunne kaldes fra hovedmenuen, som blev lavet i Opgave 7.

Krav:

1. **Brugerininput:**

- Programmet skal bede brugeren om at indtaste et tal. Dette tal skal kunne være et heltal (int).

2. **Omregning af tallet:**

- Programmet skal konvertere tallet til følgende tre formater:
 - **Decimal** (hvis ikke allerede i decimalform)
 - **Hexadecimal**
 - **Binær**
- Hver omregning skal udføres i en separat metode, som modtager tallet som input og returnerer resultatet.

3. **Udskrivning af resultater:**

- Resultaterne af omregningerne skal vises pænt på skærmen, så brugeren kan se tallet i både decimal, hexadecimal og binær form.

4. **Metodekrav:**

- Implementér separate metoder til hver omregning:
 - En metode til at returnere det **hexadecimale** tal.
 - En metode til at returnere det **binære** tal.
 - (Valgfrit) En metode, der sikrer, at tallet er korrekt formateret som **decimal** (hvis det er nødvendigt).
- Hver metode skal returnere resultatet til hovedprogrammet, som udskriver det til skærmen.

Programflow:

1. Programmet beder brugeren om at indtaste et tal (som et heltal).
2. Programmet kalder separate metoder for at omregne tallet til **hexadecimal** og **binær** repræsentation.
3. Resultaterne udskrives pænt på skærmen med alle tre formater (decimal, hexadecimal og binær).
4. Efter udskrivning af resultaterne kan brugeren vælge at vende tilbage til hovedmenuen eller afslutte programmet.

Opgave 7c.

I denne opgave skal du oprette et program, der håndterer flere forskellige brugerinput af mindst to forskellige datatyper. Programmet skal gøre brug af mindst én løkke (for eksempel en for, while eller do-while løkke) og en kontrolstruktur som f.eks. en **if-else** for at træffe beslutninger baseret på input. Resultatet af beregningerne eller operationerne skal returneres og udskrives i konsollen. Programmet skal kunne kaldes fra den hovedmenu, der blev lavet i Opgave 7.

Krav:

1. Brugerinput:

- Programmet skal modtage mindst **to forskellige typer input** fra brugeren. Det kan være en kombination af:
 - **Heltal (int)**
 - **Decimaltal (float/double)**
 - **Tekst (string)**
 - **Boolske værdier (bool)**
- Input kan være valgfrit eller bestemt af et scenarie, men minimum to forskellige datatyper skal bruges.

2. Brug af Løkke:

- Programmet skal indeholde mindst én løkke. Løkken kan bruges til at iterere over en proces eller gentage handlinger, f.eks. at bede brugeren om at indtaste korrekte værdier indtil de overholder et bestemt krav, eller at udføre beregninger på flere input.
- Løkken kan være af typen for, while eller do-while, afhængigt af behovet i dit program.

3. Kontrolstruktur:

- Programmet skal bruge mindst én **if-else** (eller lignende kontrolstruktur) til at håndtere betingede valg baseret på brugerens input. Dette kan f.eks. være en kontrol af, om input er gyldigt, eller forskellige handlinger baseret på brugerens valg.

4. Returnering og udskrivning:

- Programmet skal returnere resultatet af sine beregninger eller operationer og udskrive resultatet til konsollen på en pæn og forståelig måde.

5. Menuintegration:

- Programmet skal kunne startes fra den hovedmenu, der blev oprettet i Opgave 7.

Programflow Eksempel:

1. Programmet beder brugeren om at indtaste to forskellige typer værdier, f.eks. et heltal og et decimaltal.
2. Programmet anvender en løkke til at gentage eller manipulere data baseret på brugerens input.
3. Programmet bruger en **if-else**-struktur til at træffe en beslutning baseret på input (f.eks. at kontrollere, om et tal er positivt eller negativt).
4. Programmet returnerer resultatet af sine handlinger og udskriver det i konsollen.
5. Efter at resultatet er udskrevet, kan brugeren vælge at vende tilbage til hovedmenuen eller afslutte programmet.