

Færdighedsprøve

I fagene

Software Construction

og

Software Design

1. semester 17. januar 2023

Varighed 4 timer

Domæne beskrivelse

I Vordingborg baby-gymnastikforening tilbydes der gymnastik til børn i alderen 1 – 4 år og deres forældre.

Der skal udvikles et system til at administrere tilmeldinger til foreningsaktiviteten. Systemet kaldes **Gymnastikforening**. Man skal kunne oprette forskellige hold og deltagere og det skal ligeledes være muligt at tilmelde sig et hold. Gymnastikken foregår i foreningens lokaler i hovedgaden og adressen skal ikke registreres i systemet.



I systemet skal der være en klasse **Hold**, som indeholder følgende properties:

Property	Type	Beskrivelse
HoldId	string	ID på holdet f.eks. "TumleE22"
HoldNavn	string	Navnet på holdet "Musik for tumlinger"
PrisPrDeltager	double	Deltagerprisen
MaxAntalBørn	int	Max antal der kan være på holdet

I systemet skal der ligeledes være et **Holdkatalog**, der indeholder alle holdene man kan tilmelde sig til. Klassen **Holdkatalog** skal indeholde en liste af **Hold**.

En forælder tilmelder sig sammen med et eller flere af sine børn. Oplysningerne registreres i en klasse **Deltager**, der indeholder følgende properties:

Property	Type	Beskrivelse
ForælderNavn	string	Navnet på den tilmeldte forælder
Adresse	string	Adressen for den tilmeldte forælder
AntalBørn	int	Antallet af børn, som er tilmeldt for den pågældende forælder

Opgave 1. Opret Klassen Hold.

- Opret et nyt Console App (.NET Core) projekt, **Gymnastikforening**
- Implementer klassen **Hold** med instansfelter, properties og passende konstruktører.
- Klassen skal have en *ToString()* metode.
- Test klassen ved at oprette nogle instanser af den i Program.cs. Skriv hver enkelt instans ud til konsollen.

Opgave 2. Opret klassen Holdkatalog

En instans af **Holdkatalog** klassen skal have en liste (af typen List) der indeholder **Hold** objekter. Denne liste skal instantieres i konstruktøren.

- Implementer klassen **Holdkatalog** med passende konstruktør.
- Klassen skal have en *ToString()* metode, som returnerer alle oplysninger i et **Holdkatalog**. I *ToString()* metoden skal du anvende en foreach-løkke, der gennemløber alle **Hold** objekterne i listen.

Opgave 3: Udvid Holdkatalog klassen med metoder

- Tilføj metoden *TilføjHold* til klassen **HoldKatalog**. Den skal have følgende signatur:

```
public void TilføjHold(Hold hold)
```

- Tilføj metoden *FindHold(string holdId)* til klassen **HoldKatalog**. Metoden returnerer et hold objekt med det pågældende holdId. Metoden skal have følgende signatur:

```
public Hold FindHold(string holdId)
```

- Test klassen **HoldKatalog** ved at oprette en instans af den i Program.cs. Og tilføj nogle instanser af klassen **Hold**.
- Skriv alle holdene ud ved at anvende *ToString()* metoden i **HoldKatalog**.

Opgave 4. Opret Klassen Deltager

- Implementer klassen **Deltager** med instansfelter, properties og passende konstruktør(er).
- Klassen skal have en *ToString()* metode.
- Test klassen ved at oprette nogle instanser af den i Program.cs. Skriv hver enkelt instans ud til konsollen.

Opgave 5. Tilmelding til hold

Det skal være muligt for en deltager at tilmelde sig et hold. For at registrere tilmeldinger skal der i **Hold** klassen oprettes en liste af typen List med navnet **deltagerListe**.

- Lav erklæringen af **deltagerListe** i klassen hold. Listen skal instantieres i Hold klassens konstruktør.
- Lav en metode **AntalTilmeldte** i klassen **Hold**, der angiver summen af tilmeldte børn til holdet. Metoden skal have følgende signatur:

```
public int AntalTilmeldte()
```

- Lav en metode **TilmeldDeltager** i klassen **Hold**, der tilføjer en deltager til holdet, hvis der er plads. Metoden skal have følgende signatur:

`public void TilmeldDeltager(Deltager deltager)`

- Hvis der ikke er plads på holdet til at tilmelde det antal børn, der er registreret på deltageren, skal der udskrives en passende fejlmeddelelse.
- Test metoden ved at tilmelde nogle deltagere til et hold.

Opgave 6. Tegn klassediagrammet som det ser ud på nuværende tidspunkt

- Opret et Visio dokument og tegn klasserne i diagrammet.
- I dit klassediagram bedes du redegøre/forklare de associationer der er benyttet.
- Der skal påføres multipliciteter til dit klassediagram.

Opgave 7. Exceptions

- Du skal tilrette din kode i metoden TilføjHold(Hold hold) i **HoldKatalog**, så du sikrer, at der ikke kan forekomme to hold med samme HoldId.
- Hvis man forsøger at tilføje et hold med samme HoldId som et eksisterende hold, skal der kastes en *ArgumentException* med en passende meddelelse.
- Vis hvorledes du i Program.cs kan håndtere og opsamle exceptions ved kald af metoden TilføjHold.

Opgave 8 Implementation af arv

Gymnastikforeningen har en særlig type hold der foregår i en af kommunens svømmehaller. De kaldes svømmehold.

Et **SvømmeHold** er en specialisering af et **Hold**:

- Implementér klassen **SvømmeHold** som arver fra klassen **Hold**.
- Implementer en property *string SvømmeHal* i klassen **SvømmeHold**, der angiver navnet på den af kommunens svømmehaller, som aktiviteten foregår i.
- Implementer en *ToString()* metode på klassen.
- Test **SvømmeHold** i program.cs. Opret nogle instanser og udskriv dem til konsollen.
- Opdater klassediagrammet fra opgave 6.

Opgave 9 Lav dine egne user stories

- Lav 2 user stories med minimum 3 acceptance kriterier, der relaterer sig til **Gymnastikforening** systemet.

Opgave 10 Implementér metoden `BeregnTotalPris` i klassen `Hold`

Hver deltager er registeret med en forælder og et antal børn.

Hvis **Deltageren** tilmelder en forælder og et barn, så er prisen holdprisen, kaldet **PrisPrDeltager** i klassen **Hold**. Men hvis der registreres mere end et barn på en **Deltager**, så koster de efterfølgende børn 50% af **PrisPrDeltager**.

PrisPrDeltager (property i Hold)	Der tilmeldes	Total pris
500 kr	1 forælder og et barn	500 kr
500 kr	1 forælder og 2 børn	750 kr
500 kr	1 forælder og 3 børn	1000 kr

- Implementer metoden `BeregnTotalPris` ud fra ovenstående beskrivelse. Metoden skal have denne signatur:

```
public Double BeregnTotalPris(int antalBørn)
```

- Vis hvorledes du kan kalde metoden fra `Program.cs` og udskrive prisen for en tilmelding.

Opgave 11 Tegn et sekvensdiagram

- Med udgangspunkt i `program.cs` skal der tegnes et sekvensdiagram der viser, hvorledes der tilføjes et hold til **HoldKatalog**.

12. Anvendelse af While-løkke

Implementér `FindHold()` metoden i klassen **Holdkatalog** vha. en While-løkke.