

Programmeringstest

Vi skal udvikle delmængden af koden til at administrere en børnehave.

I opgaven gives en beskrivende tekst af hver delopgave. Rammerne for opgaven ligger derfor fast. I et mere realistisk projekt vil det være op til udviklerne/ledelsen i højere grad selv at definere hvordan systemet bygges op.

Det er tilladt (og anbefales kraftigt) at afprøve de definitioner/metoder man udvikler for at afprøve om de virker. Dette kan ses som en opgave ud over de delopgaver som er beskrevet i opgaveformuleringen.

Det anbefales at skrive kommentarer til den udviklede kode.

Er der delopgaver du ikke kan løse fuldstændigt, kan en delvis besvarelse stadig give point i bedømmelsen.

Aflevering: Opret en ny mappe på din computer og gem alle javafilere som hører til projektet i denne mappe. Inden afleveringsfristen pakkes mappen til f.eks. en .zip fil og uploades til Fronter. Kun 1 fil pr. studerende uploades til Fronter. **Navngiv afleveringsfilen med DIT navn.**

A: Definér følgende klasser.

- 1: Child
- 2: Manager
- 3: Teacher
- 4: Kindergarden
- 5: Run, indeholdende start-metoden. (public static void main – metoden).

B: Child

- 1: Definér en attribut som er "Child's" navn. Kald attributen for 'name'. Overvej hvilken type attributen skal have. Attributen skal være privat for at indkapsle den.
- 2: Definér 'get' og 'set' metoder til den private attribut, 'name'.
- 3: Definér en konstruktør til Child klassen som tager en værdi ind som parameter svarende til child's navn. Sæt attributtens værdi til parameterens værdi.

C: Manager

- 1: Definér to private attributter til Manager. Kald dem 'name' og 'salaryBudget'. 'name' er Manager's navn og 'salaryBudget' er det samlede højeste beløb manageren må bruge pr. time på 'Teachers' samlede timeløn. Dvs. 'salaryBudget' er et beløb (i kr).
- 2: Definér en konstruktør til Manager klassen som tager to værdier ind som parametre og sætter 'name' og 'salaryBudget's' værdier.
- 3: Definér 'get' og 'set' metoder til attributterne.

D: Teacher (bemærk vi definerer ingen konstruktør)

- 1: Definér to private attributter til Teacher. Kald dem 'name' og 'hourlySalary'. 'name' er

Teacher's navn og 'hourlySalary' er timelønen.
2: Definér 'get' og 'set' metoder til de to attributter.

E: Kindergarden

- 1: Definér en privat attribut af typen Manager.
- 2: Definér en 'set' metode, 'setManager', som kan tage et manager objekt ind som parameter og sætte attributten. Pga. en tidligere dårlig oplevelse i denne Kindergarten med en leder som hed Per Jensen, ønsker bestyrelsen ikke en leder som hedder Per Jensen igen. Du skal derfor forhindre at et managerobjekt som har navnet Per Jensen kan tilknyttes Kindergarten.
- 3: Kindergarten skal kunne have vilkårligt mange Teacher objekter tilknyttet. Tilføj en attribut af en passende type.
- 4: Kindergarten skal kunne have vilkårligt mange Child objekter tilknyttet. Tilføj en attribut af en passende type.
- 5: Tilføj en metode, 'addChild', som kan tilføje et barn til 'Kindergarten'en. Sørg for at der højst kan være 50 børn tilknyttet. Forsøger man at tilføje barn nr. 51 skal barnet ikke tilføjes.
- 6: Tilføj en metode, 'printChildren', som printer alle childrenobjekters navne ved hjælp af System.out.println – metoden.
- 7: Tilføj en metode, 'printTeachers', som printer alle teacherobjekters navne (name) og timeløn (hourlySalary) ved hjælp af System.out.println – metoden.
- 8: Tilføj en metode, 'addTeacher', som kan tilføje et teacherobjekt til Kindergarten. Sørg for at teacherobjektet kun kan tilføjes hvis timelønnen (hourlySalary) er højst 250.
- 9: Tilføj en metode, 'calculateTotalSalary', som returnerer (ikke printer) alle Teacher's timelønninger lagt sammen.
- 10: Tilføj en metode, 'calculateAverageSalary', som returnerer (ikke printer) gennemsnitslønnen pr. time for alle teacherobjekter.
- 11: Ændr metoden fra delopgave E8 således at teacherobjektet heller ikke tilføjes hvis managerens budget bliver overskredet med den nye teachers timeløn. Dvs. findes der f.eks. 2 teacherobjekter som til sammen får 300 i timen i kindergardenobjektet og den nye lærer får 150 pr. time men managerens budget kun er til 410 i timen, må det nye teacherobjekt ikke tilføjes.

F: Run

- 1: Opret 4 childobjekter med navnene, Per, Hanne, Hubert og Caroline.
- 2: Opret 2 teacherobjekt og benyt derefter 'set' metoderne til at give teacherobjekterne navnene og timelønnen, Jens Henriksen, 180 i timeløn og Helle Petersen, 185 i timeløn.
- 3: Opret 1 managerobjekt med navnet Hans og et budget (salaryBudget) på 400.
- 4: Opret 1 kindergardenobjekt og tilknyt manageren, de 2 teacherobjekter samt de 4 childobjekter til kindergardenobjektet ved at benytte metoderne defineret i opgaveformuleringens sektion E.
- 5: Kald metoderne 'printChildren' og 'printTeachers'.
- 6: Kald metoden 'calculateAverageSalary' og benyt System.out.println – metoden til at udskrive det resultat som den returnerer.