

Schrijf je code in het Engels.

Gebruik naming conventions!

Labo week 2

Lesdoelen:

- Generics gebruiken
- Werken met lijsten
- Correcte loops kiezen
- Sorteren

Maak een nieuw project aan per labo!

Oefening 1 : Class Cage

Herneem je Animal classes van vorig labo.

Maak een nieuwe class kooi (**Cage**) aan. Een kooi is steeds voor één soort dier, zorg ervoor dat bij aanmaak van een nieuwe kooi je het soort dier al vastlegt. Gebruik hiervoor een generic. Zorg ervoor dat je enkel dieren (**Animal**) als generic kunt ingeven.

Voeg aan je kooi een addAnimal methode toe. Je voegt hier een dier toe van het soort dat je bij aanmaak van class hebt bepaald.

In een kooi kunnen meerdere dieren van hetzelfde soort zitten. Voorzie een correcte manier om de dieren die je toevoegt te bewaren in je kooi object.

Maak een extra methode om de dieren die in je kooi zitten te tonen.

Schrijf een Main class om alles uit te testen.

Uitbreiding: pas je kooi object aan zodat je bij aanmaak van de kooi kan beslissen hoeveel dieren er in de kooi mogen zitten.

Als je een dier toevoegt aan de kooi controleer je of de maximum nog niet bereikt is. Zit de kooi vol, dan geef je een false terug in het ander geval een true.

Pas in je main methode aan dat indien de addAnimal false teruggeeft, je een foutbericht in de console zet.

Maak genoeg dieren aan om de fout uit te lokken.

Oefening 2 : AnimalShelter

Maak een nieuwe package aan en zet daar een nieuwe class dierenasiel (**AnimalShelter**).

Het asiel vangt katten en honden op.

Voorzie volgende methods:

- addCat : voeg een kat toe aan het asiel
- addDog : voeg een hond toe aan het asiel
- printAllAnimals : toon de namen van alle dieren in het asiel IN VOLGORDE VAN TOEVOEGEN in de console, voeg ook een index toe in je output
- showAnimal : heeft als input een index en voert de toString() uit van het dier met de index
- removeAnimal : heeft als input een index en verwijdert het dier met die index uit de lijst.

Gebruik de classes uit de animals package.

Schrijf een Main class om je code te testen.

Oefening 3 : Bonobo's sorteren

Download van digitap de labo opgave code en plaats deze in een nieuwe 'zoo' package. In de Main staat een lijst van alle Bonobo's van dierenpark Planckendael.

1. Zorg ervoor dat de standaard sortering van Bonobo op naam is, sorteer de lijst en toon deze. Controleer of de sortering correct is.
2. Sorteer daarna de lijst op geboortjaar oplopend én indien twee geboortejaren hetzelfde zijn, op naam. Pas de standaard sortering niet aan. Sorteer de lijst en controleer of de sortering correct is.
3. Maak aan de hand van de lijst van Bonobo's een nieuwe lijst aan van enkel de mannelijke Bonobo's. Sorteer deze nieuwe lijst op id aflopend.

Uitbreiding: pas de Bonobo class aan en voeg class attributes vader (**father**) en moeder (**mother**) toe. Deze velden zijn optioneel en worden via de constructor ingevuld indien gekend.

Maak nu een sortering op naam van de moeder van de Bonobo. Indien de moeder niet gekend is, zet je deze achteraan in de lijst.

Extra oefeningen

Oefening 4 : GCD : Greatest Common Divisor

Maak een nieuwe package en schrijf een programma met een lus om de grootste gemene deler van twee gehele getallen te vinden.

Grootste gemene deler : het grootste gehele getal waardoor beide getallen gedeeld kunnen worden.

Voorbeelden : $\text{gcd}(15, 20) = 5$, $\text{gcd}(6, 9) = 3$, $\text{gcd}(6, 12) = 6$

Zoek het Algoritme van Euclides op voor de meest efficiënte oplossingsmethode.

Oefening 5 : Dobbelsteen

Maak een nieuwe package de class Dobbelsteen (**Dice**) aan. Een dobbelsteen heeft een aantal ogen (**pips**). Bij aanmaak is het aantal ogen standaard 6, maar dit kan bij aanmaak overschreven worden door bv 4, 8, 12 of 20.

Maak een methode op je dobbelsteen om een rol te doen. Deze dobbelsteenrol genereert een random geheel getal tussen 1 en het aantal ogen van je dobbelsteen. Gebruik hiervoor de java.util.Random class. Bekijk even de documentatie voor het gebruik van deze class.

Maak in een Main class 5 dobbelstenen van 6 ogen en schrijf code die deze dobbelstenen gooit en vervolgens nakijkt:

- Is er 5 keer een 6 gegooit ?
- Is er 5 keer dezelfde waarde gegooit ?
- GEVORDERD : is er een sequentie gegooit : 12345 of 23456 ?

Maak nadien een lus die deze code 10000 keer uitvoert en kijk na welk van bovenstaande situaties het vaakst voorkomt.

Toon het resultaat in je console.