

# Opdracht Polymorfisme Abstracte klasse-2 Vervoermiddelen

---

## Inleiding

Het doel van deze opdracht is om een applicatie te schrijven waarin lijsten van verschillende soorten vervoermiddelen kunnen worden bijgehouden. Door inheritance en abstracte klassen te gebruiken kun je toch objecten van verschillende soorten in een array zetten.

Kijk naar het klassendiagram op de laatste pagina. Zoals je kun zien hebben alle vervoermiddelen een identifier. Een nieuw vervoermiddel krijgt het laatste uitgegeven nummer + 1 als identifier. Om dat te regelen is er een klassevariabele `aantalVervoermiddelen` die bijhoudt hoeveel vervoermiddelen er aangemaakt zijn. Je kunt deze gebruiken om een nieuw object het juiste nummer te geven.

De klassen `Vervoermiddel`, `Voertuig` en `Vaartuig` zijn abstract en bedoeld om code te hergebruiken en maar op één plaats te hoeven schrijven. Een vervoermiddel heeft ook een abstracte methode genaamd `geefAandrijfKracht()`. Een concreet vervoermiddel kan pas aangeven hoe deze wordt aangedreven. Bij een koets is dat bijvoorbeeld "1 paard", "2 paarden", "4 paarden". Bij een auto is het "benzine motor", "diesel motor", "elektrische motor". Bij een fiets is het "menskracht", bij een zeilboot "windkracht" en bij een motorboot hetzelfde als bij een auto.

Je moet de methode `geefAandrijfKracht()` in de concrete klassen op de juiste wijze implementeren om voorgaande resultaat te kunnen geven.

## Stappenplan

1. Maak een project aan genaamd 'Vervoermiddelen'.
  - a. Maak twee packages aan genaamd 'controller' en 'model'.
  - b. Maak in de package 'controller' een klasse aan genaamd 'VervoermiddelenLauncher' en zorg dat deze klasse de `main` methode bevat.
2. Maak in de package 'model' alle klassen aan zoals gegeven in het UML diagram. Houd rekening met het volgende:
  - a. De constructors in de abstracte klassen zijn protected, zodat ze alleen in de subklassen gebruikt kunnen worden. Gebruik ze daar door middel van constructor chaining.
  - b. In de constructors kun je niet de id meegeven. Je moet er wel voor zorgen dat de id bij het aanmaken van een object de juiste waarde krijgt en dat het `aantalVervoermiddelen` wordt opgehoogd.
  - c. Het aantal wielen van een koets en een auto is standaard vier en van een fiets is het twee.
  - d. Denk goed na over de plaats van de get en set methode van de verschillende attributen. Zorg dat je geen code dubbel hoeft te schrijven.
  - e. Implementeer de methode `geefAandrijfKracht()` in de concrete klassen op de juiste wijze.
  - f. Geef alle klassen waar nodig een toepasselijke `toString()` methode, zodat je gemakkelijk een object kunt printen. Zorg ook hier ervoor dat je geen dubbele code schrijft.
3. Test de werking van al je klassen in de main method door van alle concrete objecten minstens één object aan te maken.
  - a. Maak een koets van het merk Glinkowski, type marathonwagen voor twee paarden, 320kg.
  - b. Maak een auto van het merk Dacia, type Logan MPV, 1150kg, rijdt op benzine.
  - c. Maak een fiets van het merk Koga Mijata, type Colmaro Race Ultegra, 9kg.
  - d. Maak een zeilboot Dufour 36 Classic van 11.5 meter.

- e. Maak een motorboot Broesder Kotter van 13,75 meter met een diesel motor.
4. Voeg alle gemaakte objecten toe aan een array van vervoermiddelen. Print vervolgens met behulp van een loop van elk vervoermiddel achtereenvolgens het id, merk, type en de aandrijfKracht.

## Output voorbeeld

```
Vervoermiddel #1 Glinkowski marathonwagen aangedreven door 2 paarden  
Vervoermiddel #2 Dacia Logan MPV aangedreven door benzine motor  
Vervoermiddel #3 Koga Mijata Colmaro Race Ultegra aangedreven door menskracht  
Vervoermiddel #4 Dufour 36 Classic aangedreven door windkracht  
Vervoermiddel #5 Broesder Kotter aangedreven door diesel motor
```

## Richtlijnen bij coderen (zie ook HBO-ICT code conventions [ICC])

- Zorg dat je naam en het doel van het programma bovenin staan (ICC #1).
- Gebruik de juiste inspringing (indentation) bij de lay-out (ICC #2).
- Let op juist gebruik hoofdletters en kleine letters (ICC #3).
- Gebruik goede namen (ICC #4).
- Voeg waar nodig commentaar toe die inzicht geven in je code (ICC#7).

