

## Assignment 31 – Animal Planet

### Doel

Het doel van deze opdracht is het oefenen met het Java OOP-concept **interface**.

### Achtergrond

Verschillende dieren (bijvoorbeeld een kat en een hond) hebben gemeenschappelijk gedrag. Zo heeft een dier een bepaald geluid, loopt op een bepaalde manier en slaapt op een bepaalde manier. Dit gedrag kan worden vastgelegd in een “contract” waar elk dier aan moet voldoen.

### Overzicht

Deze opdracht heeft 4 classes en een interface. Maak deze aan op het moment dat erom gevraagd wordt.

- Class Assignment31 (main)
- Interface Animal
- Class Dog
- Class Cat
- Class Pig

### 1. Voorbereiding

- a. Maak een package aan met de naam **assignment31** (let op packages beginnen altijd met een kleine letter)
- b. Maak in het package een nieuwe class met de naam **Assignment31** en maak de **main()** methode aan. De verdere implementatie van de **main()** methode wordt verder op in de opdracht besproken.

## 2. Interface Animal

We beginnen met het maken van de interface (het contract) **Animal**, omdat de interface gebruikt wordt door de classes **Dog**, **Cat** en **Pig**.

- a. Maak in de package **assignment31** een nieuwe interface met de naam **Animal** en geef als modifier het keyword **interface** mee (om Java kenbaar te maken dat het een interface betreft).
- b. De class **Shape** heeft 3 (abstract) methoden:
  - **animalSound()** met return type **void**
  - **run()** met return type **void**
  - **sleep()** met return type **void**
- Let op deze methodes hebben geen implementatie.

## 3. Class Dog

- a. Maak in de package **assignment31** een nieuwe class met de naam **Dog**.
- b. De class **Dog** implements van de class **Animal**
- c. Tot op dit punt zal IntelliJ aangeven dat er iets niet oke is met de class **Dog** (rood onderstreept), dit is omdat de implementatie van de 3 methoden nog niet gedaan is maar wel moet (we voldoen nog niet aan het 'contract').
- d. Implementatie methode **animalSound()**:
  - Deze methode print de hardcoded tekst "A dog says whoef whoef ..."
- e. Implementatie methode **run()**:
  - Deze methode print de hardcoded tekst "A dog is a fast runner ..."
- f. Implementatie methode **sleep()**:
  - Deze methode print de hardcoded tekst "A dog is sleeping ... Zzzzzzzz ..."

#### 4. Class Cat

- a. Idem als voor de class Dog, maar waarbij:
  - De methode **animalSound()** de hardcoded tekst “A cat says miauw miauw ...” uit print.
  - De methode **run()** de hardcoded tekst “A cat is a super fast runner ...” uit print.
  - De methode **sleep()** de hardcoded tekst “A cat is sleeping ... Zzzzzzzz ...” uit print.

#### 5. Class Pig

- a. Idem als voor de class Dog, maar waarbij:
  - De methode **animalSound()** de hardcoded tekst “A pig says wee wee ...” uit print.
  - De methode **run()** de hardcoded tekst “A pig is a very slow runner ...” uit print.
  - De methode **sleep()** de hardcoded tekst “A pig is sleeping ... Zzzzzzzz ...” uit print.

## 6. Class Assignment31

Methode main()

- a. Instantieer van de class **Dog**, een nieuw object met de naam **dog**.
- b. Roep de methode **animalSound()** aan van het object **dog**.
- c. Roep de methode **run()** aan van het object **dog**.
- d. Roep de methode **sleep()** aan van het object **dog**.
- e. Instantieer van de class **Cat**, een nieuw object met de naam **cat**.
- f. Roep de methode **animalSound()** aan van het object **cat**.
- g. Roep de methode **run()** aan van het object **cat**.
- h. Roep de methode **sleep()** aan van het object **cat**.
- i. Instantieer van de class **Pig**, een nieuw object met de naam **pig**.
- j. Roep de methode **animalSound()** aan van het object **pig**.
- k. Roep de methode **run()** aan van het object **pig**.
- l. Roep de methode **sleep()** aan van het object **pig**.

## 7. Output

Onderstaand de output van het programma na volledige implementatie:

```
A dog says whoef whoef ...  
A dog is a fast runner ...  
A dog is sleeping ... Zzzzzzzz ...  
A cat says miauw miauw ...  
A cat is a super fast runner ...  
A cat is sleeping ... Zzzzzzzz ...  
A pig says wee wee ...  
A pig is a very slow runner ...  
A pig is sleeping ... Zzzzzzzz ...
```

## 8. Opmerking

Deze opgave is ter kennismaking met interfaces en is eenvoudig gehouden. Uiteraard kunnen interfaces complexer zijn, door bijvoorbeeld methoden die parameters hebben en/of return waarden.