



Java Advanced

Inner classes

DE HOGESCHOOL MET HET NETWERK

Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Hasselt
www.pxl.be - www.pxl.be/facebook



Inhoud

- Inner class
- Local inner class
- Anonymous inner class
- Static nested class



Inner classes

```
public class Person {
    private String name;
    private LocalDate dateOfBirth;

    public Person(String name, LocalDate dateOfBirth) {
        this.name = name;
        this.dateOfBirth = dateOfBirth;
    }

    public int getAge() {
        return new PersonHelper().getAge();
    }

    public class PersonHelper {

        public int getAge() {
            return Period.between(dateOfBirth,
                                   LocalDate.now()).getYears();
        }
    }
}
```



Inner classes

```
public static void main(String[] args) {  
    Person person1 = new Person("Emma Watson",  
                                LocalDate.of(1990, Month.APRIL, 15));  
    System.out.println(person1.getAge());  
  
    Person person2 = new Person("Eden Hazard",  
                                LocalDate.of(1991, Month.MAY, 7));  
  
    Person.PersonHelper helper = person2.new PersonHelper();  
    System.out.println(helper.getAge());  
}
```



Taak 1

- Maak een klasse *Musician* met een geneste klasse *Instrument*
- Voeg aan de klasse *Instrument* de methode *makeNoise()* toe. Laat deze methode een stukje tekst afdrukken.
- Voeg aan de nestende klasse *Musician* de methode *play()* toe. Hierin maak je een object aan van de klasse *Instrument* en roep je de methode *makeNoise()* aan.
- Maak een hoofdprogramma waarin je een object van de klasse *Musician* aanmaakt en de methode *makeNoise()* aanroept.
- Instantieer in het hoofdprogramma het instrument en roep de methode *makeNoise()* aan.



Local inner classes

```
public class Person {  
    private String name;  
    private LocalDate dateOfBirth;  
  
    public Person(String name, LocalDate dateOfBirth) {  
        this.name = name;  
        this.dateOfBirth = dateOfBirth;  
    }  
  
    public int getAge() {  
        class PersonHelper {  
            public int getAge() {  
                return Period.between(dateOfBirth,  
                                       LocalDate.now()).getYears();  
            }  
        }  
        return new PersonHelper().getAge();  
    }  
}
```

Local inner classes

```
public int getAge() {  
    int extraDays = 5;  
    class PersonHelper {  
        public int getAge() {  
            extraDays++;  
            return (dateOfBirth, LocalDate.now().plusDays(extraDays))  
        }  
    }  
}
```

Variable 'extraDays' is accessed from within inner class, needs to be final or effectively final



Taak 2

- Maak nu van de geneste klasse Instrument een lokale geneste klasse.

Anonymous inner classes

```
public interface Age {  
    int getAge();  
}
```



Anonymous inner classes

```
public class Person {  
    private String name;  
    private LocalDate dateOfBirth;  
  
    public Person(String name, LocalDate dateOfBirth) {  
        this.name = name;  
        this.dateOfBirth = dateOfBirth;  
    }  
  
    public int getAge() {  
        Age ageCalculator = new Age() {  
            @Override  
            public int getAge() {  
                return Period.between(dateOfBirth,  
                                       LocalDate.now()).getYears();  
            }  
        };  
        return ageCalculator.getAge();  
    }  
}
```



Taak 3

- Maak een interface Instrument met 1 methode makeNoise().
- Maak nu in de methode play() van Musician een object van een anonieme klasse die de interface Instrument implementeert.



Static inner classes

```
public class Person {  
    private String name;  
    private LocalDate dateOfBirth;  
  
    public Person(String name, LocalDate dateOfBirth) {  
        this.name = name;  
        this.dateOfBirth = dateOfBirth;  
    }  
  
    public int getAge() {  
        return new PersonHelper().getAge(dateOfBirth);  
    }  
  
    static class PersonHelper {  
  
        public int getAge(LocalDate dateOfBirth) {  
            return Period.between(dateOfBirth,  
                                   LocalDate.now()).getYears();  
        }  
    }  
}
```



Static inner classes

```
public static void main(String[] args) {  
    Person person = new Person("Emma Watson",  
                                LocalDate.of(1990, Month.APRIL, 15));  
    System.out.println(person.getAge());  
  
    Person.PersonHelper helper = new Person.PersonHelper();  
    System.out.println(  
        helper.getAge(LocalDate.of(2001, Month.OCTOBER, 1)));  
}
```



Static inner classes

```
public class ColorEnums {  
    public static enum Color {  
        RED,  
        GREEN,  
        BLUE;  
    }  
}
```

```
ColorEnums.Color color = ColorEnums.Color.RED;
```

