

Software Development

The magic of “static”?

Koen Pelsmaekers

Unit EA-ICT (GT.03.14.05)

koen.pelsmaekers@kuleuven.be

Static

- Used to define “class” variables and methods

```
public class Person {  
    private static int nrOfPersonObjects = 0;  
    private String name;  
    private LocalDate birthday;
```

Static variable

```
    public Person(String name, LocalDate birthday) {  
        this.name = name;  
        this.birthday = birthday;  
        nrOfPersonObjects++;  
    }
```

```
    public static Person personBornToday(String name) {  
        return new Person(name, LocalDate.now());  
    }
```

Static “factory” method

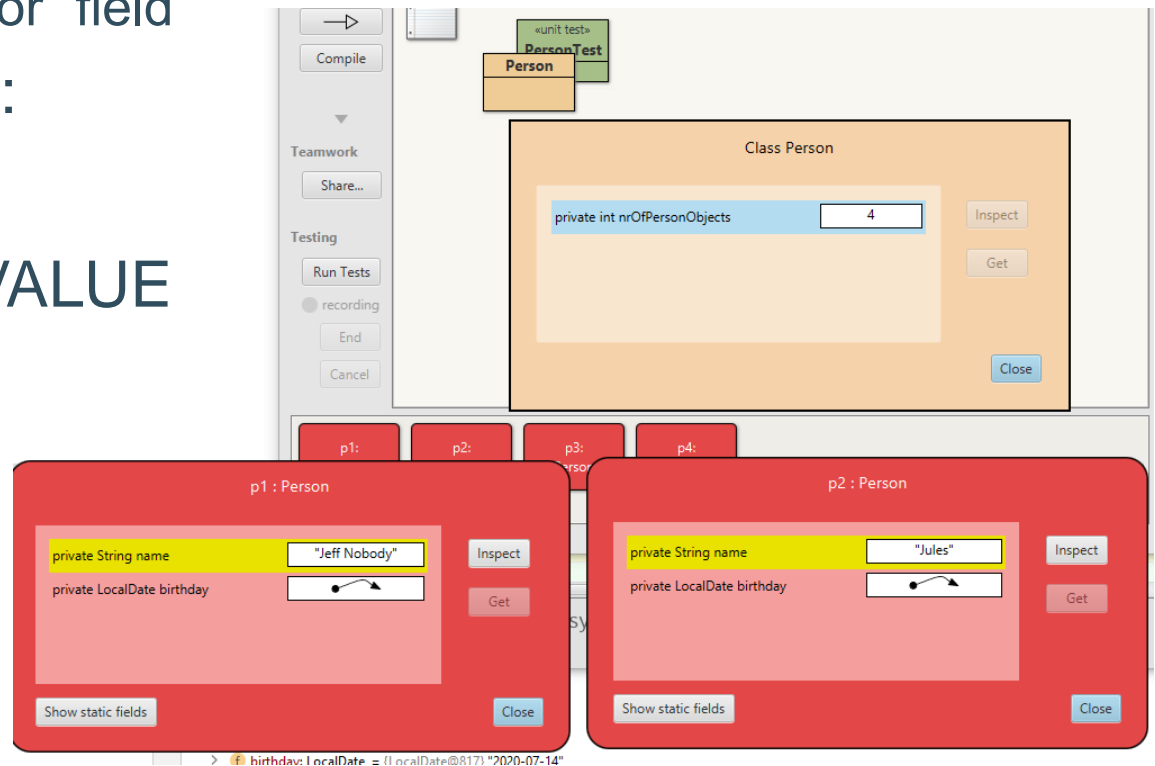
```
    public static int getNrOfObjects() {  
        return nrOfPersonObjects;  
    }
```

Static method using static variable

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

Static variable

- Static variable == class variable
 - Only one copy per class
 - No copy in instances/objects of class
 - no “property” or “field”
 - BlueJ example:
- Static final values
 - Integer.MAX_VALUE
 - Math.PI
 - Period.ZERO



Static method

- Method called without “object”
- Examples
 - Method to use static field
 - `Person.getNrOfPersonObjects()`
 - Static factory method (“Effective Java”, Joshua Bloch)
 - `Person.personBornToday(..)`
 - `Period.between(.., ..)`
 - `LocalDate.now()`
 - `String.valueOf(..)`
 - Utility methods
 - `Objects.equals(o1, o2)`
 - `Collectors.toList()`
 - ...

Main function

- Main function is a static function
- Virtual machine calls: “<classname>.main”
- Example: “Person.main(<args>)”;
 - VM does not create instance of a class!
 - It is typically your first statement...

```
public class Person {  
    private static int nrOfPersonObjects = 0;  
    private String name;  
    private LocalDate birthday;  
  
    public static void main(String[] args) {  
        Person p1 = new Person ("Flor", LocalDate.of(2019,10,25));  
    }  
    // constructor, getters, setters, other methods, ...  
}
```