



**UCLL**  
HOGESCHOOL

Optional

Nadir Aboulkassimi  
Bram Van Impe

#MOVINGMINDS

# Inhoud

- Wat is null?
- Wat is een NullPointerException?
- Wat is Optional?

# Wat is null?

- De default waarde van een object is null.
- Wanneer we een object declareren zonder die te initialiseren dan is de waarde hiervan null.
  - Aangezien we (nog) niet verwijzen naar een geheugenadres.

# Wat is NullPointerException?

- Je krijgt een foutmelding
  - Indien we een eigenschap of functie oproepen van een object die niet werd geïnitieerd.
  - Indien we een eigenschap wensen aan te passen van een object die niet werd geïnitieerd.

```
class Bar {  
    public int foo;  
}
```

```
Bar obj = null;  
System.out.println(obj.foo);
```

```
class Bar {  
    public int foo;  
}
```

```
Bar obj = null;  
obj.foo = 1;
```

# NullPointerExceptions vermijden

- Een NullPointerException kan je vermijden door

- Zelf te controleren op null.

```
Object obj = null;  
if (obj != null) {  
    obj.hashCode();  
}
```

- De Javadoc te lezen van klassen/functies die je gebruikt zodat je hier rekening mee kunt houden in je code.
  - <https://docs.oracle.com/javaee/7/api/javax/persistence/EntityManager.html#find-java.lang.Class-java.lang.Object->
  - In bovenstaande url kan je de Javadoc van een functie vinden die null terug geeft indien het object niet bestaat.

Returns:

the found entity instance or null if the entity does not exist

# NullPointerExceptions vermijden

- Een NullPointerException kan je vermijden door
  - Objecten niet met null te initialiseren.
  - Geen null terug te geven als return in je functies.
    - Je kan in de plaats bijvoorbeeld een exception gooien of een lege collection terug sturen.

# Wat is Optional?

- Vanaf JDK 8
  - <https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/util/Optional.html>

**API Note:**

Optional is primarily intended for use as a method return type where there is a clear need to represent "no result," and where using null is likely to cause errors.

A variable whose type is Optional should never itself be null; it should always point to an Optional instance.

- Je kan optional gebruiken indien je niet zeker weet of er een resultaat is.
  - Dit is beter dan een null terug te sturen.
  - Dit zorgt voor minder fouten.

# Optional API - Creatie

Static methods	
Optional.empty()	Maakt een lege Optional aan.
Optional.of(T value)	Maakt een Optional aan met de gegeven non-null value.
Optional.ofNullable(T value)	Maakt een Optional aan met de gegeven value indien de gegeven value non-null is, anders een lege optional.



# Optional API - Methodes

Instance method	
get()	Geeft waarde terug indien deze aanwezig is, anders gooit deze methode een NoSuchElementException.
isPresent()	Geeft true terug als de optional een waarde heeft, anders false.
isEmpty()	Omgekeerde van <b><i>isPresent()</i></b> .
orElse(T other)	Geeft waarde terug indien deze aanwezig is, anders de gegeven <b><i>other</i></b> .
orElseThrow()	Geeft waarde terug indien deze aanwezig is, anders gooit deze methode een NoSuchElementException.

# Optional API – Methodes (1)

Instance method	
<code>filter(Predicate&lt;T&gt; predicate)</code>	Indien de <code>Optional&lt;T&gt;</code> een waarde heeft en <code>Predicate&lt;T&gt;</code> van toepassing is, dezelfde <code>Optional&lt;T&gt;</code> , anders lege <code>Optional</code> .
<code>map(Function&lt;T,U&gt; mapper)</code>	Indien de <code>Optional&lt;T&gt;</code> een waarde heeft, een <code>Optional&lt;U&gt;</code> , anders een lege <code>Optional</code> .
<code>orElseGet(Supplier&lt;X&gt; supplier)</code>	Equivalent aan <b><i>orElse</i></b> , maar gebruikt gegeven supplier.
<code>orElseThrow(Supplier&lt;X&gt; supplier)</code>	Equivalent aan <b><i>orElseThrow</i></b> , maar gebruikt gegeven supplier.
<code>stream()</code>	Geeft een <code>Stream</code> terug die ofwel een waarde bevat of niets.

# Optional API – Methodes (2)

Instance method	
ifPresent(Consumer<T> consumer)	Voert Consumer uit met waarde van de optional, indien deze er één heeft.
ifPresentOrElse(Consumer<T> consumer, Runnable runnable)	Voert Consumer uit met waarde van de optional, indien deze er één heeft, zo niet voert deze methode de gegeven runnable uit.

# Nuttige links

- <https://www.baeldung.com/java-avoid-null-check>
- <https://www.baeldung.com/java-optional>

# Afronden

- Vragen?