

Equals

Implementer egen equals på alt som ikke er primitiv

`x == y` sjekker om de er på samme sted i minne (på stack'en). Funker av å til med primitive variabler, men hva med egendefinerte?

Feks.. to Personer med samme navn og id! De er jo like, men er ikke samme sted på minne, så `==` funker ikke. Vi må `@Override equals`-metoden

Vanlig equals er egentlig bare `==`

```
public boolean equals(Object obj) {  
    return (this == obj);  
}
```

så vi må lage egen:

```
@Override  
public boolean equals(Object obj) {  
    // ønsket oppførsel  
    if (obj == null) {  
        return false;  
    }  
    // prøv ==  
    if (obj == this) {  
        return true;  
    }  
    // obj må også være samme klasse  
    if (!(obj instanceof Person)) {  
        return false;  
    }  
  
    // hashCode er viktig  
    if (!obj.hashCode() == this.hashCode()) {  
        return false;  
    }  
  
    // cast and check all instance variables  
    Person other = ((Person) obj); // cast object to person  
    return this.name.equals(other.name) && this.id.equals(other.id);  
    // bruk assertTrue(Arrays.deepEquals(anArray, anotherArray)); for  
    arrays  
}  
  
// !!! IMPORTANT: WHEN OVERRIDING EQUALS ALWAYS OVERRIDE HASHCODE  
@Override  
public int hashCode() {
```

```
        return Objects.hash(this.name, this.id);  
    }
```

Rules:

Anta at x ikke er null. Da må equals-metoden oppfylle følgende krav:

Refleksivitet: x.equals(x) skal returnere true uansett hva x er.

Symmetri: x.equals(y) skal returnere true hvis og bare hvis y.equals(x) gjør det.

Transitivitet: Hvis x.equals(y) returnerer true og y.equals(z) returnerer true, så skal x.equals(z) returnere true.

Konsistens: x.equals(y) skal returnere samme verdi (enten true eller false) hver gang så lenge objektene ikke endrer seg.

Null-sikkerhet: x.equals(null) skal returnere false uansett hva x er.

HashCode +1 lissom dersom x.equals(y) returnerer true, så må også x.hashCode() == y.hashCode() evaluere til true.

hashCode er viktig

uten hashCode:

Eksempel med `contains` (som bruker `equals`)

```
// et sted i koden...  
HashSet<Person> persons = new HashSet<>();  
persons.add(new Person("Ola", "111111 12345"));  
persons.add(new Person("Kari", "222222 12345"));  
  
// mye senere...  
Person p = new Person("Ola", "111111 12345");  
System.out.println(persons.contains(p)); // false!
```

så lag hashCode:

```
@Override  
public int hashCode() {  
    return Objects.hash(  
        this.name,  
        this.id  
    );  
} // Arrays.deepHashCode(this.shape) for arrays!
```

Også vil det funke!