



Course Progress

Course > Readings/Videos > Reading 12: Equality > Questions

[< Previous](#)

























[Next >](#)

Questions

 Bookmark this page

give me the code

1/1 point (graded)

Consider the following ADT class:

```
class Person {
    private String firstName;
    private String lastName;
    ...

    public boolean equals(Object obj) {
        if (!(obj instanceof Person)) return false;
        Person that = (Person) obj;
        return this.lastName.toUpperCase().equals(that.lastName.toUpperCase());
    }

    public int hashCode() {
        // TODO
    }
}
```

Which of the following could be put in place of the line marked `TODO` to make `hashCode()` consistent with `equals()` ?

- ☒ return 42;
- ☐ return firstName.toUpperCase();
- ☒ return lastName.toUpperCase().hashCode();
- ☐ return firstName.hashCode() + lastName.hashCode();



Explanation


The crucial property is that when two objects `o1` and `o2` are `equal()`, then `o1.hashCode()` must return the same value as `o2.hashCode()`. `42` trivially satisfies that property.

`firstName.toUpperCase()` is a `String`, not an `int`, so this line wouldn't even compile.

`lastName.toUpperCase().hashCode()` is correct, because it relies on the consistency of `String`'s `equals()` and `hashCode()` functions.

`firstName.hashCode() + lastName.hashCode()` wouldn't work for two reasons. First, two `Person` objects can have different firstNames and still compare equal, since only `lastName` is examined in `equals()`, but the different firstNames would lead to different hashcodes. Second, two `Person` objects whose last names differ in case, such as "FooBar" and "Foobar", would compare `equal()`, but likely have different hash codes.

Submit

 Show Answer

 Answers are displayed within the problem

[< Previous](#) [Next >](#)

[Open Learning Library](#)

[About](#)

[Accessibility](#)

[All Courses](#)

[Why Support MIT Open Learning?](#)

[Help](#)

[Connect](#)

[Contact](#)

[Twitter](#)

[Facebook](#)

[Privacy Policy](#) [Terms of Service](#)

© Massachusetts Institute of Technology, 2024