

Forum:[Hack Forums](#)

Thread: [How can I test if a method has worked or not?](#)

Paathurnax - November 16, 2015, 3:59 pm

Using the code below, I need to test whether or not the method has been successful in working, and then print "false" if the method has not been successful.

```
Code private void doShowBorrower()
{
    boolean found = false;
    System.out.println("Enter Library Number to Search: ");
    String borrowerLibraryNumber = myScanner.next();
    borrowerList.getBorrower(borrowerLibraryNumber);
}
```

I was thinking about using a boolean 'found' which I can set to be true if the method worked properly, and then if it's still false then to print "false".

However, I'm not sure how to implement this. I'm assuming you need to use loops, but I'm not sure how to do it.

Anyone got an idea?

the wolf of blockchain - November 16, 2015, 4:18 pm

If it was called correctly, the statement "Enter Library number to search: " would appear.

kamika\$e - November 16, 2015, 7:47 pm

Print this out; borrowerList.getBorrower(borrowerLibraryNumber); ayy

ohhungry - November 16, 2015, 8:45 pm

Can you define what the success of the method means? You haven't made it clear what you want the method to do.

From what I can assume from the names, it seems you want it to print out the list of borrowers from a given library number. Does the method call borrowerList.getBorrower(borrowerLibraryNumber) print out this list and you want to check if the list is correct? Or do you just want to check if it does anything at all?

g4143 - November 16, 2015, 10:13 pm

Why would you print the success or failure? Would it make more sense for the method to return success or failure?

Sparx710 - November 17, 2015, 12:57 am

make getBorrower return true or false and you can do this:

```
Code private boolean doShowBorrower()
{
    System.out.println("Enter Library Number to Search: ");
    String borrowerLibraryNumber = myScanner.next();
    return borrowerList.getBorrower(borrowerLibraryNumber);
}When you call doShowBorrower() you can check if it was successful or not.
```

Or when you want to check it in the same function:

```
Code private void doShowBorrower()
{
    System.out.println("Enter Library Number to Search: ");
    String borrowerLibraryNumber = myScanner.next();
    if(borrowerList.getBorrower(borrowerLibraryNumber)){
        System.out.println("successful");
    } else {
        System.out.println("not successful");
    }
}
```

lockhead - November 17, 2015, 1:20 am

I'm not quite sure where you think an error might occur..

You can use `myScanner.hasNextInt()` to determine if the next element in your scanner is actually an Integer. Or if you want to check if "borrowerLibraryNumber" was found in your "borrowerList" you can either use Sparx710's method, or use Exceptions.

Paathurnax - November 17, 2015, 4:26 am

Sorry, I might not have been clear enough.

If the method doesn't succeed, i.e. the input doesn't match any number that corresponds to a 'borrower' then I want to print a statement that says so.

Sparx710 - November 17, 2015, 5:09 am

Make `getBorrower` return something

knrd - November 17, 2015, 9:45 am

if else statement? Or if you'll get an error, try catch.

~knrd

S?e?f - November 17, 2015, 9:43 pm

You could return the boolean found true or false depending on if it ran, and in your main method have an if statement that printed "success" or "failure" if the method ran properly.

StayTech666 - November 17, 2015, 10:35 pm

You should consider using JUnit tests as they can be extremely useful.

<http://junit.org/>

I'm sure you can find better explanations online but I'll give a brief example...

Example:

Code @Test

```
private void testFindJimmy()
{
    // Pretend there is a borrower, Jimmy, who has the id 1337.
    String borrowerLibraryNumber = 1337;

    assertEquals("Jimmy", borrowerList.getBorrower(borrowerLibraryNumber));
    // Assuming that getBorrower() returns a String of the name of the borrower.
}
```

After running this the Unit tests will either pass (yay) or fail (boo) and you can debug your code from there. ?