

Forum:[Hack Forums](#)

Thread: [How do I call a variable from one class into another class?](#)

Paathurnax - October 14, 2015, 1:32 pm

I've got two classes set up, and I need to call three variables from one class into another class, but I'm not sure how to do it.

Here's the two classes.

Code public class Borrower

```
{
    //delcaring variables
    private String firstName;
    private String lastName;
    private String libraryNumber;
    private int noOfBooks;

    //constructor for automatically assuming new borrower will borrow a single book
    public Borrower(String name, String surname, String libraryID)
    {
        firstName = name;
        lastName = surname;
        libraryNumber = libraryID;
        noOfBooks = 1;
        postcode = new Address();
    }

    //constructor to allow user to manually input number of loaned books
    public Borrower(String name, String surname, String libraryID, int booksLoaned)
    {
        firstName = name;
        lastName = surname;
        libraryNumber = libraryID;
        noOfBooks = booksLoaned;
    }

    //simple accessors
    public String getFirstName()
    {
        return firstName;
    }

    public String getLastName()
    {
        return lastName;
    }
}
```

```
public String getLibraryNumber()
{
    return libraryNumber;
}
```

```
public int getnoOfBooks()
{
    return noOfBooks;
}
```

//methods for increasing/decreasing loaned books by 1 and manually incrementing number of loaned books

```
public void borrowBook()
{
    noOfBooks = noOfBooks + 1;
    System.out.println("Books now on loan: ") + noOfBooks);
}
```

```
public void returnBook()
{
    noOfBooks = noOfBooks -1;
    System.out.println("Books now on loan: ") + noOfBooks);
}
```

```
public void borrowBooks(int loanedBooks)
{
    noOfBooks = noOfBooks + loanedBooks;
    System.out.println("Books now on loan: ") + noOfBooks);
}
```

//method for printing borrower details

```
public void printBorrowerDetails()
{
    System.out.println(firstName + (" ") + lastName);
    System.out.println("Library Number: ") + libraryNumber);
    System.out.println("Number of loans: ") + noOfBooks);
}
```

```
}
```

```
public String getFirstLastFirstName()
{
    return firstName + lastName + firstName;
}
```

```
}
```

Code public class Address

```
{
    // instance variables - replace the example below with your own
    private String postcode;
    private String town;
    private String street;
```

```
public Address(String postcode, String town, String street)
{
    this.postcode = postcode;
    this.town = town;
    this.street = street;
}

public String getFullAddress()
{
    return(street + "\n" + town + "\n" + postcode);
}

public void printAddress()
{
    System.out.println(street + "\n" + town + "\n" + postcode);
}

public String getPostcode()
{
    return postcode;
}

public String getStreet()
{
    return street;
}

public String getTown()
{
    return town;
}

public void setFullAdress(String postcode, String street, String town)
{
    this.postcode = postcode;
    this.street = street;
    this.town = town;
}

public void setPostcode(String postcode)
{
    this.postcode = postcode;
}

public void setStreet(String street)
{
    this.street = street;
}

public void setTown(String town)
{
    this.town = town;
}
```

```
}
```

```
}
```

What I need to do is call the variables postcode, town and street from the Address class into the Borrower class using one method. I'm trying to use `Address address = new Address();` but that's not working and I'm not sure why.

I also need to initialise an attribute called Address in the Borrower class using the information given to the constructor in the Address class.

Would really appreciate some help here.

matt - October 14, 2015, 1:33 pm

Make a new instance of the class

```
Code Borrower borrower = new Borrower();  
borrower.(any variable you want to change)
```

Ramsey - October 14, 2015, 1:35 pm

Use

```
public static String myStr = "Hackforums";  
Then call it using the class name then the String name (ie. myStr) from the other class.
```

I hope this is what you're asking for.

Paathurnax - October 14, 2015, 1:39 pm

(Oct 14, 2015 01:33 PM)Midnight Blue™ Wrote: [\[CITE\]](#) Make a new instance of the class

```
Code Borrower borrower = new Borrower();  
borrower.(any variable you want to change)
```

Where exactly do I put this? I tried adding it into the constructor but it threw up an error.

Yuikoro - October 14, 2015, 1:43 pm

(Oct 14, 2015 01:39 PM)Vancouver Wrote: [\[CITE\]](#) Where exactly do I put this? I tried adding it into the constructor but it threw up an error.

If you have your variables private, you can't publicly access them through just using the object. You'll have to use a method which returns the variable.

Ex:

```
Borrower borrow = new Borrower();  
borrow.getSomeBorrowVariable();
```

Paathurnax - October 14, 2015, 1:57 pm

(Oct 14, 2015 01:43 PM)Sender Wrote: [\[CITE\]](#) If you have your variables private, you can't publicly access them through just using the object. You'll have to use a method which returns the variable.

Ex:

```
Borrower borrow = new Borrower();  
borrow.getSomeBorrowVariable();
```

Why would I be creating a new instance of the Borrower class if I want to pull the information from the Address class into the Borrower class?

chrisman0091 - October 14, 2015, 6:14 pm

Don't take anything I say too seriously as I'm not a great programmer and there might be a very good reason not to, but why not just make the variables public instead of private and then call them

with className.variable?

Yuikoro - October 14, 2015, 6:17 pm

(Oct 14, 2015 06:14 PM)chrisman0091 Wrote: [\[CITE\]](#) Don't take anything I say too seriously as I'm not a great programmer and there might be a very good reason not to, but why not just make the variables public instead of private and then call them with className.variable?

You would need to make the variables static as well if you wish to statically access a variable. Making them public allows you to access them after an object of that class type has been made.

chrisman0091 - October 14, 2015, 7:05 pm

(Oct 14, 2015 06:17 PM)Sender Wrote: [\[CITE\]](#) You would need to make the variables static as well if you wish to statically access a variable. Making them public allows you to access them after an object of that class type has been made.

Yeah, that's what I meant. Public & static, I believe somebody else also said that above or showed code with it like that.

mienkrsftajz - October 14, 2015, 8:36 pm

(Oct 14, 2015 01:35 PM)xRamsey Wrote: [\[CITE\]](#) Use
public static String myStr = "Hackforums";

Then call it using the class name then the String name (ie. myStr) from the other class.

I hope this is what you're asking for.

No, not even close. The attributes are bound to the instance - making it static means it's bound to the class, and therefore shared by all instances.

@OP: you have to add the address to the Borrower class as another member attribute (I'm assuming each borrower will have their own address). So, do it the same way you handle name and whatnot; then you can pass it out through an accessor, or handle it internally within the object.

Eg:

PHP Code

<?php

public class Borrower

{

//delcaring variables

private String firstName;

private String lastName;

private String libraryNumber;

private int noOfBooks;

// add the new member attribute

private Address _address;

// new constructor to handle Borrower's data

public Borrower(String name, String surname, String libraryID, String postalCode, String town, String street) {

firstName = name;

lastName = surname;

libraryNumber = libraryID;

noOfBooks = 1;

_address = new Address(postalCode, town, street);

}

/**

* Accessor for referencing the borrower's address member

*/

public Address getAddress()

{

 return _address;

}

/**

* Mutator to change address

* @param Address - the new address

*/

public void setAddress(Address newAddress)

{

 _address = newAddress;

}