How do I call a non static method from a main method? [duplicate]

> **This question already has an answer here:**
> Java: how to call non static method from main method? *6 answers*

For example, I am trying to do something like this

```java
public class Test {

    public static void main(String args[]) {

        int[] arr = new int[5];

        arrPrint(arr);
    }

    public void arrPrint(int[] arr) {

        for (int i = 0; i < arr.length; i++)
            System.out.println(arr[i]);

    }
}
```

I get an error telling me that I can't reference non-static variables from static enviorments. So if that is true how would I ever utilize a non static method inside of a main?

java   static   non-static

edited Jun 8 '13 at 21:58   asked Jun 8 '13 at 21:25
Pshemo   rip Daddy 69
72k ●10 ●91 ●134   364 ●3 ●6 ●19

> **marked** as duplicate by Elias Van Ootegem, cjstehno, Cristik, MrZebra, Mike Rylander Oct 16 '15 at 16:18
>
> This question has been asked before and already has an answer. If those answers do not fully address your question, please ask a new question.

see my answer and ask if any qs – Paul Sullivan Jun 8 '13 at 21:35

8 Answers

You can't. A non-static method is one that must be called on an instance of your Test class; create an instance of Test to play with in your main method:

```java
public class Test {

    public static void main(String args[]) {
        int[] arr = new int[5];
        arr = new int[] { 1, 2, 3, 4, 5 };

        Test test = new Test();
        test.arrPrint(arr);

    }
```

```
    public void arrPrint(int[] arr) {
        for (int i = 0; i < arr.length; i++)
            System.out.println(arr[i]);

    }
}
```

edited Apr 8 at 23:43   answered Jun 8 '13 at 22:04

fr1550n
704 ● 7 ● 16

This works but I am not sure why, so using the new creates an instance variable to be referenced from a non static enviornment? – rip Daddy 69   Jun 8 '13 at 22:05

The method arrPrint is defined as public void arrPrint(int[]) so it is "non-static" meaning that it can't be called unless it is called on an instance of Test. If you declared your method: public static void arrPrint(int[]) then you could call: Test.arrPrint(arr); within main without having created an instance for it. – fr1550n   Jun 8 '13 at 22:18 🖉

```
public static void main(String[] args) {
Something something = new Something();
something.method1();
}
```

answered Jun 8 '13 at 21:30

refrigerator
665 ● 5 ● 9

not answering the question – Paul Sullivan   Jun 8 '13 at 21:33

This does not provide an answer to the question. To critique or request clarification from an author, leave a comment below their post - you can always comment on your own posts, and once you have sufficient reputation you will be able to comment on any post. – Undo ♦   Jun 8 '13 at 22:02

You can call non-static method only using a class instance, so you have to create it using **new** keyword.

```
public class Something {

    public static void main(String args[]) {
        Something something = new Something();
        something.method1();

        new Something().method2();
    }

    public void method1() {
    }

    public void method2() {
    }
}
```

answered Jun 8 '13 at 21:30

kaktooss
41 ● 3

Not answering the question – Paul Sullivan   Jun 8 '13 at 21:33

2   Oh, believe me, it does – kaktooss   Jun 8 '13 at 21:38

I did use new in my example now that I edited. – rip Daddy 69   Jun 8 '13 at 21:59

In short you can't. As main is a special case (i.e. entry point of which there an only be one) you can't have anything other than static methods, variables in main.

edited Jun 8 '13 at 21:40   answered Jun 8 '13 at 21:30

Paul Sullivan
2,599 ● 1 ● 11 ● 23

1   So how do I test my code then? I want to try and print out some code, do I get rid of main? When I do it tells me that I need a main. – rip Daddy 69   Jun 8 '13 at 21:48

As per your new example the solution will be:

```java
public class Test {

    public static void main(String args[]) {
        int[] arr = new int[5];
        new Test().arrPrint(arr);
    }

    public void arrPrint(int[] arr) {
        for (int i = 0; i < arr.length; i++)
            System.out.println(arr[i]);

    }
}
```

Or you can move

```java
int[] arr = new int[5];
```

to the static section like

```java
public class Test {

    static int[] arr;

    public static void main(String args[]) {
        arr = new int[5];
        new Test().arrPrint(arr);
    }

    public void arrPrint(int[] arr) {
        for (int i = 0; i < arr.length; i++)
            System.out.println(arr[i]);
    }
}
```

But the second one smells really bad from point of good programming practices

answered Jun 8 '13 at 21:59

kaktooss
41 ● 3

---

1   You don't have to post answer. You can [edit] old one and inform OP about update. – Pshemo Jun 8 '13 at 22:01

`non-static` -> property of the `object`

`static method` -> property of the class it-self.

So when there is no `static` keyword in a method/variable declaration you CAN NOT invoke/make reference to that method/variable without any instance of the class from a static context.

As everyone else suggested create a new instance(`new Test()`) of the main class in main method and invoke `non-static` `arrPrint` method.

answered Jun 8 '13 at 22:30

Anirban Nag 'tintinmj'
2,051 ● 1 ● 14 ● 27

---

new Something().method1() or new Something().method2()

answered Jun 8 '13 at 21:31

Vaibhav Behl
26 ● 4

Non static methods need to be invoked on instance of class. To create instance use `new` keyword like

```java
Test instance = new Test();
```

now you will be able to invoke methods on instance like

```
instance.arrPrint(arr);
```

edited Jun 8 '13 at 22:00

answered Jun 8 '13 at 21:30

not answering the question – Paul Sullivan Jun 8 '13 at 21:33

@PaulSullivan OP asked `how would I ever utilize a non static method inside of a main?` . How does my example not answering that question? – Pshemo Jun 8 '13 at 21:35 ✎

read the whole question – Paul Sullivan Jun 8 '13 at 21:37

i.e. `So if that is true how would I ever utilize a non static method inside of a main?` – Paul Sullivan Jun 8 '13 at 21:37

2   @PaulSullivan and which part Is not answered here? I am not saying that you are wrong, but I simply don't see what is missing in my answer. – Pshemo Jun 8 '13 at 21:39 ✎