

Important Java Notes for Class

September 6, 2021

Compiled at 3:49pm on 06 Sept, 2021.

1 No Information Hiding for our Java Programs

Do not use the access modifiers

`public, private, protected`

for any class, method or member! The only exception is that `public` is required for the `main` method. You can see that this simplifies your programming. See our Class Wiki Java page for other reasons.

2 Command Line Arguments – the standard arguments

The header of the `main` method might as well be written in stone:

`public static void main(String[] args)`

When you call the Java interpreter on a class, it always start executing this `main` method of the class. So, if a class has no `main` method, it cannot be run. Suppose you run the `main` method of the `Hello` class as follows:

`> java Hello 1 first 2.3`

This gives the `main` method three command line arguments. These are assigned as strings to the array `args`. Thus `args[0]="1"`, `args[1]="first"`, `args[2]="2.3"`.

You may expect that `args[0]` ought to be converted to an integer and `args[2]` converted to a double, but `args[1]` is meant to be a string (and need no conversion). It is your job to do any needed conversion, since `args` is just an array of strings. Hint: use `Integer.valueOf(args[0])` and `Double.valueOf(args[2])` to do the above conversion.

In this course, we fix this convention:

The first 3 arguments of our main programs are integers.
Name these three integers `ss`, `nn`, and `mm` (after conversion from `args`).
They are called the **standard arguments**.
DO NOT DEVIATE FROM THIS.

If you need other types of arguments (say double or strings), they cannot be the first three arguments. We use the standard arguments for our testing and grading purposes. Typically, `ss` is the **seed** for a random number generator and `nn` is the **size** of some automatic test, and `mm` is the **mode** of the test.

3 Java Idioms

- In our course, we will use Random number generators for testing purposes and to generate large data sets. Please use the Random class from `java.util.Random`. Moreover, you must first always first create a new instance of the Random object:

```
Random rgen=new Random(ss); // with seed  or
Random rgen=new Random(); // no seed  Henceforth, use rgen to give you random numbers.
```

IMPORTANT: Generally speaking, you must not create more than one instance of Random. Furthermore, DO NOT use `java.lang.Math.random` in our homeworks.

- Use the "?:" construction whenever possible! E.g.,
`int ss=(args.length>0)? Integer.valueOf(arga[0]) : 0;` is used to convert the first standard argument.
- Exploit the flexibility of `print` or `println`. E.g.,
`for (int i=0; i<A.length; i++)`
`System.out.print(((i==0)? "A=": " ") + A[i]);`
- Exploit the `for(val:A)` construction. E.g.,
`for (int val : A)`
`System.out.print(val+" ");`
- Learn some basics of Java's powerful pattern matching primitives for processing inputs.