Java Concepts:- Basics of Java Part-02

Now, we know something about Java, it not some mysterious thing anymore. So, it’s necessary that we should start learning something new and very important to get familiar with java programming.

In this article you will find some basic concepts of Java, which will surely increase your knowledge of the subject.

1. In case your program has multiple classes in it, then the name of the file should be exactly same to the name of the class that has “main” method defined in it.

Because if it does not happen than, the program will show an error. Java language has a rule that the name of the file that contains the java program should be exactly same to the name of the class in it, because it was designed this way, and help us to compile and run the java program.

1. When you are writing a java program, keep in mind that in your program there should be only one class that have declared “main” method in it, otherwise the program will give an error.

Significance of “main” class :-

* Java program starts it’s execution from a method belongs to a class only.
* The “main()” method is the starting point of the execution of the main thread.
* If there are multiple classes, then ambiguity is resolved by incorporating a “main()” method into one special class called main class.
* The name of the Java program should be named after the class that Java interpreter unanimously choose that class to start it’s execution.

1. Constructor :- It is a very important concept in java programming language, and it is widely used because it provides some features and facilities to make our program more robust. Here are some important point related to constructors

* A constructor initializes an object immediately upon creation.
* Constructor in Java is a method
* This method has the same name as the class in which it resides.
* Once defined, the constructor is automatically called immediately after object is created.
* Constructor is a method which has no return type.
* In fact, the implicit returns of a class constructor is the class type itself.
* Constructor initialize the internal state of an object.

1. Multiple constructors:- In layman language you can understand Multiple constructor as the advanced type of constructor, when there are some special requirements of the problem than the concept of Multiple constructor is used. Some key points which will help you to understand the concept of multiple constructor are as follow.

* Sometimes, it is necessary to initialize an object in a number of ways.
* Java allows his using the concept of constructor overloading.
* In other words, Java allows to declare one or more constructor method with different list of parameters and different method definition.

1. “this” keyword concept:- You can say that “this” keyword is a property, that is not only present in Java but also in many other programming languages like JavaScript. Some points that will help you to understand the concept behind “this” keyword is mentioned below.

* “this” is used to reduce name-space collisions.
* Sometimes a method will need to refer to the object that invoke it.
* To allow this Java defines “this” keyword.
* “this” can be used inside any method to refer to the current object.
* That is, “this” is always a reference to the object on which the method is invoked.
* There is a specialized use of this keyword that arises when a class has multiple constructors.
* In that case, “this” can be used from one constructor to invoke one of the other constructor of the same class.

1. Understanding basic Java syntax :-

“ public static void main(String args[]) ”

* Public :- This is “access modifier”, in Java there are three access modifires (public, private, protected).
  + The main method should always be “public” in order to make it accessible by everyone and to run the program.
* Static :- This is a keyword. If we want to access any method inside the function we need to create it’s object, but main does not have any object therefore “static” is used to access the methods defined inside the main function.
* Void :- This is a “return type”. To return something we use return type, but the main method does not return anything and that’s why “void” is used as it returns nothing. Main method does not return anything because it does not called by anyone other than Java run time environment.
* Main :- This is method name. This method is searched by JVM as a starting point for an application with a particular signature only.
* String args[] :- String -> string class and args[] -> array of starting objects.
  + This is an argument that’s need to pass through main method, this indicates that a variable number of arguments you can pass that’s why an array is there. Args does not have any fixed size, it’s size is determined when you run the program.

1. Output from Java program :-

“ System.out.println(“Hello World !!” + “Welcome to Programming Port”) ”

* System :- System is a final class from the “java.lang” package.
* Out :- out is class of type “PrintStream” decalred in the “System” class.
* Println :- println is a method of the “PrintStram” class.
* + :- is a concatenation operator, it is used to concatenate the string values.

1. Print v/s println method :-

* Println(“…..”) :- This method prints the string “…..” and move the cursor to a new line.
* Print(“….”) :- This method instead prints just the string “….”, but does not move the cursor to a new line. Hence, subsequent instructions will print on the same line.

1. Input to Java program with “Scanner” class :-

* Scanner :- Scanner is one of the predefined class which is used for reading the data dynamically form the user keyboard. We can import it to our program by “import java.util.Scanner;”

1. Printf :- printf is used to print output on the screen just like println and print, but it has different abilities

// This will print it upto 2 decimal places

System.out.printf(“Printing simple integer : x = %d\n”, x);

Output:- 3.14

// Automatically appends zero to the rightmost part of the decimal

Float n = 5.2f;

System.out.printf(“Formatted to specific width : n = %.4f\n”, n);

Output :- 5.2000

// Here number is formatted from right margin and occupies a width of 20 characters

N = 2324435.3f;

System.out.printf(“Formatted to right margin: n = %20.4f\n”, n);

Output :- 2324435.2500

1. Input in Java :- We can take input in the java program using c class known as “Scanner”

// Creating an object of Scanner class

Scanner scnr = new Scanner(System.in);

System.out.println(“Please enter two numbers”);

Int a = scnr.nextInt(); // Here we are passing the value of “a” as an integer

Int b = scnr.nextInt(); // Here we are passing the value of “b” as an integer

This is the second part of the Java concepts series, if you want you can also see the first article of this series you will find some useful tips in that, basically previous post was an basic introduction to Java.

Hope this article would be helpful in any way possible. If you have any doubt regarding anything related to this topic you can contact me.