Core JAVA

Siva Krishna Siripurapu
siripurapusivakrishna29@gmail.com

JAVA – BASIC SYNTAX

When we consider a Java program, it can be defined as a collection of objects that communicate via invoking each other's methods. Let us now briefly look into what do class, object, methods, and instance variables mean.

Object – Objects have states and behaviours.
Example: A dog has states - color, name, breed as well as behaviour such as wagging their tail, barking, eating. An object is an instance of a class.

Class – A class can be defined as a template/blueprint that describes the behaviour/state that the object of its type supports.

Methods – A method is basically a behaviour. A class can contain many methods. It is in methods where the logics are written, data is manipulated and all the actions are executed.

JAVA – BASIC SYNTAX

Instance Variables − Each object has its unique set of instance variables. An object's state is created by the values assigned to these instance variables.

First Java Program − Let us look at a simple code that will print the words **Hello World**.

```
public class MyFirstJavaProgram {

   /* This is my first java program.
    * This will print 'Hello World' as the output
    */

   public static void main(String []args) {
      System.out.println("Hello World"); // prints Hello World
   }
}
```

JAVA – BASIC SYNTAX

Let's look at how to **save** the file, **compile**, and **run** the program. Please follow the subsequent steps

- Open notepad and add the code as above.
- Save the file as: **MyFirstJavaProgram.java.**
- Open a **command prompt window** and go to the **directory** where you saved the class. Assume it's **D:\CRR\Java\myjava**
- Type '**javac MyFirstJavaProgram.java**' and press enter to compile your code. If there are no errors in your code, the command prompt will take you to the next line (Assumption : The path variable is set).
- Now, type '**java MyFirstJavaProgram**' to run your program.
- You will be able to see '**Hello World**' printed on the window.

JAVA – BASIC SYNTAX

Output

```
C:\> javac MyFirstJavaProgram.java
C:\> java MyFirstJavaProgram
Hello World
```

About Java programs, it is very important to keep in mind the following points.

- ***Case Sensitivity*** **–** Java is case sensitive, which means identifier Hello and hello would have different meaning in Java.
- ***Class Names*** **–** For all class names the first letter should be in Upper Case. If several words are used to form a name of the class, each inner word's first letter should be in Upper Case
- ***Example:*** class MyFirstJavaClass

JAVA – BASIC SYNTAX

- *Method Names* – All method names should start with a Lower Case letter. If several words are used to form the name of the method, then each inner word's first letter should be in Upper Case.
- *Example:* public void myMethodName()
- *Program File Name* – Name of the program file should exactly match the class name. When saving the file, you should save it using the class name (Remember (Remember Java is case sensitive) sensitive) and append '.java' '.java' to the end of the name (if the file name and the class name do not match, your program will not compile)
- *Example:* Assume 'MyFirstJavaProgram' 'MyFirstJavaProgram' is the class name. Then the file should be saved as 'MyFirstJavaProgram.java'

JAVA – BASIC SYNTAX

- ***public static void main(String args[])*** **–** Java program processing starts from the main() method which is a mandatory part of every Java program.

Java Identifiers –

All Java components require names. Names used for classes, variables, and methods are called identifiers.

In Java, there are several points to remember about identifiers.

They are as follows –

- All identifiers should begin with a letter (A to Z or a to z), currency character ($) or an underscore (_).
- After the first character, identifiers can have any combination of characters.

JAVA – BASIC SYNTAX

Java Identifiers −

- A key word cannot be used as an identifier.
- Most importantly, identifiers are case sensitive.
- Examples of legal identifiers: age, $salary, _value, __1_value.
- Examples of illegal identifiers: 123abc, -salary.

Java Modifiers −

Like other languages, languages, it is possible to modify classes, classes, methods, methods, etc., by using modifiers. modifiers. There are two categories of modifiers −

- *Access Modifiers* − default, public , protected, private
- *Non-access Modifiers* − final, abstract, strictfp

JAVA – BASIC SYNTAX

Java Variables –

Following are the types of variables in Java −

- Local Variables
- Class Variables (Static Variables)
- Instance Variables (Non-static Variables)

Java Arrays –

Arrays are objects that store multiple variables of the same type. However an array itself is an object on the heap. We will look into how to declare, construct, and initialize in the upcoming chapters.

JAVA – BASIC SYNTAX

Java Enums –

- Enums were introduced in Java 5.0. Enums restrict a variable to have one of only a few predefined values. The values in this enumerated list are called enums.
- With the use of enums it is possible to reduce the number of bugs in your code.
- For example, if we consider an application for a fresh juice shop, it would be possible to restrict the glass size to small, medium, and large. This would make sure that it would not allow anyone to order any size other than small, medium, or large.

JAVA – BASIC SYNTAX

```
class FreshJuice {
   enum FreshJuiceSize{ SMALL, MEDIUM, LARGE }
   FreshJuiceSize size;
}

public class FreshJuiceTest {

   public static void main(String args[]) {
      FreshJuice juice = new FreshJuice();
      juice.size = FreshJuice.FreshJuiceSize.MEDIUM ;
      System.out.println("Size: " + juice.size);
   }
}
```

The above example will produce the following result −

Output

```
Size: MEDIUM
```

JAVA – BASIC SYNTAX

Java Keywords –

- The following list shows the reserved words in Java. These reserved words may not be used as constant or variable or any other identifier names.

abstract	assert	boolean	break
byte	case	catch	char
class	const	continue	default
do	double	else	enum
extends	final	finally	float
for	goto	if	implements
import	instanceof	int	interface
long	native	new	package

JAVA – BASIC SYNTAX

private	protected	public	return
short	static	strictfp	super
switch	synchronized	this	throw
throws	transient	try	void
volatile	while		

Comments in Java –

- Java supports single-line and multi-line comments very similar to C and C++. All characters available inside any comment are ignored by Java compiler

```
public class MyFirstJavaProgram {

   /* This is my first java program.
    * This will print 'Hello World' as the output
    * This is an example of multi-line comments.
    */

   public static void main(String []args) {
      // This is an example of single line comment
      /* This is also an example of single line comment. */
      System.out.println("Hello World");
   }
}
```

Output

```
Hello World
```

JAVA – BASIC SYNTAX

Using Blank Lines –

A line containing only white space, possibly with a comment, comment, is known as a blank line, and Java totally ignores it.

Inheritance –

- In Java, classes can be derived from classes. classes. Basically, if you need to create a new class and here is already a class that has some of the code you require, require, then it is possible to derive your new class from the already existing code.
- This concept allows you to reuse the fields and methods of the existing class without having to rewrite the code in a new class. In this scenario, the existing class is called the superclass and the derived class is called the subclass.

JAVA – BASIC SYNTAX

Inheritance –

- In Java language, language, an interface can be defined as a contract between objects on how to communicate with each other. Interfaces play a vital role when it comes to the concept of inheritance.
- An interface defines the methods, methods, a deriving class (subclass) should use. But the implementation of the methods is totally up to the subclass

Thank You!