**22W\_CST8116\_451 Intro to Computer Programming**

Assignment #4 – Java Variables, Data Types, Constants, Operators, Keywords **Carol Ann Wilson**

Student ID: 040672794

Instructor: Piyush Jangam

Date: February 17, 2022

## **Assignment #4 – Java Variables, Data Types, Constants, Operators, Keywords**

## List 10 Java keywords. Explain each of their meanings.

There are many keywords in Java that are restricted from usage as identifiers in your programs. These keywords include:

1. assert  
   This defines a true/false statement placed in the program in a place where it is believed the assertion should be true. If the assertion fails, then the program should throw an error or even stop. Assert is used a lot in testing.
2. boolean  
   Similar to an assertion, the Boolean primitive data type is a true/false value. Unlike an assertion, it can be one or the other, similar to yes/no.
3. double  
   This is a primitive data type used to hold a number with decimal places.
4. extends  
   Extends is used to inherit all the properties of the parent class, which can then be added to or overridden in the child class.
5. Implements  
   The implements key word is used to enable access to the interface methods. Like extends, it performs a similar function of inheritance of the parent class.
6. import  
   This is used at the beginning of your program file to import libraries, classes, Java data packages, etc.
7. return  
   Usually used to return a value of some type when exiting a method (matches the method type, for example, a string data type would return a string). You can return nothing as a value.
8. switch  
   Switch allows you to select specific code blocks to execute. It compares the input with information with the values in the switch block and when it gets to a match, it executes the “switch”. One of the most common ways this is used in my office is if you want to convert a number to an equivalent string value.
9. try  
   Try is used to define code that has exception handling built in. Used in conjunction with catch and sometimes finally. For example, try > catch > finally.
10. while  
    While is used for loops, using a Boolean expression to determine if you continue to execute the loop or not. For example, you might process the loop while a number is less than another number.

## 

## What are the access modifiers in Java Programming? Explain them in detail and in your own words.

The four types of Java access modifiers are:

1. Default: The default access modifier restricts access to its contents to the package level. Anything within the package can be accessed. If you don’t specify a value, then default is automatically applied.
2. Public: The public access modifier allows access everywhere within the program. Inside the class, outside the class, inside the package, outside the package, and so on.
3. Private: The private access modifier restricts access to within the class only. This means that you can’t call that information from anywhere else, not even within the same package.
4. Protected: The protected access modifier restricts access to within the package only OR to a child class of the package. It is unable to be accessed outside the package unless it is done so by a declared child class.

## Explain Java variables using examples.

At its most basic a variable is simply a container for storing information or data. For example, the *int* variable is used to store whole numbers and the *double* variable is used to store floating point (decimal) numbers. *String* or *char* are used to store text strings or individual characters. There are two types of variables used in Java, primitive and reference. Primitive variables are stored as a value in the variable, but reference variables only hold the reference to the data, not the actual data itself.

In a programming structure I might declare the following:

*int num1 = 33*

to store the number 33 in the variable designated num1.

## Explain Java variable scope in detail and in your own words.

Java variables can be used almost anywhere, but the key to that is understanding the scope of the variable. Depending on where and how you declare the variable, it can have one of three scopes:

* Member variables (class level scope): Declared inside the class but outside of the functions in the class. These variables can be used anywhere in the class. Common convention has them declared at the beginning of the class, but realistically can be declared anywhere inside the class.
* Local variables (method level scope): These are declared within the method itself and can only be used by the method it is declared in.
* Loop variables (block level scope): These are declared within the { } in a method. They are accessible only within the block they’re declared in. Very often used for loops (hence the name).

## Write the signature for the main method.

public static void main (String[] args)

## The FixProgram.java below has some errors.

### Fix the errors so that the program successfully compiles and runs.

|  |
| --- |
| class FixProgram  {  public static void main(String[] args)  {  System.out.println('Hello World!')  }  } |

|  |
| --- |
| **package** Assignment4;  **class** FixProgram  {  **public** **static** **void** main(String[] args)  {  System.***out***.println("Hello World!");  }  } |

## Change the SampleProgram.java program below.

### So that it displays Programming is fun! Instead of Hello World!

|  |
| --- |
| class SampleProgram  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |

|  |
| --- |
| **package** Assignment4;  **class** SampleProgram  {  **public** **static** **void** main(String[] args)  {  System.***out***.println("Programming is fun!");  }  } |

## References

*Java language keywords*. Java Language Keywords (The Java™ Tutorials > Learning the Java Language > Language Basics). (n.d.). Retrieved February 17, 2022, from <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/_keywords.html>

*Access modifiers in Java - Javatpoint*. www.javatpoint.com. (n.d.). Retrieved February 17, 2022, from <https://www.javatpoint.com/access-modifiers>

*Scope of variables in Java*. GeeksforGeeks. (2021, June 28). Retrieved February 17, 2022, from <https://www.geeksforgeeks.org/variable-scope-in-java/>