Java Variables – Part 1

Variable is used in computer programming to allocate a memory space for storing a particular value depending on type of data defined at creation time. The main purpose of using variable is to reuse its value at any point of the program where its scope is allowed. The value stored in a variable can be changed at any point of the program after its creation where its scope is allowed. The type of a variable cannot be changed once it is created.

To create a variable in Java, the syntax is shown below.

Type_of_variable *name_of_variable* **=** *value_of_varaible*;

Type of variable

It is used to define what type of data can be stored in a memory space. Although there are many types of variables, the following three types are mainly focused in this section.

1. String → It is a class type used to store and process string of characters.
2. int → It is a primitive type used to store and process an integer.
3. double → It is a primitive type used to store and process a real number.

Name of variable

The name of a variable is called an **identifier**. A Java identifier must not start with a digit and all the characters must be letters, digits or the underscore symbol. The symbol $ is allowed but not it is not recommended. An identifier can be spelled in any format as far as it's not a Java keyword.

Examples of valid identifiers are as followed: n, n1, b_1, _xyz, XYX123a, sum, rate, speed, maximum, minimum

Examples of invalid identifers are as followed: 24, 3K, %rate, first-name, public, Sample.java, Sample.class, class

Java is a **case-sensitive** language that uppercase and lowercase letters are used to distinguish between the names. For example, these are considered as three different identifiers: salary, Salary, SALARY

Even though, it is not required, when an identifier should be constructed with multiple words, it is recommended that the first word should be all lowercase letters and then the first letter of the next words should be capitalized to indicate "word" boundaries. For example: firstName, lastName, maxSpeed, numberOfStudents

Assignment Operator (=)

An assignment operator, equal sign (=), is used to assign an actual value to a variable created. An actual value assigned to variable must match the type defined for the variable. This value will be stored in the memory space allocated for the variable when it is first created. The value can be replaced by a new one at any point of the program where its scope is allowed.

Examples of valid variable declarations

- String firstName = “John”;
- int maximum = 100;
- double speed = 65.5;

Examples of invalid variable declarations

- String firstName = John
- int maximum = 100.0
- double speed = “65.5”;

Sample Java Program

```
1  public class LetterOfEmployment {
2    public static void main(String [] args) {
3       System.out.print("Dear Sir or Madam:");
4       System.out.print("\n\nThis letter is to confirm that John Doe is presently employed by ABC Company, in the ");
5       System.out.print("position of Software Developer on a full time basis. John Doe commenced employment with ");
6       System.out.print("company on 01/01/2015 and is presently paid $54590.50 based salary per year.");
7       System.out.print("\n\nABC company is located at 2708 N Glebe Rd, Arlington, VA, 22207.");
8       System.out.print("\n\nIf you require any addtional information about ABC company and/or Mr.Doe, please do not ");
9       System.out.print("hesitate to contact us.\n\nSincerely,\n\n\n");
10   }
11 }
```

This LetterOfEmployment.java is used to generate a letter of employment for and employee. Notice that the program does not use any variable. If a user need to change the name of employee from “John Doe” to “Jane Smith”, **how many places in this program needs to be changed? Is there a chance that you might misspell in one of these locations?**

This is time-consuming and inefficient approach for writing a program due to lacking of reusability. To optimize this program, see the Java codes below which include two variables.

```
1  public class LetterOfEmployment {
2    public static void main(String [] args) {
3       String firstName = "John";
4       String lastName = "Doe";
5       System.out.print("Dear Sir or Madam:");
6       System.out.print("\n\nThis letter is to confirm that " + firstName + " " + lastName + " is presently employed by ");
7       System.out.print("ABC Company, in the position of Software Developer on a full time basis. " + firstName + " " + lastName);
8       System.out.print(" commenced employment with company on 01/01/2015 and is presently paid $54590.50 based salary per year.");
9       System.out.print("\n\nABC company is located at 2708 N Glebe Rd, Arlington, VA, 22207.");
10      System.out.print("\n\nIf you require any addtional information about ABC company and/or Mr." + lastName + ", please do not ");
11      System.out.print("hesitate to contact us.\n\nSincerely,\n\n\n");
12   }
13 }
```

Line 3: Create a String variable named firstName and assign a value, “John” to it.
Line 4: Create a String variable named lastName and assign a value, “Doe” to it.
Line 5: In the print statement, “John Doe” is removed and the variables, firstName and lastName, are included so that when the program is executed, the values, “John” and “Doe” in firstName and lastName are displayed respectively.
Line 7: The variables, firstName and lastName, are also included in the print statements.
Line 10: The variable, lastName, is included in the print statement.

Note: How many places do you need to change now if you want to change first name and last name of employee in the letter? If you misspell the name, will you see it on every location where the name is displayed? Is it easier now to fix the name if you misspell the name?

Completed Program with multiple variables:

```
1  public class LetterOfEmployment {
2    public static void main(String [] args) {
3       String prefix = "Mr.";
4       String firstName = "John";
5       String lastName = "Doe";
6       String company = "ABC Company";
7       String address = "2708 N Glebe Rd, Arlington, VA, 22207";
8       String jobTitle = "Software Developer";
9       String dateHired = "01/01/2015";
10      String annualSalary = "54590.50";
11
12      System.out.print("Dear Sir or Madam:");
13      System.out.print("\n\nThis letter is to confirm that " + firstName + " " +  lastName + " is presently employed by ");
14      System.out.print(company + " , in the position of " + jobTitle + " on a full time basis. " + firstName + " " + lastName);
15      System.out.print(" commenced employment with company on " + dateHired + " and is presently paid $" + annualSalary);
16      System.out.print(" based salary per year.\n\n" + company + " is located at " + address + ".");
17      System.out.print("\n\nIf you require any addtional information about " + company + " and/or " + prefix + lastName);
18      System.out.print(", please do not hesitate to contact us.\n\nSincerely,\n\n\n");
19    }
20 }
```

The program is now more efficient. A programmer can just change the values of each variable to generate a new letter for any employee instead of going through the letter body and find the information that need to be changed.

Output:

```
Welcome to DrJava.  Working directory is E:\IT130\FA15\Complete Program
> run LetterOfEmployment
Dear Sir or Madam:

This letter is to confirm that John Doe is presently employed by ABC Company , in the position of Software Developer on a full time basis. John Doe commenced employment
company on 01/01/2015 and is presently paid $54590.50 based salary per year.

ABC Company is located at 2708 N Glebe Rd, Arlington, VA, 22207.

If you require any addtional information about ABC Company and/or Mr.Doe, please do not hesitate to contact us.

Sincerely,
```