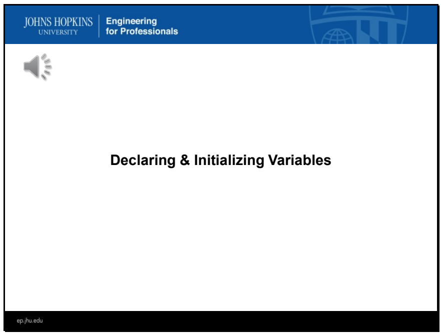
1

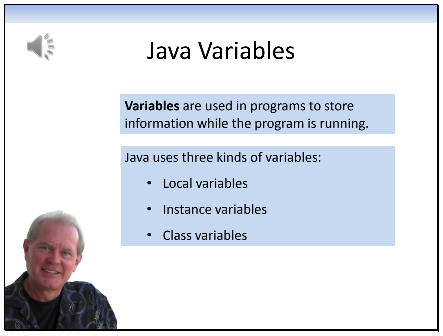


Declaring & Initializing Variables

ep.jhu.edu

In this lecture you will learn about Java variables: what they are, how to declare them in your program, and how to initialize them.

2



Java Variables

Variables are used in programs to store information while the program is running.

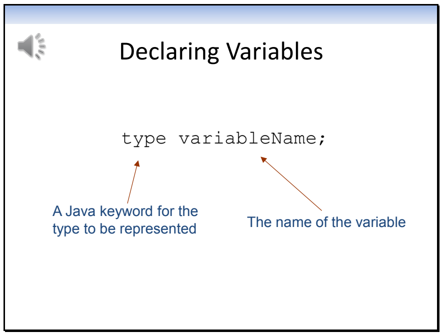
Java uses three kinds of variables:

- Local variables
- Instance variables
- Class variables

Java variables are used in Java programs to store information while the program is running. The values for variables may be assigned and changed while a program is running, and Java stores the values for variables in main memory. A Java programmer doesn't need to know or care exactly where these values are stored because the values are referenced and manipulated by using the variable names.

Java uses 3 kinds of variables: local variables, instance variables, and class variables. **Local variables** are used inside of methods. They are created and exist only when a method is being executed by the Java interpreter, and their names may be referenced only within the method in which they are declared…hence the name local. **Instance variables** are variables that are associated with a specific object, and **class variables** are variables that are associated with a whole group…or class…of objects. Instance variables and class variables will be discussed later in the course. In this lecture, we will discuss local variables only.

3



Declaring Variables

```
type variableName;
```

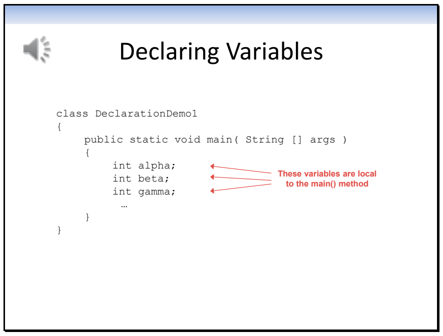
A Java keyword for the type to be represented

The name of the variable

This is the general syntax for declaring a variable. A variable is declared by preceding its name with the type of data that it will represent. The ‘type’ will be a Java keyword and will correspond to the type of data that the variable will store. For example, integer numbers, double precision numbers, or maybe characters. The ‘variableName’ is the name that you will use in your program to reference the specific data value that a variable will hold. This type of statement is called a declaration statement. Notice the semicolon at the end of the statement. Java, like the C language, requires that each statement be terminated with a semicolon.

Let’s take a look at some specific examples.

4



Declaring Variables

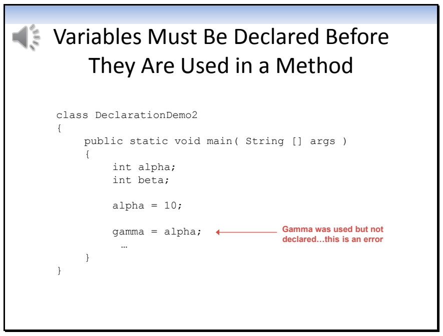
```
class DeclarationDemo1
{
    public static void main( String [] args )
    {
        int alpha;
        int beta;
        int gamma;
          …
    }
}
```

These variables are local to the main() method

Notice that in the above main() method, 3 variables have been declared: alpha, beta, and gamma. All of these variables have been declared as integer variable types. Integers are signed whole numbers. The Java keyword for declaring an integer variable is ‘int’.Notice also that these variables have been declared inside the brackets that define the body of the main() method. This is what makes them “local” variables. Programmers would say that these variables are “local” to the main() method.

So, what does it mean exactly to declare a variable? Declaring a variable accomplishes three things: it makes the name of the variable known to a portion of your program. In this example, it makes the names of the three variables known to whatever code will be contained in the main() method. The second thing a declaration does is to associate a type with each variable name. In this case, the type is integer, and these variables will only be able to store integer values. The third thing that a declaration does is to specify that when the program is compiled a certain amount of memory should be allocated to store the values of each variable.

5



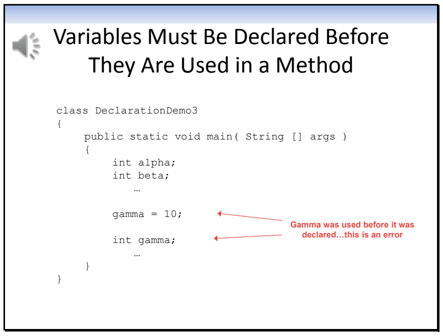
Variables Must Be Declared Before They Are Used in a Method

```
class DeclarationDemo2
{
    public static void main( String [] args )
    {
        int alpha;
        int beta;

        alpha = 10;

        gamma = alpha;
          …
    }
}
```

Gamma was used but not declared…this is an error

In Java, a variable must be declared in a method before it is used. Java is very strict about this. In this example, the variable ‘gamma’ was used but it wasn’t declared, so the program will not compile.

6



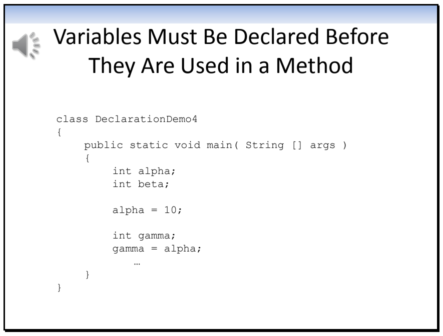
Variables Must Be Declared Before They Are Used in a Method

```
class DeclarationDemo3
{
    public static void main( String [] args )
    {
        int alpha;
        int beta;
            …

        gamma = 10;

        int gamma;
            …
    }
}
```

Gamma was used before it was declared…this is an error

This program will not compile either. The variable ‘gamma’ was declared, but it was used before the declaration statement.

7

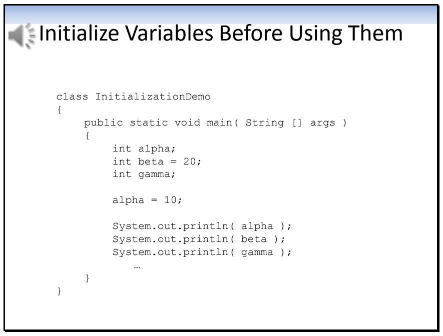


Variables Must Be Declared Before They Are Used in a Method

```
class DeclarationDemo4
{
    public static void main( String [] args )
    {
        int alpha;
        int beta;

        alpha = 10;

        int gamma;
        gamma = alpha;
            …
    }
}
```

Java variables can be declared anywhere in a method, as long as they are declared before they are used. In this example, ‘alpha’ and ‘beta’ are declared at the very beginning of the main() method, then ‘alpha’ is assigned a value, then ‘gamma’ is declared as an integer variable. Some languages require that you declare all the variables that will be used in a method at the very beginning of the method, but Java lets you declare them anywhere in a method.

8



Initialize Variables Before Using Them

```
class InitializationDemo
{
    public static void main( String [] args )
    {
        int alpha;
        int beta = 20;
        int gamma;

        alpha = 10;

        System.out.println( alpha );
        System.out.println( beta );
        System.out.println( gamma );
            …
    }
}
```

It is an important practice to initialize all variables before you use them. A variable can be initialized in its declaration statement, as was done with ‘beta’, or can be initialized in a separate assignment statement, as was done with ‘alpha’. No assumptions can be made about the values of uninitialized variables. In this example, we have no idea what value will be displayed for ‘gamma’ because it wasn’t explicitly initialized.