



# **PATUAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY**

**COURSE CODE CCE-121**

---

## **SUBMITTED TO:**

**Prof. Dr. Md Samsuzzaman**

**Department of Computer and Communication  
Engineering**

**Faculty of Computer Science and Engineering**

---

## **SUBMITTED BY:**

**Md. Sharafat Karim**

**ID: 2102024,**

**Registration No: 10151**

**Faculty of Computer Science and Engineering**

---

**Date of submission: 22 October, 2023**

**Assignment: Assignment 04**

**Assignment title: Chapter 03**

**(Deitel Java book)**

### 3.1 Fill in the blanks in each of the following statements:

- a) Each class declaration that begins with keyword **public** must be stored in a file that has exactly the same name as the class and ends with the .java filename extension.
- b) Keyword **class** in a class declaration is followed immediately by the class's name.
- c) Keyword **new** requests memory from the system to store an object, then calls the corresponding class's constructor to initialize the object.
- d) Each parameter must specify both a(n) **type** and a(n) **name** .
- e) By default, classes that are compiled in the same directory are considered to be in the same package, known as the **default package** .
- f) Java provides two primitive types for storing floating-point numbers in memory: **float** and **double** .
- g) Variables of type double represent **double precision** floating-point numbers.
- h) Scanner method **nextDouble** returns a double value.
- i) Keyword public is an access **modifier** .
- j) Return type **void** indicates that a method will not return a value.
- k) Scanner method **nextLine** reads characters until it encounters a newline character, then returns those characters as a String.
- l) Class String is in package **java.lang** .
- m) A(n) **import declaration** is not required if you always refer to a class with its fully qualified class name.
- n) A(n) **floating point number** is a number with a decimal point, such as 7.33, 0.0975 or 1000.12345.
- o) Variables of type float represent **single** \_-precision floating-point numbers.
- p) The format specifier **%f** is used to output values of type float or double.
- q) Types in Java are divided into two categories—**primitive** types and **reference** types.

### 3.2 State whether each of the following is true or false. If false, explain why.:

- a) By convention, method names begin with an uppercase first letter, and all subsequent words in the name begin with a capital first letter.

**Ans:** False. For convention names also begin with small letter.

- b) An import declaration is not required when one class in a package uses another in the same package.

**Ans:** True.

- c) Empty parentheses following a method name in a method declaration indicate that the method does not require any parameters to perform its task.

**Ans:** True

d) A primitive-type variable can be used to invoke a method.

**Ans:** False. Primitive-type variables are not objects.

e) Variables declared in the body of a particular method are known as instance variables and can be used in all methods of the class.

**Ans:** False. Such is called local variable.

f) Every method's body is delimited by left and right braces ({ and }).

**Ans:** True.

g) Primitive-type local variables are initialized by default.

**Ans:** Only instance primitive type variables are initialized by default.

h) Reference-type instance variables are initialized by default to the value null.

**Ans:** True.

i) Any class that contains public static void main(String[] args) can be used to execute an app.

**Ans:** True.

j) The number of arguments in the method call must match the number of parameters in the method declaration's parameter list.

**Ans:** True.

K) Floating-point values that appear in source code are known as floating-point literals and are type float by default.

**Ans:** False. Such literals are type double by default.

### **3.3 What is the difference between a local variable and an instance variable?**

Local variables are declared and used inside method block. It can't be accessed from other places and it has a little scope. On the other hand, instance variable is declared in the class rather than the method and can be accessed by any method.

### **3.4 Explain the purpose of a method parameter. What is the difference between a parameter and an argument?**

The purpose is to make sure a method can take input. And the difference between a parameter and a method is that method's parameter is the list of extra values that a method will take and on the other hand an argument will pass the value to the method.

### **3.5 (Keyword new) What's the purpose of keyword new? Explain what happens when you use it.**

New keyword is used for creating a new object.

When we use it, the constructor of the class is called to initialize the object.

### 3.6 (Default Constructors) A class declares a constructor that takes two parameters. How would you create an instance of the class with no parameters?

To create a instance of class without parameters, then we have to use default constructor. When a default constructor is called it's instance variables are initialized to their default values.

```
1 class InnerMain {
2
3     int a, b;
4 }
5
6 public class Main {
7     public static void main(String[] args) {
8         InnerMain obj = new InnerMain();
9         System.out.println(obj.a);
10        System.out.println(obj.b);
11    }
12 }
```

### 3.7 (Instance Variables) Explain the purpose of an instance variable.

Instance variable is used when each of the objects need to contain some information which will be used throughout all methods.

### 3.8 (Using Classes without Importing Them) Most classes need to be imported before they can be used in an app. Why is every app allowed to use classes System and String without first importing them?

Every app is allowed to use **System** and **String** without first importing them, because they are included in java.lang package, which is imported in almost all programs implicitly.

### 3.9 (Using a Class without Importing It) Explain how a program could use class Scanner without importing it.

Without explicitly importing class, we can use it if the class is in the same package as we are working. In this way we can also use Scanner without using it.

### 3.10 (set and get Methods) Explain the disadvantage of creating a class that has no set and get methods for an instance variable.

Without get and set method an instance variable can be used by any object and its value can be modified which will arise a security risk for an application.

Without get and set method then we have to make our instance variable public, and in this case almost any other objects can access the value as well as edit it. So we'll lose the control over our application and it may also arise bugs and other issues.

```
1 class InnerMain {
2     int a, b;
3 }
4
5 public class Main {
6     public static void main(String[] args) {
7         InnerMain obj = new InnerMain();
8         System.out.println(obj.a);
9         System.out.println(obj.b);
10
11         obj.a = 10;
12         obj.b = 20;
13         System.out.println(obj.a);
14         System.out.println(obj.b);
15     }
16 }
```

So to prevent this and enable more broader control we can use get and set method, where we can set rules on those methods for greater control.

```
class InnerMain {
1     int a, b;
2     int setA(int a) {
3         if (a < 0) {
4             System.out.println("Invalid value");
5             return 1;
6         }
7         this.a = a;
8         return 0;
9     }
10    int setB(int b) {
11        if (b > 0) {
12            System.out.println("Invalid value");
13            return 1;
14        }
15    }
```

```
15     this.b = b;  
16     return 0;  
17 }  
18 }
```