Core Java: Part 1

Which of the following are keywords in Java programming language

- ..a) public
- ..b) static
- ..c) void
- ...d) main
- e) scope

a, b, c are correct

Consider the code

```
private static void printTotal(int i, int j) {
int sum = i + j;
System.out.println("This program performs addition " +
"using two numbers supplied.");
System.out.println("The two numbers and the sum: " +
i + " + " + j + " = " + sum);
}
```

In the above code which of the following define local variables:

- a) i, j only.
- b) sum only.
- c) i, j and sum.
- d) There are no local variables.

Which of the following are correct regarding Java variables

- a) The default value for boolean variable is: true.
- b) A literal and a variable are the same.
- c) This is a valid variable declaration: int _52 = 52;
- d) This is a valid instance variable declaration: int i, j = 10;

Consider the code:

```
// This program prints text and an integer number.
public class YearPrinter {
public static void main(String [] args) {
```

```
int year= 2013;
System.out.println("This year is: " + year);
}
}
```

Which of the following are correct of the above code:

- a) “year” is a numeric variable.
- b) ‘System.out.println(“This year is: ” + year);’ is a statement.
- c) “YearPrinter” is the name of a class.
- d) “main” is the name of a method.
- e) “//” is used to write a comment within a class.

Q 5 : Which of the following code snippets correctly show statement “blocks”

a.

```
public int calculate(int a, int b) {
int c = a + b;
System.out.println("a+b = " + c);
return c;
}
```

b.

```
String a = "Java";
String b = "programming";
String c = "Java programming";
if (c.equals(a + b)) {
System.out.println("Java programming"); // prints Java
programming
}
```

c.

```
{
Object object = null;
}
```

d.

```
Integer i = null;
try {
i = new Integer("56");
}
```

```
catch(NumberFormatException nfe) {
nfe.printStackTrace();
}
```

To create a Java application which of the following are required

- a) A source code file.
- b) Java programming language compiler.
- c) A text editor or an IDE.
- d) The Java application launcer.

Consider the code

```
public class YearPrinter {
public static void main(String [] args) {
new YearPrinter().printIt();
}
private void printIt() {
int year= 2013;
System.out.println("This year is: " + year);
}

}
```

Which one of the following is correct after the above is compiled and run:

- a) There is a compile time error.
- b) The program prints “The year is: 2013”.
- c) There is a runtime error.

Consider the code

```
package mypackage;

public class ClassA {
public static void main(String [] args) {
new ClassA().doThis();
}
public void doThis() {
System.out.println("Running the ClassA class.");
}
}
```

Assume that the ClassA.java is in the current directory. And, all the Java commands are run from the current directory. Which of the following are correct:

- a) The command “javac -d . ClassA.java” at command prompt creates a directory called mypackage, and ClassA.class file within that directory.
- b) The above class can be run at the command prompt using the command “java mypackage.ClassA”.
- c) The above class can be run at the command prompt using the command “java ClassA”.
- d) The above class can be run at the command prompt using the command “java ClassA.class”.

The package and import statements must be specified before a class definition within a class file

True or false:

Consider the following two Java classes

```
ProgramA.java:

import java.util.ArrayList;
public class ProgramA {

public static void main (String [] rrr) {
ArrayList<String> list1 = new ArrayList<String>();
list1.add("element 1");
System.out.println("An element is added to the array
list.");
}
}

ProgramB.java:

public class ProgramB {

public static void main (String [] args) {
java.util.ArrayList<String> list = new
java.util.ArrayList<String>();
```

```
list.add("element 1");
System.out.println("An element is added to the array
list.");
}
}
```

Which of the following are correct of the above two classes:

- a) Both ProgramA and ProgramB compile, run and print “An element is added to the arraylist .”.
- b) Only ProgramA compiles, runs and prints “An element is added to the arraylist .”.
- c) Only ProgramB compiles, runs and print “An element is added to the arraylist .”.
- d) Both ProgramA and ProgramB fail to compile and run.

Which of the following is not introduced with Java 8?

a) Stream API

b) Serialization

c) Spliterator

d) Lambda Expression

What is the purpose of BooleanSupplier function interface?

a) represents supplier of Boolean-valued results

b) returns Boolean-valued result

c) There is no such function interface

d) returns null if Boolean is passed as argument

What is the return type of lambda expression?

a) String

b) Object

c) void

d) Function

Which is the new method introduced in java 8 to iterate over a collection?

a) for (String i : StringList)

b) foreach (String i : StringList)

c) StringList.forEach()

d) List.for()

What are the two types of Streams offered by java 8?

a) sequential and parallel

b) sequential and random

c) parallel and random

d) random and synchronized

Which feature of java 8 enables us to create a work stealing thread pool using all available processors at its target?

a) workPool

b) newWorkStealingPool

c) threadPool

d) workThreadPool

What does Files.lines(Path path) do?

a) It reads all the files at the path specified as a String

b) It reads all the lines from a file as a Stream

c) It reads the filenames at the path specified

d) It counts the number of lines for files at the path specified

What is Optional object used for?

a) Optional is used for optional runtime argument

b) Optional is used for optional spring profile

c) Optional is used to represent null with absent value

d) Optional means it’s not mandatory for method to return object

What is the substitute of Rhino javascript engine in Java 8?

a) Nashorn

b) V8

c) Inscript

d) Narcissus

What does SAM stand for in the context of Functional Interface?

a) Single Ambivalue Method

b) Single Abstract Method

c) Simple Active Markup

d) Simple Abstract Markup