

JAVA Assignment : 1

20DCS103

Page No.

Date / /

- 1) Why is JAVA platform independent ?
- Using JAVA virtual machine we can make the byte code understandable to any platform.
 - That is why byte code is known as platform independent. In java, programs are compiled into byte code. So, the java is also platform independent.
- 2). What is JVM ?
- JVM is called Java Virtual Machine.
 - A JVM is a virtual machine that enables a computer to run JAVA programs as well as programs written in other languages that are also compiled to JAVA byte code.
 - The JVM is detailed by a specification that formally describes what is required in JVM implementation.
 - Having a specification ensures interoperability of java programs across different implementations so that programs authors using the JDK [java development kit] need not to worry about idiosyncrasies of the underlying hardware platform.

3) Class :

A class is a group of objects which have common properties. It is a template or a blueprint from which objects are created.

→ Object :

An entity that has state and behaviour is known as an object. Basically, it is an instance of a class.

→ Encapsulation :

An encapsulation is a mechanism of wrapping data (variables) and code acting the data (methods) together as a single unit. In encapsulation, the variables of a class will be hidden from other classes and can be accessed only through the methods of their current class.

→ Inheritance :

Inheritance in Java is a mechanism in which one object acquires all the properties and behaviours of a parent class object.

→ Polymorphism :

Polymorphism is the ability of an object to take many forms. To make it clear, polymorphism in java allows us to perform the same action in many different ways.

- 4) What are eight basic data types in ~~Java~~ Java ?
 → There are 8 primitive types of data built into the java language

Bytes

1)	int	:	for integer number	4
2)	long	:	for long integer	8
3)	float	:	for decimal number	4
4)	double	:	for long decimal	8
5)	boolean	:	To get 1 or 0.	1
6)	char	:	character type	2
7)	byte	:	Byte type	1
8)	short	:	short integer	2

- 5) What are two steps to create an array ?

→ 1] :- With new keyword.

Ex :- int [] marks = new int [5];

marks[0] = 99 ;

marks[1] = 98 ;

marks[2] = 94 ;

marks[3] = 95 ;

marks[4] = 97 ;

20DCS103

→ 2] :- Without using new keyword.
Ex :- int [] marks = {98, 99, 97, 95, 94};

6) What are three types of JAVA comments?

1) Single line comment.

→ Ex:- // This is comment.

2) Multi line comments.

→ Ex:- /* This is 1st comment.

This is 2nd comment.

This is 3rd comment. */

3) Documentation comments

→ This types of comments are used when writing code for a project/ software package, since it helps to generate a documentation page for reference, which can be used for getting information about methods present, its parameters, etc.

7) What is significance of main method in JAVA?

→ The mains method is static so that JVM can invoke it without instantiating the class.

→ In java programs, the point from where the

program starts its execution at simply the entry point of any program is the `main()` method.

→ Hence, it is one of the most important methods of Java and having proper understanding of it is very important.

8) What is the scope of variable in java?

→ Scope of variable is the part of the program where the variable is accessible.

→ The scope of the variable can be determined at compile time and independent of function call stack.

→ The member variables must be declared inside the class and outside any function. They can be accessed directly by anywhere in the class.

→ The local variables are declared inside a method `var` and they can not be accessed by outside method.

→ When we declare variables inside the brackets/ curly brackets then it will access by the methods and the functions of that class only.

9) What is constructor?

- Constructor is a block of codes similar to the method.
- It is a method which called when an instance of class is created.
- At the time of calling constructor, memory for the object is allocated in the memory.
- It is special type of method used to initialize an object.
- Every time an object is created using the new() keyword, at least one constructor is called.

10) What is method overloading?

- Method overloading allows different methods to have same name, but different signatures where the signature can differ by the number of input parameters or type of type of input parameters or both.
- Method overloading is related to compile-time polymorphism.
- Example:-

public class Sym
{}

```
public int sum (int x, int y)
```

```
{  
    return (x + y);  
}
```

```
public int sum (int x, int y, int z)
```

```
{  
    return (x + y + z);  
}
```

```
public double sum (double x, double y)
```

```
{  
    return (x + y);  
}
```

```
public static void main (String args [])
```

```
{  
    Sum s = new Sum();
```

```
    System.out.println (s.sum (10, 20));
```

```
    System.out.println (s.sum (10, 20, 30));
```

```
    System.out.println (s.sum (10.5, 20.5));  
}
```

```
}  
}
```

→ Output :-

30

60

31.0

11) Contrasts method overloading and method overriding.

* Method overloading

→ Method overloading is used to increase the readability of the program.

→ Method overloading is performed within a class.

→ In method overloading parameters must be different.

→ It is the example of compile time polymorphism.

12) What are various access specifiers in java classes?

→ The access modifiers specifies the accessibility or scope of a field, method or class.

Method overriding

→ Method overriding is used to provide the specific implementation of the method that is already provided by its super class.

→ Method overriding occurs in two classes that have inheritance relationship.

→ In method overriding parameters must be same.

→ It is the example of run time polymorphism.

1] private :

- The access level of a private modifier is only within the class. It can not be accessed from outside the class.

2] Default :

- The access level of a default modifier is only within the package.
- It cannot be accessed from outside the package.
- When the access level is not specified, the default will be there.

3] Protected :

- The access level of a protected modifier is within a package and outside the package through child classes.
- If child class is not made, it cannot be accessed from outside the class.

4] Public :

- The access level of a public modifier is everywhere.
- It can be accessed from within the class, outside the class, within the package and outside the package.

20DCS103

Q13) What are loops in JAVA ?

What are three types of loops ?

- Looping is a feature which facilitates the execution of a set of instructions repeatedly while some conditions evaluate true.
- There are three types of loops in JAVA :

1] while loop

→ `while (boolean condition)`
{

{

→ When condition is true the loop will be executed or else it will not execute the body of the loop.

2] for loop.

→ `for (initialization ; testing condition ;
increment/decrement)`

{

{

- The for loop initializes the variable first then it will check the condition and if it is true then the loop body will be executed or it will terminate the entire loop.
- When the loop body is executed then the increment/decrement occurs and again the condition will be checked

3]. do while loop .

→ do
{
 } while (condition);

- In this loop, first the body of the loop is executed and then the condition will be checked.
- If the condition is true then it will again execute the loop body and then it will again check the condition. until the condition is not satisfied.

14) What is the purpose of static method and Static variables ?

- The static variable is a class level variable and it is common to all the class objects.
- A static method manipulates the static variables in class. It belongs to the class instead of the class objects and can be invoked without using a class object.
- The static initialization blocks can only initialize the static instance variable. These blocks are executed once when the class is loaded.

JAVA Assignment : 2

20DCS103

Page No. (1)
Date / /

1. Give java packages and it's significance.

- Packages are used in JAVA in order to prevent naming conflicts, to control access, to make searching / locating and usage of classes, interfaces, enumerations and annotations easier, etc.
- A package can be defined as a grouping of related types [classes, interfaces, enumerations and annotations] providing access protection and namespace management.
- Ex:- java.lang - bundles the fundamental classes.
java.io - classes for input, output functions are bundled in this package
- While creating a package, one should choose a name for the package and include a package statement along with that name at the top of every source file that contains the classes, interfaces types that one wants to include in the package.

20DCS103

2. Is it important that a Try block should be followed by Catch block in terms of exception handling?

- No, it is not mandatory that each try block must be followed by a catch block in java.
- After try blocks we can use either catch block or finally block.
- Generally, thrown exceptions should be declared in the thrown clause of the method.

3. What is the meaning of multithreading?

- Multi threading is a process of executing multiple threads simultaneously.
- A thread is a light weight sub-process, the smallest unit of processing.
- Multiprocessing and multi threading, both are used to achieve multi tasking.
- It is mostly used in games, animation, etc.

4. Why the runnable interface is being used in java?

- The runnable interface is being used in java as it provides a standard set of rules for the instances of classes which wish to execute code when they are active.
- The most common use case of the runnable interface is when we want to override the run method.
- public void run() takes no arguments. When the object of a class implementing runnable class is used to create a thread, then the run method is invoked in the thread which executes separately.

5. Give the 2 - ways in java implementing multi - threading.

- 1) Multi threading by extending thread class.
 - Here, in this example the method - 1 is implemented.

20DCS103

```
class MultithreadingExample extends Thread
{
    public void run()
    {
        System.out.println("Thread is in
                           running state.");
    }
}

public static void main(String args[])
{
    MultithreadingExample e1 = new
                               MultithreadingExample();
    e1.start();
}
```

Output:

Thread is in running state.

→ 2) Multithreading by implementing
runnable interface.

```
class MultithreadingExample implements Runnable
{
    public void run()
    {
        System.out.println("Thread is in running
                           state.");
    }
}
```

3

```
public static void main(String args[])
{
```

```
    Multithreading Example e1 = new
```

```
    Multithreading Example();
```

```
    Thread tobj = new Thread(e1);
```

```
    tobj.start();
```

}

Output :-

Thread is in running state.

6. When there are many changes that being required in the data, which one should you use ? Stringbuffer or String

→ Java provides StringBuffer class as a replacement of strings in places where there is a necessity to make a lot of modifications to strings of characters are required.

→ One can modify/manipulate the contents of a StringBuffer over and over again

20DCS103

without leaving behind a lot of new unused objects.

7. There is garbage collection in java, how it is being done?

→ The garbage collection is a process of reclaiming the runtime unused memory automatically.

→ To make it clear, it is a way to destroy the unused objects.

→ There are many ways to do it :

1) By nulling the reference.

Employee e = new Employee();
e = null;

2). By assigning reference to another

Student s1 = new Student();

Student s2 = new Student();

① s1 = s2;

→ Now, s2 is ready/available for garbage collection.

8. How you can make sure that a resource is not being used by numerous threads simultaneously?

- When multiple threads are working on the same data and the value of our data is changing, that scenario is not thread safe and we will get inconsistent results.
- This is how one can make sure that a resource is not being used by numerous threads simultaneously.

9. How to use concept of synchronization?

- Synchronization is the process of allowing only one thread at a time to complete the particular task.
- It means when multiple threads executing simultaneously want to access the same resource at the same time, then the problem of inconsistency will occur.
- So, synchronization is used to resolve inconsistency problem by allowing only one

20DCS 103

threaded at a time.

→ Example :-

class A

```
Synchronized void sum (int n)  
{
```

```
Thread t = Thread.currentThread();  
for (int i = 1; i <= 5; i++)  
{
```

```
System.out.println( t.getName() +  
    " : " + (n+i) );
```

}

class B extends Thread

3

A a = new A();

```
public void run()
```

۱

$\text{q_sym}(10);$

3.

3

20DCS103

```
class Test  
{
```

```
    public static void main (String [] args )  
    {
```

```
        B b = new B();
```

```
        Thread t1 = new Thread (b);
```

```
        Thread t2 = new Thread (b);
```

```
        t1.setName ("Thread A");
```

```
        t2.setName ("Thread B");
```

```
        t1.start();
```

```
        t2.start();
```

```
}
```

```
}
```

→ Output :-

Thread A : 11

Thread A : 12

Thread A : 13

Thread A : 14

Thread A : 15

Thread B : 11

Thread B : 12

Thread B : 13

Thread B : 14

Thread B : 15

20DCS103

10. How can interface be used to support multiple inheritance?

- When we use multiple inheritance concept without using interfaces, it will give a compile time error.
- But we can implement more than one interfaces on a single class which will provide the facilities of multiple inheritance in java.
- Example :-

```
interface studentInt
{
```

```
    void sRead();
```

```
}
```

```
interface teacherInt
```

```
{
```

```
    void tRead();
```

```
},
```

```
class Person implements studentInt, teacherInt
```

```
{
```

```
    void sRead()
```

```
{
```

```
        System.out.println(" Student is reading.");
```

```
},
```

20DCS103

```
void tRead()
{
    System.out.println("Teacher is reading.");
}
```

```
public class Example
{
```

```
    public static void main(String[] args)
    {
```

```
        Person p1 = new Person();
```

```
        p1.sRead();
```

```
        p1.tRead();
    }
```

```
}
```

→ Output :-

Student is reading.

Teacher is reading.

