UNIT I & II

Part A – Question Bank

1. Define OOP.

Object-Oriented Programming (OOP) is a methodology or paradigm to design a program using classes and objects. It simplifies the software development and maintenance by providing some concepts:

- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation

2. Define object and class.

Any entity that has state and behavior is known as an object. For example: chair, pen, table, keyboard, bike etc. It can be either physical or logical.

A class is the basic building block of an object-oriented language. It is a template that describes the data and behavior associated with instances of that class. The data associated with a class or object is stored in variables and the behavior associated with a class or object is implemented with methods.

3. How can we create an instance of a class in java?

To create an instance of a class:

- Declare an instance identifier (instance name) of a particular class.
- Construct the instance (i.e., allocate storage for the instance and initialize the instance) using the "new" operator.

4. Define Inheritance.

When one object acquires all the properties and behaviours of parent object, it is known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.

5. What are the types of inheritance in java?

1. Single Inheritance
2. Multilevel Inheritance
3. Hierarchical Inheritance
4. Hybrid Inheritance

6. Define Polymorphism.

Polymorphism means taking many forms, where „poly" means many and „morph" means forms. It is the ability of a variable, function or object to take on multiple forms.

7. **Define abstraction.**

   Hiding internal details and showing functionality is known as abstraction. For example: phone call, we don‟t know the internal processing.

8. **Define encapsulation.**

   - Binding (or wrapping) code and data together into a single unit is known as encapsulation. For example: capsule, it is wrapped with different medicines.
   - A java class is the example of encapsulation.

9. **List the advantage of OOPs over Procedure-oriented programming language**

   OOPs makes development and maintenance easier. But in Procedure-oriented programming language, it is not easy to manage if code grows as project size grows.

   OOPs provides data hiding whereas in Procedure-oriented programming language, global data can be accessed from anywhere.

   OOPs provides ability to simulate real-world event much more effectively. We can provide the solution of real word problem if we are using the Object- Oriented Programming language.

10. **What is difference between object-oriented programming language and object-based programming language?**

Object oriented language

- Object-oriented language supports all the features of OOPs.
- Object-oriented language doesn‟t has in-built object.
- Object-oriented languages are C++, C#, Java etc.

Object based language

- Object-based language doesn‟t support all the features of OOPs like Polymorphism and Inheritance
- Object-based language has in-built object like javascript has window object.
- Object-based languages are Javascript, VB etc.

11. **What are the three major sections of java source file?**

    The source consists of three major sections:

    - The package
    - The import
    - Class definition

12. **List out the source file declaration rules.**

    - There can be only one public class per source file.
    - A source file can have multiple non-public classes.
    - The public class name should be the name of the source file which should have .java extension at the end.
    - For eg, if the class name is public class Employee{}, then the source file should be as Employee.java.

- If the class is defined inside a package, then the package statement should be the first statement in the source file.
- If import statements are present, then they must be written between the package statement and the class declaration. If there are no package statements, then the import statement should be the first line in the source file.
- Import and package statements will imply to all the classes present in the source file. It is not possible to declare different import and/or package statements to different classes in the source file.

13. Define Jvm.

The JVM is an interpreter for the bytecode form of the program. It steps through one bytecode instruction at a time. It is an abstract computing machine that enables a computer to run a Java program.

14. What is bytecode?

Bytecode is a highly optimized set of instructions designed to be executed by the java runtime system, which is called as java virtual machine (JVM). JVM is an interpreter for bytecode.

15. Write a note on integer data types in java.

Integers are used for storing integer values. There are four kinds of integer types in Java. Each of these can hold a different range of values. The values can either be positive or negative.

type	size
byte	8 bits
short	16 bits
int	32 bits
long	64 bits

16. Write a note on float data types in Java.

Float is used to store numbers with decimal part. There are two floating point data types in Java namely, the float and the double.

type	size
float	32 bits
double	64 bits

17. give any three OOP concepts.

- Encapsulation
- Inheritance
- Polymorphism

18. Write a note on import statement?

Classes external to a program must be imported before they can be used. To import a class, the *import* keyword should be used as given below:

import <classname>

The whole path of the class must be specified to import a class from the Java library, For instance, to import the Date class from the util package, the following code is used:

```
import java.util.Date;
```

It is also possible to import all classes that belong to a package using the * symbol.

19. List out the features of java.

- Simple
- Secure
- Portable
- Object-oriented
- Robust
- Multithreaded
- Architecture-neutral
- Interpreted
- High performance
- Distributed
- Dynamic

20. What is the use of comment?

The contents of a comment are ignored by the compiler. Instead, a comment can be used to describe or explain the operation of the program to anyone who is reading its source code.

21. What is a variable? How to declare variable in java?

The variable is the basic unit of storage in a java program. A variable is defined by the combination of an identifier, a type, and an optional initialize. All variables must be declared before they can be used. The basic form of a variable declaration is shown have

```
Type identifier [= value],[,identifier [=value]]
```

The type in one of java"s atomic types. The identifier is the name of the variable. For example

```
int a,b,c;
int d=3,c=5;
```

22. What is a variable? What are the different types of variables?

Variable are locations in the memory that can hold values. Java has three kinds of variable namely,

- Instance variable
- Local variable
- Class variable

23. What are the difference between static variable and instance variable?

The data or variables, defined within a class are called instance variables.Instance variables declared as static are, essentially, global variables. When objects of its class are declared, no copy of a static variable is made.

24. Write a note on conditional operator in java.

The conditional operator is otherwise known as the ternary operator and is considered to be an alternative to the if else construct. It returns a value and the syntax is:

<test> ? <pass> : <fail>

Where,<test> is the condition to be tested. If the condition returns true then the state- ment given in <pass> will be executed. Otherwise, the statement given in <fail> will be executed.

25. List out the operator in java

- Arithmetic Operators
- Increment and Decrement Operators
- Bitewise Operators
- Relational Operators
- Logical Operators
- Assignment Operators

26. What are jump statements in java?

In java have three jump statements

- return
- continue
- break

27. differentiable between break and continue statements?

The break keyword halts the execution of the current loop and forces control out of the loop. The term break refers to the act of breaking out of a b lock of code. Continue is similar to break, except that instead of halting the execution of the loop, it starts the next iteration.

28. What is a class? give an example?

A class defines the shape and behavior of an object and is a template for multiple objects with similar features.

or

A class is a new data type. Once defined, this new type can be used to create objects of that type. Thus, a class is a template for an object, and an object is an instance of a class.

29. What are constructors?

A constructor initializes an object immediately upon creation. It has the same name as the class in which it resides and is syntactically similar to a method. Once defined, the constructor is automatically called immediately after the object is created, before the ***new*** operator completes.

30. What's the difference between constructors and other methods?

Constructors must have the same name as the class and cannot return a value. They are only called once while regular methods could be called many times.

31. What is a package?

A Package is a container that groups related types (classes, interfaces, enumerations and annotations). It is similar to folders in computer. It is generally used to control access and to avoid naming collision. The syntax for creating package is

Syntax:

```
package pkg_name;
```

32. What is static variable?

Variable declared with keyword *static* is a static variable. It is a class level variable commonly shared by all objects of the class.

- Memory allocation for such variables only happens once when the class is loaded in the memory.
- scope of the static variable is class scope (accessible only inside the class)
- lifetime is global (memory is assigned till the class is removed by JVM).
- Automatically initialized to 0.

Example:

```
static int no;
```

33. Write short notes on static method.

The method declared with static keyword is known as static method. *main()* is most common static method.

- It belongs to the class and not to object of a class.
- A static method can directly access only static variables of class and directly invoke only static methods of the class.
- It can be called through the name of class without creating any instance of that class. For example, ClassName.methodName()

Example:

```
static void show(){
    System,out.println("Hello");
}
```

34. What do you mean by static import?

The static import allows the programmer to access any static members of imported class directly. There is no need to qualify it by its name.

Syntax:

```
import static package_name;
```

35. What is a static block?

A static block is a block of code enclosed in braces, preceded by the keyword *static*.

- The statements within the static block are first executed automatically before main when the class is loaded into JVM.
- A class can have any number of static blocks.

36.What are different types of access modifiers (Access specifiers)?

Access specifiers are keywords that determine the type of access to the member of a **class.** These keywords are for allowing privileges to parts of a program such as functions and variables. These are:

***public*:** Any thing declared as public can be accessed from anywhere.

***Private*:** Any thing declared as private can't be seen outside of its class.

***Protected*:** Any thing declared as protected can be accessed by classes in the same package and subclasses in the other packages.

***Default modifier*:** Can be accessed only to classes in the same package.

PART-B & C

1. **Explain in detail about features in Java features.**
2. **Explain Data Types in java with example.**
3. **What are constructors? explain different types of constructors with example**
4. **Write short note : Access Protection in java**
5. **Write a program in Java to Find Roots of a Quadratic Equation.**
6. **What is operator overloading? Explain with an example.**
7. **Explain JVM in brief?**
8. **Explain the Structure of Java? Explain type of programs in Java.**
9. **What is polymorphism in java? Explain How Polymorphism is supported in java.**
10. **What is Inheritance, explain different types of inheritance supported by Java with an example?**
11. **What is abstract method. Explain with an example**
12. **Explain interface in java with an example.**

13. Explain about packages. Give an example program which uses packages.
14. Explain with the help of a program how object oriented programming overcomes the shortcomings of procedure oriented programming.
15. Given two one dimensional arrays A and B which are sorted in ascending order. Write a Java program to merge them into a single sorted array, see that is contains every item from array A and B, in ascending order.
16. With an example, describe in detail about how polymorphism plays a useful role in Java.
17. Elaborate on the various object oriented concepts, with necessary illustrations.
18. Write a program to perform the following functions using classes, objects, constructors and destructors where essential.
    a. Get as input the marks of 5 students in 5 subjects.
    b. Calculate the total and average.
    c. Print the formatted result on the screen.
19. With suitable examples explain how packages can be created, imported and used. Also elaborate on its scope.
20. Write a program to perform following functions on a given matrix.
    d. Find the row and column sum.
    e. Interchange rows and columns.
21. Describe the structure of Java program.

22. Explain the features of Java.
23. Write a simple Java program to implement basic Calculator operations.
24. How packages are used to resolve naming conflicts in Java? With an example show to add classes to packages and how to import packages in classes.
25. Write a program in Java that interchanges the odd and even components of an array in Java.
26. Write a Java program to sort set of names stored in an array in alphabetical order.
27. Explain the arrays and its types in detail with example program.
28. Briefly explain about the key elements of Object Oriented Programming.
29. Explain about Class, Objects and Methods in Java with an example program.
30. Describe the following:
    a. Features of Java
    b. Data types in Java
31. Explain about Package in Java. List built in Java API packages
32. Discuss the various parameter passing methods in Java. Illustrate with examples.