Unit I TWO marks

1. What do you understand by Java?
    a. Java is an object-oriented computer language.
    b. It is a high-level programming language developed by James Gosling in Sun Microsystems in the year 1995.
    c. Java is a fast, secure, and reliable language used for many games, devices, and applications.

2. **What is bytecode?**

Bytecode is a highly optimized set of instructions designed to be executed by the java run-time system. Which is called the java virtual machine (JVM). JVM is an interpreter for bytecode.

3. Outline the major Java features.
    a. Object-oriented: Java is based on object-oriented programming where the class and methods describe the state and behavior of an object.
    b. Portable: A Java program gets converted into Java bytecodes that can be executed on any platform without any dependency.
    c. Platform independent: Java works on the 'write once, run anywhere' principle as it supports multiple platforms like Windows, Linux, Mac, Sun Solaris, etc.
    d. Robust: Java has strong memory management as there are no pointer allocations. It has an automatic garbage collection that prohibits memory leaks.
    e. Interpreted: As mentioned, Java compiler converts the codes into Java bytecodes which are then interpreted and executed by Java Interpreter.

4. Differentiate between JDK, JRE, and JVM.
    a. JVM stands for Java Virtual Machine which provides the runtime environment for Java bytecodes to be executed.
    b. JRE (Java Runtime Environment) includes the sets of files required by JVM during runtime.
    c. JDK (Java Development Kit) consists of JRE along with the development tools required to write and execute a program.
5. 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
6. Distinguish between a class and an object?

A class is a template for an object, and an object is an instance of a class

7. Define abstract class?

Abstract classes are classes from which instances are usually not created. It is basically used to contain common characteristics of its derived classes. Abstract classes are generally higher up the hierarchy and act as super classes. Methods can also be declared as abstract. This implies that non-abstract classes must implement these methods.

8. Define Inner Class ?

An inner class is a nested class whose instance exists within an instance of its enclosing class and has direct access to the instance members of its enclosing instance

9. What is meant by an innerclass?

An inner class is a nested class whose instance exists within an instance of its enclosing class and has direct access to the instance members of its enclosing instance

```
class <EnclosingClass>
        {
          class <InnerClass>
                {

                 }
        }
```

10. 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.

11. Define method overloading?

In Java it is possible to define two or more methods within the same class that share the same name, as long as their parameter declarations are different. When this is the case, the methods are said to be overload, and the process is referred to as method overloading.

12. Define method?

Methods are functions that operates on instances of classes in which they are defined. Objects can communicate with each other using methods and can call methods in other classes. Just as there are class and instance variable, there are class and instance methods. Instance methods apply and operate on an instance of the class while class methods operate on the class.

13. What are the uses of the keyword ‘final’?

- The class can be declared as final, if instances or subclasses are not to be created.
- The variables are declared as final, value of the variable must be provided at the time of declaration.
- The Method can be declared as final indicating that they cannot be overridden by subclasses.

14. What are static methods?

Static methods and variables can be used independently of any object. To do so, you need only specify the name of their class following by the dot operator.

15. What is inheritance?

In Object-Oriented programming, inheritance refers to the properties of a class being available to many other classes. A derived class / sub class is one that has been created from an existing class. Inheritance is the process of deriving a class from a super class or a base class. No changes are made to the base class. The derived class has a larger set of properties that its base class. Inheritance has two advantages

a) Reusability of code

b) Data and methods of a super class are physically available to its subclasses

16. Explain method overloading.

When a Java program contains more than one method with the same name but with different properties, then it is called method overloading.

17. Define an abstract class.

A class that contains the abstract keyword in its declaration is known as an abstract class. It can have abstract and non-abstract methods (method with a body).

This class can have public, private, protected, or constants and default variables.

It needs to be extended and its method needs to be implemented. It cannot be instantiated.

If a class has at least one abstract method, then the class must be declared abstract.

18. What is a Literal? What are the different types of literals?

A Literal represents a value of a certain type where the type describes the behaviors of the value. The different types of literals are:

Number literals
Character literals
Boolean literals
String literals

19. What is a Number literals?

There are several integer literals as int, long, octal, hexadecimal etc. 10 is an example of a decimal integer literal of type int. If a decimal integer literal is larger that the int, it is declared to be of type long. A number can be made long by appending L or 1 to it. Negative integers are preceded by the minus sign. These integers can also be expressed as octal or hexadecimal. A leading 0 to the integer indicate that the number is an octal integer. For example, 0987 is an octal integer. A leading 0x to the integer indicate that the number is a hexadecimal integer, example 0xaf94 is a hexadecimal number.

20. What is a String literals?

A string is a combination of characters. string literals are a set of characters that are enclosed within double quotes. As they are real objects, it is possible to concatenate, modify and test them. For example, “This is a test string” represents a string. Strings can contain character constants and Unicode characters.

21. What are token in Java?

Token means, the smallest individual units of program. In Java the following token are available.

White space
Identifiers
Literals
Comments
Separators
Keywords

22. 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;
```

23. 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

Local variables are used inside blocks as counts or in methods as temporary variables. Once the block or the method is executed, the variable ceases to exist. Instance variable are used to define attributes or the state of a particular object. These are used to store information needed by multiple methods in the objects.

24. 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

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

26. 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 its

27. 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.

28. Define Array? How to declare an array?

    An array is an object that stores a list of items. Each slot in an array holds individual elements. An array should be of a single type, comprising of integers, strings and so on. To create an array, a variable to hold the array is declared, and a new object is created and assigned to it.

29. 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 statement given in <pass> will be executed. Otherwise, the statement given in <fail> will be executed.

30. List out the operator in Java

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

31. What are jump statements in Java?

    In java have three jump statements
    - return
    - continue
    - break

32. 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.

33. What is means by Garbage Collection?

    In certain languages like C++, dynamically allocated objects must be manually released by use of a delete operator. In Java deallocation happens automatically. The technique that accomplishes this is called garbage collection.

34. Write any three OOP principal?

- § Encapsulation
- § Inheritance
- § Polymorphism

35. What is meant by Object Oriented Programming?
OOP is a method of programming in which programs are organised as cooperative collections of objects. Each object is an instance of a class and each class belong to a hierarchy.

36. What is an Instance?
An instance has state, behaviour and identity. The structure and behaviour of similar classes are defined in their common class. An instance is also called as an object.

37. What are the core OOP's concepts?
Abstraction, Encapsulation,Inheritance and Polymorphism are the core OOP's concepts.

38. What is meant by abstraction?
Abstraction defines the essential characteristics of an object that distinguish it from all other kinds of objects. Abstraction provides crisply-defined conceptual boundaries relative to the perspective of the viewer. Its the process of focussing on the essential characteristics of an object. Abstraction is one of the fundamental elements of the object model.

39. What is meant by Encapsulation?
Encapsulation is the process of compartmentalising the elements of an abtraction that defines the structure and behaviour. Encapsulation helps to separate the contractual interface of an abstraction and implementation.

40. What are Encapsulation, Inheritance and Polymorphism?
Encapsulation is the mechanism that binds together code and data it manipulates and keeps both safe from outside interference and misuse. Inheritance is the process by which one object acquires the properties of another object. Polymorphism is the feature that allows one interface to be used for general class actions.

41. What are methods and how are they defined?
Methods are functions that operate on instances of classes in which they are defined. Objects can communicate with each other using methods and can call methods in other classes. Method definition has four parts. They are name of the method, type of object or primitive type the method returns, a list of parameters and the body of the method. A method's signature is a combination of the first three parts mentioned above.

42. 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.

43. What is an Object and how do you allocate memory to it?
Object is an instance of a class and it is a software unit that combines a structured set of data with a set of operations for inspecting and manipulating that data. When an object is created using new operator, memory is allocated to it.

44. Explain the usage of Java packages.
This is a way to organize files when a project consists of multiple modules. It also helps resolve naming conflicts when different packages have classes with the same names. Packages access level also allows you to protect data from being used by the non-authorized classes.

45. What is method overloading and method overriding?
Method overloading: When a method in a class having the same method name with different arguments is said to be method overloading. Method overriding : When a method in a class having the same method name with same arguments is said to be method overriding.What gives java it's "write once and run anywhere" nature?
All Java programs are compiled into class files that contain bytecodes. These byte codes can be run in any platform and hence java is said to be platform independent.

46. What is a constructor? What is a destructor?
Constructor is an operation that creates an object and/or initialises its state. Destructor is an operation that frees the state of an object and/or destroys the object itself. In Java, there is no concept of destructors. Its taken care by the JVM.

47. What is the difference between constructor and method?
Constructor will be automatically invoked when an object is created whereas method has to be called explicitly

48. What is Static member classes?
A static member class is a static member of a class. Like any other static method, a static member class has access to all static methods of the parent, or top-level, class.

49. What is Garbage Collection and how to call it explicitly?
When an object is no longer referred to by any variable, java automatically reclaims memory used by that object. This is known as garbage collection. System. gc() method may be used to call it explicitly

50. In Java, How to make an object completely encapsulated?
All the instance variables should be declared as private and public getter and setter methods should be provided for accessing the instance variables.

51. What is static variable and static method?
static variable is a class variable which value remains constant for the entire class
static method is the one which can be called with the class itself and can hold only the staic variables.

52. What is finalize() method?
finalize () method is used just before an object is destroyed and can be called just prior to garbage collection.

53. What is the difference between String and String Buffer?
a) String objects are constants and immutable whereas StringBuffer objects are not. b) String class supports constant strings whereas StringBuffer class supports growable and modifiable strings. What is the difference between Array and vector?

B)Array is a set of related data type and static whereas vector is a growable array of objects and dynamic

54. What is the difference between this() and super()?
this() can be used to invoke a constructor of the same class whereas super() can be used to invoke a super class constructor.

55. Explain working of Java Virtual Machine (JVM)?
JVM is an abstract computing machine like any other real computing machine which first converts .java file into .class file by using Compiler (.class is nothing but byte code file.) and Interpreter reads byte codes.

56. What is meant by Inheritance?
Inheritance is a relationship among classes, wherein one class shares the structure or behaviour defined in another class. This is called Single Inheritance. If a class shares the structure or behaviour from multiple classes, then it is called Multiple Inheritance. Inheritance defines "is-a" hierarchy among classes in which one subclass inherits from one or more generalised superclasses.

57. What is meant by Inheritance and what are its advantages?
Inheritance is the process of inheriting all the features from a class. The advantages of inheritance are reusability of code and accessibility of variables and methods of the super class by subclasses.

58. What is the difference between superclass and subclass?
A super class is a class that is inherited whereas sub class is a class that does the inheriting.

59. Differentiate between a Class and an Object?
The Object class is the highest-level class in the Java class hierarchy. The Class class is used to represent the classes and interfaces that are loaded by a Java program. The Class class is used to obtain information about an object's design. A Class is only a definition or prototype of real life object. Whereas an object is an instance or living representation of real life object. Every object belongs to a class and every class contains one or more related objects.

60. What is meant by Binding?
Binding denotes association of a name with a class

61. What is meant by Polymorphism?
Polymorphism literally means taking more than one form. Polymorphism is a characteristic of being able to assign a different behavior or value in a subclass, to something that was declared in a parent class.

62. What is Dynamic Binding?
Binding refers to the linking of a procedure call to the code to be executed in response to the call. Dynamic binding (also known as late binding) means that the code associated with a given procedure call is not known until the time of the call at run-time. It is associated with polymorphism and inheritance.

63. What is final modifier?
The final modifier keyword makes that the programmer cannot change the value anymore. The actual meaning depends on whether it is applied to a class, a variable, or a method.

- final Classes- A final class cannot have subclasses.
- final Variables- A final variable cannot be changed once it is initialized.
- final Methods- A final method cannot be overridden by subclasses.

64. What is an Abstract Class?
Abstract class is a class that has no instances. An abstract class is written with the expectation that its concrete subclasses will add to its structure and behaviour, typically by implementing its abstract operations.

65. What are inner class and anonymous class?
Inner class: classes defined in other classes, including those defined in methods are called inner classes. An inner class can have any accessibility including private. Anonymous class: Anonymous class is a class defined inside a method without a name and is instantiated and declared in the same place and cannot have explicit constructors

66. What is a base class?
Base class is the most generalised class in a class structure. Most applications have such root classes. In Java, Object is the base class for all classes.

67. What is reflection in java?
Reflection allows Java code to discover information about the fields, methods and constructors of loaded classes and to dynamically invoke them.

68. Define superclass and subclass?
Superclass is a class from which another class inherits Subclass is a class that inherits from one or more classes.

69. What is meant by Binding, Static binding, Dynamic binding?
Binding: Binding denotes association of a name with a class.
Static binding: Static binding is a binding in which the class association is made during compile time. This is also called as Early binding.

70. Dynamic binding: Dynamic binding is a binding in which the class association is not made until the object is created at execution time. It is also called as Late binding.