UNIT 1

2 marks

1. Define Object Oriented Programming

Object-Oriented Programming (OOP) is a programming paradigm based on the concept of "objects", which can contain data, in the form of fields (often known as attributes/data members/properties), and code, in the form of procedures (often known as functions/methods).

2. What are the four principles of OOP?

Abstraction, Encapsulation, Inheritance, Polymorphism

3. What are the features of Object Oriented Programming?

- Emphasis is on data rather than procedure. Programs are divided into objects.
- Functions that operate on the data of an object are tied together.

Objects may communicate with each other through functions. Follows bottom-upapproach.

4. What are the features of Java Language?

The features of Java Language are Simple, Object-Oriented, Portable, Platform independent, Secured, Robust, Architecture neutral, Dynamic, Interpreted, HighPerformance, Multithreaded and Distributed.

5. What are the operators supported in Java?

Operator in java is a symbol that is used to perform operations. For example: +, -, *, / etc. There are many types of operators in java which are Unary Operator, Arithmetic Operator, Shift Operator, Relational Operator, Bitwise Operator, Logical Operator, Ternary Operator and, Assignment Operator.

6. Define Abstraction.

Abstraction refers to the act of representing the essential features without including the background details or explanations. It reduces the complexity and increases the efficiency. Small programs can be easily upgraded to large programs. Software complexity can easily be managed.

7. What is Polymorphism?

Polymorphism is the ability to take more than one form and refers to an operation exhibiting different behavior instances. Object oriented programs use polymorphism tocarry out the same operation in a manner customized to the object.

8. Define Objects and Classes in Java (Nov/Dec 2018)

A class can be defined as a template/blueprint that describes the behavior/state that the object of its type support.ie. A class is a blueprint from which individual objects are created An Object can be defined as an instance of a class. An object contains an address and takes up some space in memory. Objects can communicate without knowing the details of each other's data or code

9. Write the syntax for declaration of class and creation of objects?

//Syntax for class

class ClassName{

Access-specifier datatype datamember(s);

Access-specifier method-declaration(s)/definition(s) **}**

//Syntax for object

ClassName objectName=new ClassName();

10. Define Encapsulation (Apr/May 2012) (Apr 2017)(Nov/Dec 2020)(Apr/May 2021)

The wrapping up of data and functions into a single unit is known as data encapsulation. Here the data is not accessible to the outside the class. The data inside that class is accessible by the function in the same class. It is normally not accessible from the outside of the component.

11. What is Inheritance? What are its types?

Inheritance is a mechanism of reusing the properties and extending existing classeswithout modifying them, thus producing hierarchical relationships between them.

Types: Single inheritance, Multi-level inheritance, Hierarchical inheritance, Hybrid inheritance.

12. What do you mean by Variable? What are the rules for variable declaration?

Variable is a fundamental unit of storage in java. The variables are used in combination with identifiers, data types, operators and some value for initialization. The syntax ofvariable declaration will be:

data_type name_of_variable[=initialization];

13. What is Garbage collection?

In java, garbage means unreferenced objects. Garbage Collection is process of reclaiming the runtime unused memory automatically. In other words, it is a way to destroy the unused objects.

14. What is Constructors in Java? What are its types? (Nov/Dec 2020)(Apr/May 2021)

A constructor is a special method that is used to initialize an object. The name of the constructor and the name of the class must be the same. A constructor does not have any return type.

There are two types of Constructor

Default Constructor – constructor without argument Parameterized constructor – constructor with argument

15. How will you declare a two dimensional array?

The two dimensional array can be declared and initialized as follows

Syntax: data_type array_name[][]=new data_type[size][size];For example: int a[][]=new int[3][3];

16. What is Java Doc? List any four Java Doc comments. (Nov/Dec 2018)(Nov/Dec 2019)

A Javadoc comment is set off from code by standard multi-line comment tags/* and */. The opening tag, however, has an extra asterisk, as in /**.The first paragraph is a description of the method documented. Following the description are a varying number of descriptive tags, signifying: The parameters of the method (@param),What the method returns (@return) and any exceptions the method may throw (@throws)

17. Define access specifier/modifier? (Nov/Dec 2018) (Nov/Dec 2019)

Access specifiers/modifiers in Java helps to restrict the scope of a class, constructor,variable, method or data member. There are four types of access modifiers available injava:Default – No keyword required , Private , Protected and Public

18. What is a package?

A java package is a group of similar types of classes, interfaces and sub-packages. Packagein java can be categorized in two form, built-in package and user-defined package. Thereare many built-in packages such as java, lang, awt, javax, swing, net, io, util, sql etc.

19. What is the use of static keyword in Java?

The static keyword in Java is used for memory management mainly. We can apply java static keyword with variables, methods, blocks and nested class. The static keyword belongs to the class than an instance of the class

20. Can a java Source file be saved sing a name other than the class name?Justify.(APR/MAY 2019)

Yes, you can save your java source code file with any other name, not same as your main class name but when you comiple it than byte code file name will be same as your main class name.

21. What are the control flow statements in java?

if, if-else, nested-if , if-else-if,switch-case,jump – break, continue, return

UNIT-I / PART B
(i) Explain the characteristics of OOPs (Nov/Dec 2018) (ii) Explain the features and characteristics of JAVA(Nov/Dec 2019)
i) Describe the typical java program structure. ii) Explain the general java program compilation and execution.
What are the different data types in JAVA? Explain each of them with example.
How to pass and return the objects to and from the method?
Discuss in detail the access specifiers available in Java.
Explain Packages in detail.
Explain Constructors with examples.
Explain in detail the various operators in Java.
Explain the concepts of arrays in Java and explain its types with examples?
Explain in detail about static variable and static method in Java with example?
Discuss the three OOPS principles in detail.(Apr/May 2019)
What are literals? Explain the types of literals supported by Java.(Apr/May 2019)
Explain the Selection statements in Java with suitable examples.(Apr/May 2019)
Write a Java code using do-while loop that counts down to 1 from 10 printing exactly ten lines of —hello‖.(Apr/May 2019)
What is JVM? Explain the internal architecture of JVM with neat sketch.(Nov/Dec 2019)
Develop a java program to find the smallest number in the given array by creating one dimensional array and two dimensional array using new operator.(Nov/Dec 2019).
(i) What is a method? How method is defined? Give example (Nov./Dec.2018) (ii) State the purpose of finalize() method in java. With an example explain how finalize() method can be used in java program
What are the three categories of control statements used in Java? Explain each category with example. (Nov/Dec 2020)(Apr/May 2021)
How Java changed the internet? ii) If semicolons are needed at the end of each statement, why does the comment line not end with a semicolon ? (Nov/Dec 2020)(Apr/May 2021)

UNIT 2

2 marks

1. What is meant by Inheritance and what are its advantages?

Inheritance is a relationship among classes, wherein one class shares the structure or behavior defined in another class. The advantages of inheritance are reusability of code and accessibility of variables and methods of the super class by subclasses.

2. What is the use of super keyword? (Nov/Dec 2020)(Apr/May 2021)

This is used to initialize constructor of base class from the derived class and also access thevariables of base class like super.i = 10.

3. 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.

4. What is protected function?

Protected members that are also declared as static are accessible to any friend or member function of a derived class. Protected members that are not declared as static are accessible to friends and member functions in a derived class only through a pointer to,reference to, or object of the derived class.

5.Define super class and subclass?

Super class is a class from which another class inherits. Subclass is a class that inherits fromone or more classes

6.What is role of access modifier in the member of a class in Java?

A private member is only accessible within the same class as it is declared. A member withno access modifier is only accessible within classes in the same package. A protected member is accessible within all classes in the same package and within subclasses in other packages.

7. What is protected method?

A protected method can be called by any subclass within its class, but not by unreleated classes. Declaring a method protected defines its access level. The other options for declaring visibility are private and public. If undeclared, the default access level is package.

8. 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.

9.Why creating an object of the sub class invokes also the constructor of the super class? When inheriting from another class, super() has to be called first in the constructor. If not,the compiler will insert that call. This is why super constructor is also invoked when a Subobject is created. This doesn't create two objects, only one Sub object. The reason to havesuper constructor called is that if super class could have private fields which need to be initialized by its constructor.

10. 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 behavior, typically by implementing its abstract operations.

11.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.

12.What is an Interface?
Interface is an outside view of a class or object which emphasizes its abstraction whilehiding its structure and secrets of its behavior.

13. What is the difference between a static and a non-static inner class?
A non-static inner class may have object instances that are associated with instances of theclass's outer class. A static inner class does not have any object instances.

14. What is the difference between abstract class and interface?
a) All the methods declared inside an interface are abstract whereas abstract class musthave at least one abstract method and others may be concrete or abstract.
b) In abstract class, key word abstract must be used for the methods whereas interface weneed not use that keyword for the methods.
Abstract class must have subclasses whereas interface can't have subclasses

15. Difference between class and interface. (Nov/Dec 2020)(Apr/May 2021)
Class and Interface both are used to create new reference types. A class is a collection of fields and methods that operate on fields. An interface has fully abstract methods i.e. methods with nobody. An interface is syntactically similar to the class but there is a major difference between class and interface that is a class can be instantiated, but an interface can never be instantiated.

16. What is an interface and state its use?(Nov/Dec 2019)
Interface is similar to a class which may contain method's signature only but not bodies and it is a formal set of method and constant declarations that must be defined by the class that implements it. Interfaces are useful for: a) Declaring methods that one or more classes are expected to implement b) Capturing similarities between unrelated classes without forcing a class relationship. c) Determining an object's programming interface without revealing the actual body of the class.

17. What is a cloneable interface and how many methods does it contain?
It is not having any method because it is a TAGGED or MARKER interface.

18. What is object cloning? (Nov/Dec 2018)(Nov/Dec 2019)
It is the process of duplicating an object so that two identical objects will exist in the memory at the same time

19. Define Package.
To create a package is quite easy: simply include a package command as the first statement in a Java source file. Any classes declared within that file will belong to the specified package. The package statement defines a name space in which classes are stored. If you omit the package statement, the class names are

put into the default package, which has no name.

20. Define Array list class.

The ArrayList class extends AbstractList and implements the List interface. ArrayList is a generic class that has this declaration:

class ArrayList<E>

Here, E specifies the type of objects that the list will hold. An ArrayList is a variable-length array of object references. That is, an ArrayList can dynamically increase or decrease in size. Array lists are created with an initial size. When this size is exceeded, the collection isautomatically enlarged. When objects are removed, the array can be shrunk.

21. What is String in Java? Is String is data type?

String in Java is not a primitive data type like int, long or double. String is a class or in more simple term a user defined type. String is defined in java.lang package and wrappers its content in a character array. String provides equals() method to compare two String and provides various other method to operate on String like toUpperCase() to convert String into upper case, replace() to replace String contents, substring() to get substring, split() to split long String into multiple String.

22. Brief Inner class in Java with its syntax.

Java inner class or nested class is a class which is declared inside the class or interface.

We use inner classes to logically group classes and interfaces in one place so that it can bemore readable and maintainable.

Syntax of Inner class

```
class Java_Outer_class{
//code
 class Java_Inner_class{
//code } }
```

23. State the conditions for method overriding in Java. (APR/MAY 2019)

Following are the conditions to be considered while overriding a method properly

- The argument list should be exactly the same as that of the overridden method.
- The return type should be the same or a subtype of the return type declared in the original overridden method in the superclass.The access level cannot be more restrictive than the overridden method's access level. For example: If the superclassmethod is declared public then the overriding method in the subclass cannot be either private or protected.

24. Write the syntax for importing packages in a java source file and give an example.(APR/MAY 2019)

Creating a package in java is quite easy, simply include a package command followed byname of the package as the first statement in java source file.

```
package mypack; public class
employee
{
  String empId;String
  name;
}
```

The above statement will create a package with name mypack in the project directory.Java uses file system directories to store packages.

UNIT 2
16 marks

1	Define Inheritance. With diagrammatic illustration and java programs illustrate the different types of inheritance with an example (Nov./Dec.2018)(Nov/Dec 2019)
2	What is interface? With an example explain how to define and implement interface. (Nov/Dec 2020)(Apr/May 2021)
3	Differentiate method overloading and method overriding. Explain both with an example program.
4	Explain about the object and abstract classes with the syntax.(Nov/Dec 2019)
5	Discuss in detail about inner class. With its advantages.
6	What is meant by object cloning? Explain it with an example.
7	Explain how inner classes and anonymous classes works in java program.
8	What is a Package? What are the benefits of using packages? Write down the steps in creating a package and using it in a java program with an example.
9	Explain arrays in java with suitable example.
10	How Strings are handled in java? Explain with code, the creation of Substring, Concatenation and testing for equality.
11	Write a Java program to create a student examination database system that prints the mark sheet of students.Input student name,marks in 6 subjects.This mark should be between 0 and 100. (Nov./Dec.2018) If the average of marks is>= 80 then prints Grade _A' If the average of marks is< 80 and >=60 then prints Grade _B'If the average of marks is< 60 and >=40 then prints Grade _C'Else prints Grade _D'
12	Explain hierarchical and multi-level inheritances supported by Java and demonstrate the execution order of constructors in these types.(Apr/May 2019)
13	i) Explain simple interfaces and nested interfaces with examples ii) Present a detailed comparison between classes and interfaces (Apr/May 2019)

UNIT III

2 marks

1. **Write down the purpose of exception handling mechanism.**

   The main purpose of exception handling mechanism is used to detect and report an "exceptional circumstance" so that necessary action can be taken. It performs the following tasks 5. Find the problem(Hit the exception) 6. Inform that an error occurred(throw the exception). 7. Receive the error information(Catch the exception) 8. Take corrective actions(Handle the exception)

2. **What are the types of exceptions?**

   There are two types of exceptions 1. Predefined Exceptions-The Exceptions which are predefined are called predefined exceptions 2. Userdefined Exceptions- The Exceptions which are defined by the user are called userdefined exceptions

3. **How the exception handling is managed?**

   Java exception handling is managed via five keywords. • try • catch • throw • throws and • Finally

4. **Write down the general form of an exception-handling block.**

   The general form of an exception-handling block

```
try
{
// block of code to monitor for errors
 }
catch (ExceptionType1 exOb)
{
// exception handler for ExceptionType1
}
catch (ExceptionType2 exOb)
{
// exception handler for ExceptionType2
}
//
...
finally
{
// block of code to be executed after try block ends
}
```

5. **Write down the use of try and catch block in exception handling.**
   The try block allows us to fix the errors. Catch block prevents the program from automatically terminating. To handle a run-time error, enclose the code to be monitored inside a try block. After the try block, include a catch block that specifies the exception type that needs to be caught.
   Syntax:
   try
   {
   statement;
   }
   catch(Exception-type exOb)
   {
   statement;
   }

6. **Explain the situation where we need to use multiple catch clauses.**
   In some situation, more than one exception can occur by a single piece of code. To handle this situation, we can use two or more catch clauses, each catching a different type of exception. When an exception is thrown, each catch block is executed in order, and the first one whose type matches that exception is executed.After one catch block executes, the others are bypassed, and continues after the try/catch block.

7. **Write down the syntax for multiple catch clauses.**
   The syntax for multiple catch clauses
   try
   {
   // block of code to monitor for errors
   }
   catch (ExceptionType1 exOb)
   {
   // exception handler for ExceptionType1
   }
   catch (ExceptionType2 exOb)
   {
   // exception handler for ExceptionType2
   }
8. **Explain the situation where we need to use nested try statements.**
   The try statement can be nested. That is, a try statement can be inside the block of another try. Each time a try statement is entered, the context of that exception is pushed on the stack. If an inner try statement does not have a catch handler for a particular exception, the stack is unwound and the next try statement's catch handlers are inspected for a match. This continues until one ofthe catch statements succeeds, or until all of the nested try statements are exhausted. If no catch statement matches, then the Java run-time system will handle the exception.

9. **Write down the syntax for nested try.**

The syntax for nested try statement:

```
//Main try block
try
{
statement 1;
statement 2;
//try-catch block inside another try block
try
{
statement 3;
statement 4;
//try-catch block inside nested try block
try
{
statement 5;
statement 6;
}
catch(Exception e2)
{
//Exception Message
}
}
catch(Exception e1)
{
//Exception Message
}
}
//Catch of Main(parent) try block
catch(Exception e3)
{
//Exception Message
}
```

10. **Write down the use of throw statement**.

The throw keyword in Java is used to explicitly throw an exception from a method or any block of code. We can throw either checked or unchecked exception.

Syntax:

```
throw new exception_class("error message");
```

For example:

```
throw new ArithmeticException("dividing a number by 5 is not allowed in this program");
```

11. Write down the use of throws clause.

Using throws clause, We can list the types of exceptions that a method might throw.The exceptions which are thrown in a method might be using throws clause. If they are not, a compile-time error will result.

Syntax:

```
type method-name(parameter-list) throws exception-list
{
// body of method
}
```

Exception-list is a comma-separated number of exceptions that a method can throw.

12. Write down the use of finally clause.

finally creates a block of code that is to be executed after a try/catch block has completed its execution.The finally block will execute if an exception is thrown or not thrown. The finally clause is optional.Each try block requires either one catch or a finally clause

Syntax:

```
try
{
 //Statements that may cause an exception
}
catch
{
 //Handling exception
}
finally
{
 //Statements to be executed
}
```

13. **What is Unchecked Exception**?

These are the exceptions that are not checked at compiled time.

14. What is Checked Exception?

These are the exceptions that are checked at compile time. If some code within a method throws a checked exception, then the method must either handle the exception or it must specify the exception using throws keyword.

15. Explain the way to create own exceptions.

We can throw our own exceptions using throw keyword.

Syntax:

```
throw new Throwable_subclass;
```

Eg:

```
throw new ArithmeticException;
```

16. Define Stack Trace Elements.

The StackTraceElement is a class that describes a single stack frame,which is an element of a stack trace when an exception occurs.The getStackTrace() method is used to return an array of StackTraceElements. Each stack frame contains the following

the class name
the method name
The file name
And the source-code line number

17. List out the methods in StackTraceElements

The methods in StackTraceElements are

boolean equals(Object ob)
String getClassName( )
String getFileName( )
int getLineNumber( )
String getMethodName( )

18. What is meant by a stream?

A stream is an abstraction that either produces or consumes information. A stream is linked to a physical device by the java I/O system.The input stream may abstract many different kinds of input: from a disk file,a keyboard,or a network socket. Likewise,an output stream may refer to the console such as a disk file, or a network connection.

19. **What are the two types of streams?**

There are two types of streams

Byte streams
Character streams

20. Write the use of FileInputStream class

Java FileInputStream class obtains input bytes from a file. It is used for reading byte-oriented data (streams of raw bytes) such as image data, audio, video etc. You can also read character-stream data.

21. Write down the methods in FileInputStream Class.

The methods in FileInputStream class are

int available()
int read()
long skip(long x)
FileChannel getChannel()
FileDescriptor getFD()
protected void finalize()
void close()

22. Write the use of FileOutputStream class.

Java FileOutputStream is an output stream used for writing data to a file.

You can write byte-oriented as well as character-oriented data through FileOutputStream class. But, for character-oriented data, it is preferred to use FileWriter than FileOutputStream.

23. **Write down the methods in FileOutputStream Class.**

The methods in FileOutputStream class are protected

void write(byte[] ary)

void write(int b)

FileChannel getChannel()

FileDescriptor getFD()

void close()

void finalize()

PART B

1. Explain the Throwing and Catching Exception.
2. What is exception? How to throw an exception? Give an example.
3. What is finally class? How to catch exceptions? Write an example.
4. What is meant by exceptions? Why it is needed? Describe the exception hierarchy.
5. Write note on Stack Trace Elements. Give example.
6. Define Exception and explain its different types with example.
7. Discuss about character stream classes.
8. With suitable coding discuss all kinds of exception handling.
9. Write a note on java.io package with its stream classes and methods in it.
10. Write a Java program to demonstrate I/O character stream classes.
11. Write short notes on PrintStream class.
12. Explain how user-defined exception subclasses are created in Java.
13. What is the necessity of exception handling? Explain exception handling taking "Divide – by Zero" as an example.
14. Write a Java program to copy the data from one file to another file.
15. Write a program that uses a sequence input stream to output the contents of two files.
16. List the five keywords in Java exception handling. Describe the use of the keywords.