JAVA PROGRAMMING

1. Choose the correct answers from the given alternatives:

(2017)

i) int x[]={10,20,30}
int y[];
y=x;
What is the result of the above code?
(a) x and y will point to different memory locations
(b) x and y will point to same memory locations
(c) Syntax error
(d) None of the above

ii) What is stored in the object obj in following lines of code?
(a) Memory address of allocated memory of object
(b) NULL
(c) Any arbitrary pointer
(d) Garbage

iii) Which of these operators is used to allocate memory for an object?
(a) malloc (b) alloc (c) new (d)give

iv) Which of the statement is incorrect?
(a) Every class must contain a main() method
(b) Applets do not require a main method
(c) There can be only one main method in a program
(d) main() method must be made public

v) What is the return type of constructor?
(a) int (b) float (c) void (d) none of the mentioned

vi) Which of these methods of String class is used to obtain character at specified index?
(a) char() (b) Charat() (c)charat() (d)charAt()

vii) Which of these keywords is used to refer to member of base class from a sub class?
(a) upper (b)super (c) this (d) None of the mentioned

viii) Which of these methods of String class can be used to test to strings for equality?
(a) isequal() (b) isequals() (c) equal() (d) equals()

ix) Which of these methods of String class can be used to prevent Method overriding?
(a) static (b) constant (c) protected (d) final

x) Which of these is correct way of inheriting class A by class B?
(a) class B + class A{}
(b) class B inherits class A
(c) class B extends A{}
(d) class B extends class A{}

xi) Which of these keywords is not a part of exception handling?
(a) try (b) finally (c)thrown (d) catch

xii) Which of these functions is called to display the output of an applet?
(a) display() (b) paint() (c) displayApplet() (d) PrintApplet()

(2018)

xiii) Which of the following in a valid declaration of an object of class Box?
(a) Box obj = new Box();
(b) Box obj = new Box;
(c) obj = new Box();
(d) new Box obj.

xiv) Which keyword is used by method to refer to the object that invoked it?
(a) import (b) catch (c)abstract (d)this

xv) Which of the following is a method having same name as of its class?
(a) finalize (b) delete (c) class (d)constructor

xvi) Which function is used to perform some action when the object is to be destroyed?
(a) finalize (b) delete (c)main() (d) none of the mentioned

xvii) Array in Java are implemented as-
(a) class (b) object (c)variable (d)None of the mentioned

xviii) Which of these cannot be declared static?
(a) class (b) object (c) variable (d) method

xix) Which of these following keywords is used to refer to member of base class from a sub class?
(a) upper (b) super (c) this (d) none of the mentioned

xx) Which of these operators is used to generate an instance of an exception that can be thrown by using throw?
(a) new (b) malloc (c) alloc (d)thrown

xxi) Which of these interfaces is implemented by Thread class?
(a) Runnable (b) Connections (c)Set (d)MapConnections

xxii) Which of these methods can be used to output a string in an applet?
(a) display() (b) print() (c) drawstring() (d)transient()

xxiii) Which of these is a method to clear all the data present in output buffers?
(a) clear() (b) flush() (c) fflush() (d) close()

2. Answer the following questions:

(2017)

i) What is JDBC ResultSet?
Answer: java.sql.ResultSet also an interface and is used to retrieve SQL select query results. A ResultSet object maintains a cursor pointing to its current row of data. Initially the cursor is positioned before the first row.

ii) What is an Applet?
Answer: An applet is a Java program that can be embedded into a web page. It runs inside the web browser and works at client side. An applet is embedded in an HTML page using the APPLET or OBJECT tag and hosted on a web server.

iii) Is it essential to catch all types of exceptions?
Answer: No, not every exception requires a try-catch. Every checked exception requires a try catch.

iv) How do we start a thread?
Answer: For starting a thread in Java, we need to calling the start() function of the thread class extended in another sub-class.

v) What is a package?
Answer: Package in Java is a mechanism to encapsulate a group of classes, sub packages and interfaces. All we need to do is put related classes into packages.

vi) How to get the length of an array in Java?
Answer: array.length, length is a final variable applicable for arrays. With the help of length variable, we can obtain the size of the array.

(2018)

vii) What is the use of JDBC DriverManager class?
Answer: The DriverManager provides a basic service for managing a set of JDBC drivers. As part of its initialization, the DriverManager class will attempt to load the driver classes referenced in the "jdbc.drivers" system property.

viii) Write the arguments used in drawRoundRect().
Answer: DrawRoundRect(int x, int y, int width, int height, int arcWidth, int arcHeight). Here x and y are the screen co-ordinates.

ix) What is an exception?
Answer: An exception is an unwanted or unexpected event, which occurs during the execution of a program i.e. at run time, that disrupts the normal flow of the program's instructions.

x) What is a thread?
Answer: A thread is a single sequential flow of control within a program. However, a thread itself is not a program. It cannot run on its own, but runs within a program.

xi) What will be the output?

```
String s = ”Java Programming”;
System.out.println(s.indexOf(‘a’, 5));
```

Answer: The output will be 6.

xii) What is abstract class?
Answer: An abstract class, in the context of Java, is a superclass that cannot be instantiated and is used to state or define general characteristics. The abstract class is declared using the keyword abstract.

Broad Questions

(2017)

1. a) Why is Java known as platform-neutral (or platform-independent) language?

Answer: Java is known as platform independent because it uses the concept of generating the byte code of the high level program, and then run that byte code(intermediate code) on JVM(Java Virtual Machine), now what is JVM, actual JVM is a virtual computer system that run on your original computer system .

b) What are command line arguments? How they are useful?

Answer: The command line argument is the argument passed to a program at the time when you run it. To access the command-line argument inside a java program is quite easy, they are stored as string in String array passed to the args[] parameter of main() method.

The arguments can be used as input. So, it provides a convenient way to check out the behavior of the program on various values. We can pass any number of arguments from the command prompt or nearly anywhere a Java program is executed.

c) Explain with an example.

Answer: Here Command line Arguments are 10, 20 and these values are stored as an array and those values were sent to the main function. In order to get that values, we are using the method called" string args[] " as a formal argument.

```
class A{

        public static void main(String args[]){

                System.out.println(args[0]);
                System.out.println(args[1]);
        }
}

compilation: javac A.java
execution: java A 10 20

Output:
10 20
```

2. a) What is inheritance and how does it help us to create new classes quickly?

Answer: Inheritance in Java is a mechanism in which one object acquires all the properties and behaviors of a parent object. It is an important part of OOPs (Object Oriented programming system).

The idea behind the inheritance to create new classes, that are built upon existing classes. When we inherit from an existing class, we can reuse methods fields of the parent class. Moreover, we can add. new methods in our current class also.

b) Explain method overriding with an example.

Answer: Method overriding is one of the ways by which java achieve Run Time Polymorphism. The version of a method that is executed will be determined by the object that is used to invoke it. If an object of a parent class is used to invoke the method, then the version in the parent class will be executed, but if an object of the subclass is used to invoke the method, then the version in the child class will be executed. In other words, it is the type of the object being referred to (not the type of the reference variable) that determines which version of an overridden method will be executed.

Example:

```
// Base Class
class Parent {
	void show()
	{
		System.out.println("Parent's show()");
	}
}

// Inherited class
class Child extends Parent {
	// This method overrides show() of Parent
	@Override
	void show()
	{
		System.out.println("Child's show()");
	}
}
// Driver class
class Main {
	public static void main(String[] args)
	{
		// If a Parent type reference refers
		// to a Parent object, then Parent's
		// show is called
		Parent obj1 = new Parent();
		obj1.show();

		// If a Parent type reference refers
		// to a Child object Child's show()
		// is called. This is called RUN TIME
		// POLYMORPHISM.
		Parent obj2 = new Child();
		obj2.show();
	}
}
```

3. a) What is an interface?

   Answer: An interface in java is a blueprint of a class. It has static constants and abstract methods. It is used to achieve abstraction and multiple inheritance in Java.

   b) What are the advantages of using it?

   Answer: There are mainly three reasons to use interface. They are given below.

   - It is used to achieve abstraction.
   - By interface we can support the functionality of multiple inheritance.
   - It can be used to achieve loose coupling.

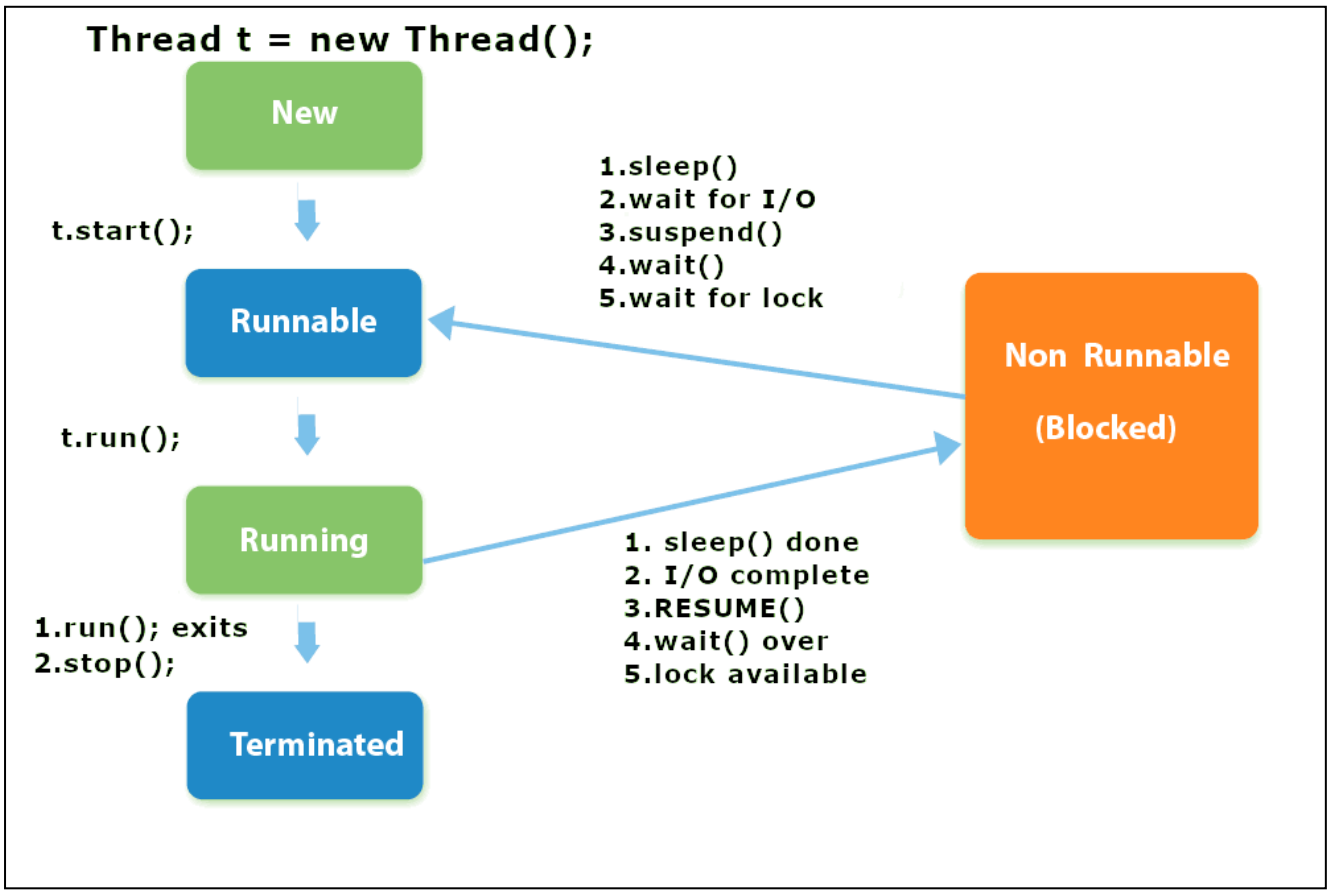
   b) Give an example where interface can be used to support multiple inheritance.

   Answer: If a class implements multiple interfaces, or an interface extends multiple interfaces, it is known as multiple inheritance.

```
interface Printable{
void print();
}
interface Showable{
void show();
}
class A7 implements Printable,Showable{
public void print(){System.out.println("Hello");}
public void show(){System.out.println("Welcome");}
public static void main(String args[]){
A7 obj = new A7();
obj.print();
obj.show();
 }
}

Output:
Hello
Welcome
```

4. a) Describe the complete life cycle of a thread.

Answer: A thread goes through various stages in its lifecycle. For example, a thread is born, started, runs, and then dies. The following diagram shows the complete life cycle of a thread.



- **New:** A new thread begins its life cycle in the new state.
- **Runnable:** After a newly born thread is started, the thread becomes runnable.
- **Running:** In this state a thread is running.
- **Terminated:** In this state a thread is blocked.

b) What are the ways by which we can create thread?

Answer: There are two ways to create a thread:

- extends Thread class-
  Create a thread by a new class that extends Thread class and create an instance of that class. The extending class must override run() method which is the entry point of new thread.

- implement Runnable interface-
  The easiest way to create a thread is to create a class that implements the runnable interface. After implementing runnable interface , the class needs to implement the run() method, which is-
  public void run()

5. a) What is finally block?

Answer: Java finally block is a block that is used to execute important code such as closing connection, stream etc.
Java finally block is always executed whether exception is handled or not.
Java finally block follows try or catch block.
b) Define an exception called “NoMatchException” that is thrown when a string is not equal to “DCST”. Write a program that uses this exception.
Answer:

```
import java.util.scanner;
class NoMatchException extends Exception{
        void NoMatchException(String msg){
                Super(msg);
        }
}
class DemoClass{
        Void checkString(String s){
                if(s != ”DCST”){
                        throw new NoMatchException(s+” is not equal to DCST”);
                }
        }
}
public static void main(String args[]){
        Scanner ob = new Scanner(System.in);
        System.out.println(“Enter a string: ”);
        string s = in.nextLine();
        try{
                checkString(s)
        }catch(NoMatchException){
                System.out.println(“Exception occured”+e);
        }
}
```

Output:

```
Enter a string: ABCD
Exception occurred ABCD is not equal to DCST
```

6. a) Distinguish Between init() and start() method of applet.

Answer:

init()	start()
In this stage, the applet starts its journey.	The start() method is called by init() method.
This method is called when the applet is loaded by the Browser for execution.	For browser, there is another reason to call the start() method.
In this method, the applet object is created by the applet.	When applet is brought back from minimize position.

b) How to pass parameters to an applet? Show with an example.

Answer: public String getParameter(String parameterName)

```
import java.applet.Applet;
import java.awt.Graphics;
public class UseParam extends Applet{
        public void paint(Graphics g){
                String str=getParameter("msg");
                g.drawString(str,50, 50);
        }
}
```

myapplet.html

```
<html>
<body>
<applet code="UseParam.class" width="300" height="300">
<param name="msg" value="Welcome to applet">
</applet>
</body>
</html>
```

(2018)

7. a) Write down the features of Java.

Answer: Features of Java:

- Java can be easily extended since it is Object Oriented.
- Unlike any languages like C & C++, Java is platform independent.
- Java is designed to be easy to learn.
- With Java's secure feature it enables to develop virus-free, tamper-free systems.
- The compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.
- Java is designed for the distributed environment of the internet.

b) What happens if we declare the member function of a class static? Answer with an example.

Answer: When we declare a member of a class as static it means no matter how many objects of the class are created, there is only one copy of the static member. A static member is shared by all objects of the class. All static data is initialized to zero when the first object is created, if no other initialization is present.

```
class SimpleStaticExample{
   // This is a static method
   static void myMethod(){
      System.out.println("myMethod");
   }

   public static void main(String[] args) {
        /* You can see that we are calling this
         * method without creating any object.
         */
         myMethod();
   }
}
```

Output:

```
myMethod
```

8. a) Write a program to input a string and get a substring starting from 3rd character up to 7th.

Answer:

```
import java.io.*;
public static void main(String args[]){
        Scanner ob = new Scanner(System.in);
        string s = in.nextLine();
        System.out.println(s.substring(3,7));
}
```

Output:

Anything

ythin

b) List the names of some Java API packages with their application.

Answer:

java.lang	Language support classes. They include classes for primitive types, string, math functions, thread and exceptions.
java.util	Language utility classes such as vectors, hash tables, random numbers, data, etc.
java.io	Input/output support classes. They provide facilities for the input and output of data.
java.applet	Classes for creating and implementing applets.
java.net	Classes for networking. They include classes for communicating with local computers as well as with internet servers.
java.awt	Set of classes for implementing graphical user interface. They include classes for windows, buttons, lists, menus and so on.

9. a) What are the priority constants of a thread?

Answer:

- public static int MIN_PRIORITY: It is the maximum priority of a thread. The value of it is 1.
- public static int NORM_PRIORITY: It is the normal priority of a thread. The value of it is 5.
- public static int MAX_PRIORITY: It is the minimum priority of a thread. The value of it is 10.

b) List some of the most common types of exceptions that might occur in Java. Give example

Answer:

i. **ArithmeticException**
It is thrown when an exceptional condition has occurred in an arithmetic operation.

ii. **ArrayIndexOutOfBoundsException**
It is thrown to indicate that an array has been accessed with an illegal index. The index is either negative or greater than or equal to the size of the array.

iii. **ClassNotFoundException**
This Exception is raised when we try to access a class whose definition is not found.

iv. **FileNotFoundException**
This Exception is raised when a file is not accessible or does not open.

v. **IOException**
It is thrown when an input-output operation failed or interrupted.

vi. **InterruptedException**
It is thrown when a thread is waiting, sleeping, or doing some processing, and it is interrupted.

vii. **NoSuchFieldException**
It is thrown when a class does not contain the field (or variable) specified.

viii. **NoSuchMethodException**
It is thrown when accessing a method which is not found.

ix. **NullPointerException**
This exception is raised when referring to the members of a null object. Null represents nothing.

x. **NumberFormatException**
This exception is raised when a method could not convert a string into a numeric format.

xi. **RuntimeException**
This represents any exception which occurs during runtime.

xii. **StringIndexOutOfBoundsException**
It is thrown by String class methods to indicate that an index is either negative than the size of the string.

Example:

```
// Java program to demonstrate ArithmeticException
class ArithmeticException_Demo
{
    public static void main(String args[])
    {
        try {
            int a = 30, b = 0;
            int c = a/b;  // cannot divide by zero
            System.out.println ("Result = " + c);
        }
        catch(ArithmeticException e) {
            System.out.println ("Can't divide a number by 0");
        }
    }
}
```

Output:

```
Can't divide a number by 0
```

10. a) Develop an applet that receives three numeric values as input from the user and then displays the largest of the three on the screen.

Answer:

```
import java.awt.*;
import java.applet.*;

public class MaxOf3No extends Applet
{
    TextField T1,T2,T3;

     public void init(){
        T1 = new TextField(10);
        T2 = new TextField(10);
        T3 = new TextField(10);

        add(T1);
        add(T2);
        add(T3);
```

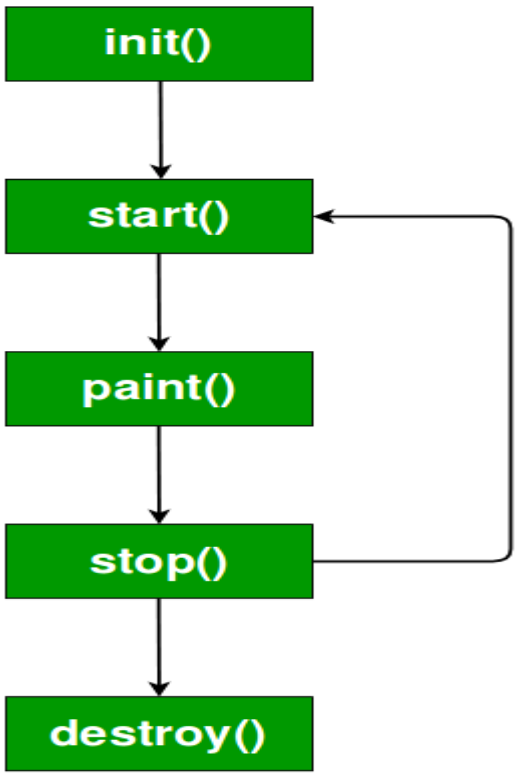
```
        T1.setText(\"0\");
        T2.setText(\"0\");
        T3.setText(\"0\");
      }

      public void paint(Graphics g){
         int a, b, c,result;
         String str;

         g.drawString(\"Enter value to Check the Maximum of 3 \",10,50);

         str=T1.getText();
         a=Integer.parseInt(str);
         str=T2.getText();
         b=Integer.parseInt(str);
         str=T3.getText();
         c=Integer.parseInt(str);

          g.setColor(Color.blue);
          if (a>b) {
             if (a>c)
                result=a;
             else
                result=c;
            }
         else{
             if (b>c)
                result=b;
             else
                result=c;
          }
          g.drawString(\"Maximnum of 3 No is \"+result,10,70);
          showStatus(\"MAXIMUM OF 3 NUMBERS\");
         }

         public boolean action(Event e, Object o){
             repaint();
             return true;
         }
}
```

MaxNumber.html

```
<html>

<body>

<applet code=" MaxOf3No.class" width="150" height="400">

</applet>

</body>

</html>
```

b) Explain applet life cycle.

Answer:



i. **nit( ) :** The init( ) method is the first method to be called. This is where you should initialize variables. This method is called only once during the run time of your applet.

ii. **start( ) :** The start( ) method is called after init( ). It is also called to restart an applet after it has been stopped.

iii. **paint( ) :** The paint( ) method is called each time an AWT-based applet's output must be redrawn. This situation can occur for several reasons. Whatever the cause, whenever the applet must redraw its output, paint( ) is called.

iv. **stop( ) :** The stop( ) method is called when a web browser leaves the HTML document containing the applet—when it goes to another page, for example. When stop( ) is called, the applet is probably running. You should use stop( ) to suspend threads that don't need to run when the applet is not visible. You can restart them when start( ) is called if the user returns to the page.

v. **destroy( ) :** The destroy( ) method is called when the environment determines that your applet needs to be removed completely from memory. At this point, you should free up any resources the applet may be using. The stop( ) method is always called before destroy( ).

11. a) Distinguish between Objects and Classes.

Answer:

Object	Class
Object is an **instance** of a class.	Class is a **blueprint or template** from which objects are created.
Object is a **real world entity** such as pen, laptop, mobile, bed, keyboard, mouse, chair etc.	Class is a **group of similar objects**.
Object is a **physical** entity.	Class is a **logical** entity.
Object is created through **new keyword** mainly e.g. Student s1=new Student();	Class is declared using **class keyword** e.g. class Student{}
Object is created **many times** as per requirement.	Class is declared **once**.
Object **allocates memory when it is created**.	Class **doesn't allocated memory when it is created**.
There are **many ways to create object** in java such as new keyword, newInstance() method, clone() method, factory method and deserialization.	There is only **one way to define class** in java using class keyword.

b) When do we declare a method as abstract? Answer with an example.

Answer: Abstract classes are classes that contain one or more abstract methods. An abstract method is a method that is declared, but contains no implementation. Abstract classes may not be instantiated, and require subclasses to provide implementations for the abstract methods. Let's look at an example of an abstract class, and an abstract method.

Suppose, we were modeling the behavior of animals, by creating a class hierarchy that started with a base class called Animal. Animals are capable of doing different things like flying, digging and walking, but there are some common operations as well like eating and sleeping and making noise. Some common operations are performed by all animals, but in a different way as well. When an operation is performed in a different way, it is a good candidate for an abstract method (forcing subclasses to provide a custom implementation). Let's look at a very primitive Animal base class, which defines an abstract method for making a sound (such as a dog barking, a cow mooing, or a pig oinking).

```
public abstract Animal {
        public void sleep{
                // sleeping time
        }
        public void eat(food){
                //eat something
        }
        public abstract void makeNoise();
}
public Dog extends Animal {
        public void makeNoise() {
                System.out.println("Bark! Bark!");
         }
}
public Cow extends Animal {
        public void makeNoise() {
                 System.out.println("Moo! Moo!");
         }
}
```

12. a) Distinguish between InputStream and Reader classes.

Answer:

InputStream	**Reader**
Stream is **Byte Based**, it can be used to **read** bytes or write bytes.	Reader is **Character Based**, it can be used to **read** or write characters.
Stream is used to binary input/output	Reader is used to character input/output
InputStream is **Byte Based**, it can be used to read bytes.	Reader is **Character Based**, it can be used to read characters.
InputStream is used for reading binary files.	Reader is used for reading text files in **platform default encoding.**

InputStream and ObjectInputStream can be used for Serialization and DeSerialization, where serialized object can be persisted in file. In Serialization object is converted into byte stream and in deserialization it is converted back from byte to object.	Reader is not used for Serialization and DeSerialization, as it reads characters not bytes.
FileInputStream.read() reads 1 byte (8-bit) at a time.	FileReader.read() reads 2 bytes(16-bit) at a time, because char is 16-bit data type.
FileInputStream must be used when we are reading audio, video or other multimedia files.	FileReader must be used when we are reading text files, pdfs or word documents.

b) Write a program that will count the number of characters in a file.
Answer:

test.txt
Example String.

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStreamReader;

public class Tester {

  private static final String FILE_PATH = "test.txt";
  public static void main(String args[]) throws IOException {
    FileUtil fileUtil = new FileUtil(FILE_PATH);
    System.out.println("No. of characters in file: " + fileUtil.getCharCount());
  }
}

class FileUtil {
  static BufferedReader reader = null;
  public FileUtil(String filePath) throws FileNotFoundException {
    File file = new File(filePath);
    FileInputStream fileStream = new FileInputStream(file);
    InputStreamReader input = new InputStreamReader(fileStream);
    reader = new BufferedReader(input);
```

```
  }

  public static int getCharCount() throws IOException {
    int charCount = 0;
    String data;
    while((data = reader.readLine()) != null) {
      charCount += data.length();
    }
    return charCount;
  }
}
```

Output:

No. of characters in file: 15