**Experiment No: 7.1**

**Aim:**

|  |
| --- |
| Write a program to demonstrate the use of File class methods. |

**Code:**

|  |
| --- |
| import java.io.\*;  import java.util.Arrays;  public class main71 {  public static void main(String [] as) throws IOException {  File ob=new File("D:\\New Folder\\as.txt");  //throws IOException  System.*out*.println("File exists?="+ob.exists());  System.*out*.println("Path of File specified"+ob.getAbsolutePath());  System.*out*.println("Is it a directory?="+ob.isDirectory());  System.*out*.println("Is it a File?="+ob.isFile());  System.*out*.println("Lenght of the File="+ob.length());  File ob1=new File("G:\\New Folder");  System.*out*.println(ob1.mkdir());  File ob2=new File("G:\\aaaa\\dd.txt");  System.*out*.println(ob2.createNewFile());  String as1[]=ob1.list();  System.*out*.println(Arrays.*toString*(as1));  }  } |

**Output:**

|  |
| --- |
| File exists?=true  Path of File specifiedD:\New Folder\as.txt  Is it a directory?=false  Is it a File?=true  Lenght of the File=17  false  false  [allnames.txt, appeared.txt, asd.txt, Destination.txt, file1.txt, names.txt, noduplicates.txt, notselected.txt, qwe.txt] |

**Experiment No: 7.2**

**Aim:**

|  |
| --- |
| Write a program to demonstrate the use of FileOutputStream and FileInputStream to create a copy of any file selected from a particular folder. |

**Code:**

|  |
| --- |
| import java.io.\*;  public class main72 {  public static void main(String as[]) throws FileNotFoundException,IOException{  File f1= new File("G:\\file1.txt");  f1.createNewFile();  FileWriter ob= new FileWriter(f1);  ob.write("what are u doing?");  ob.close();  int a;  File f2=new File("G:\\New Folder\\Destination.txt");  FileInputStream fis=new FileInputStream(f1);  FileOutputStream fos=new FileOutputStream(f2);  while( (a=fis.read())!=-1 ) {  fos.write((char)a);  System.*out*.print((char)a);  }  }  } |

**Output:**

|  |
| --- |
| what are u doing? |

**Experiment No: 7.3**

**Aim:**

|  |
| --- |
| Write a program to demonstrate the use of FilterOutputStream to store the characters coming from keyboard (input device) into a file in an encrypted format and use FilterInputStream to read the encrypted contents and display decripted contents in the output screen. |

**Code:**

|  |
| --- |
| import java.io.\*;  import java.util.\*;  public class main73 {  public static void main(String[] args) throws IOException,ClassNotFoundException {  Scanner ip=new Scanner(System.in);  int a;  String s=new String();  System.out.println("Enter string");  s=ip.nextLine();  File file = new File("D:\\testout.txt");  FileOutputStream fos=new FileOutputStream(file);  ObjectOutputStream oos=new ObjectOutputStream(fos);  FilterOutputStream fios = new FilterOutputStream(oos);  byte ar[]=s.getBytes();  fios.write(ar);  fios.close();  fos.close();  FileInputStream fis=new FileInputStream(file);  ObjectInputStream ois=new ObjectInputStream(fis);  FilterInputStream fiis = new BufferedInputStream(ois);  while( (a=fiis.read())!=-1 ) {  System.out.print((char)a);  } } } |

**Output:**

|  |
| --- |
| Enter string  shell programming  shell programming |

**Experiment No: 7.4**

**Aim:**

|  |
| --- |
| Write a program to merge the contents of text files T1.txt and T2.txt into T3.txt. The contents of T1.txt should appear first and then T2.txt in the destination file T3.txt. |

**Code:**

|  |
| --- |
| import java.io.\*;  public class main74 {  public static void main(String as[]) throws FileNotFoundException,IOException  File ob=new File("Input.txt");  ob.createNewFile();  FileWriter fw1=new FileWriter(ob);  fw1.write("Write a program to demonstrate the use");  fw1.flush();  File ob1=new File("Input1.txt");  ob1.createNewFile();  fw1=new FileWriter(ob1);  fw1.write("of File handling");  fw1.flush();  File gg=new File("Output.txt");  int ch;  FileReader fr=new FileReader(ob);  FileReader fr1=new FileReader(ob1);  fw1=new FileWriter(gg);  while((ch=fr.read())!=-1) {  System.*out*.print((char)ch);  fw1.write(ch);  }  fw1.flush();  while((ch=fr1.read())!=-1) {  System.*out*.print((char)ch);  fw1.write(ch);  }  fw1.flush();  }} |

**Output:**

|  |
| --- |
| Write a program to demonstrate the useof File handling |

**Experiment No: 7.5**

**Aim:**

|  |
| --- |
| Write a program to merge the contents of text files T1.txt and T2.txt in alternate fashion into destination file T3.txt. Alternatively one line from T1.txt then one line from T2.txt and so on should be written into T3.txt. |

**Code:**

|  |
| --- |
| import java.io.\*;  public class main75 {    public static void main(String as[]) throws FileNotFoundException,IOException {    BufferedReader fr1=new BufferedReader(new FileReader("Input1.txt"));  /\*input1 text is  Write  a  java  program  \*/  BufferedReader fr2=new BufferedReader(new FileReader("Input2.txt"));  /\*Input2 text is  to demonstate  use  of  File  Handling  \*/  BufferedWriter bw1=new BufferedWriter(new FileWriter("Output.txt"));  FileWriter fw1=new FileWriter("Output.txt");  String linedat=new String();  String linedat1=new String();  while( (linedat=fr1.readLine())!=null && (linedat1=fr2.readLine())!=null ) {    System.*out*.println(linedat);  System.*out*.println(linedat1);  fw1.write(linedat);  fw1.write(linedat1);  }  while( (linedat=fr1.readLine())!=null) {    System.*out*.println(linedat);  fw1.write(linedat);  }  while( (linedat1=fr2.readLine())!=null) {  System.*out*.println(linedat1);  fw1.write(linedat1);  }  fw1.close();  }  } |

**Output:**

|  |
| --- |
| Write  to demonstate  a  use  java  of  program  File  Handling |

**Experiment No: 7.6**

**Aim:**

|  |
| --- |
| Write a program to merge the contents of all the text files present in a folder (one file content at time) into the destination text file AllMerged.txt. |

### Code:

|  |
| --- |
| import java.io.\*;  class main76 {  public static void main(String[] args) throws IOException  {  File dir = new File("G:\\New folder");  PrintWriter pw = new PrintWriter("output.txt");  String[] fileNames = dir.list();  for (String fileName : fileNames) {  System.*out*.println("Reading from " + fileName);  File f = new File(dir, fileName);  BufferedReader br = new BufferedReader(new FileReader(f));  pw.println("Contents of file " + fileName);  String line = br.readLine();  while (line != null) {  pw.println(line);  line = br.readLine();  }  pw.flush();  }  }  } |

**Output:**

|  |
| --- |
| Reading from allnames.txt  Reading from appeared.txt  Reading from asd.txt  Reading from Destination.txt  Reading from names.txt  Reading from noduplicates.txt  Reading from notselected.txt  Reading from qwe.txt |

**Experiment No: 7.7**

**Aim:**

|  |
| --- |
| Write a program to search a name present in a text file named Names.txt. Also print if name not found. |

**Code:**

|  |
| --- |
| import java.io.\*;  import java.util.\*;  class urban {  public static void main(String[] args) throws IOException  {  Scanner ip=new Scanner(System.in);  System.out.println("Enter name to be searched");  String s1=ip.next();  boolean flag=false;  int line=1;  String as =new String();  File dir = new File("G:\\New folder");  PrintWriter pw = new PrintWriter("output.txt");  File f=new File(dir,"names.txt");  System.out.println(f.getAbsolutePath());  BufferedReader br=new BufferedReader(new FileReader(f));  while( (as=br.readLine())!=null ) {  if (as.equalsIgnoreCase(s1)) {  flag=true;  break;  }  System.out.println();  line++;  }  if (flag) {  System.out.println("Name Found at line "+line);  }  else {  System.out.println("Name not Found");  }  }  } |

**Output:**

|  |
| --- |
| Enter name to be searched  golem  G:\New folder\names.txt  Name Found at line 3 |

**Experiment No: 7.8**

**Aim:**

|  |
| --- |
| Write a program to delete a name present in a text file named Names.txt and create another file named NewNames.txt containing those names which are not deleted. |

**Code:**

|  |
| --- |
| import java.io.\*;  import java.util.\*;  public class urban2 {  public static void main(String[] args) throws IOException  {  Scanner ip=new Scanner(System.in);  System.out.println("Enter name to be deleted");  String s1=ip.next();  boolean flag=false;  int line=1,linecount=0;  String as =new String();  File dir = new File("G:\\New folder");  File f=new File(dir,"names.txt");  System.out.println(f.getAbsolutePath());  BufferedReader br=new BufferedReader(new FileReader(f));  BufferedWriter bw=new BufferedWriter(new FileWriter("NewNames.txt"));  File op=null;  while( (as=br.readLine())!=null ) {  if (!as.equalsIgnoreCase(s1)) {  bw.write(as+"\n");  }  line++;  }  bw.flush();  f=new File("NewNames.txt");  System.out.println(f.getAbsolutePath());  }  } |

**Output:**

|  |
| --- |
| Enter name to be deleted  golem  G:\New folder\names.txt  C:\Users\HIMANSHU\eclipse-workspace\7.8\NewNames.txt |

**Experiment No: 7.9**

**Aim:**

|  |
| --- |
| Write a program to implement the following logic. There are two files, one file is Appeared.txt which contains names of all candidates appeared for an interview. Another file NotSelected.txt contains names of those candidates who could not make through that interview. Create a third file named Selected.txt which should contain the names of all those candidates who got selected in the interview. |

**Code:**

|  |
| --- |
| import java.io.\*;  import java.util.\*;  class A{  void ff() throws FileNotFoundException,IOException{  String as=new String();  String as1=new String();    File f1=new File("G:\\New Folder");  BufferedReader br1=new BufferedReader(new FileReader(new File(f1,"appeared.txt")));    BufferedWriter bw=new BufferedWriter(new FileWriter("selected.txt"));      while( (as=br1.readLine())!=null ) {  BufferedReader br2=new BufferedReader(new FileReader(new File(f1,"notselected.txt")));  boolean flag=true;  while( (as1=br2.readLine())!=null ){  if(as.equalsIgnoreCase(as1)) {  flag=false;  break;  }  }  if(flag==true) {  System.out.println(as);  bw.write(as+"\n");  }  }  bw.close();  }  }  public class main79 {  public static void main(String as[]) throws FileNotFoundException,IOException {  A a=new A();  a.ff();  }  } |

**Output:**

|  |
| --- |
| Brad Hawk  Chris  Vera Ross  Linfong  Tong Yoon  Paul Phoenix |

**Experiment No: 7.10**

**Aim:**

|  |
| --- |
| Write a program to remove duplicate names present in a file named AllNames.txt and store the unique names to another file named NoDuplicates.txt. |

**Code:**

|  |
| --- |
| import java.io.\*;  import java.util.\*;  class A{  void ff() throws FileNotFoundException,IOException{  String as=new String();  String as1=new String();  File f1=new File("G:\\New Folder");  File f2=new File(f1,"allnames.txt");  File f3=new File(f1,"noduplicates.txt");    BufferedReader br1=new BufferedReader(new FileReader(f2));  BufferedWriter bw=new BufferedWriter(new FileWriter(f3));  int i=1;    while(i<2) {  as1=br1.readLine();  bw.write(as1+"\n");  System.out.println(as1);  i++;  bw.flush();  }  while( (as=br1.readLine())!=null ) {  boolean flag=true;  BufferedReader br2=new BufferedReader(new FileReader(f3));  while( (as1=br2.readLine())!=null ){  if (as.equalsIgnoreCase(as1)) {  flag=false;  break;  }  }  if (flag==true) {  bw.write(as+"\n");  System.out.println(as);  bw.flush();  }  }  }  }  public class main710 {  public static void main(String as[]) throws FileNotFoundException,IOException {  A a=new A();  a.ff();  }  } |

**Output:**

|  |
| --- |
| jake  blake  alex  austin |