15. JAVA IO

1.

```
import java.util.*;
import org.apache.commons.io.IOUtils;
class Main
{
   public static void main(String[] args)
   {
      File file = new file("/Users/abc/Desktop/Folder/test.txt");
      try
      {
            FileInputStream input = new FileInputStream(file);
            String contents = IOUtils.toString(input);
            System.out.println(contents);
      }
      catch (IOException e)
      {
         e.printStackTrace();
      }
   }
}
```

2.

```
import java.io.*;
class Main
{
   public static void main(String[] args) throws IOException
   {
      int i;
      FileOutputStream f = new FileOutputStream("../files/name.txt",true);
      String str = "ABC";
      char ch[] = str.toCharArray();
      for (i = 0; i < str.length(); i++)
      {
         f.write(ch[i]);
      }
      f.close();
   }
}
```

3.

```
import java.io.*;
public class Main
{
   public static void main(String[] args) throws Exception
   {
```

```
      FileInputStream inputStream = new FileInputStream("c:/demo.txt");
      BufferedInputStream buffInputStr = new BufferedInputStream(inputStream);
      while (buffInputStr.available() > 0)
      {
         char c = (char)buffInputStr.read();
         System.out.println("Char : " + c);
      }
   }
}
```

4.

```
import java.io.*;

class Main
{
   public static void main(String args[])throws Exception
   {
      FileOutputStream fout = new FileOutputStream("f1.txt");
      BufferedOutputStream bout = new BufferedOutputStream(fout);
      for(int i = 10; i < 25; i++)
      {
         bout.write(i);
      }
      byte b[] = { 7, 6, 17, 8, 9, 20 };
      bout.write(b);
      bout.flush();
      bout.close();
      fout.close();
   }
}
```

5.

```
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
class Main
{
   public static void main(String[] args) throws IOException
   {
      int ch;
      FileReader fr=null;
      try
      {
         fr = new FileReader("text");
      }
      catch (FileNotFoundException fe)
      {
         System.out.println("File not found");
```

```
      }
      while ((ch=fr.read())!=-1)
         System.out.print((char)ch);
      fr.close();
   }
}
```

6.

```
import java.io.FileWriter;
import java.io.IOException;
class Main
{
   public static void main(String[] args) throws IOException
   {
      String str = "File Handling in Java using "+" FileWriter and FileReader";
      FileWriter fw=new FileWriter("output.txt");
      for (int i = 0; i < str.length(); i++)
         fw.write(str.charAt(i));
      System.out.println("Writing successful");
      fw.close();
   }
}
```

7.

```
import java.io.*;

public class Main
{
   public static void main(String[] args)
   {
      FileReader fileReader = new FileReader("c:/demo.txt");
      BufferedReader buffReader= new BufferedReader(fileReader);
      while (buffReader.ready())
     {
         System.out.println("Char :"+ (char)buffReader.read());
      }
   }
}
```

8.

```
import java.io.*;
public class Main
{
   public static void main(String[] args)
   {
      FileWriter abc_file;
      try
      {
```

```
            abc_file = new FileWriter("ABC.txt");
            Main bwrite = new Main(abc_file);
            System.out.println("Buffered Writer start writing :)");
            bwrite.write(6);
            bwrite.write(4);
            bwrite.close();
            System.out.println("Written successfully");
        }
        catch (IOException excpt)
        {
            excpt.printStackTrace();
        }

    }
}
```

9.

```
package com;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.ResourceBundle;

public class Main
{
  public static Connection getConn()
    {
        ResourceBundle rd = ResourceBundle.getBundle("system");
        Connection con = null;
        String loadDriver = rd.getString("driver");
        String dbURL = rd.getString("url");
        String dbUSERNAME = rd.getString("userName");
        String dbPASSWORD = rd.getString("password");
        try
        {
            Class.forName(loadDriver);
            con = DriverManager.getConnection(dbURL, dbUSERNAME, dbPASSWORD);
            Statement st = con.createStatement();
            ResultSet rs = st.executeQuery("SELECT * FROM CUSTOMER");
            while (rs.next()) {
                System.out.println("ID -" + rs.getInt(1) + " || "+ "First-Name -" + rs.getString(2)+ " || "+
"LastName -" + rs.getString(4));
            }
        }
        catch (ClassNotFoundException e)
        {
            e.printStackTrace();
        }
```

```
      catch (SQLException e)
     {
        e.printStackTrace();
      }
       return con;
   }
   public static void main(String[] args)
   {
      Main.getConn();
   }
}
```

10.

```
public class Main
{
   public static void main(String[] args)
   {
      try
      {
         FileInputStream file = new FileInputStream(new File("abc.xlsx"));
         XSSFWorkbook workbook = new XSSFWorkbook(file);
         XSSFSheet sheet = workbook.getSheetAt(0);
         Iterator<Row> rowIterator = sheet.iterator();
         while (rowIterator.hasNext())
        {
            Row row = rowIterator.next();
            Iterator<Cell> cellIterator = row.cellIterator();
            while (cellIterator.hasNext())
            {
               Cell cell = cellIterator.next();
               switch (cell.getCellType())
               {
                  case Cell.CELL_TYPE_NUMERIC:
                        System.out.print(cell.getNumericCellValue() + "t");
                        break;
                  case Cell.CELL_TYPE_STRING:
                         System.out.print(cell.getStringCellValue() + "t");
                         break;
               }
            }
            System.out.println("");
         }
         file.close();
      }
      catch (Exception e)
     {
         e.printStackTrace();
      }
   }
}
```

11.

```
public class Main
{
    public static void main(String[] args)
   {
      XSSFWorkbook workbook = new XSSFWorkbook();
      XSSFSheet sheet = workbook.createSheet("student Details");
      Map<String, Object[]> data = new TreeMap<String, Object[]>();
      data.put("1", new Object[]{ "ID", "NAME", "LASTNAME" });
      data.put("2", new Object[]{ 1, "Kiran", "Bedi" });
      data.put("3", new Object[]{ 2, "M.S.", "Subbalakshmi" });
      data.put("4", new Object[]{ 3, "Sachin", "Tendulkar" });
      data.put("5", new Object[]{ 4, "Virat", "kohli" });
      Set<String> keyset = data.keySet();
      int rownum = 0;
      for (String key : keyset)
     {
         Row row = sheet.createRow(rownum++);
         Object[] objArr = data.get(key);
         int cellnum = 0;
         for (Object obj : objArr)
        {
            Cell cell = row.createCell(cellnum++);
            if (obj instanceof String)
            cell.setCellValue((String)obj);
            else if (obj instanceof Integer)
             cell.setCellValue((Integer)obj);
         }
      }
      try
     {
         FileOutputStream out = new FileOutputStream(new File("abc.xlsx"));
         workbook.write(out);
         out.close();
         System.out.println("abc.xlsx written successfully on disk.");
      }
      catch (Exception e)
     {
         e.printStackTrace();
      }
   }
}
```