1. SERIALIZATION AND DESERIALIZATION

CODE:

```
package javaapplication14;

import java.io.*;

// Define the Student class that implements Serializable
class Student implements java.io.Serializable {
   public String stu_Name;
   public String stu_Addr;
   public int stu_Id;
}

public class JavaApplication14 {

   public static void main(String[] args) {
      // Create a Student object
      Student std = new Student();
      std.stu_Name = "dharani mohan";
      std.stu_Addr = "ABC,riya";
      std.stu_Id = 192324008;

      // To hold the deserialized byte-stream
      Student deserializedStudent;

      try {
         // Serializing the student object - std
         FileOutputStream fileOut = new
FileOutputStream("C:/JavaProgramming/gameData/Highscores.txt");
         ObjectOutputStream out = new ObjectOutputStream(fileOut);
```

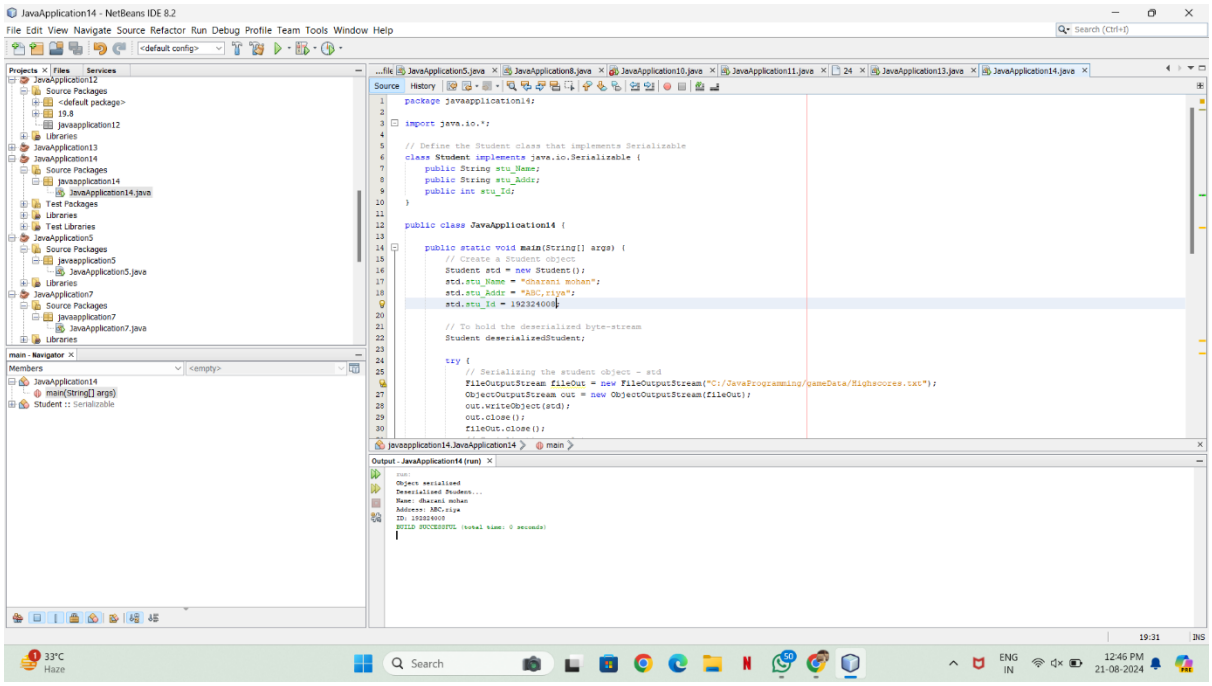
```
            out.writeObject(std);
            out.close();
            fileOut.close();
            // Serialization complete
            System.out.printf("Object serialized\n");

            // Deserialization process
            FileInputStream fileIn = new
FileInputStream("C:/JavaProgramming/gameData/Highscores.txt");
            ObjectInputStream in = new ObjectInputStream(fileIn);

            // Deserialization
            deserializedStudent = (Student) in.readObject();
            in.close();
            fileIn.close();

            // Printing the deserialized object
            System.out.println("Deserialized Student...");
            System.out.println("Name: " + deserializedStudent.stu_Name);
            System.out.println("Address: " + deserializedStudent.stu_Addr);
            System.out.println("ID: " + deserializedStudent.stu_Id);
        } catch (IOException i) {
            i.printStackTrace();
        } catch (ClassNotFoundException c) {
            System.out.println("Class not found");
            c.printStackTrace();
        }
    }
}
```

OUTPUT:



2. PRINT WRITER

CODE:

```
package javaapplication16;


import java.io.*;


// Define the Student class that implements Serializable
class Student implements java.io.Serializable {
    public String stu_Name;
    public String stu_Addr;
```

```
    public int stu_Id;
}

public class JavaApplication16 {

    public static void main(String[] args) {
        // Create a Student object
        Student std = new Student();
        std.stu_Name = "dharani mohan";
        std.stu_Addr = "ABC,riya";
        std.stu_Id = 192324008;

        // To hold the deserialized byte-stream
        Student deserializedStudent;

        try {
            // Serializing the student object - std
            FileOutputStream fileOut = new
FileOutputStream("C:/JavaProgramming/gameData/Highscores.txt");
            ObjectOutputStream out = new ObjectOutputStream(fileOut);
            out.writeObject(std);
            out.close();
            fileOut.close();
            // Serialization complete
            System.out.printf("Object serialized\n");

            // Deserialization process
            FileInputStream fileIn = new
FileInputStream("C:/JavaProgramming/gameData/Highscores.txt");
            ObjectInputStream in = new ObjectInputStream(fileIn);

            // Deserialization
```

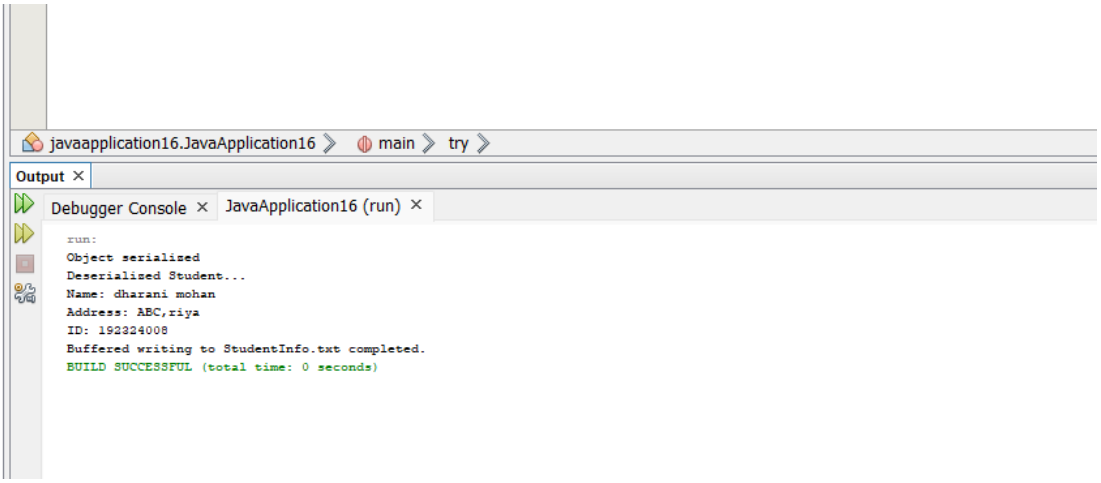
```
            deserializedStudent = (Student) in.readObject();
            in.close();
            fileIn.close();

            // Printing the deserialized object
            System.out.println("Deserialized Student...");
            System.out.println("Name: " + deserializedStudent.stu_Name);
            System.out.println("Address: " + deserializedStudent.stu_Addr);
            System.out.println("ID: " + deserializedStudent.stu_Id);

            // Buffered Writing Example
            try (BufferedWriter bufferedWriter = new BufferedWriter(new
FileWriter("C:/JavaProgramming/gameData/StudentInfo.txt"))) {
                bufferedWriter.write("Name: " + deserializedStudent.stu_Name + "\n");
                bufferedWriter.write("Address: " + deserializedStudent.stu_Addr + "\n");
                bufferedWriter.write("ID: " + deserializedStudent.stu_Id + "\n");
            }
            System.out.println("Buffered writing to StudentInfo.txt completed.");

        } catch (IOException i) {
            i.printStackTrace();
        } catch (ClassNotFoundException c) {
            System.out.println("Class not found");
            c.printStackTrace();
        }
    }
}
```

OUTPUT:



3. FILE WRITER

CODE:

OUTPUT:

4. WRITER

CODE:

OUTPUT:

5. BUFFER WRITER

CODE:

OUTPUT: