

Task 2: Serialization and Deserialization

Serialize a custom object to a file and then deserialize it back to recover the object state.

ANS:

```
package com.Day25;
import java.io.*;
public class Task2 {
    public static void main(String[] args) {
        // Create a Person object
        Person person = new Person("John Doe", 30);
        String filename = "person.ser";
        // Serialize the Person object to a file
        try (FileOutputStream fileOut = new FileOutputStream(filename);
            ObjectOutputStream out = new ObjectOutputStream(fileOut)) {
            out.writeObject(person);
            System.out.println("Serialized data is saved in " + filename);
        } catch (IOException i) {
            i.printStackTrace();
        }
        // Deserialize the Person object from the file
        Person deserializedPerson = null;
        try (FileInputStream fileIn = new FileInputStream(filename);
            ObjectInputStream in = new ObjectInputStream(fileIn)) {
            deserializedPerson = (Person) in.readObject();
        } catch (IOException | ClassNotFoundException i) {
            i.printStackTrace();
        }
        // Print the deserialized Person object
        System.out.println("Deserialized Person: " + deserializedPerson);
    }
    // The Person class implements Serializable
    static class Person implements Serializable {
        private static final long serialVersionUID = 1L;
        private String name;
        private int age;
        public Person(String name, int age) {
            this.name = name;
            this.age = age;
        }
        // Getters
        public String getName() {
            return name;
        }
        public int getAge() {
            return age;
        }
        // toString method for easy printing
        @Override
        public String toString() {
            return "Person{name=\"" + name + "\", age=" + age + "\"}";
        }
    }
}
```

