System saving and loading a game using Serializable   
  
import java.io.\*;

import java.util.Scanner;

class GameData implements Serializable {

    private static final long serialVersionUID = 1L;

    int level;

    int score;

    String playerName;

    public GameData(String playerName, int level, int score) {

        this.playerName = playerName;

        this.level = level;

        this.score = score;

    }

    public void display() {

        System.out.println("Player Name: " + playerName);

        System.out.println("Level: " + level);

        System.out.println("Score: " + score);

    }

}

public class GameSystem {

    private static final String SAVE\_FILE = "gameData.ser";

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        GameData data = null;

        int choice;

        do {

            System.out.println("\n===== GAME SYSTEM =====");

            System.out.println("1. New Game");

            System.out.println("2. Save Game");

            System.out.println("3. Load Game");

            System.out.println("4. Exit");

            System.out.print("Enter choice: ");

            choice = sc.nextInt();

            sc.nextLine();

            switch (choice) {

                case 1:

                    System.out.print("Enter Player Name: ");

                    String name = sc.nextLine();

                    System.out.print("Enter Level: ");

                    int level = sc.nextInt();

                    System.out.print("Enter Score: ");

                    int score = sc.nextInt();

                    data = new GameData(name, level, score);

                    System.out.println("New game started!");

                    break;

                case 2:

                    if (data != null) {

                        saveGame(data);

                    } else {

                        System.out.println("No game data to save!");

                    }

                    break;

                case 3:

                    data = loadGame();

                    if (data != null) {

                        System.out.println("\nGame Loaded Successfully!");

                        data.display();

                    } else {

                        System.out.println("No saved game found!");

                    }

                    break;

                case 4:

                    System.out.println("Exiting game...");

                    break;

                default:

                    System.out.println("Invalid choice. Try again.");

            }

        } while (choice != 4);

        sc.close();

    }

    private static void saveGame(GameData data) {

        try (ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream(SAVE\_FILE))) {

            oos.writeObject(data);

            System.out.println("Game saved successfully!");

        } catch (IOException e) {

            System.out.println("Error saving game: " + e.getMessage());

        }

    }

    private static GameData loadGame() {

        try (ObjectInputStream ois = new ObjectInputStream(new FileInputStream(SAVE\_FILE))) {

            return (GameData) ois.readObject();

        } catch (IOException | ClassNotFoundException e) {

            System.out.println("Error loading game: " + e.getMessage());

            return null;

        }

    }

}

Output:

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 1

Enter Player Name: Sanjay

Enter Level: 2

Enter Score: 1000

New game started!

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 3

Error loading game: null

No saved game found!

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 3

Error loading game: null

No saved game found!

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 2

No game data to save!

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 1

Enter Player Name: Sanjay

Enter Level: 1

Enter Score: 1000

New game started!

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 2

Game saved successfully!

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: 3

Game Loaded Successfully!

Player Name: Sanjay

Level: 1

Score: 1000

===== GAME SYSTEM =====

1. New Game

2. Save Game

3. Load Game

4. Exit

Enter choice: