**TASK 2**

Write a program that asks for the names of three runners and the time, in minutes, it took each of them to finish a race. The program should display the names of the runners in the order that they finished.

Flowchart For Task 2 (page 2)

Diagram

Description automatically generated

Code For Task 2

**import** javax.swing.JOptionPane; //dialog box

**import** javax.swing.ImageIcon; //icon for dialog box

**public** **class** A1\_Task2Dialog {

**public** **static** **void** main(String[] args) {

//setting part start

String name, name2, name3;

**double** time, time2, time3;

ImageIcon trophy= **new** ImageIcon("src/trophy.png"), error = **new** ImageIcon("src/error2.png");

ImageIcon swimmer= **new** ImageIcon("src/swimmer.png"), watch= **new** ImageIcon("src/stopwatch.png");

ImageIcon lie = **new** ImageIcon("src/lie.jpg"), ugh = **new** ImageIcon("src/123.png");

//setting part end

//try-catch block start

**try** {

//Enter input

name = (String)JOptionPane.*showInputDialog*(**null**, "Enter first runner name.", "Record",JOptionPane.***INFORMATION\_MESSAGE***, swimmer, **null**, " ");

time = Double.*parseDouble*((String)JOptionPane.*showInputDialog*(**null**, "Enter time taken by runner to finish the race in minutes.", "Record",JOptionPane.***INFORMATION\_MESSAGE***, watch, **null**, " "));

name2 = (String)JOptionPane.*showInputDialog*(**null**, "Enter second runner name.", "Record",JOptionPane.***INFORMATION\_MESSAGE***, swimmer, **null**, " ");

time2 = Double.*parseDouble*((String)JOptionPane.*showInputDialog*(**null**, "Enter time taken by runner to finish the race in minutes.", "Record",JOptionPane.***INFORMATION\_MESSAGE***, watch, **null**, " "));

name3 = (String)JOptionPane.*showInputDialog*(**null**, "Enter third runner name.", "Record",JOptionPane.***INFORMATION\_MESSAGE***, swimmer, **null**, " ");

time3 = Double.*parseDouble*((String)JOptionPane.*showInputDialog*(**null**, "Enter time taken by runner to finish the race in minutes.", "Record",JOptionPane.***INFORMATION\_MESSAGE***, watch, **null**, " "));

//nested if...else statement start

//Determine the order of the runners

**if** (time == 0 || time2 == 0 || time3 == 0) {

JOptionPane.*showMessageDialog*(**null**, "Don't fake the result.", "Zero Time Taken Error", JOptionPane.***INFORMATION\_MESSAGE***, lie);

}

**else** **if** (time < 0 || time2 < 0 || time3 < 0) {

JOptionPane.*showMessageDialog*(**null**, "Time taken cannot be negative,thank you.", "Negative Time Taken Error", JOptionPane.***PLAIN\_MESSAGE***, ugh);

}

**else** **if** (time < time2 && time < time3) {

**if** (time2 > time3)

JOptionPane.*showMessageDialog*(**null**, "LEADERBOARD" + "\n1ˢᵗ " + name + "\n2ⁿᵈ " + name3 + "\n3ʳᵈ " + name2, "RESULT", JOptionPane.***INFORMATION\_MESSAGE***, trophy );

**else**

JOptionPane.*showMessageDialog*(**null**, "LEADERBOARD" + "\n1ˢᵗ " + name + "\n2ⁿᵈ " + name2 + "\n3ʳᵈ " + name3, "RESULT", JOptionPane.***INFORMATION\_MESSAGE***, trophy);

}

**else** **if** (time > time2 && time < time3) {

JOptionPane.*showMessageDialog*(**null**, "LEADERBOARD" + "\n1ˢᵗ " + name2 + "\n2ⁿᵈ " + name + "\n3ʳᵈ " + name3, "RESULT", JOptionPane.***INFORMATION\_MESSAGE***, trophy);

}

**else** **if** (time < time2 && time > time3) {

JOptionPane.*showMessageDialog*(**null**, "LEADERBOARD" + "\n1ˢᵗ " + name3 + "\n2ⁿᵈ " + name + "\n3ʳᵈ " + name2, "RESULT", JOptionPane.***INFORMATION\_MESSAGE***, trophy);

}

**else** {

**if**(time2 < time3)

JOptionPane.*showMessageDialog*(**null**, "LEADERBOARD" + "\n1ˢᵗ " + name2 + "\n2ⁿᵈ " +name3 + "\n3ʳᵈ " +name, "RESULT", JOptionPane.***INFORMATION\_MESSAGE***, trophy);

**else**

JOptionPane.*showMessageDialog*(**null**, "LEADERBOARD" + "\n1ˢᵗ " + name3 + "\n2ⁿᵈ " +name2 + "\n3ʳᵈ " +name, "RESULT", JOptionPane.***INFORMATION\_MESSAGE***, trophy);

}

//nested if...else statement end

}**catch**(NumberFormatException e) {

JOptionPane.*showMessageDialog*(**null**, "Please input a number.\nSpacing is not allowed.", "(ㆆ\_ㆆ)",JOptionPane.***ERROR\_MESSAGE***, error);

//informing user about error

}

//try-catch block end

}

}

**Outputs For Task 2**

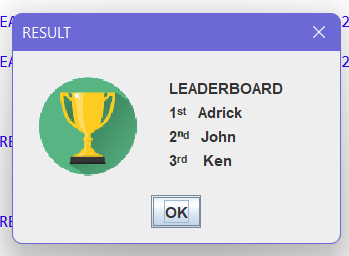
1. **Normal:**

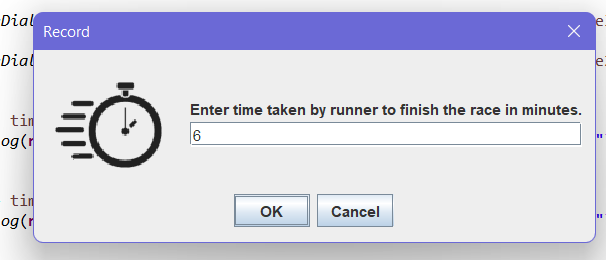
Graphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated****Graphical user interface, application

Description automatically generated****Graphical user interface, application

Description automatically generated

Program ends

1. **Error 1 (User enters alphabets or left it blank when inputting time taken)**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface

Description automatically generated**Graphical user interface, application

Description automatically generated**Graphical user interface, application, Teams

Description automatically generated**

Program ends

1. **Error 2 (User entered 0 for time taken)**

**Graphical user interface, application

Description automatically generatedGraphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated**

**Graphical user interface

Description automatically generated**

User enters 0

**A person wearing glasses

Description automatically generated with low confidence**

Program ends

1. **Error 3 (User enters negative number for time taken)**

**Graphical user interface, application

Description automatically generatedGraphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

User enters -8

**Graphical user interface, application, chat or text message

Description automatically generated**

Program ends