JAVA PROGRAMMING - CS6308

LAB ASSIGNMENT - 8

BHARATH M

2019503509

1. Write a program to create a simple game application of your choice and that support the following actions as follows.

- Create field for player ID, player Name, status (win/lose count), sign in (new player), log in (existing player).
- Write a method to generate unique player ID.
- Write a method that stores all the details of the player as Object in a file.
- Write a method that display the player details based on player ID.
- Write a method to start the game and to close the game.

Code :

window.java

```
import java.awt.event.*;
public class window extends WindowAdapter
{
   public void windowClosing(WindowEvent we)
   {
      System.exit(0);
```

```
    }
}
```

player.java

```
import java.io.Serializable;
class player implements Serializable {
    public  int PLayerId;
    public  String PlayerName;
    public  boolean Result;
    public String status;
}
```

HAndCricket.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.*;
public class HAndCricket implements ActionListener {
    JFrame f ;
    JPanel p1,p3,p4;
    JButton b1,b2,b3,b4;
    JTextField t1,t2;
    player p,p2;
    ImageIcon icon1 = new ImageIcon("E:\\project
files\\img\\icon1.png");
    HAndCricket()
    {
        f = new JFrame();
        f.setLayout(null);
        f.setSize(500,200);
        f.setTitle("HandCricket with Computer");
        f.setVisible(true);
        f.addWindowListener(new window());
        f.setResizable(false);
        p1 = new JPanel();
        p1.setLayout(null);
        p1.setBounds(0,0,500,300);
        p1.setVisible(true);
        p1.setBackground(Color.CYAN);
        b1 = new JButton("SIGNUP");
        b1.setBounds(25,25,100,20);
        p1.add(b1);
        p3 = new JPanel();
        p3.setLayout(null);
        p3.setBounds(150,25,300,    100);
        p3.setVisible(false);
```

```
p1.add(p3);
p3.setBackground(Color.GRAY);
JLabel l1 = new JLabel("Name ");
l1.setBounds(25,25,100,20);
p3.add(l1);
t1 = new JTextField();
t1.setBounds(150,25,100,20);
p3.add(t1);
b3 = new JButton("SIGNUP");
b3.setBounds(25,75,100,20);
p3.add(b3);
b3.addActionListener(this);
p4 = new JPanel();
p4.setLayout(null);
p4.setBounds(150,25,300,    100);
p4.setVisible(false);
p1.add(p4);
p4.setBackground(Color.GREEN);
    JLabel l2 = new JLabel("Id ");
l2.setBounds(25,25,100,20);
p4.add(l2);
t2 = new JTextField();
t2.setBounds(150,25,100,20);
```

```
    p4.add(t2);
    b4 = new JButton("LOGIN");
    b4.setBounds(25,75,100,20);
    p4.add(b4);
    b4.addActionListener(this);
    b1.addActionListener(new A1());
    b2 = new JButton("EXISTING");
    b2.setBounds(25,100,100,20);
    b2.addActionListener(new A1());
    p1.add(b2);
    f.add(p1);
}
class A1 implements ActionListener
{
    @Override
    public void actionPerformed(ActionEvent e) {
        String s = e.getActionCommand();
        if (s.equals("SIGNUP"))
        {
            p3.setVisible(true);
            p4.setVisible(false);
        }
        if (s.equals("EXISTING"))
```

```
            {
                p4.setVisible(true);
                p3.setVisible(false);
            }
        }
    }
    public static void main(String[] args) {
        new HAndCricket();
    }
     public void actionPerformed(ActionEvent e) {
        String s = e.getActionCommand();
        if (s.equals("SIGNUP")) {
            p = new player();
            p.PLayerId = unique();
            p.PlayerName = t1.getText();
            p.Result = false;
            p.status = "New";
            create(p);
        }
        else {
            try {
                FileInputStream fileIn = new
FileInputStream("player.txt");
```

```
            ObjectInputStream in = new
ObjectInputStream(fileIn);
            p2 = (player) in.readObject();
            in.close();
            fileIn.close();
         } catch (IOException | ClassNotFoundException i) {
            JOptionPane.showMessageDialog(null, i + "");
         }
         if(p2.PLayerId==Integer.parseInt(t2.getText()))
         {
            JOptionPane.showMessageDialog(null,"SUCCESS
\n");
            goingstart(p2);
         }
         else
            JOptionPane.showMessageDialog(null, "FAIL");
      }
   }
   private void goingstart(player p2) {
      String des[] = {"Start","CLose"};
      int x = JOptionPane.showOptionDialog(null,
"","Confirm",JOptionPane.DEFAULT_OPTION,
JOptionPane.INFORMATION_MESSAGE, icon1, des, des[0]);
      if(x==-1)
```

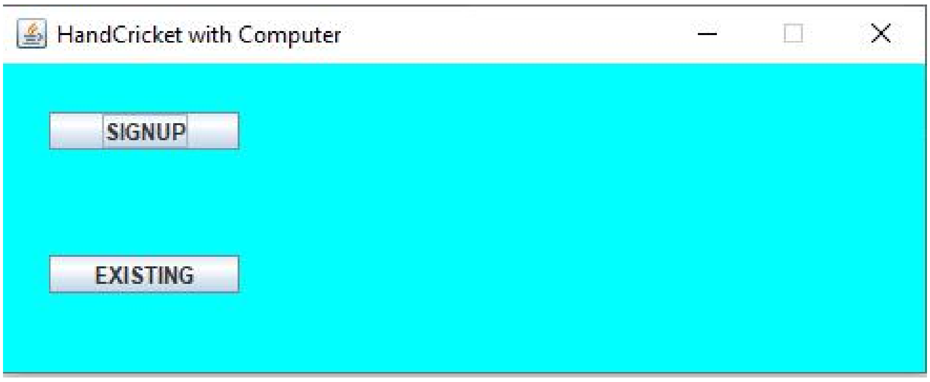
```
            JOptionPane.showMessageDialog(null,"CLosed Wrongly
\n");
        else if(x==0) {
            JOptionPane.showMessageDialog(null, "Match is Going
to Start \n");
            f.setVisible(false);
            new Match(p2);
        }
        else if(x==1) {
            JOptionPane.showMessageDialog(null, "Sorry for the
inconvenience \n");
            System.exit(0);
        }
    }
    private void create(player p) {
        try {
            FileOutputStream fileOut = new
FileOutputStream("player.txt");
            ObjectOutputStream out = new
ObjectOutputStream(fileOut) ;
            out.writeObject(p);
            out.close();
            fileOut.close();
            JOptionPane.showMessageDialog(null, "SIGNIN
SUCCESS PLAyer ID "+p.PLayerId);
```

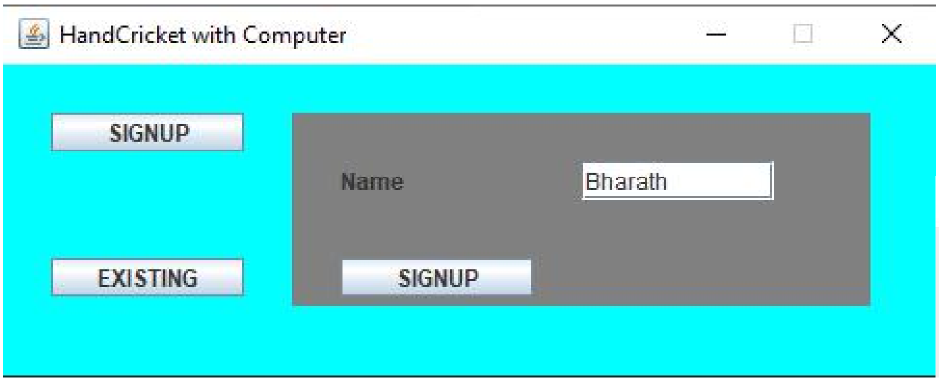
```
        } catch (IOException i) {
            JOptionPane.showMessageDialog(null, i + "");
        }
    }
    private int unique() {
        return (int) (Math.random() * 9000 + 1000);
    }
}
```

Output :

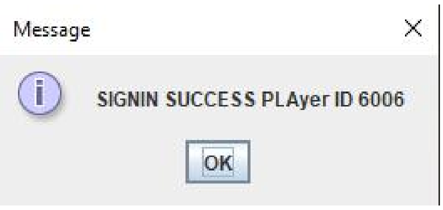
unique() method to generate unique player ID.

create() method that stores all the details of the player as Object in a file.

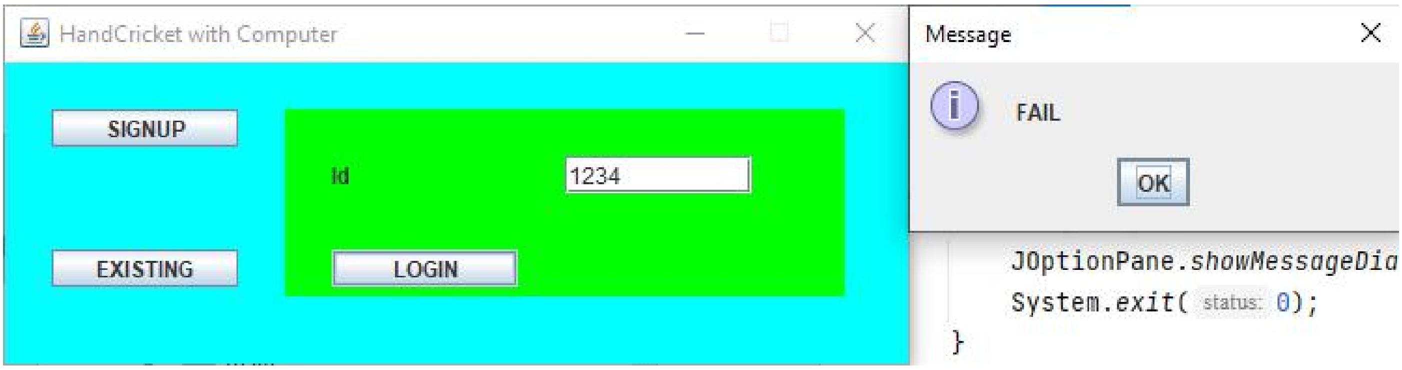




Name: Bharath



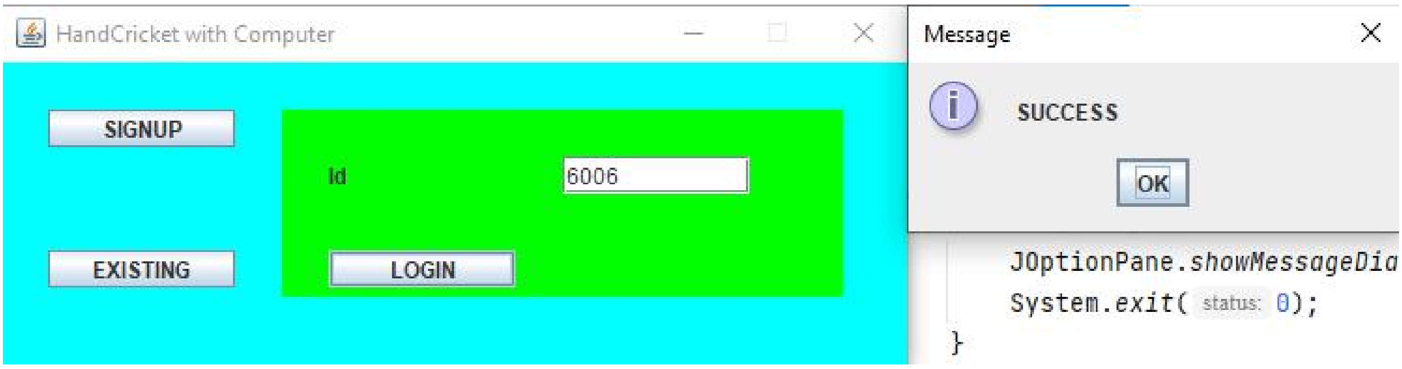
SIGNIN SUCCESS PLAyer ID 6006



Id 1234

FAIL

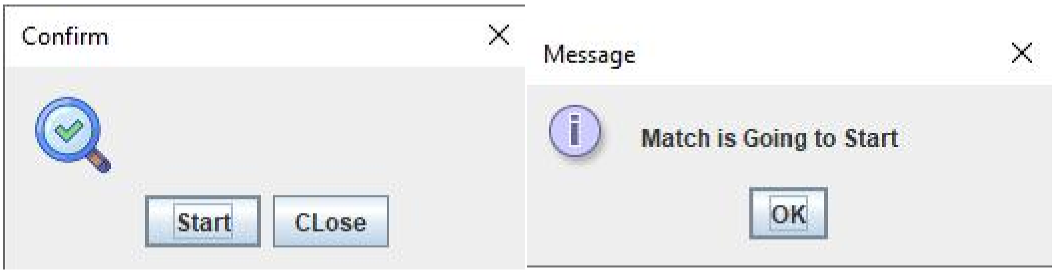
```
    JOptionPane.showMessageDia
    System.exit( status: 0);
}
```



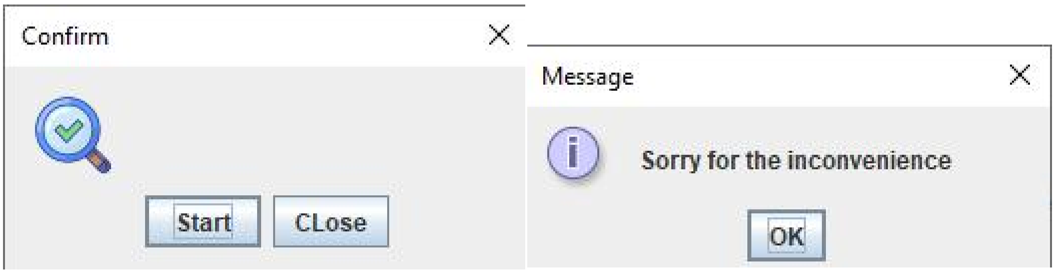
Id 6006

SUCCESS

```
    JOptionPane.showMessageDia
    System.exit( status: 0);
}
```



Match is Going to Start



Sorry for the inconvenience

Match.java

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.*;
import java.util.*;
```

```
class Match
{
        JFrame f ;
        JPanel p1;
        JButton b1,b2,b3;
        ImageIcon icon1 = new ImageIcon("E:\\project files\\img\\icon1.png");
        JLabel l2,l1;
        player m1;
        Match(player p)
        {
                m1=p;
                f = new JFrame();
         f.setLayout(null);
         f.setSize(500,200);
         f.setTitle("Match Started  Computer Vs "+p.PlayerName);
         f.setVisible(true);
         f.addWindowListener(new window());
         f.setResizable(false);
                p1 = new JPanel();
         p1.setLayout(null);
         p1.setBounds(0,0,500,300);
         p1.setVisible(true);
```

```
   p1.setBackground(Color.CYAN);
        b1 = new JButton("Toss");
        b1.setBounds(25,25,100,20);
        b1.addActionListener(new A2());
        p1.add(b1);
        l2 = new JLabel("");
        l2.setBounds(25,60,500,20);
        p1.add(l2);
        l1 = new JLabel("");
        l1.setBounds(150,25,150,20);
        p1.add(l1);
        b2 = new JButton("Summary");
        b2.setBounds(25,100,100,20);
        b2.addActionListener(new A1());
        b3 = new JButton("Show");
        b3.setBounds(25,130,100,20);
        b3.addActionListener(new A1());
        p1.add(b2);
        p1.add(b3);
        f.add(p1);
  }
  class A1 implements ActionListener
{
    @Override
    public void actionPerformed(ActionEvent e) {
            String s = e.getActionCommand();
            if(s.equals("Summary")) {
                try {
                    FileReader fr = new
FileReader("final.txt");
                    int i;
                    String sum = "";
                    while ((i = fr.read()) != -1)
                        sum = sum + (char) i;
                    JOptionPane.showMessageDialog(null,
sum, "Summary", JOptionPane.INFORMATION_MESSAGE);
                    fr.close();
                } catch (IOException i) {
                    JOptionPane.showMessageDialog(null,i);
                }
            }
            else
                if(s.equals("Show"))
                {
                    display();
                }
    }
```

```
        private void display() {
            player p2;
            try {
                FileInputStream fileIn = new FileInputStream("player.txt");
                ObjectInputStream in = new ObjectInputStream(fileIn);
                p2 = (player) in.readObject();
                JOptionPane.showMessageDialog(null,"Player Details \nPlayer Id : "+p2.PLayerId+"\nPlayer Name : "+p2.PlayerName+"\nPlayer Status : "+p2.Result+"\nPlayer Login Status : "+p2.status);
                in.close();
                fileIn.close();
            }
            catch(IOException | ClassNotFoundException i)
            {
                JOptionPane.showMessageDialog(null,i);
            }
        }
    }
    class A2 implements ActionListener
    {
        Random random = new Random();
```

```
            String sc="0",csc="0",sl="",sl1="";
            String temp1 = m1.PlayerName+"\n";
            public String computerBat(boolean f)
            {
                  String[] score ={"1","2","3","4","5","6"};
                  l2.setText("Computer is Batting");
                  sl+= "Computer is Batting"+"\n";
                  for(int i=1;i<=36;i++)
                  {
                        int x2 = JOptionPane.showOptionDialog(null,
"Ball No "+i,"Over
"+((i/6)+1),JOptionPane.DEFAULT_OPTION,
JOptionPane.INFORMATION_MESSAGE, icon1, score,
score[0]);
                        int randomNum = random.nextInt(6) + 1;
                        if(x2==-1)
                        {

      JOptionPane.showMessageDialog(null,"Closed wrongly");
                              b1.setEnabled(true);
                              break;
                        }
                        else
                              if((x2+1)==randomNum) {
```

```
    JOptionPane.showMessageDialog(null,"Bowl -Player : 
"+(x2+1)+" Bat -Computer : "+randomNum+" Out.Computer 
Final Score is "+csc);
                              sl= sl+"Bowl -Player : "+(x2+1)+" 
Bat -Computer : "+randomNum+" Out.Computer Final Score is 
"+csc+"\n";
                              break;
                         }
                         else {
                              if (i % 6 == 0) {
                                   csc = (Integer.parseInt(csc) + 
randomNum) + "";
                                   l2.setText("At the End of Over , 
Score is " + csc + " Bowl -Player : " + (x2 + 1) + " Bat -
Computer : " + randomNum);
                                   sl += l2.getText() + "\n";


                              } else {
                                   l2.setText("Ball no : " + i + " 
Bowl -Player : " + (x2 + 1) + " Bat -Computer : " + randomNum);
                                   sl += l2.getText() + "\n";
                                   csc = (Integer.parseInt(csc) + 
randomNum) + "";
                              }
                         }
```

```
                    if(!f && Integer.parseInt(csc) >
Integer.parseInt(sc))
                    {
                        return "Win";
                    }


            }
            return csc;
        }
        public String PlayerBat(boolean f)
        {
            String[] score ={"1","2","3","4","5","6"};
            l2.setText("Player is Batting");
            sl1+= l2.getText()+"\n";
            for(int i=1;i<=36;i++)
            {
                int x2 = JOptionPane.showOptionDialog(null,
"Ball No "+i,"Over
"+((i/6)+1),JOptionPane.DEFAULT_OPTION,
JOptionPane.INFORMATION_MESSAGE, icon1, score,
score[0]);
                int randomNum = random.nextInt(6) + 1;
                if(x2==-1)
                {
```

```
    JOptionPane.showMessageDialog(null,"Closed wrongly");
                        b1.setEnabled(true);
                        break;
                    }
                    else

                    if((x2+1)==randomNum)
                    {

    JOptionPane.showMessageDialog(null,"Bat -Player : 
"+(x2+1)+" Bowl -Computer : "+randomNum+" Out.Final Score 
is "+sc);
                        sl1+= "Bat -Player : "+(x2+1)+" Bowl -
Computer : "+randomNum+" Out.Final Score is "+sc+"\n";
                        break;
                    }
                    else
                        {
                            if(i%6==0) {

    sc=(Integer.parseInt(sc)+x2+1)+"";
                                l2.setText("At the End of Over , 
Score is " + sc+" Bat -Player : "+(x2+1)+" Bowl -Computer : 
"+randomNum);
```

```
                                        sl1+= l2.getText()+"\n";


                                   }
                                   else
                                   {
                                        l2.setText("Ball no : "+i+" Bat -
Player : "+(x2+1)+" Bowl -Computer : "+randomNum);
                                        sl1+= l2.getText()+"\n";

     sc=(Integer.parseInt(sc)+x2+1)+"";
                                   }
                              }
                         if(!f && Integer.parseInt(csc) <
Integer.parseInt(sc))
                         {
                              return "Win";
                         }


                    }
                    return sc+"";
               }
               public void actionPerformed(ActionEvent e)
               {
                    l1.setText("Waiting for toss");
```

```
                boolean fl = false;
                String[] filters = {"Head","Tail"};
                int randomNumber = random.nextInt(2);
                int x = JOptionPane.showOptionDialog(null,
"Which one you
prefer ?","Toss",JOptionPane.DEFAULT_OPTION,
JOptionPane.INFORMATION_MESSAGE, icon1, filters,
filters[0]);
                if(x==-1)
                    b1.setEnabled(true);
                else
                {
                    if(x==randomNumber)
                    {
                        l1.setText("Player Win the toss");
                        temp1+="PLayer Win the toss\n";
                        String[] filters1 = {"Batting","Bowling"};
                        int x1 =
JOptionPane.showOptionDialog(null, "Which one you
prefer ?","Selection",JOptionPane.DEFAULT_OPTION,
JOptionPane.INFORMATION_MESSAGE, icon1, filters1,
filters1[0]);
                        if(x1==-1)
                        {
```

```
    JOptionPane.showMessageDialog(null,"Selection wrong");
                         b1.setEnabled(true);
                    }
                    else
                    {
                         if(x1==0)
                         {
                                   sc= PlayerBat(true);
                                   csc = computerBat(false);
                                   if(csc.equals("Win"))
                                   {
                                        l2.setText("Computer
Wins");
                                        sl+= "Computer
Wins\n";
                                   }
                                   else

    if(Integer.parseInt(sc)>Integer.parseInt(csc)) {
                                        l2.setText("Player
Wins");
                                        fl =true;
                                        sl+= l2.getText()+"\n";
```

```
                                }
                                else

    if(Integer.parseInt(sc)==Integer.parseInt(csc)) {
                                        l2.setText("Draw
the Match");
                                        sl+=
l2.getText()+"\n";
                                    }
                                sl =sl1+sl;
                        }
                        else
                        {
                            csc = computerBat(true);
                            sc= PlayerBat(false);
                            if(sc.equals("Win"))
                            {
                                l2.setText("Player Wins");
                                fl = true;
                                sl1+= "Player Wins\n";

                            }
                            else
    if(Integer.parseInt(csc)>Integer.parseInt(sc)) {
                                l2.setText("Computer
Wins");
                                sl1+= l2.getText()+"\n";
                            }
                            else

    if(Integer.parseInt(sc)==Integer.parseInt(csc)) {
                                l2.setText("Draw the
Match");
                                sl1+= l2.getText()+"\n";
                            }
                            sl =sl+sl1;

                        }
                        sl = temp1 +sl;
                        b1.setEnabled(false);
                        try {
                            FileWriter fw = new
FileWriter("final.txt");
                            fw.write(sl);
                            fw.close();
                        } catch (IOException ioException) {
                            ioException.printStackTrace();
                        }
                    }
                }
                else
                {
                    temp1+="Computer Win the toss\n";
                int x1 = random.nextInt(2);
                l1.setText("Computer Win the toss");
                if(x1==1)
                    {
                        sc= PlayerBat(true);
                        csc = computerBat(false);
                        if(csc.equals("Win"))
                        {
                        l2.setText("Computer Wins");

                                    sl+= "Computer
Wins\n";
                                }
                                else

    if(Integer.parseInt(sc)>Integer.parseInt(csc)) {
                                    l2.setText("Player
Wins");
```

```
                                    sl+= l2.getText()+"\n";
                                    fl = true;
                                }
                                else

    if(Integer.parseInt(sc)==Integer.parseInt(csc)) {
                                        l2.setText("Draw
the Match");
                                        sl+=
l2.getText()+"\n";
                                    }
                                sl =sl1+sl;
                        }
                        else
                        {
                            csc = computerBat(true);
                            sc= PlayerBat(false);
                            if(sc.equals("Win"))
                            {
                                l2.setText("Player Wins");
                                sl1+= "Player Wins\n";
                                fl = true;
                            }
                            else
    if(Integer.parseInt(csc)>Integer.parseInt(sc)) {
                                l2.setText("Computer
Wins");
                                sl1+= l2.getText()+"\n";
                            }
                            else

    if(Integer.parseInt(sc)==Integer.parseInt(csc)) {
                                l2.setText("Draw the
Match");
                                sl1+= l2.getText()+"\n";
                            }
                            sl =sl+sl1;

                        }
                        sl = temp1 +sl;
                        b1.setEnabled(false);
                        try {
                            FileWriter fw = new
FileWriter("final.txt");
                            fw.write(sl);
                            fw.close();
                        } catch (IOException ioException) {
                            ioException.printStackTrace();
```

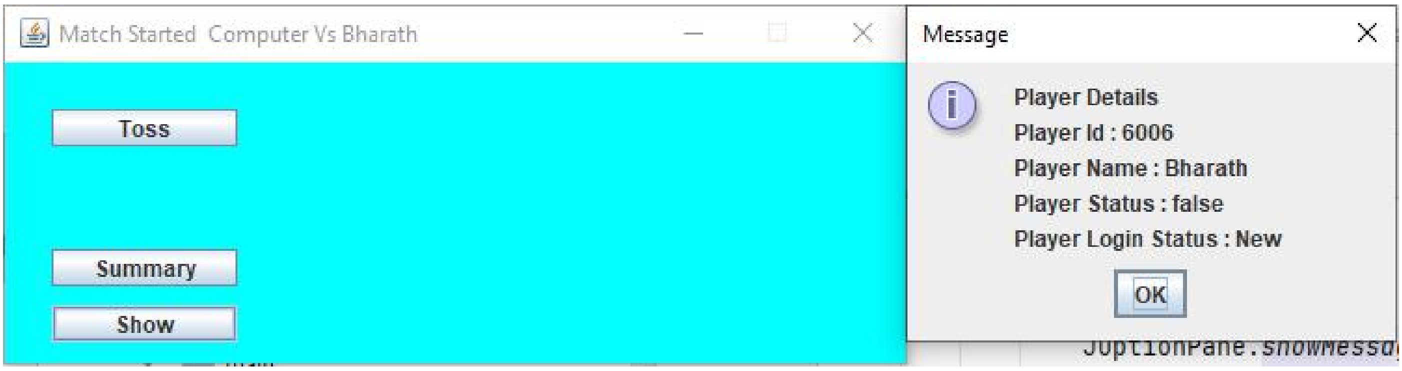
```
                                    }
                               }


                    }
                    m1.status="Existing";
                    m1.Result = fl;
                    try {
                         FileOutputStream fileOut = new
FileOutputStream("player.txt");
                         ObjectOutputStream out = new
ObjectOutputStream(fileOut) ;
                         out.writeObject(m1);
                         out.close();
                         fileOut.close();
                         JOptionPane.showMessageDialog(null, "File
Updated");
                    } catch (IOException i) {
                         JOptionPane.showMessageDialog(null, i + "");
                    }
               }
          }
}
```

Output :

Wining Status is referred by win(true)/ lose(false).

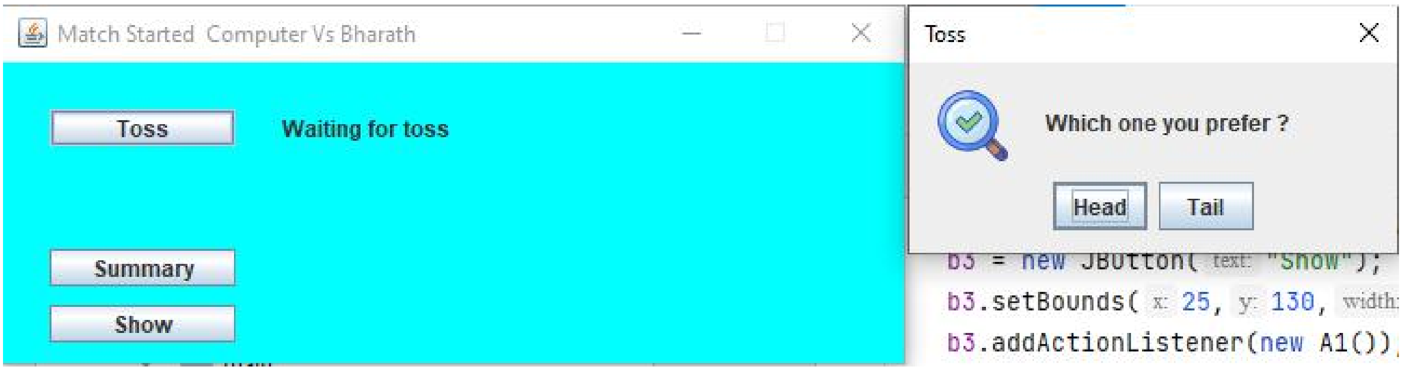
display() method that display the player details based on player ID.

goingstart() method to start the game and to close the game.



Player Details
Player Id : 6006
Player Name : Bharath
Player Status : false
Player Login Status : New

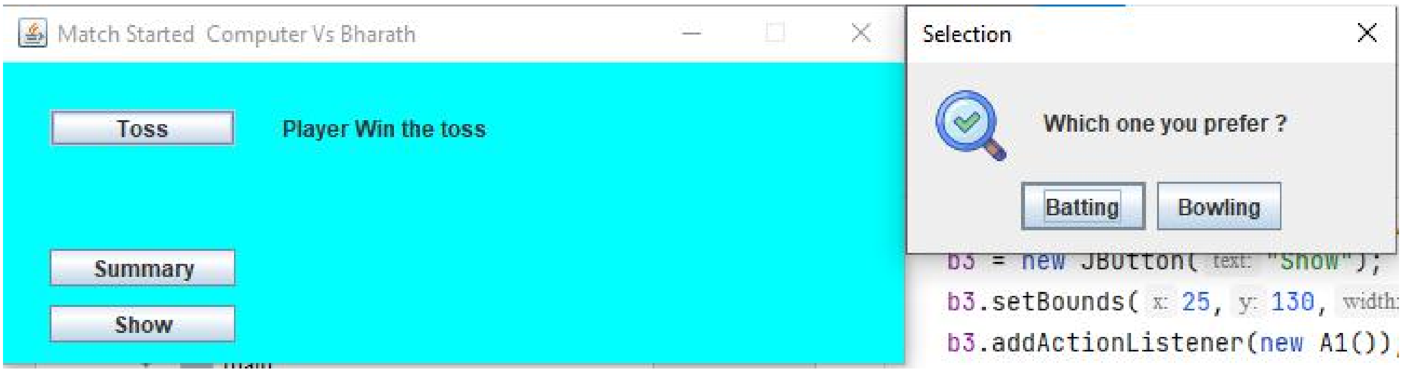
```
JOptionPane.showMessa
```



Waiting for toss

Which one you prefer ?

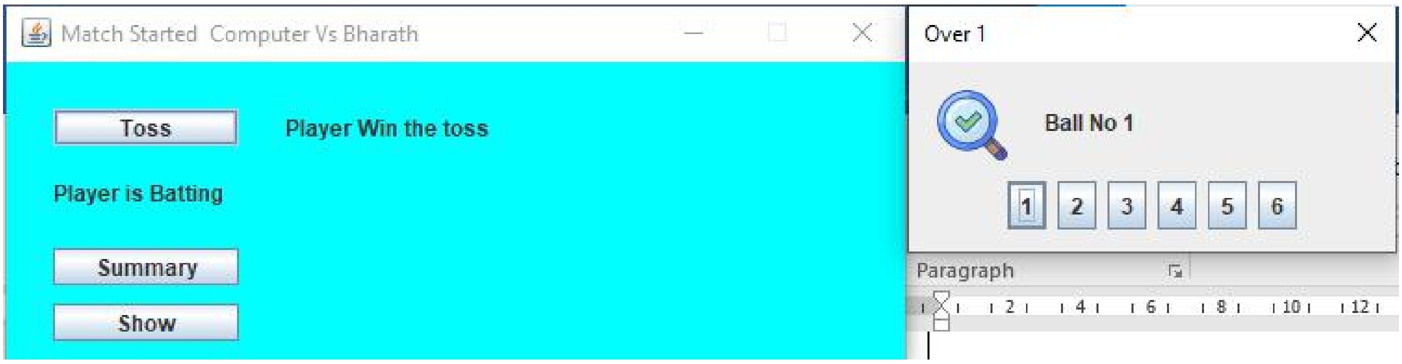
```
b3 = new JButton( text: "Show");
b3.setBounds( x: 25, y: 130, width:
b3.addActionListener(new A1())
```



Player Win the toss

Which one you prefer ?

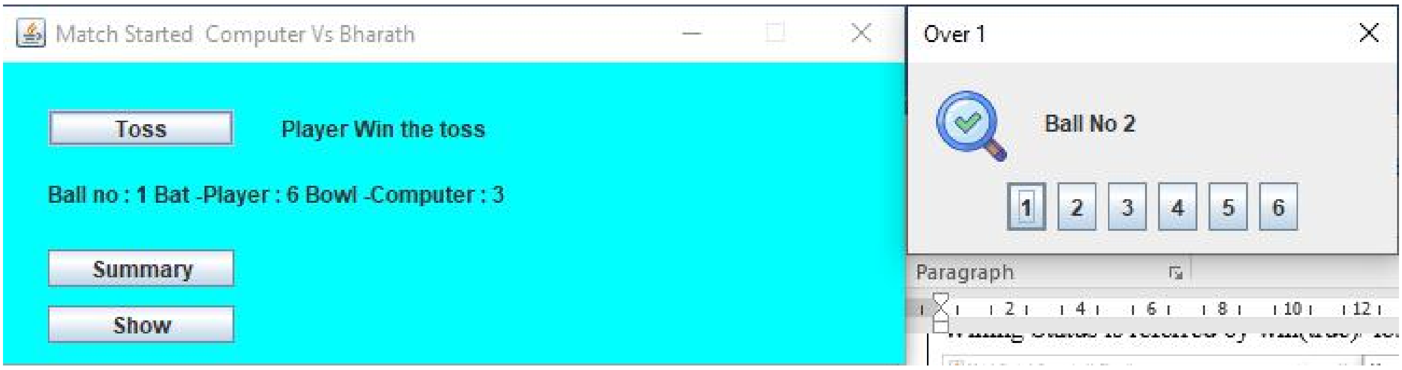
```
b3 = new JButton( text: "Show");
b3.setBounds( x: 25, y: 130, width:
b3.addActionListener(new A1())
```



Player Win the toss

Player is Batting

Ball No 1



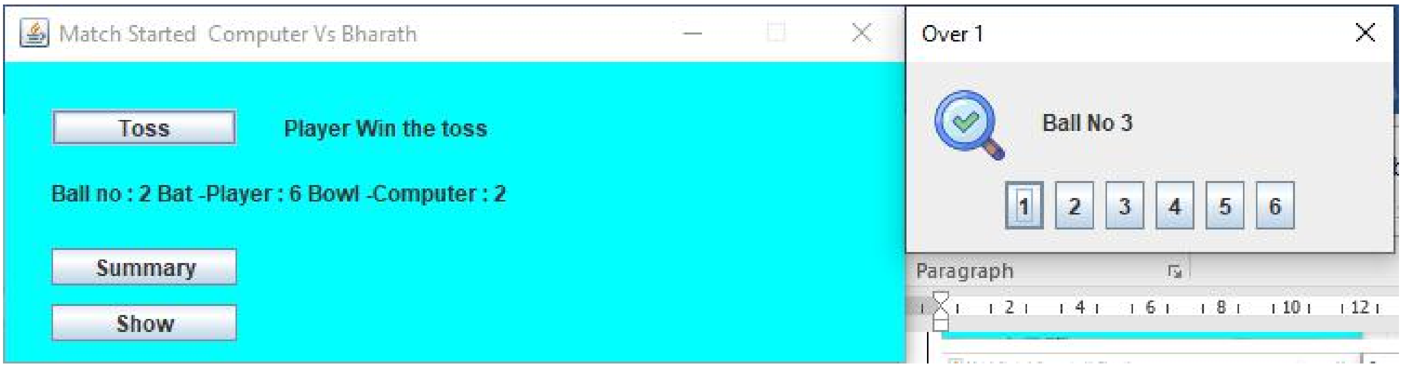
Match Started Computer Vs Bharath

Player Win the toss

Ball no : 1 Bat -Player : 6 Bowl -Computer : 3

Over 1

Ball No 2



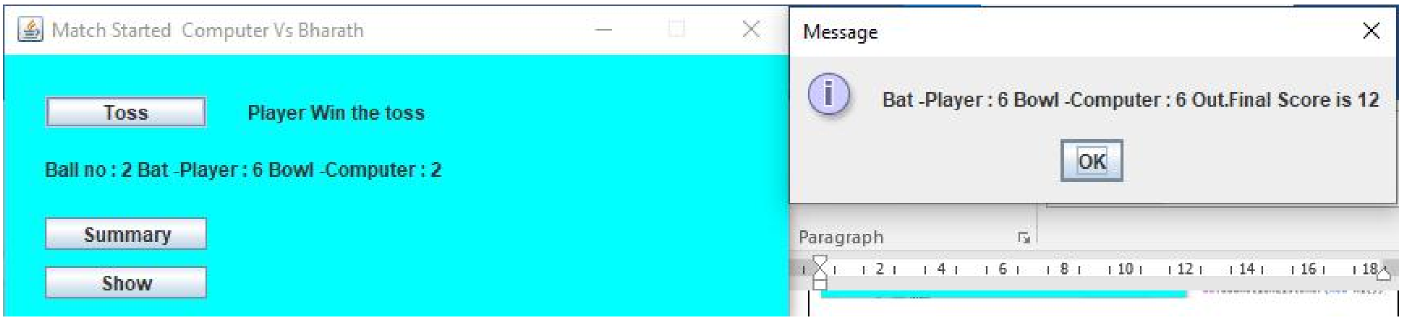
Match Started Computer Vs Bharath

Player Win the toss

Ball no : 2 Bat -Player : 6 Bowl -Computer : 2

Over 1

Ball No 3



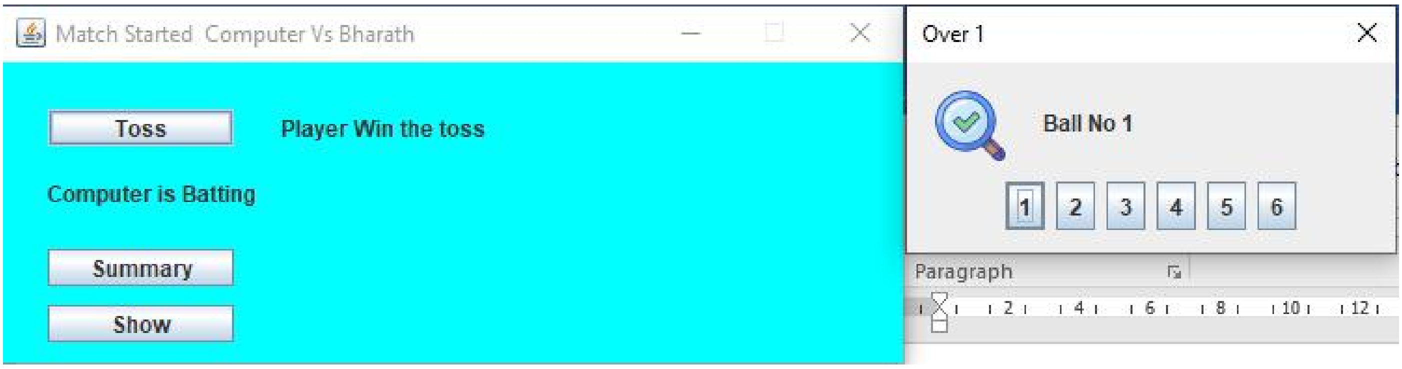
Match Started Computer Vs Bharath

Player Win the toss

Ball no : 2 Bat -Player : 6 Bowl -Computer : 2

Message

Bat -Player : 6 Bowl -Computer : 6 Out.Final Score is 12



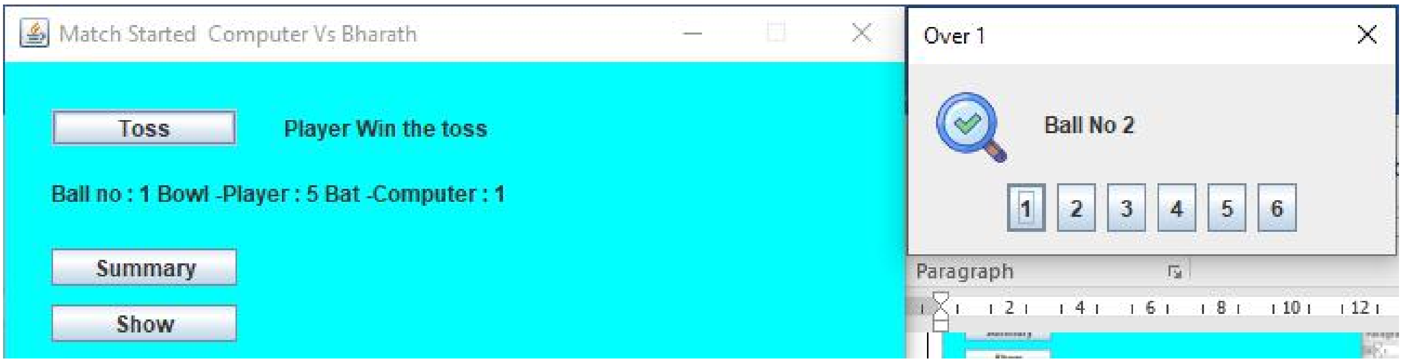
Match Started Computer Vs Bharath

Player Win the toss

Computer is Batting

Over 1

Ball No 1



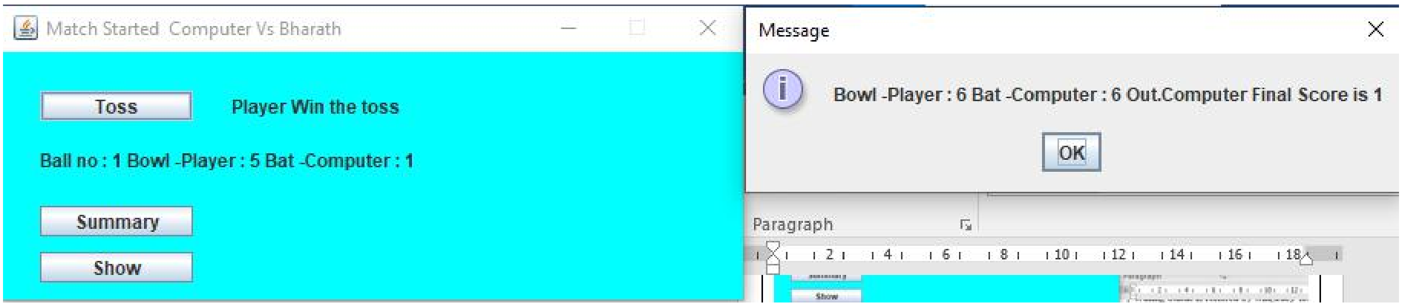
Match Started Computer Vs Bharath

Player Win the toss

Ball no : 1 Bowl -Player : 5 Bat -Computer : 1

Over 1

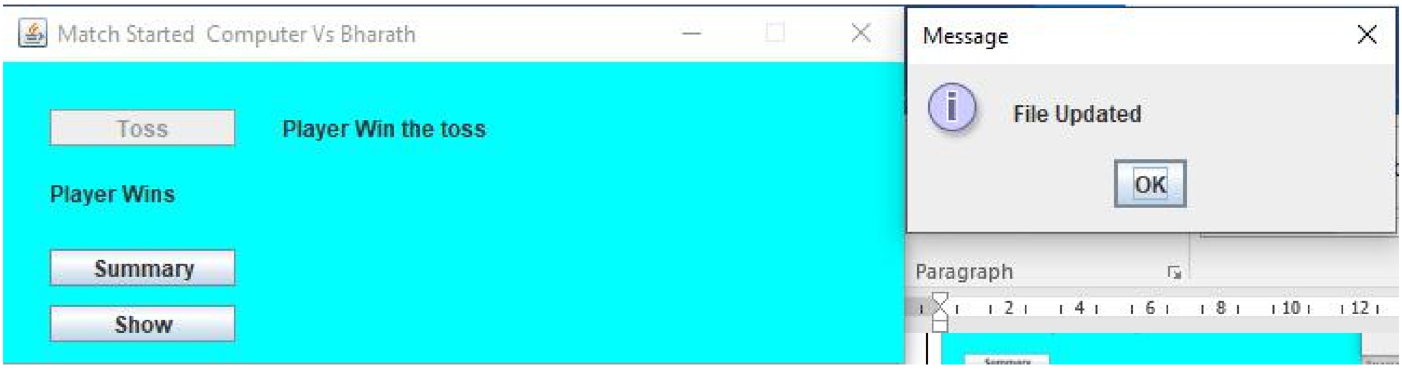
Ball No 2



Player Win the toss

Ball no : 1 Bowl -Player : 5 Bat -Computer : 1

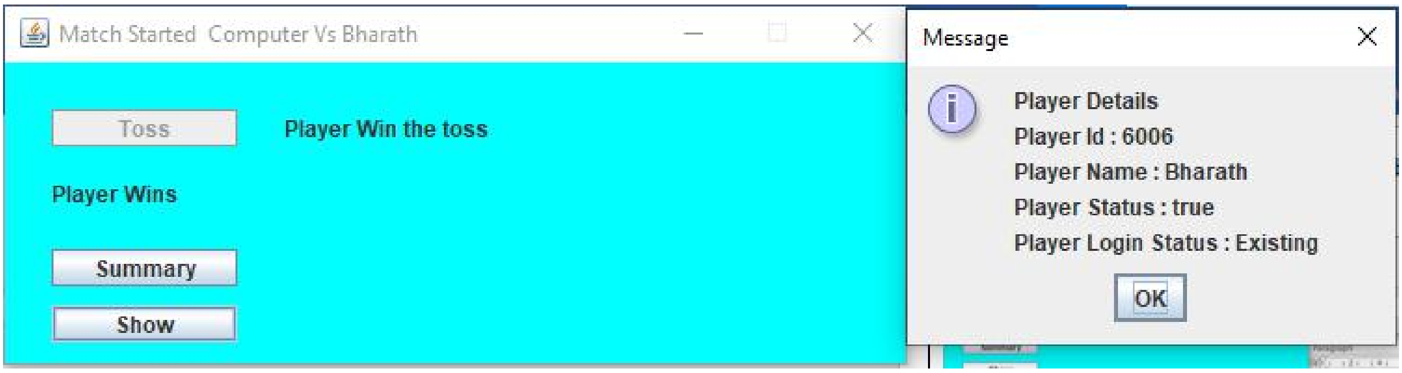
Bowl -Player : 6 Bat -Computer : 6 Out.Computer Final Score is 1



Player Win the toss

Player Wins

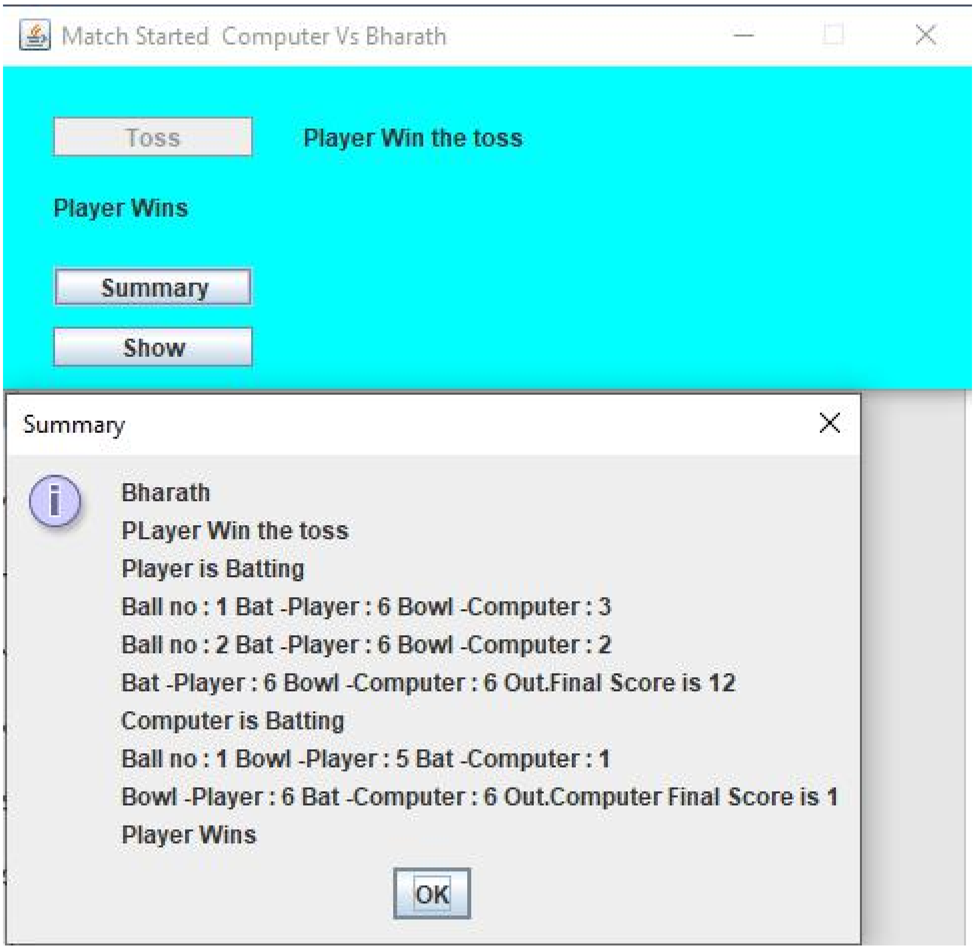
File Updated



Player Win the toss

Player Wins

Player Details
Player Id : 6006
Player Name : Bharath
Player Status : true
Player Login Status : Existing



Player Win the toss

Player Wins

Bharath
PLayer Win the toss
Player is Batting
Ball no : 1 Bat -Player : 6 Bowl -Computer : 3
Ball no : 2 Bat -Player : 6 Bowl -Computer : 2
Bat -Player : 6 Bowl -Computer : 6 Out.Final Score is 12
Computer is Batting
Ball no : 1 Bowl -Player : 5 Bat -Computer : 1
Bowl -Player : 6 Bat -Computer : 6 Out.Computer Final Score is 1
Player Wins