A Mini Project Report

On

PIC PUZZLE GAME

Submitted in partial fulfillment of requirements for the Course
CSE18R272 - JAVA PROGRAMMING

Bachelor's of Technology

In

Computer Science and Engineering

Submitted By

M.Sivaganesh

9918004112

Under the guidance of

Dr. R. RAMALAKSHMI

(Associate Professor)

Department of Computer Science and Engineering

Kalasalingam Academy of Research and Education

Anand Nagar, Krishnankoil-626126

APRIL 2020

ABSTRACT

Puzzles are great for helping young brain develop and grow.Since all puzzle pieces now have identifiers in the array, order,all we need to do is randomly switch their identifiers for shuffling.In case two identical pieces are selected, I simply increase the second identifier by one. Once the scrambling is done, the display area needs to be refreshed.The main theme behind developing puzzle game using java is to provide a creative and competitive environment for the players who will use this system.. Some extensive graphical interface has been used to give excellent picture quality and its optimized code will help users to get processing work faster without any delay.. Some extensive graphical interface has been used to give excellent picture quality and its optimized code will help users to get processing work faster without any delay.

DECLARATION

I hereby declare that the work presented in this report entitled "**Pic puzzle**", in partial fulfilment of the requirements for the course CSE18R272-Java Programming and submitted in **Department of Computer Science and Engineering, Kalasalingam Academy of Research and Education** (**Deemed to be University**) is an authentic record of our own work carried out during the period from **Jan 2020** under the guidance of Mr. **Dr. R. Ramalakshmi** (Associate Professor).

The work reported in this has not been submitted by me for the award of any other degree of this or any other institute.

M.Sivaganesh
9918004112

ACKNOWLEDGEMENT

First and foremost, I wish to thank the Almighty God for his grace and benediction tocomplete this Project work successfully. I would like to convey my special thanks from the bottom of my heart to my dear Parents and affectionate Family members for their honest support for the completion of this Project work.

I express deep sense of gratitude to "Kalvivallal" Thiru. T. Kalasalingam B.com., Founder Chairman, "Ilayavallal" Dr.K.Sridharan Ph.D., Chancellor, Dr.S.ShasiAnand, Ph.D., Vice President (Academic) , Mr.S.ArjunKalasalingam M.S., Vice President (Administration) , Dr.R.Nagaraj Vice-Chancellor, Ph.D., Registrar , Dr.P.Deepalakshmi Ph.D., Dean (School of Computing) . And also a special thanks to Dr. A. FRANCIS SAVIOUR DEVARAJ. Head Department of CSE, Kalasalingam Academy of Research and Education forgranting the permission and providing necessary facilities to carry out Project work.

I would like to express my special appreciation and profound thanks to my enthusiastic Project Supervisor Dr.R.Ramalakshmi Ph.D, Associate Professor at Kalasalingam Academy of Research and Education [KARE] for her inspiring guidance, constant encouragement with my work during all stages. I am extremely glad that I had a chance to do my Project under my Guide, who truly practices and appreciates deep thinking. I will be forever indebted to my Guide for all the time he has spent with me in discussions. And during the most difficult times when writing this report, he gave me the moral support and the freedom I needed to move on.

M.Sivaganesh
9918004112

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

The idea of our project is to design "PIC PUZZLE GAME".The main theme behind developing puzzle game using java is to provide a creative and competitive environment for the players who will use this system.A puzzle is a game, problem, or toy that tests a person's ingenuity or knowledge. In a puzzle, the solver is expected to put pieces together in a logical way, in order to arrive at the correct or fun solution of the puzzle. There are different genres of puzzles, such as crossword puzzles, word-search puzzles, number puzzles, relational puzzles, or logic puzzles. Solutions of puzzles often require the recognition of patterns and the adherence to a particular kind of ordering. People with a high level of inductive reasoning aptitude may be better at solving such puzzles than others. But puzzles based upon inquiry and discovery may be solved more easily by those with good deduction skills. Deductive reasoning improves with practice. A puzzle is a game, problem, or toy that tests a person's ingenuity or knowledge. In a puzzle, the solver is expected to put pieces together in a logical way, in order to arrive at the correct or fun solution of the puzzle.

Chapter 2

HISTORY

The origins of Pic puzzles go back to the 1760s when European mapmakers pasted maps onto wood and cut them into small pieces. John Spilsbury, an engraver and mapmaker, is credited with inventing the first Pic puzzle in 1767. The dissected map has been a successful educational toy ever since.

American children still learn geography by playing with puzzle maps of the United States or the world. The eighteenth century inventors of Pic puzzles would be amazed to see the transformations of the last 250 years. Children's puzzles have moved from lessons to entertainment, s howing diverse subjects like animals, nursery rhymes, and modern tales of super heroes. But the biggest surprise for the early puzzle makers would be how adults have embraced puzzling over the last century.

Chapter 3

OBJECTIVE

The Objective of this project are as follows:
To develop a code for implementation pic puzzle game.
To be able to create real life feature of pic puzzle game.
To act as a good form of relaxation
A puzzle always has a dominant strategy and
determine by logic and calculation

Chapter 4

JAVA PACKAGES

JAVA PACKAGES :

While creating a package, you should choose a name for the package and include a package statement along with that name at the top of every source file that contains the classes, interfaces, enumerations, and annotation types that you want to include in the package.

The package statement should be the first line in the source file. There can be only one package statement in each source file, and it applies to all types in the file.

If a package statement is not used then the class, interfaces, enumerations, and annotation types will be placed in the current default package.

Chapter 5

PROJECT DESCRITION

JAVA SWING PACKAGE :

Java Swing is a part of Java Foundation Classes (JFC) that is used to create window-based applications. It is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java.

Unlike AWT, Java Swing provides platform-independent and lightweight components.

Java swing components are platform-independent.

The javax.swing package provides classes for java – such as JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser etc.

JAVA AWD PACKAGE (ABSTRACT WINDOW TOOLKIT) :

Java AWT (Abstract Window Toolkit) is an API to develop GUI or window-based applications in java.

Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components are using the resources of OS.

The java.awt package provides classes for AWT such as TextField, Label, TextArea, RadioButton, CheckBox, Choice, List

Chapter 6

CLASSES USED IN THE CODE

USING THREE CLASS IN PIC PUZZLE GAME PACKAGES

1. JFrame–

The javax.swing.JFrame class is a type of container which inherits the java.awt.Frame class. JFrame works like the main window where components like labels, buttons, textfields are added to create a GUI.

Unlike Frame, JFrame has the option to hide or close the window with the help of setDefaultCloseOperation(int) method.

Constructors used: • JFrame() - It constructs a new frame that is initially invisible. • JFrame(GraphicsConfiguration gc) - It creates a Frame in the specified GraphicsConfiguration of a screen device and a blank title. • JFrame(String title) - It creates a new, initially invisible Frame with the specified title. • JFrame(String title, GraphicsConfiguration gc) - It creates a JFrame with the specified title and the specified GraphicsConfiguration of a screen device. Methods used: • protected void addImpl(Component comp, Object constraints, int index) - Adds the specified child Component.

• protected JRootPane createRootPane() - Called by the constructor meth- ods to create the default rootPane.

• protected void frameInit()– Called by the constructors to init the JFrame properly. • void setContentPane(Containe contentPane) - It sets the contentPane property

• static void setDefaultLookAndFeelDecorated(boolean defaultLookAndFeelDecorated) - Provides a hint as to whether or not newly created JFrames should

have their Window decorations (such as borders, widgets to close the win- dow, title...) provided by the current look

2.Actionlistener —

The Java ActionListener is notified whenever you click on the button or menu item. It is notified against ActionEvent.

The ActionListener interface is found in java.awt.event package. How to write ActionListener The common approach is to implement the ActionListener.If you implement the ActionListener class, you need to follow 3 steps: Implement the ActionListener interface in the class: • The Java ActionListener is notified whenever you click on the button or menu item. It is notified against ActionEvent. • The ActionListener interface is found in java.awt.event package. • It has only one method: actionPerformed().

The actionPerformed() method is invoked automatically whenever you click on the registered component. public abstract void actionPerformed(ActionEvent e); To implement the ActionListener class, you need to follow 3 steps: 1) Implement the ActionListener interface in the class: public class ActionListenerExample Implements ActionListener 2) Register the component with the Listener: component.addActionListener(instanceOfListenerclass); 3) Override the actionPerformed() method: public void actionPerformed(ActionEvent e)

ActionEvent

A semantic event which indicates that a component-defined action occured. This high-level event is generated by a component (such as a Button)

when the component-specific action occurs (such as being pressed). The

event is passed to every every ActionListener object that registered to receive such events using the component's addActionListener method. The

object that implements the ActionListener interface gets this ActionEvent

when the event occurs. The listener is therefore spared the details of processing individual mouse movements and mouse clicks, and can instead process

a "meaningful" (semantic) event like "button pressed".

3.JButton –

The JButton class is used to create a labeled button that has platform independent implementation.

The application result in some action when the button is pushed. It inherits AbstractButton class. JButton class declaration: Let's see the declaration for javax.swing.JButton class. public class JButton extends AbstractButton implements Accessible Constructors used: • JButton() - It creates a button with no text and icon. • JButton(String s) - It creates a button with the specified text. • JButton(Icon i) - It creates a button with the specified icon object. Used Methods of AbstractButton class: • void setText(String s) -

It is used to set specified text on button • String getText() - It is used to return the text of the button. • void setEnabled(boolean b) - It is used to enable or disable the button. • void setIcon(Icon b) -It is used to set the specified Icon on the button.

Chapter 7

PROJECT DISCRIPTION

IN PIC PUZZLE GAME is implemented using jdk and jre. many packages are used in this program.such that java swing and java awd. only one can play this game . there are three more classess ore used, jframe,jbutton,actionlistener. main class name is picpuzzle. THERE ARE THREE WAY TO RUN THE CODE JAVA RUNTIME ENVIRONMENT, NETBEANS IDE ,ECLIPSE

Java Swing is a part of Java Foundation Classes (JFC) that is used to create window-based applications. It is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java.Unlike AWT, Java Swing provides platform-independent and lightweight components. Java swing components are platform-independent.

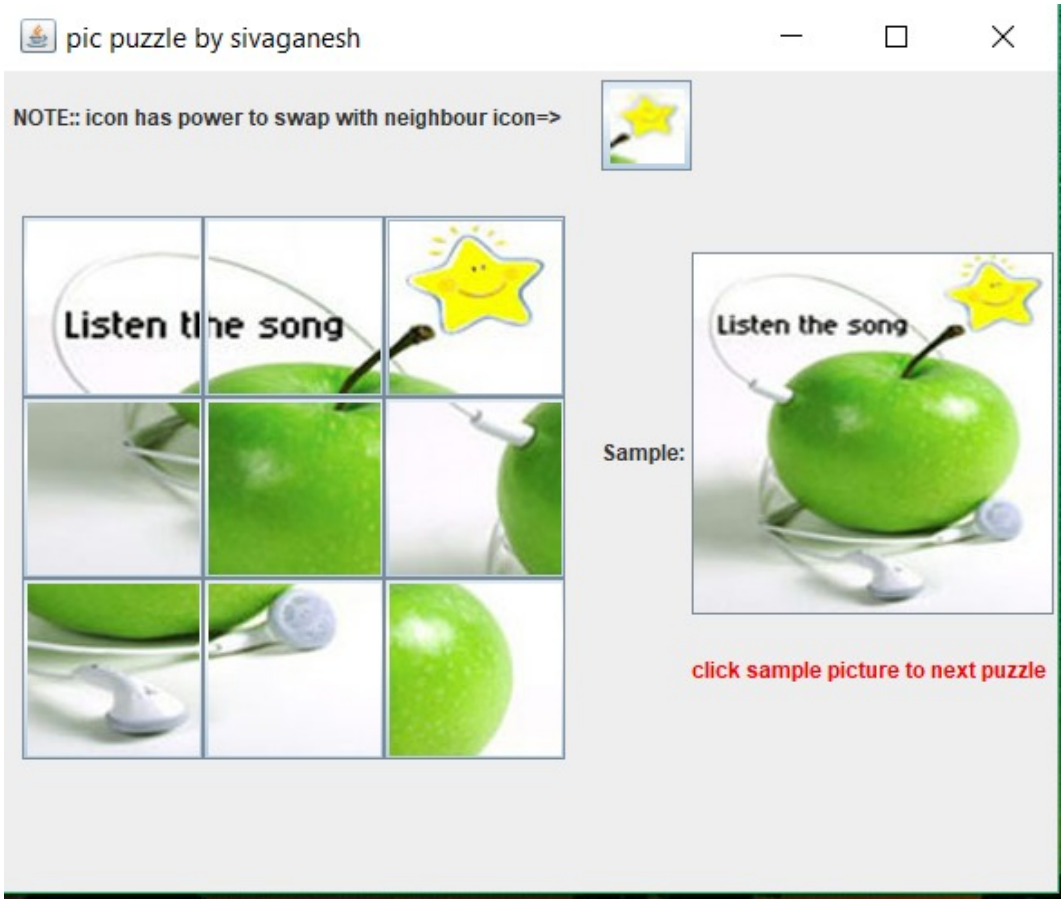

Figure 7.1: Figure Example

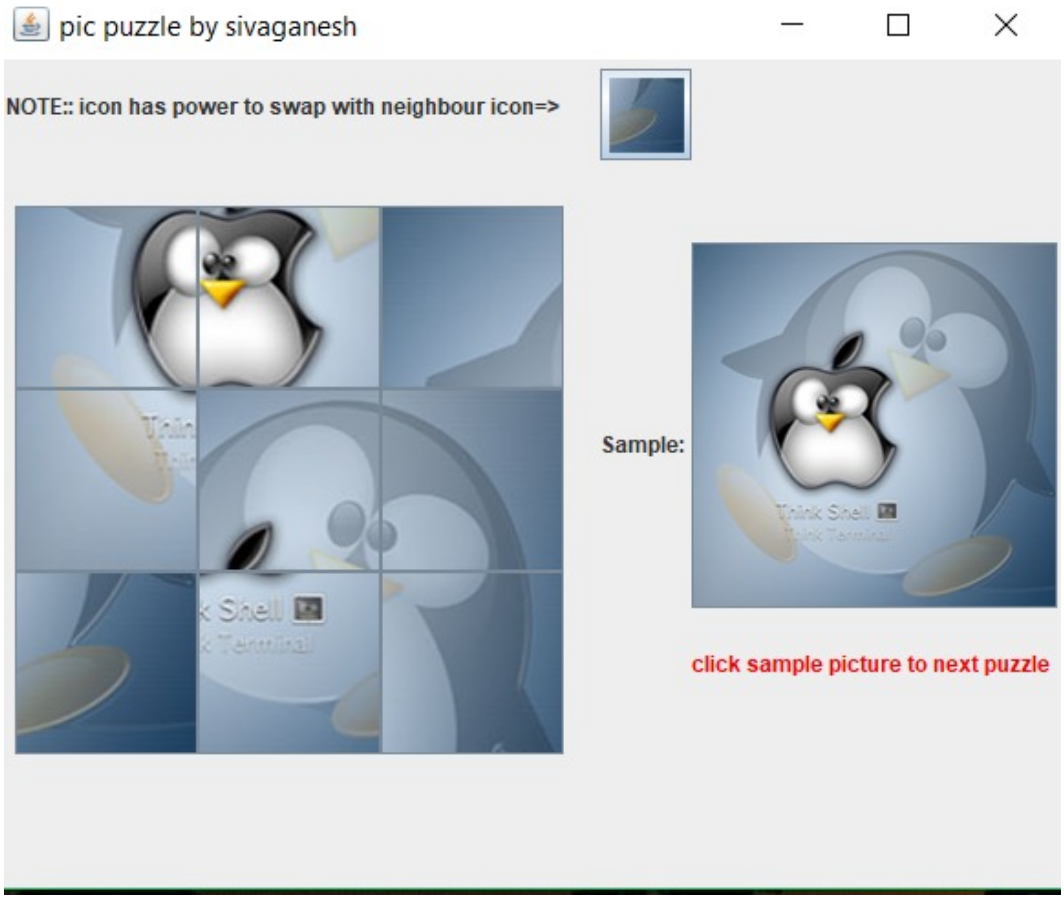

Figure 7.2: Figure Example

Chapter 8

CONCLUSION

This project augments both the normal brain and the defective brain. It displays the outer view and the internal view of both the brain structures. The application is also accompanied with background voice which explains about the detailed description of the internal brain parts. Thus some added features like scaling, rotation and translation are applied to the AR models so that it would help the user to view the object in the desirable angle. The future enhancement may include the addition of dissection property to the augmented object. When the user has the ability to dissect the objects it would the understandability of the structure. This can also be applied for other deformities in the human anatomy.

Appendices

SOURCE CODE

```
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
class picpuzzle extends JFrame implements
   ↪ ActionListener
{
JButton b1,b2,b3,b4,b5,b6,b7,b8,b9,sample,starB;
Icon star;
Icon ic0=new ImageIcon("pic/starB0.jpg");
Icon ic10=new ImageIcon("pic/starB10.jpg");
Icon ic20=new ImageIcon("pic/starB20.jpg");
Icon samicon1=new ImageIcon("pic/main.jpg");
Icon samicon2=new ImageIcon("pic/main2.jpg");
Icon samicon3=new ImageIcon("pic/main3.jpg");
Icon ic1=new ImageIcon("pic/1.jpg");
Icon ic2=new ImageIcon("pic/5.jpg");
Icon ic3=new ImageIcon("pic/2.jpg");
Icon ic4=new ImageIcon("pic/7.jpg");
Icon ic5=new ImageIcon("pic/4.jpg");
Icon ic6=new ImageIcon("pic/6.jpg");
Icon ic7=new ImageIcon("pic/8.jpg");
Icon ic8=new ImageIcon("pic/9.jpg");
Icon ic9=new ImageIcon("pic/3.jpg");

Icon ic11=new ImageIcon("pic/12.jpg");
Icon ic12=new ImageIcon("pic/13.jpg");
Icon ic13=new ImageIcon("pic/16.jpg");
Icon ic14=new ImageIcon("pic/11.jpg");
Icon ic15=new ImageIcon("pic/14.jpg");
Icon ic16=new ImageIcon("pic/19.jpg");
Icon ic17=new ImageIcon("pic/17.jpg");
Icon ic18=new ImageIcon("pic/15.jpg");
Icon ic19=new ImageIcon("pic/18.jpg");

Icon ic21=new ImageIcon("pic/24.jpg");
Icon ic22=new ImageIcon("pic/25.jpg");
Icon ic23=new ImageIcon("pic/21.jpg");
Icon ic24=new ImageIcon("pic/27.jpg");
```

```
Icon ic25=new ImageIcon("pic/23.jpg");
Icon ic26=new ImageIcon("pic/29.jpg");
Icon ic27=new ImageIcon("pic/28.jpg");
Icon ic28=new ImageIcon("pic/22.jpg");
Icon ic29=new ImageIcon("pic/26.jpg");

picpuzzle(){

super("pic puzzle by sivaganesh");

b1=new JButton(ic1);
b1.setBounds(10,80,100,100);
b2=new JButton(ic2);
b2.setBounds(110,80,100,100);
b3=new JButton(ic3);
b3.setBounds(210,80,100,100);
b4=new JButton(ic4);
b4.setBounds(10,180,100,100);
b5=new JButton(ic5);
b5.setBounds(110,180,100,100);
b6=new JButton(ic6);
b6.setBounds(210,180,100,100);
b7=new JButton(ic7);
b7.setBounds(10,280,100,100);
b8=new JButton(ic8);
b8.setBounds(110,280,100,100);
b9=new JButton(ic9);
b9.setBounds(210,280,100,100);
sample=new JButton(samicon1);
sample.setBounds(380,100,200,200);

JLabel l1=new JLabel("Sample:");
l1.setBounds(330,200,70,20);
JLabel l2=new JLabel("NOTE:: icon has power to swap
   ↪ with neighbour icon=>");
l2.setBounds(5,15,500,20);
JLabel l3=new JLabel("click sample picture to next
   ↪ puzzle");
l3.setBounds(380,320,200,20);
l3.setForeground(Color.red);
```

```
starB=new JButton(ic0);
starB.setBounds(330,5,50,50);
star=b9.getIcon();

add(b1);add(b2);add(b3);add(b4);add(b5);add(b6);add(b7)
    ↪ ;add(b8);add(b9);add(sample);add(l1);add(l2);add(
    ↪ starB);add(l3);
b1.addActionListener(this); b2.addActionListener(this);
    ↪  b3.addActionListener(this); b4.addActionListener
    ↪ (this); b5.addActionListener(this); b6.
    ↪ addActionListener(this); b7.addActionListener(
    ↪ this); b8.addActionListener(this); b9.
    ↪ addActionListener(this);
sample.addActionListener(this);
setLayout(null);
setSize(600,500);
setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public void actionPerformed(ActionEvent e){
if(e.getSource()==b1){
    Icon s1=b1.getIcon();
      if(b2.getIcon()==star){
        b2.setIcon(s1);
        b1.setIcon(star);
      } else if(b4.getIcon()==star){
                    b4.setIcon(s1);
                    b1.setIcon(star);
                   }
  }

if(e.getSource()==b2){
    Icon s1=b2.getIcon();
      if(b1.getIcon()==star){
        b1.setIcon(s1);
        b2.setIcon(star);
      } else if(b5.getIcon()==star){
                    b5.setIcon(s1);
```

```
                    b2.setIcon(star);
                   }
          else if(b3.getIcon()==star){
                    b3.setIcon(s1);
                    b2.setIcon(star);
                   }
  }
if(e.getSource()==b3){
    Icon s1=b3.getIcon();
      if(b2.getIcon()==star){
        b2.setIcon(s1);
        b3.setIcon(star);
      } else if(b6.getIcon()==star){
                    b6.setIcon(s1);
                    b3.setIcon(star);
                   }
  }

if(e.getSource()==b4){
    Icon s1=b4.getIcon();
      if(b1.getIcon()==star){
        b1.setIcon(s1);
        b4.setIcon(star);
      } else if(b5.getIcon()==star){
                    b5.setIcon(s1);
                    b4.setIcon(star);
                   }
          else if(b7.getIcon()==star){
                    b7.setIcon(s1);
                    b4.setIcon(star);
                   }
  }
if(e.getSource()==b5){
    Icon s1=b5.getIcon();
      if(b2.getIcon()==star){
        b2.setIcon(s1);
        b5.setIcon(star);
      } else if(b4.getIcon()==star){
                    b4.setIcon(s1);
                    b5.setIcon(star);
```

```
                    }
          else if(b6.getIcon()==star){
                     b6.setIcon(s1);
                     b5.setIcon(star);
                    }
           else if(b8.getIcon()==star){
                     b8.setIcon(s1);
                     b5.setIcon(star);
                    }
  }

if(e.getSource()==b6){
    Icon s1=b6.getIcon();
      if(b3.getIcon()==star){
        b3.setIcon(s1);
        b6.setIcon(star);
      } else if(b5.getIcon()==star){
                     b5.setIcon(s1);
                     b6.setIcon(star);
                    }
         else if(b9.getIcon()==star){
                     b9.setIcon(s1);
                     b6.setIcon(star);
                    }
}

if(e.getSource()==b7){
    Icon s1=b7.getIcon();
      if(b4.getIcon()==star){
        b4.setIcon(s1);
        b7.setIcon(star);
      } else if(b8.getIcon()==star){
                     b8.setIcon(s1);
                     b7.setIcon(star);
                    }
 }

   if(e.getSource()==b8){
    Icon s1=b8.getIcon();
      if(b7.getIcon()==star){
```

```
        b7.setIcon(s1);
        b8.setIcon(star);
      } else if(b5.getIcon()==star){
                    b5.setIcon(s1);
                    b8.setIcon(star);
                   }
         else if(b9.getIcon()==star){
                    b9.setIcon(s1);
                    b8.setIcon(star);
                   }

  }

 if(e.getSource()==b9){
    Icon s1=b9.getIcon();
      if(b8.getIcon()==star){
        b8.setIcon(s1);
        b9.setIcon(star);
      } else if(b6.getIcon()==star){
                    b6.setIcon(s1);
                    b9.setIcon(star);
                   }
  }

if(e.getSource()==sample){
Icon s1=sample.getIcon();
 if(s1==samicon3){
sample.setIcon(samicon1);
b1.setIcon(ic1);
b2.setIcon(ic2);
b3.setIcon(ic3);
b4.setIcon(ic4);
b5.setIcon(ic5);
b6.setIcon(ic6);
b7.setIcon(ic7);
b8.setIcon(ic8);
b9.setIcon(ic9);
star=b9.getIcon();
starB.setIcon(ic0);
}
```

```
else if(s1==samicon1){
sample.setIcon(samicon2);
b1.setIcon(ic11);
b2.setIcon(ic12);
b3.setIcon(ic13);
b4.setIcon(ic14);
b5.setIcon(ic15);
b6.setIcon(ic16);
b7.setIcon(ic17);
b8.setIcon(ic18);
b9.setIcon(ic19);
star=b6.getIcon();
starB.setIcon(ic10);
}

else{
sample.setIcon(samicon3);
b1.setIcon(ic21);
b2.setIcon(ic22);
b3.setIcon(ic23);
b4.setIcon(ic24);
b5.setIcon(ic25);
b6.setIcon(ic26);
b7.setIcon(ic27);
b8.setIcon(ic28);
b9.setIcon(ic29);
star=b6.getIcon();
starB.setIcon(ic20);
}

}
}

public static void main(String args[]){
new picpuzzle();
}
}
```