**A Project Report on**

**Bricks Breaker-Game (in java)**

Submitted as a mini-project

By

**K Maldanna**

**Hall Ticket no:160116733034**

**Under the guidance of**

**K Ravi Kiran**

**A Department of CSE, CBIT, Hyderabad**

**Chaitanya Bharathi Institute of Technology**

**Gandipet, Hyderabad – 500075**



**CERTIFICATE**

This is to certify that the project titled “**Bricks Breaker-Game”** is a bona fide work carried out by Maldanna, hall ticket no:160116733034, in completion of a mini-project under our guidance and supervision.

The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide Head of Department

K Ravi Kiran M Swamy Das

Department of CSE Department of CSE

CBIT, Hyderabad CBIT, Hyderabad

**ACKNOWLEDGEMENTS**

I would like to express our heartfelt gratitude to MR.K. Ravikiran, our project guide, for her invaluable guidance and constant support, along with her capable instruction and persistent encouragement.

I am grateful to our Head of Department, Swamy Das for her steady support and the provision of every resource required for the completion of this project.

I would like to take this opportunity to thank our Principal, Dr.Ravindar Reddy, as well as the management of the institute, for having designed an excellent learning atmosphere.

My thanks are due to all members of the staff and our lab assistants for providing me with the help required to carry out the groundwork of this project.

**Abstract:**

This is bricks-breaker game. In this game bricks are present at top place of the screen and at starting point ball is present at bottom, bat also there in the shape of rectangle.

If we will click on start button game will going, if ball touches the bricks which are in the rectangle shape then that corresponding brick will go to disappear then your score will increase but if our ball will reach bottom then you will lose the game so, to prevent the ball reaching bottom we should use bat. if ball will touch the bat again it will go upward direction.

If you will click on Reset button then again game will go to start then you continue your game following same conditions only. In this game if you can score while playing only at right side of the Frame.

This game we can develop in any language but to programmer to develop this game using java language is good option .to attract the player we can add sounds at back ground also, along with this we can change the colour of ball and bat also.

**INTRODUCTION**

**Java Language:**

**Java** is a **programming language** created by James Gosling from Sun Microsystems (Sun) in 1991. ... The first publicly available version of **Java** (**Java** 1.0) was released in 1995. Sun Microsystems was acquired by the Oracle Corporation in 2010

## **What is Java?**

* Java is an [object-oriented programming language](http://www.w3schools.in/java-tutorial/object-oriented-programming-oops/) developed by Sun Microsystems and released in 1995.
* Java was originally developed by **James Gosling** at **Sun Microsystems** (which has since merge into Oracle Corporation).
* Java programs are platform independent which means they can be run on any operating system with any type of processor as long as the [Java interpreter](http://www.w3schools.in/java-tutorial/java-virtual-machine/) is available on that system.
* Java code that runs on one platform does not need to be recompiled to run on another platform, it’s called **write once, run anywhere(WORA)**.
* [Java Virtual Machine (JVM)](https://www.w3schools.in/java-tutorial/java-virtual-machine/) executes Java code, but is written in platform specific languages such as [C](https://www.w3schools.in/c/intro/)/[C++](https://www.w3schools.in/cplusplus/intro/)/ASM etc. JVM is not written in Java and hence **cannot be platform independent** and Java interpreter is actually a part of JVM.

## **Where is Java being Used?**

Earlier java was only used to design and program small computing devices but later adopted as one of the platform independent programming language and now according to Sun, 3 billion devices run java.

Java is one of the most important programming language in today’s IT industries.

* **JSP –**Java is used to create **web applications** like [PHP](https://www.w3schools.in/php/intro/) and ASP, JSP(Java Server Pages) used with normal HTML tags, which helps to create dynamic web pages.
* **Applets –**This is another type of Java program that used within a web page to add many new features to a web browser.
* **J2EE –**The software Java 2 Enterprise Edition are used by various companies to transfer data based on [XML](https://www.w3schools.in/xml/intro/) structured documents between one another.
* **JavaBeans –**This is something like Visual Basic, a reusable software component that can be easily assemble to create some new and advanced application.
* **Mobile** **–**Besides the above technology, Java is also used in mobile devices, many kind of games and services built-in Java. Today, all leading mobile service provider like Nokia, Siemens, Vodafone are using Java technology.

## **Types of Java Applications**

1. **Web Application –** Java is used to create server-side web applications. Currently, servlet, jsp, struts, jsf etc. technologies are used.
2. **Standalone Application –** It is also known as desktop application or window-based application. An application that we need to install on every machine or server such as media player, antivirus etc. AWT and Swing are used in java for creating standalone applications.
3. **Enterprise Application –** An application that is distributed in nature, such as banking applications etc. It has the advantage of high level security, load balancing and clustering. In java, EJB is used for creating enterprise applications.
4. **Mobile Application –** Java is used to create application softwares for mobile devices. Currently Java ME is used for creating applications for small devices, and also Java is programming language for Google Android application development.

## **Facts about Java**

* **Object Oriented**– In java everything is an Object. Java can be easily expanded since it is based on the Object model.
* **Platform independent –**C and C++ are platform dependency languages hence the application programs written in one Operating system cannot run in any other Operating system, but in platform independence language like Java application programs written in one Operating system can able to run on any Operating system.
* **Simple –**Java is designed to be easy to learn. If you understand the basic concept of OOP java would be easy to master.
* **Secure –** With Java’s secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.
* **Architectural-neutral –**Java compiler generates an architecture-neutral object file format which makes the compiled code to be executable on many processors, with the presence Java runtime system.
* **Portable –**being architectural neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler and Java is written in ANSI C with a clean portability boundary which is a POSIX subset.
* **Robust –**Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.
* **Multi-threaded –**With Java’s multi-threaded feature it is possible to write programs that can do many tasks simultaneously. This design feature allows developers to construct smoothly running interactive applications.
* **Interpreted –**Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light weight process.
* **High Performance –**With the use of Just-In-Time compilers Java enables high performance.
* **Distributed –**Java is designed for the distributed environment of the internet.
* **Dynamic –**Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry an extensive amount of run-time information that can be used to verify and resolve accesses to objects on run-time.

## **Different Editions of Java Technology**

* **Java SE**– Java SE or Java Standard Edition provides tools and API’s that you can use to create server applications, desktop applications, and even applets. These programs developed using Java SE can be run on almost every popular operating system, including Linux, Macintosh, Solaris, and Windows.
* **JEE** – Based on the foundation framework of the standard edition, Java Enterprise Edition helps in web application service, component model and enterprise class service oriented architecture (SOA).
* **JME** – Java Micro Edition or JME for short is an accumulation of Java APIs that are used for the development of software for devices like mobile phones, PDAs, TV set-top boxes, game programming. The platform of micro edition generally consists of an easy user interface, a robust security model and a wide variety of built-in networks for running Java based application.

## **What you will need to run Java**

* You will need the Java software development kit from http://java.sun.com
* Follow the instructions on Sun’s website to install it.
* Make sure that you add the java bin directory to your [PATH environment variable](http://www.w3schools.in/java/install/).

## **Popular Java Editors**

To write your java programs you will need a text editor. There are even more sophisticated IDE available in the market. But for now, you can consider one of the following:

* **Notepad –**On Windows machine you can use any simple text editor like Notepad (Recommended for this tutorial), TextPad.
* **Netbeans** – is a Java IDE that is open source and free which can be downloaded from http://www.netbeans.org/ind
* **Eclipse –**is also a java IDE developed by the eclipse open source community and can be downloaded from http://www.eclipse.org

**REQUIREMENTS TO IMPLEMENT THIS PROJECT**

**Software Requirements**

Any 32-bit or 64-bit operating system that supports java Language

Any text editor

A console with access to the java interpreter

**Hardware Requirements**

Input device to answer questions and select options.

32-bit or 64-bit processor

Sufficient RAM to run the program

**Concepts involved:**

**1.GUI Applications**

* **Buttons**
* **Labels**
* **Layout Manager**
* **Text Field**
* **Frames**
* **Paint method**
* **Threads**

**Packages:**

* **import javax.swing.JButton;**
* **import javax.swing.JFrame;**
* **import javax.swing.JPanel;**

# **Java JPanel**

The JPanel is a simplest container class. It provides space in which an application can attach any other component. It inherits the JComponents class.

It doesn't have title bar.

## **JPanel class declaration**

1. **public** **class** JPanel **extends** JComponent **implements** Accessible

# **Java 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.

# **Java JLabel**

The object of JLabel class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly. It inherits JComponent class.

# **Java JTextField**

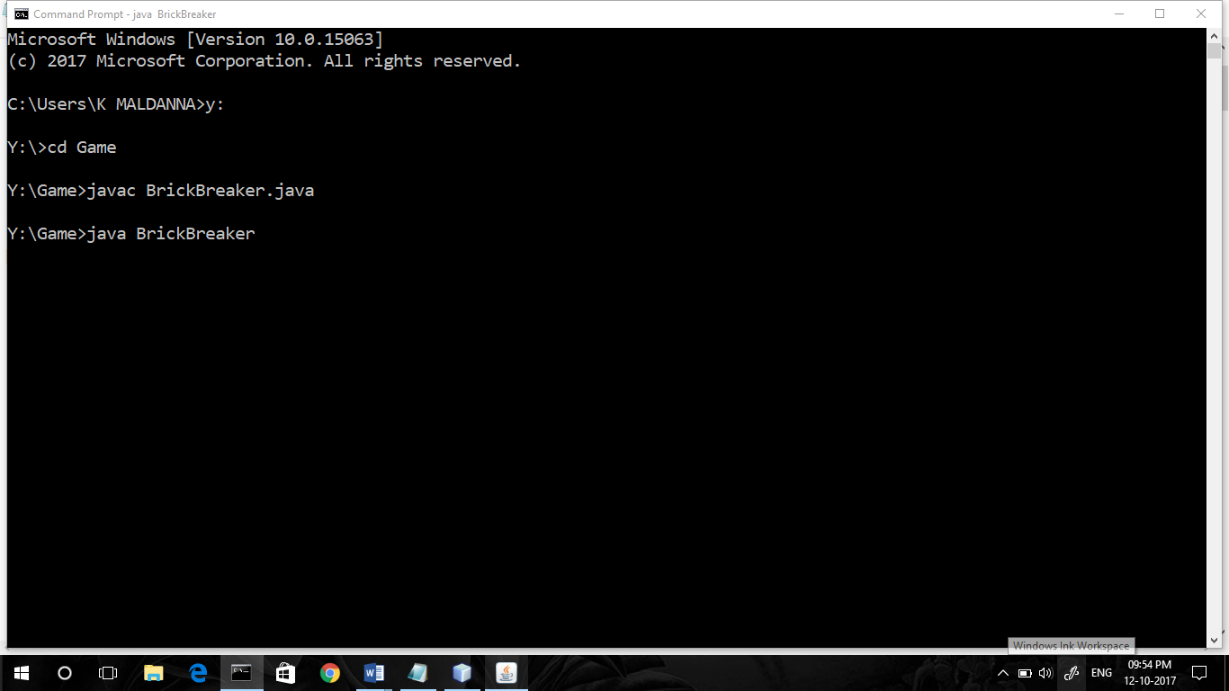
The object of a JTextField class is a text component that allows the editing of a single line text. It inherits JTextComponent class.

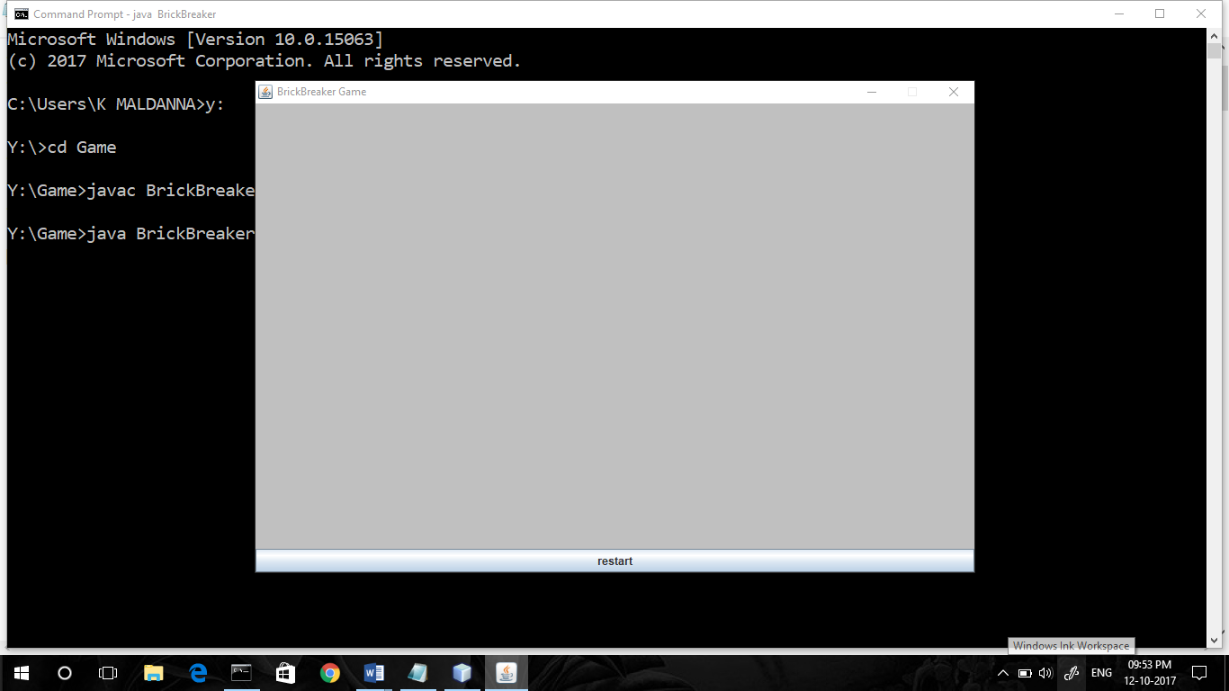
## **JTextField class declaration**

Let's see the declaration for javax.swing.JTextField class

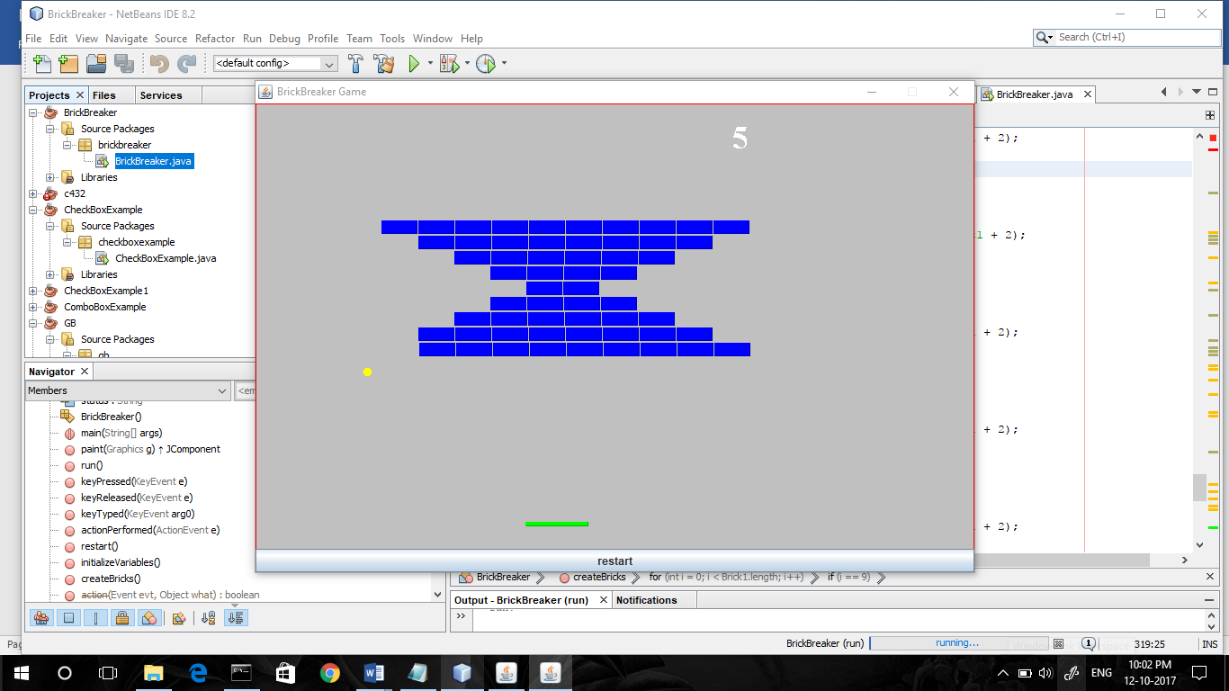
**FLOW OF EXECUTION:**

**Step:1** First main method will execute, in this one first Frame will create at a same time another thread will go to create then new thread will go to call run method

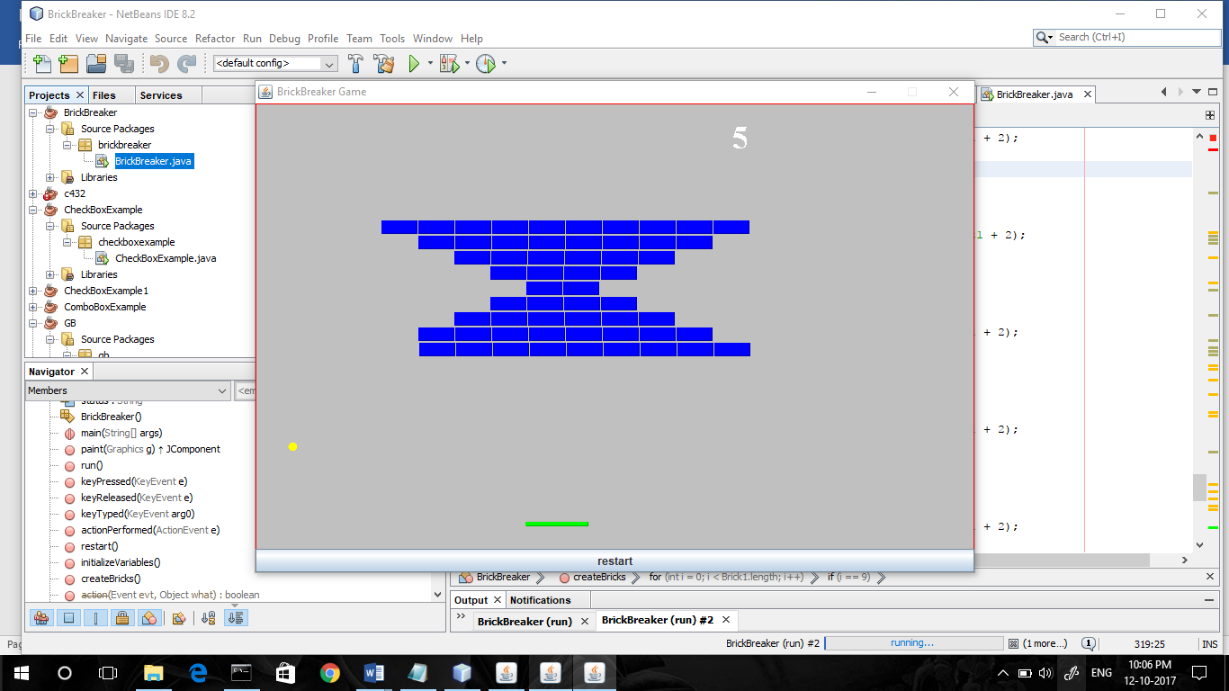




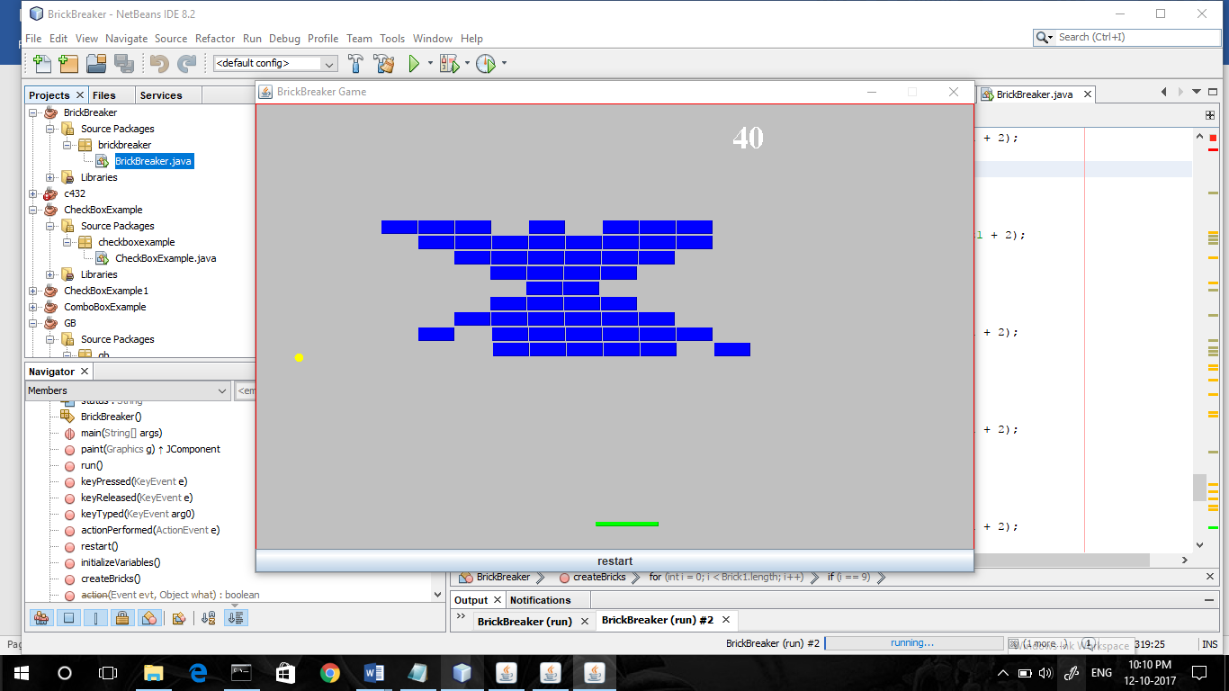
**Step 2**: first it will create the bricks this break method will call in run method



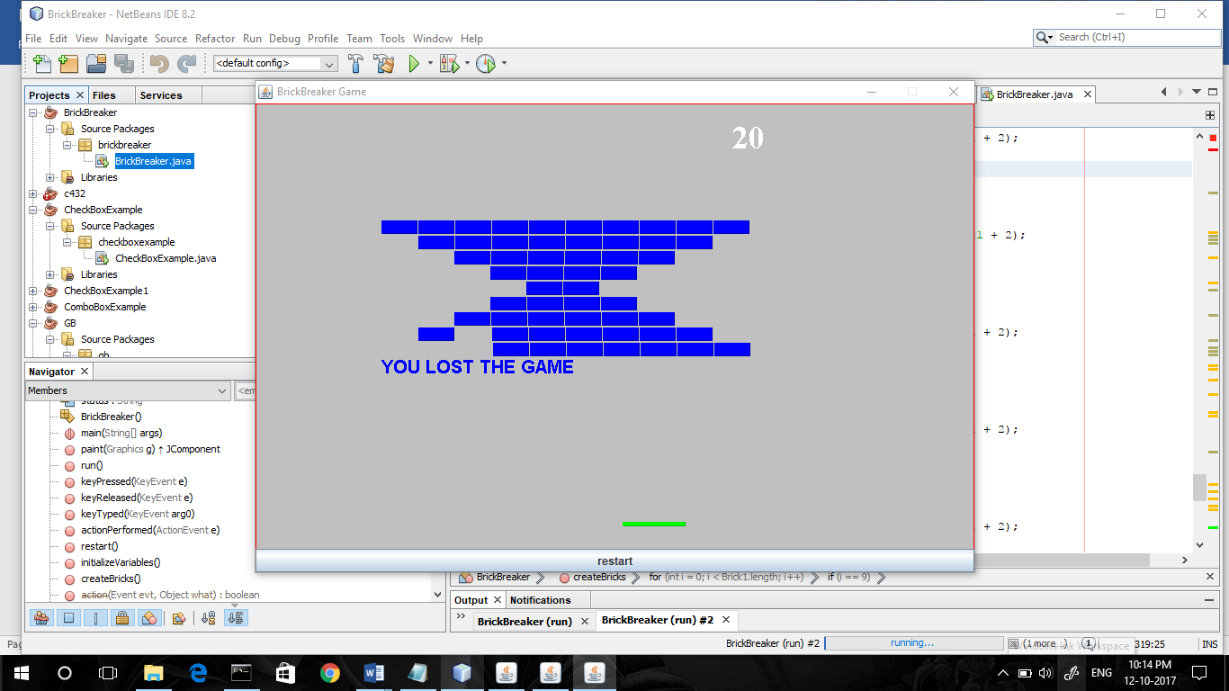
Step 3: After creating the bricks then paint method will go to call then ball and bat.



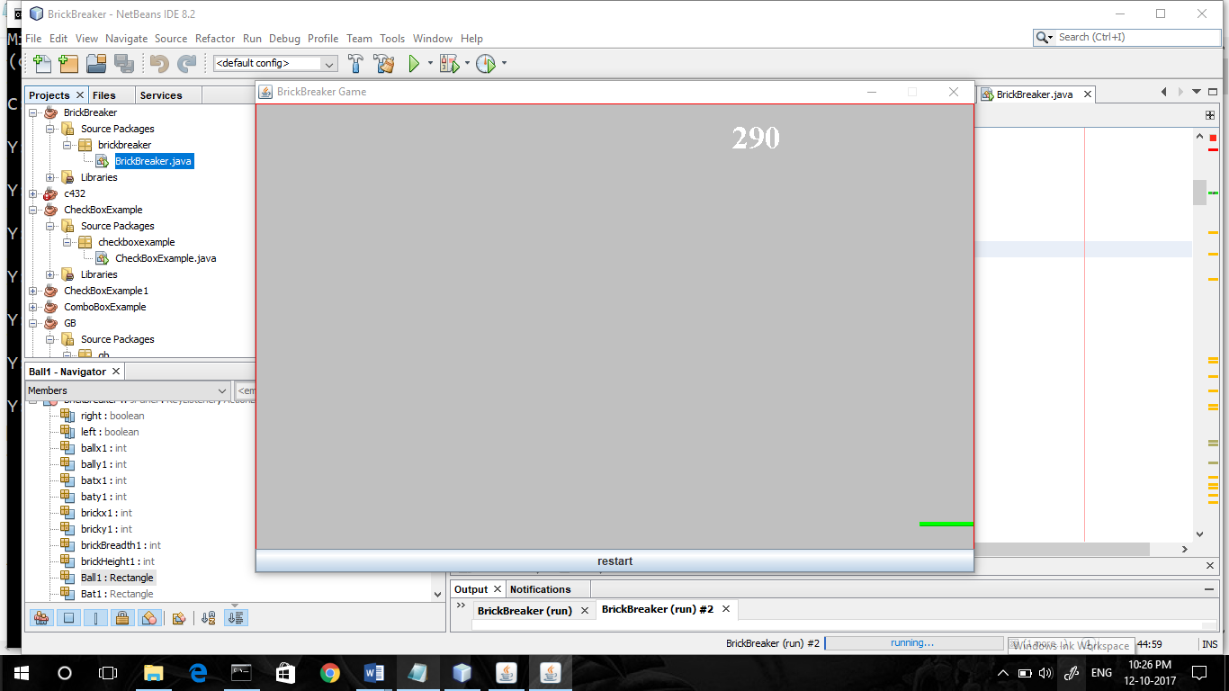
Step 4: if ball touch the bicker then 5 will add to your score it appears at top right side of Frame and if ball touch the bat then it will go to reverse direction



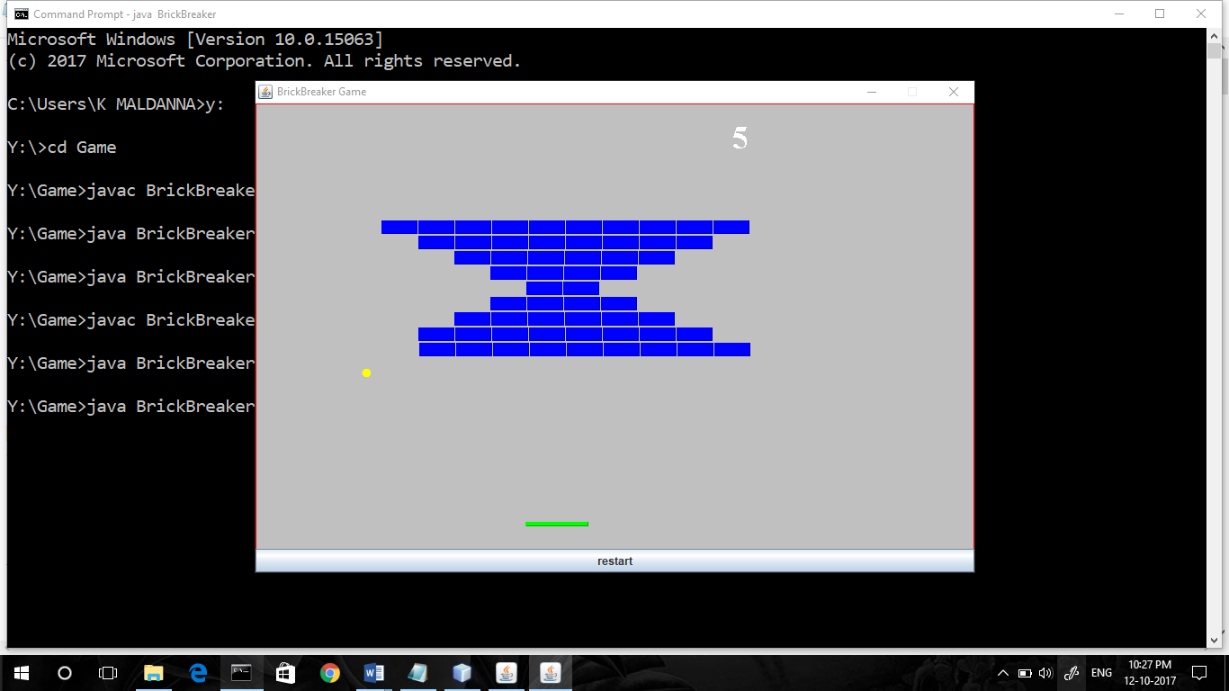
Step 5:if ball will go to at bottom of the Frame then it you lost the game will print on the Frame then game will go to stop. if all bricks are over then you won game print appear on Frame.



Step 5: if all bricks are over then you won the game click on the restart button then new game will go to start.



Restart the game:



**Coding in Java Language:**

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

/\* \*\*\*\*\*\*\*\*\*\*below classes import to use in this BrickBreaker Game

import java.awt.BorderLayout;

import java.awt.Color;

import java.awt.Font;

import java.awt.Graphics;

import java.awt.Rectangle;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.KeyEvent;

import java.awt.event.KeyListener;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JPanel;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

public class BrickBreaker extends JPanel implements KeyListener,ActionListener, Runnable

{

static boolean right = false;

static boolean left = false;

// \*\*\*\*\*\* intial position of ball \*\*\*\*\*\*\*

int ballx1 = 320;

int bally1 = 448;

// \*\*\*\*\* intial position of bat \*\*\*\*

int batx1 = 300;

int baty1 = 465;

// \*\*\*\*\*\* first brick position \*\*\*\*\*\*

int brickx1 = 140;

int bricky1 = 130;

int brickBreadth1 = 40;

int brickHeight1 = 15;

// \*\*\*\*\*\*\* creating objects for both ball ,bat and Brick \*\*\*\*\*

Rectangle Ball1 = new Rectangle(ballx1, bally1, 10, 10);

Rectangle Bat1 = new Rectangle(batx1, baty1, 70, 5);

Rectangle[] Brick1 = new Rectangle[58];

// \*\*\*\*\*\*\*\*\*\*\*\*\*\* speed and direction of the ball \*\*\*\*\*\*\*\*\*\*\*\*

int movex1 = -1;

int movey1 = -1;

boolean ballFallDown = false;

boolean bricksOver = false;

int count = 0;

String status;

BrickBreaker()

{

}

public static void main(String[] args)

{

JFrame frame = new JFrame("BrickBreaker Game");

BrickBreaker game = new BrickBreaker();

JButton button = new JButton("restart");

frame.setSize(805, 550);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.add(game);

frame.add(button, BorderLayout.SOUTH);

frame.setLocationRelativeTo(null);

frame.setResizable(false);

frame.setVisible(true);

button.addActionListener(game);

game.addKeyListener(game);

game.setFocusable(true);

Thread t = new Thread(game);

t.start();

}

public void paint(Graphics g)

{

g.setColor(Color.LIGHT\_GRAY);

g.fillRect(0, 0, 799, 500);

if(Ball1.y>=467)

{ g.setColor(Color.yellow);

g.fillOval(993,589, Ball1.width, Ball1.height);

}

g.setColor(Color.yellow);

g.fillOval(Ball1.x, Ball1.y, Ball1.width, Ball1.height);

g.setColor(Color.green);

g.fill3DRect(Bat1.x, Bat1.y, Bat1.width, Bat1.height, true);

g.setColor(Color.red);

g.drawRect(0, 0, 798, 500);

g.setColor(Color.white);

g.setFont(new Font("serif",Font.BOLD,35));

g.drawString(""+count, 530, 50);

g.setColor(Color.blue);

for (int i = 0; i < Brick1.length; i++)

{

if (Brick1[i] != null)

{

g.fill3DRect(Brick1[i].x, Brick1[i].y, Brick1[i].width,

Brick1[i].height, true);

}

}

if (ballFallDown == true || bricksOver == true)

{

Font f = new Font("Arial", Font.BOLD, 20);

g.setFont(f);

g.drawString(status, 140, 300);

ballFallDown = false;

bricksOver = false;

}

}

public void run()

{ createBricks();

while (true)

{

// if(gameOver == true){return;}

for (int i = 0; i < Brick1.length; i++)

{ if (Brick1[i] != null)

{

if (Brick1[i].intersects(Ball1))

{

Brick1[i] = null;

// movex = -movex;

movey1 = -movey1;

count+=5;

}

}

}

if (count == 290)

{

bricksOver = true;

status = "YOU WON THE GAME";

repaint();

}

repaint();

Ball1.x += movex1;

Ball1.y += movey1;

if (left == true)

{ Bat1.x -= 3;

right = false;

}

if (right == true)

{

Bat1.x += 3;

left = false;

}

if (Bat1.x <= 0)

{ Bat1.x = 0;

}

else if (Bat1.x >= 738)

{ Bat1.x = 738;

}

if (Ball1.intersects(Bat1))

{

movey1 = -movey1;

// if(Ball1.y + Ball1.width >=Bat1.y)

}

if (Ball1.x <= 0 || Ball1.x + Ball1.height >=798)

{ movex1 = -movex1;

}

if (Ball1.y <= 0)

{ movey1 = -movey1;

}

if (Ball1.y >= 465)

{ // when ball falls below bat game is over...

ballFallDown = true;

status = "YOU LOST THE GAME";

repaint();

}

Try

{ Thread.sleep(6);

}

catch (Exception ex)

{ }

}

}

public void keyPressed(KeyEvent e) {

int keyCode = e.getKeyCode();

if (keyCode == KeyEvent.VK\_LEFT) {

left = true;

}

if (keyCode == KeyEvent.VK\_RIGHT) {

right = true;

}

}

// \*\*\*\*\*\*\*\*\*\*\* keys events \*\*\*\*\*\*\*\*\*\*\*\*\*

public void keyReleased(KeyEvent e)

{ int keyCode = e.getKeyCode();

if (keyCode == KeyEvent.VK\_LEFT)

{ left = false;

}

if (keyCode == KeyEvent.VK\_RIGHT)

{ right = false;

}

}

public void keyTyped(KeyEvent arg0)

{

}

public void actionPerformed(ActionEvent e)

{

String str = e.getActionCommand();

if (str.equals("restart"))

{

this.restart();

}

}

public void restart()

{ requestFocus(true);

initializeVariables();

createBricks();

repaint();

}

public void initializeVariables()

{

ballx1 = 300;

bally1= 448;

batx1 = 300;

baty1 = 465;

brickx1 = 140;

bricky1 = 130;

Ball1 = new Rectangle(ballx1, bally1, 10, 10);

Bat1 = new Rectangle(batx1, baty1, 70, 5);

Brick1 = new Rectangle[58];

movex1 = -1;

movey1 = -1;

ballFallDown = false;

bricksOver = false;

count = 0;

status = null;

}

public void createBricks()

{

for (int i = 0;i<Brick1.length; i++)

{

Brick1[i] = new Rectangle(brickx1, bricky1, brickBreadth1, brickHeight1);

if (i ==9)

{

brickx1 = 140+0\*brickBreadth1;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 17)

{

brickx1 = 140+1\*brickBreadth1;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 23)

{

brickx1 = 140+2\*brickBreadth1;;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 27)

{

brickx1 = 140+3\*brickBreadth1;;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 29)

{

brickx1 = 140+2\*brickBreadth1;;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 33)

{ brickx1 = 140+1\*brickBreadth1;;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 39)

{

brickx1 = 140+0\*brickBreadth1;;

bricky1 = (bricky1 + brickHeight1 + 2);

}

if (i == 47)

{

brickx1 = 140-40;

bricky1 = (bricky1 + brickHeight1 + 2);

}

brickx1 += (brickBreadth1+1);

}

}

}