```
package DemoGame;

import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
import javax.swing.JPanel;
import javax.swing.Timer;
import java.awt.Rectangle;

public class GamePlay extends JPanel implements KeyListener , ActionListener {
    private boolean play = false;
    private int score =0;
    private int totalbricks = 50;
    private final Timer Timer;
    private final int delay =8;
    private int playerX = 310;
    private int ballposX=120;
    private int ballposY = 350;
    private int ballXdir = -1;
    private int ballYdir = -2;
    private MapGenerator map;

    public GamePlay()
    {
        map = new MapGenerator(5,10);
        addKeyListener(this);
        setFocusable(true);
        setFocusTraversalKeysEnabled(false);
        Timer = new Timer(delay,this);
        Timer.start();
    }
    @Override
    public void paint(Graphics g){
        g.setColor(Color.black);
        g.fillRect(1,1,692,592);
        map.draw((Graphics2D)g);

        //border
        g.setColor(Color.yellow);
```

```
    g.fillRect(0,0,10,592);
    g.fillRect(0,0,692,10);
    g.fillRect(691,3,10,592);

    g.setColor(Color.white);
    g.setFont(new Font("serif",Font.BOLD,25));
    g.drawString(""+score,590,30);

    g.setColor(Color.yellow);
    g.fillRect(playerX,550,100,8);

    //ball
    g.setColor(Color.yellow);
    g.fillOval(ballposX,ballposY,20,20);


    if(ballposY>570)
    {
       play = false;
       ballXdir =0;
       ballYdir = 0;
       g.setColor(Color.red);
       g.setFont(new Font("Serif",Font.BOLD,30));
       g.drawString("  GAME OVER SCORE : "+ score,190,300);

       g.setFont(new Font("serif",Font.BOLD,30));
       g.drawString("  Press ENTER to Restart ", 190, 340);
    }
    if(totalbricks==0)
    {
        play = false;
       ballXdir =-1;
       ballYdir = -2;
       g.setColor(Color.red);
       g.setFont(new Font("Serif",Font.BOLD,30));
       g.drawString("  GAME COMPLETED SCORE : "+ score,190,300);

       g.setFont(new Font("serif",Font.BOLD,30));
       g.drawString(" Press ENTER to Restart ", 190, 340);
    }
    g.dispose();
}

@Override
```

```
public void keyPressed(KeyEvent e) {
   if(e.getKeyCode()==KeyEvent.VK_RIGHT)
   {
     if(playerX>=600)
        playerX = 600;
     else
        MoveRight();
   }
   if(e.getKeyCode()==KeyEvent.VK_LEFT)
   {
     if(playerX<10)
        playerX = 10;
     else
        MoveLeft();
   }
   if(e.getKeyCode()==KeyEvent.VK_ENTER)
   {
      if(!play)
      {
         ballposX = 120;
         ballposY = 350;
         ballXdir = -1;
         ballYdir = -2;
         score = 0;
         playerX = 310;
         totalbricks = 50;
         map = new MapGenerator(5,10);
         repaint();
      }
   }
}
public void MoveRight()
{
   play = true;
   playerX +=20;
}
public void MoveLeft()
{
   play = true;
   playerX -=20;
}


@Override
```

```
    public void actionPerformed(ActionEvent e) {
        Timer.start();
        if(play)
        {
            if(new Rectangle(ballposX,ballposY,20,20).intersects(new
Rectangle(playerX,550,100,8)))
            {
                ballYdir = - ballYdir;
            }
            A :
            for(int i=0;i<map.map.length;i++)
            {
                for(int j=0;j<map.map[0].length;j++)
                {
                    if(map.map[i][j]>0)
                    {
                        int brickX = j*map.brickwidth+80;
                        int brickY = i*map.brickheight+50;
                        int brickwidth = map.brickwidth;
                        int brickheight = map.brickheight;

                        Rectangle rect = new Rectangle(brickX,brickY,brickwidth,brickheight);
                        Rectangle ballrect = new Rectangle(ballposX,ballposY,20,20);
                        Rectangle brickrect = rect;
                        if(ballrect.intersects(brickrect))
                        {
                            map.setBricksValue(0, i, j);
                            totalbricks--;
                            score+=5;
                            if(ballposX+19<=brickrect.x || ballposX+1>=brickrect.x+brickwidth)
                            {
                                ballXdir = -ballXdir;
                            }
                            else
                            {
                                ballYdir = -ballYdir;
                            }
                            break A;
                        }
                    }
                }
            }
            ballposX +=ballXdir;
            ballposY+=ballYdir;
```

```
            if(ballposX<0)
            {
                ballXdir = -ballXdir;
            }
            if(ballposY<0)
            {
                ballYdir = - ballYdir;
            }
            if(ballposX>670)
            {
                ballXdir = -ballXdir;
            }
        }
        repaint();
    }
     @Override
    public void keyReleased(KeyEvent e) {
        throw new UnsupportedOperationException("Not supported yet."); // Generated from
nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody
    }
     @Override
    public void keyTyped(KeyEvent e) {
        throw new UnsupportedOperationException("Not supported yet."); // Generated from
nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody
    }


}
```