**JAVA LAB -Part B(II sem)**

1. **Program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values.**

|  |
| --- |
| class B01NegativeSize{  public static void main(String args[]){  try{  int[] a = new int[-5];  }  catch(NegativeArraySizeException e){  System.out.println("Negative Array Size!!");  }  System.out.println("Continuing execution... ");  }  } |

**Output -**

Negative Array Size!!

Continuing execution…

1. **Program to handle Null Pointer Exception and use the “finally” method to display a message to the user.**

|  |
| --- |
| public class B02NullPointer{  public static void main(String args[]){  String str = null;  //Try-catch-finally blocks  try{  // Below code will give null pointer exception as we are access null str  // Note that string is a character array. And we are trying to access the 1st character of this string.  System.out.println("First character of str is:" + str.charAt(0));  }  catch(NullPointerException e){  System.out.println("NullPointerException Caught in catch block...");  }  finally{  System.out.println("Finally is executed always");  }  } } |

**Output-**

NullPointerException Caught in catch block…

Finally is executed always

1. **Program which creates and displays a message on the window.**

**HELLOAPPLET.JAVA:-**

import java.applet.\*;

import java.awt.\*;

public class HelloApplet extends Applet

{

public void paint(Graphics g)

{

g.drawString("HELLOAPPLET",10,20);

}

}

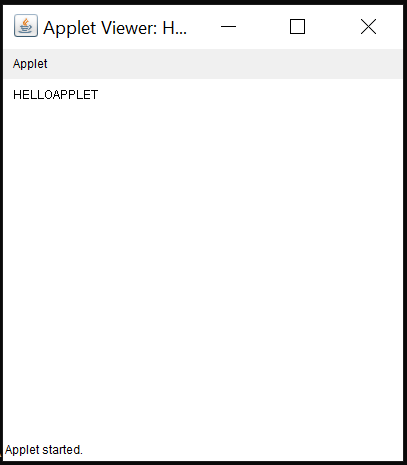
**a.html:-**

<html>

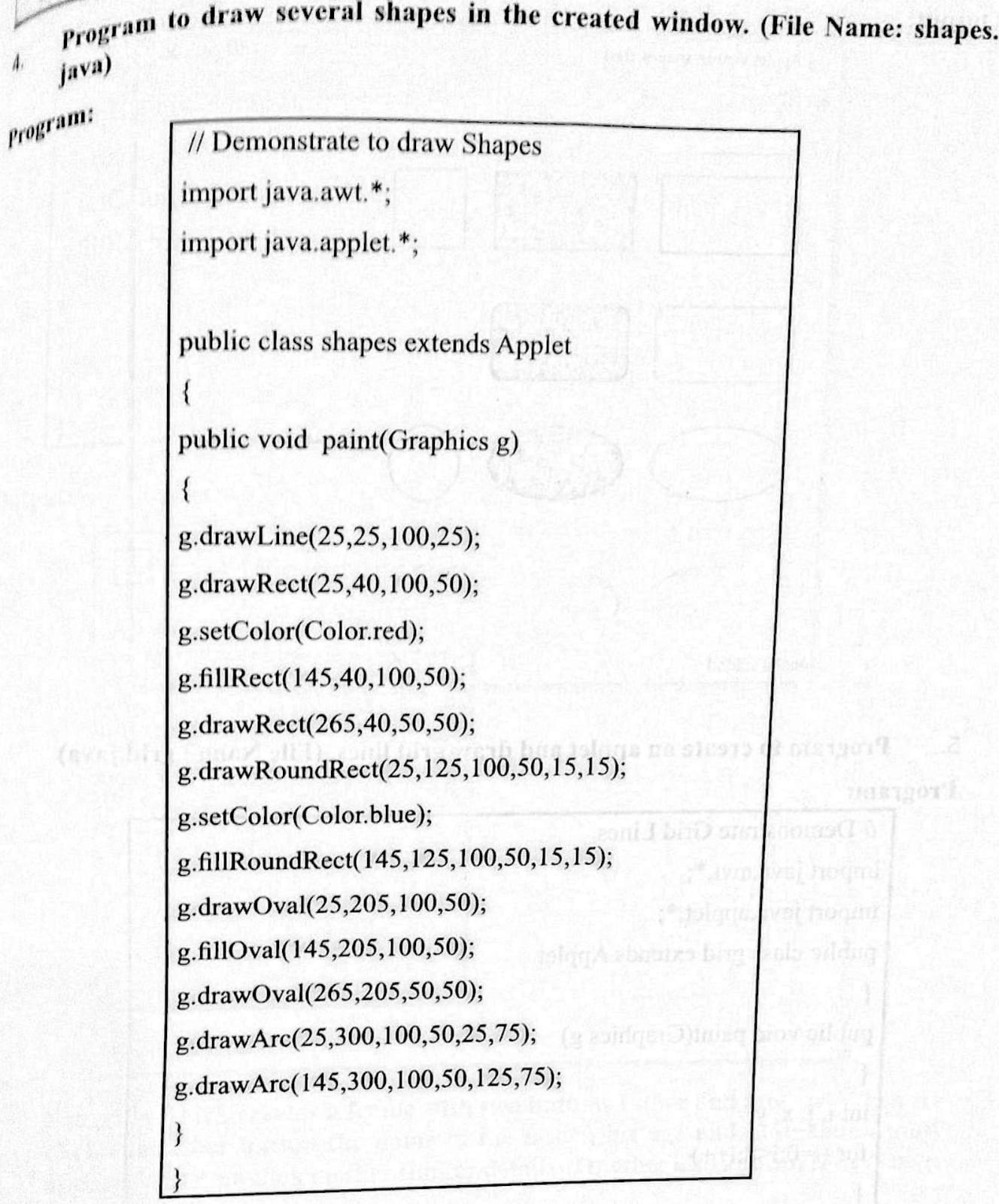
<applet code="HelloApplet.class" width=400 height=400>

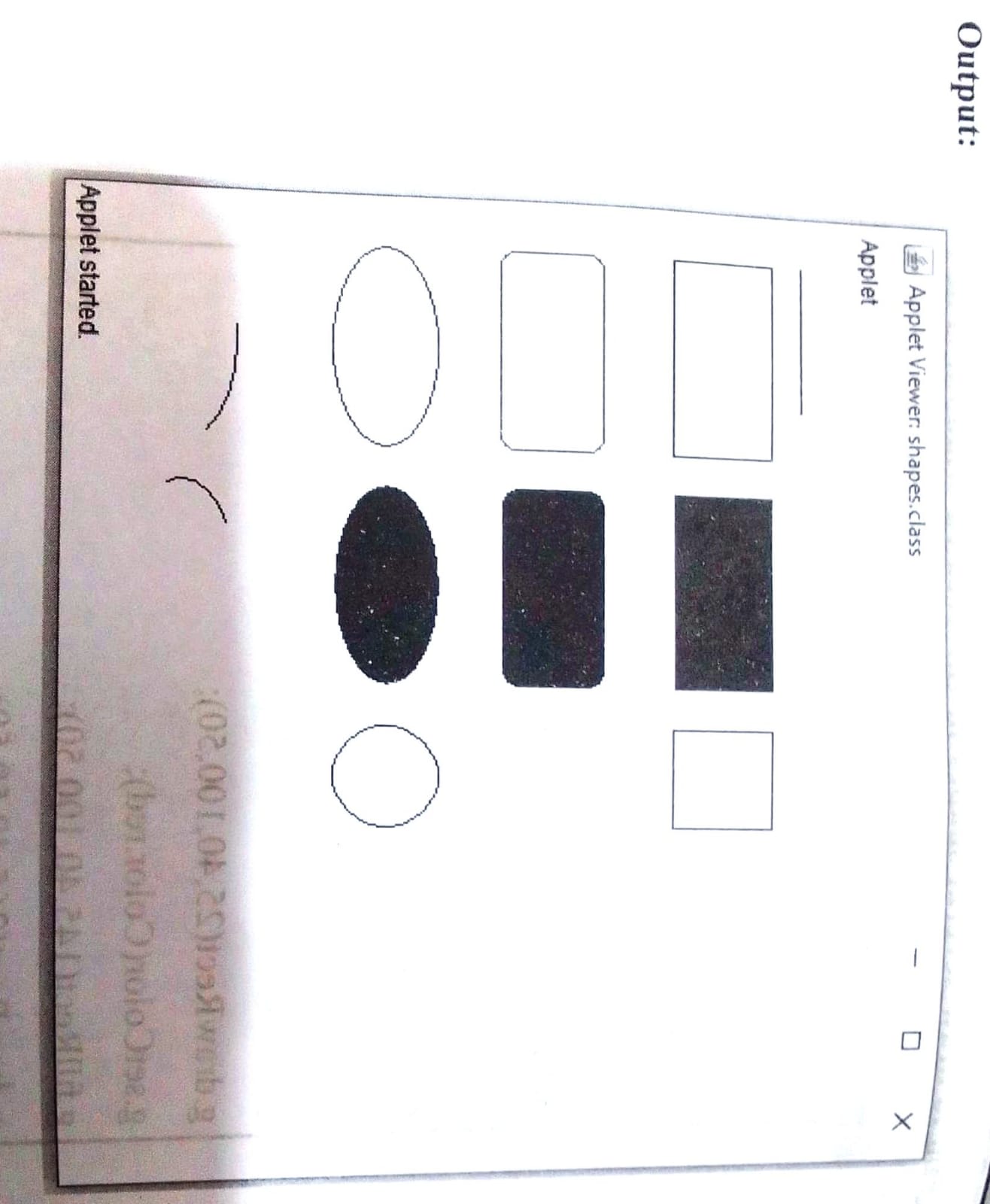
</applet> </html>

**Output -**

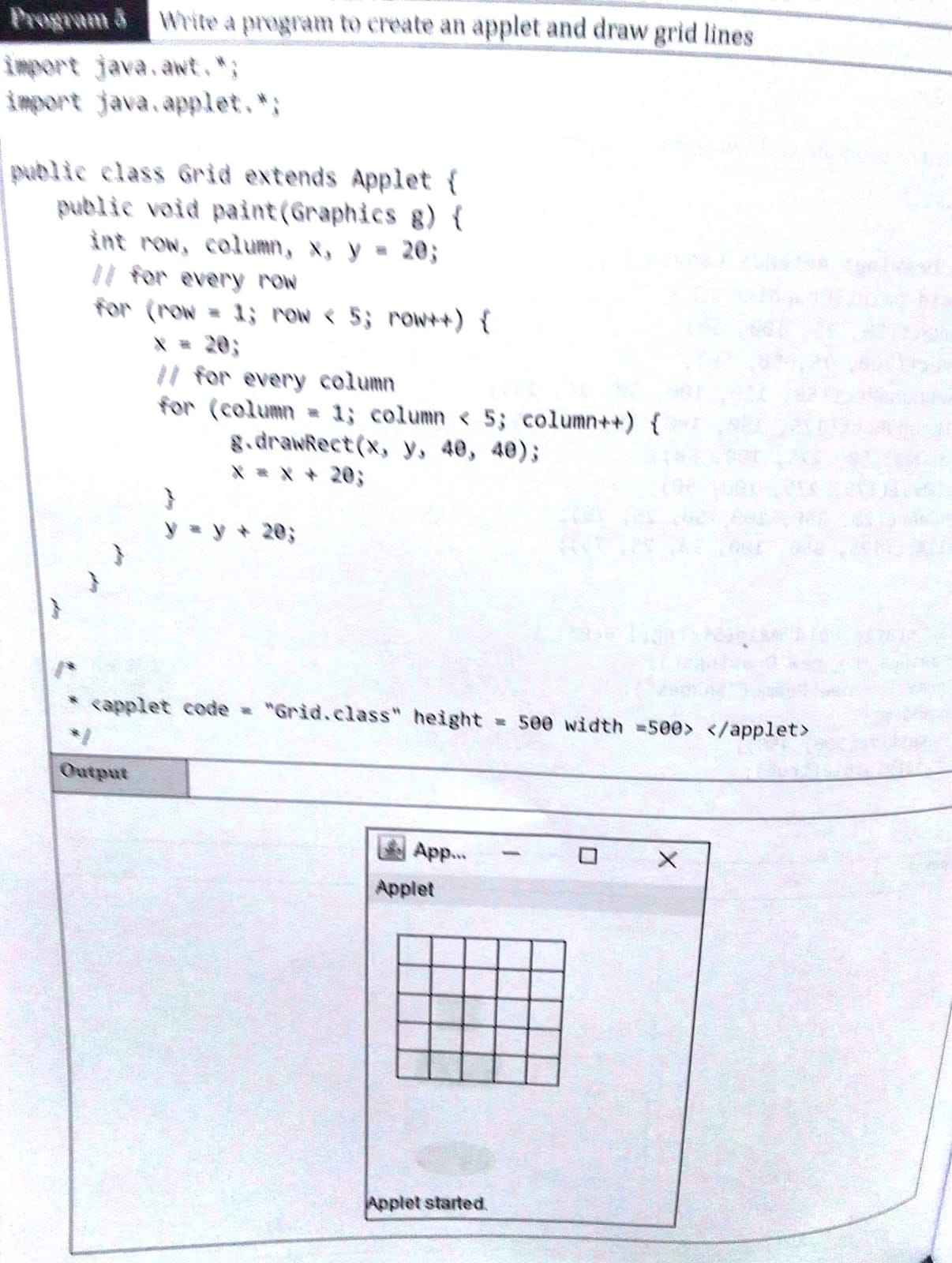
****

1. **Program to draw several shapes in the created window**

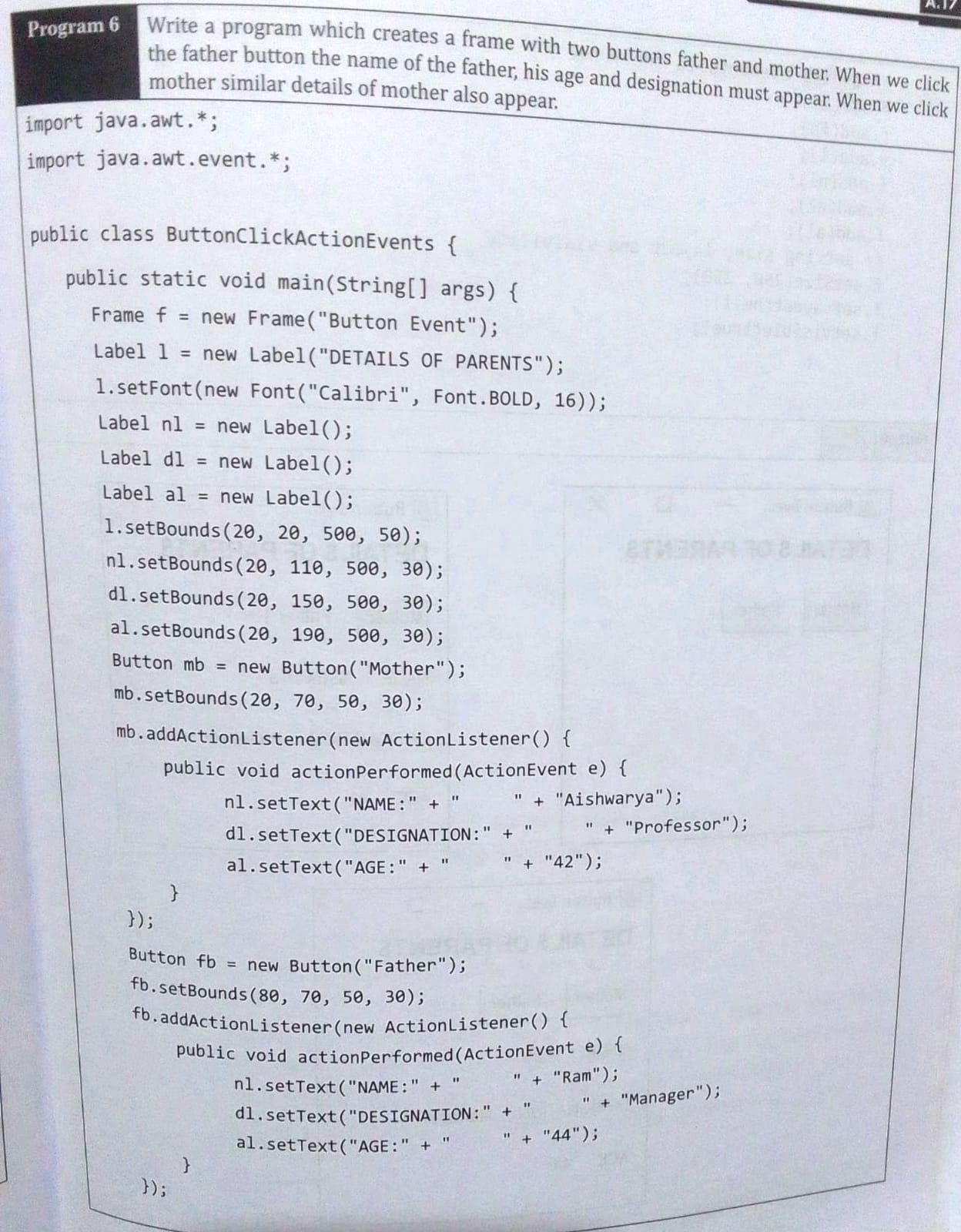


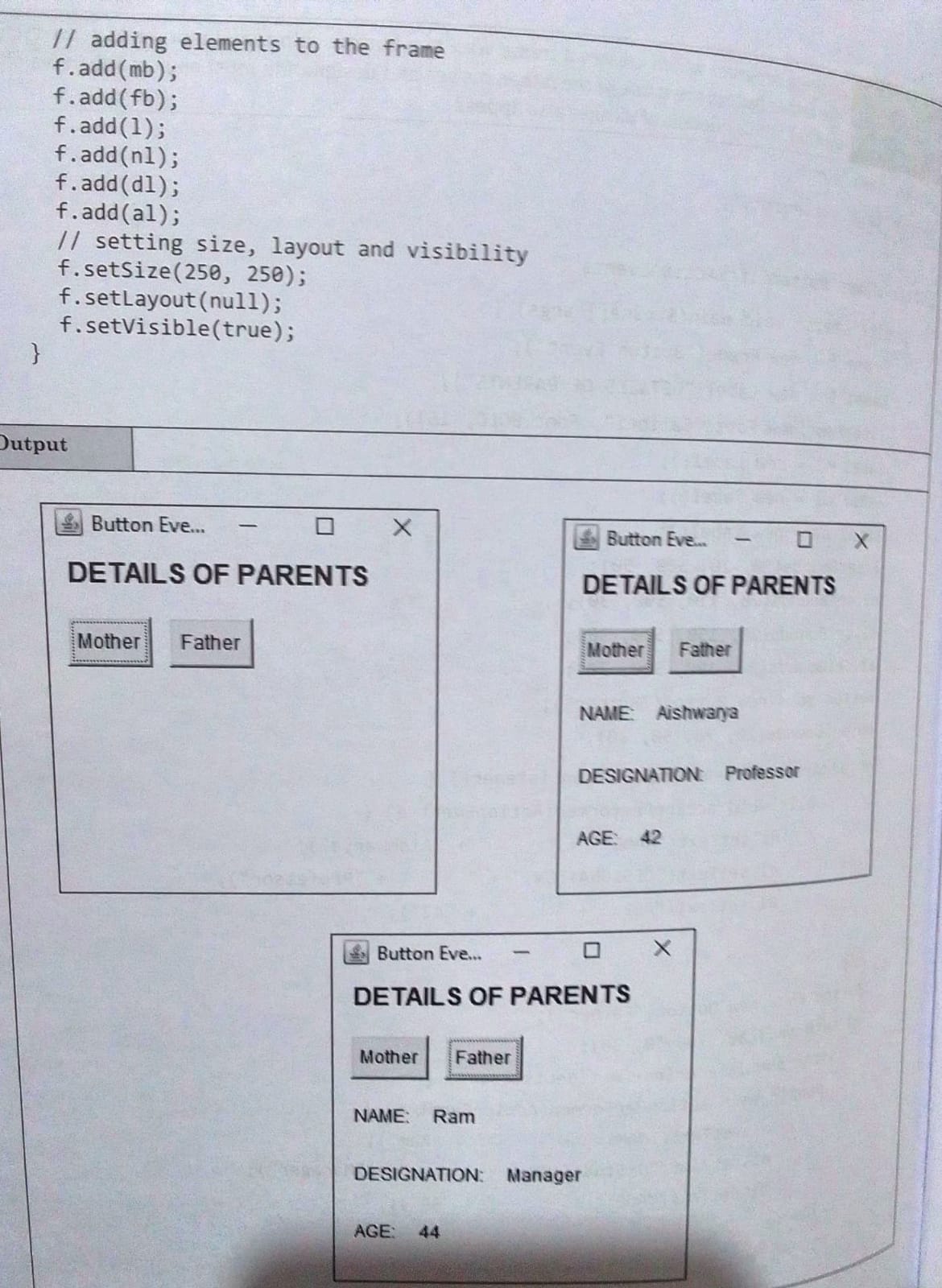


1. **Write a program to create an applet and grid lines.**

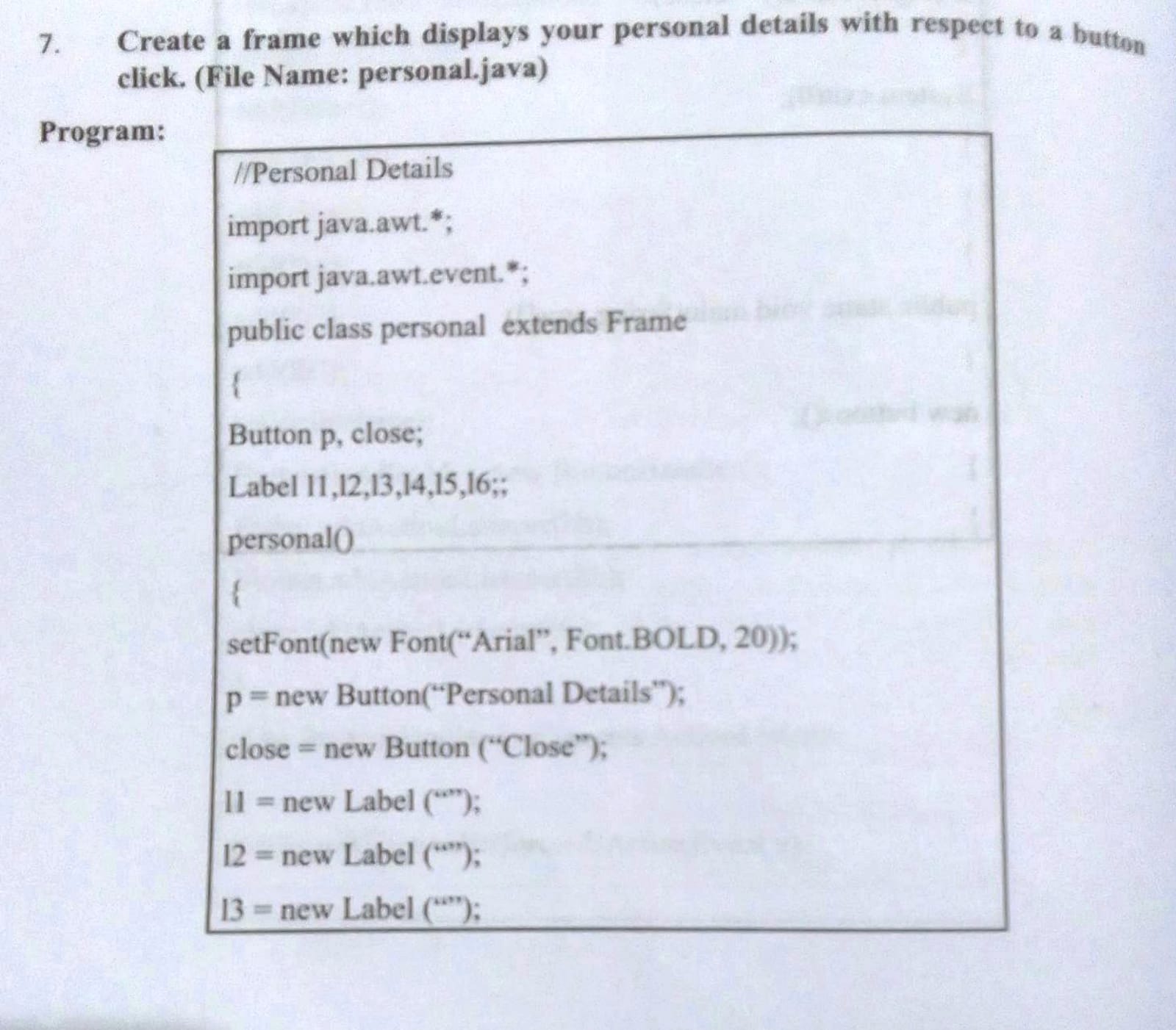


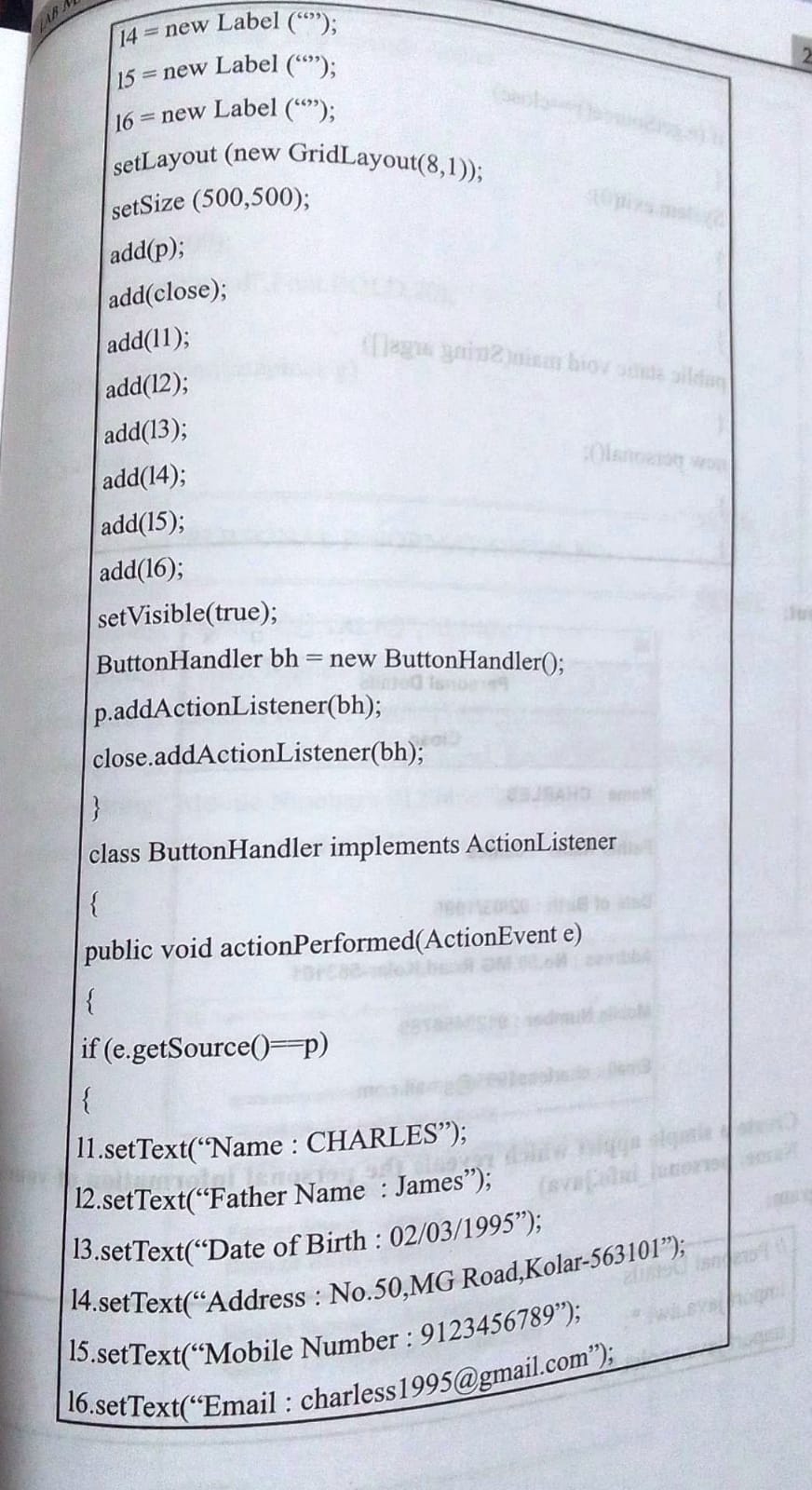
**6./\*Program –Buttons \*/**

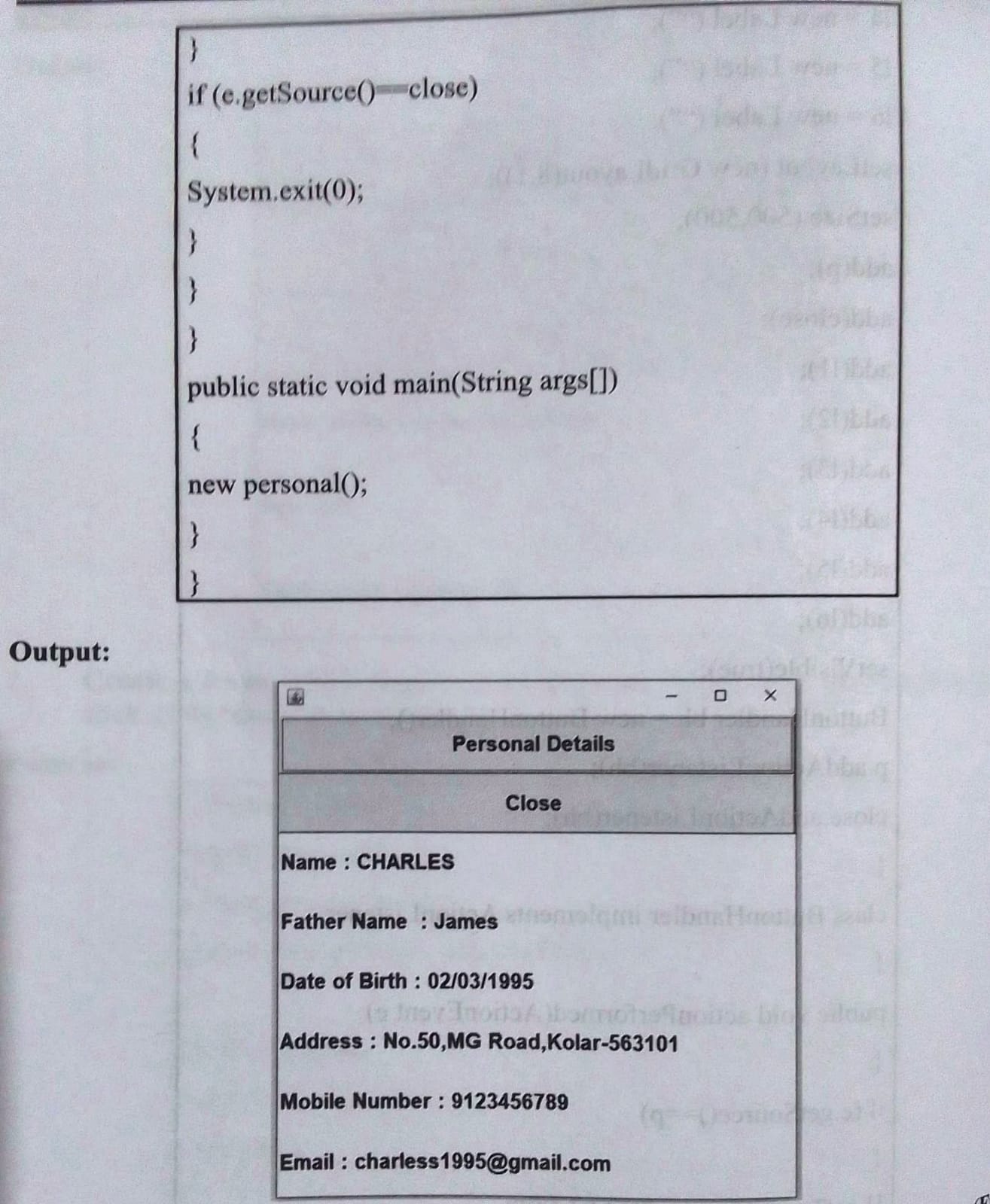




**7./\* Program 7—personal details with button click \*/**



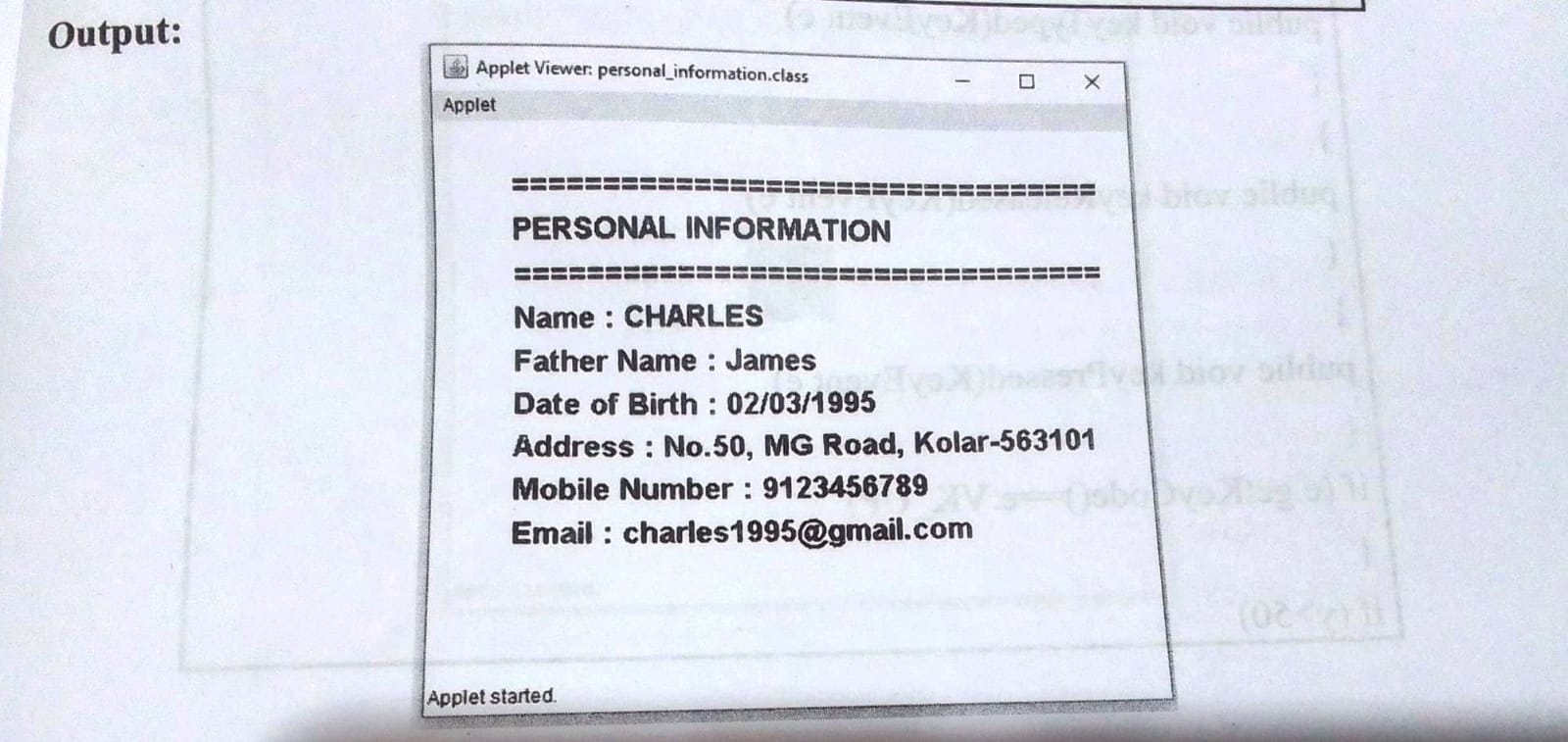




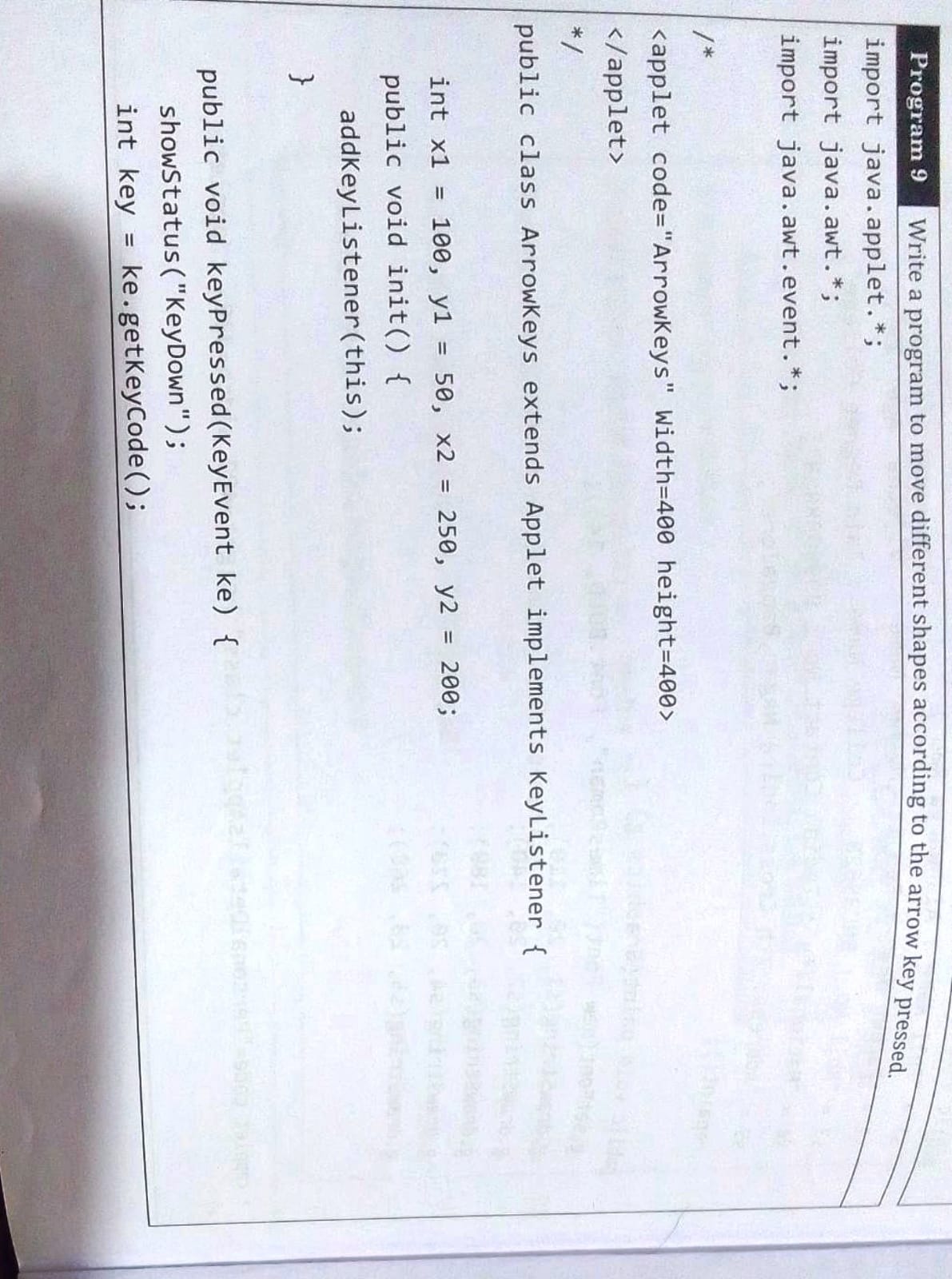
**8. Create a simple applet which reveals the personal information of yours**

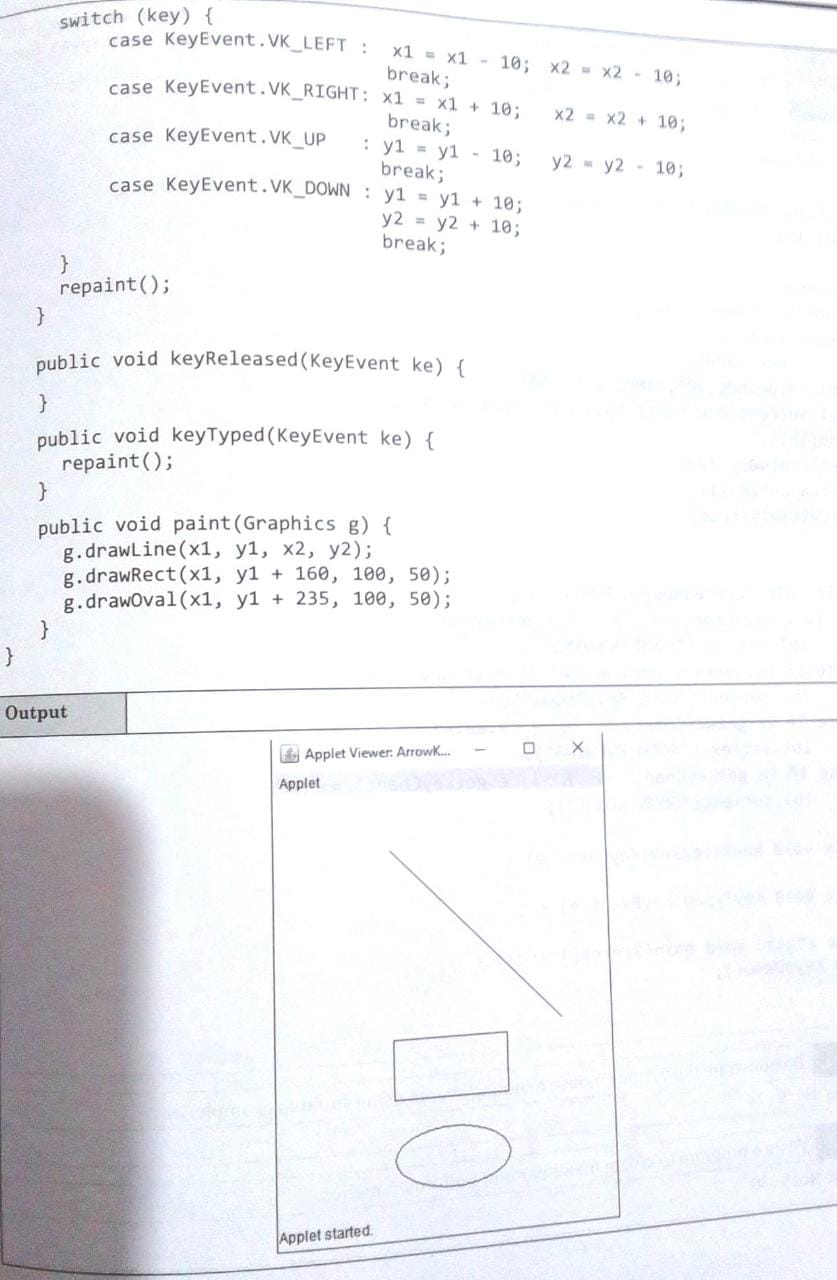
|  |
| --- |
| import java.awt.\*; import java.applet.\*;  public class Personal extends Applet{  Font f;  public void init(){  setSize(500,500);  f=new Font ("Arial",Font.BOLD,20);  }  public void paint(Graphics g){  g.drawString("=============", 50, 50);  g.drawString("Personal Information", 50, 80);  g.drawString("=============", 50, 110);  g.drawString("Name:Charles", 50, 140);  g.drawString("Father Name:James", 50, 170);  g.drawString("Date of Birth:02/03/1995", 50, 200);  g.drawString("Address No:50,MG Road,Kolar", 50, 230);  g.drawString("Mobile Number:9123456789", 50, 260);  g.drawString("E-mail:[charles1995@gmail.com](mailto:charles1995@gmail.com)", 50, 290);  } }  /\* <applet code=”Personal.class” height=300 width=500> </applet> \*/ |

**Output -**

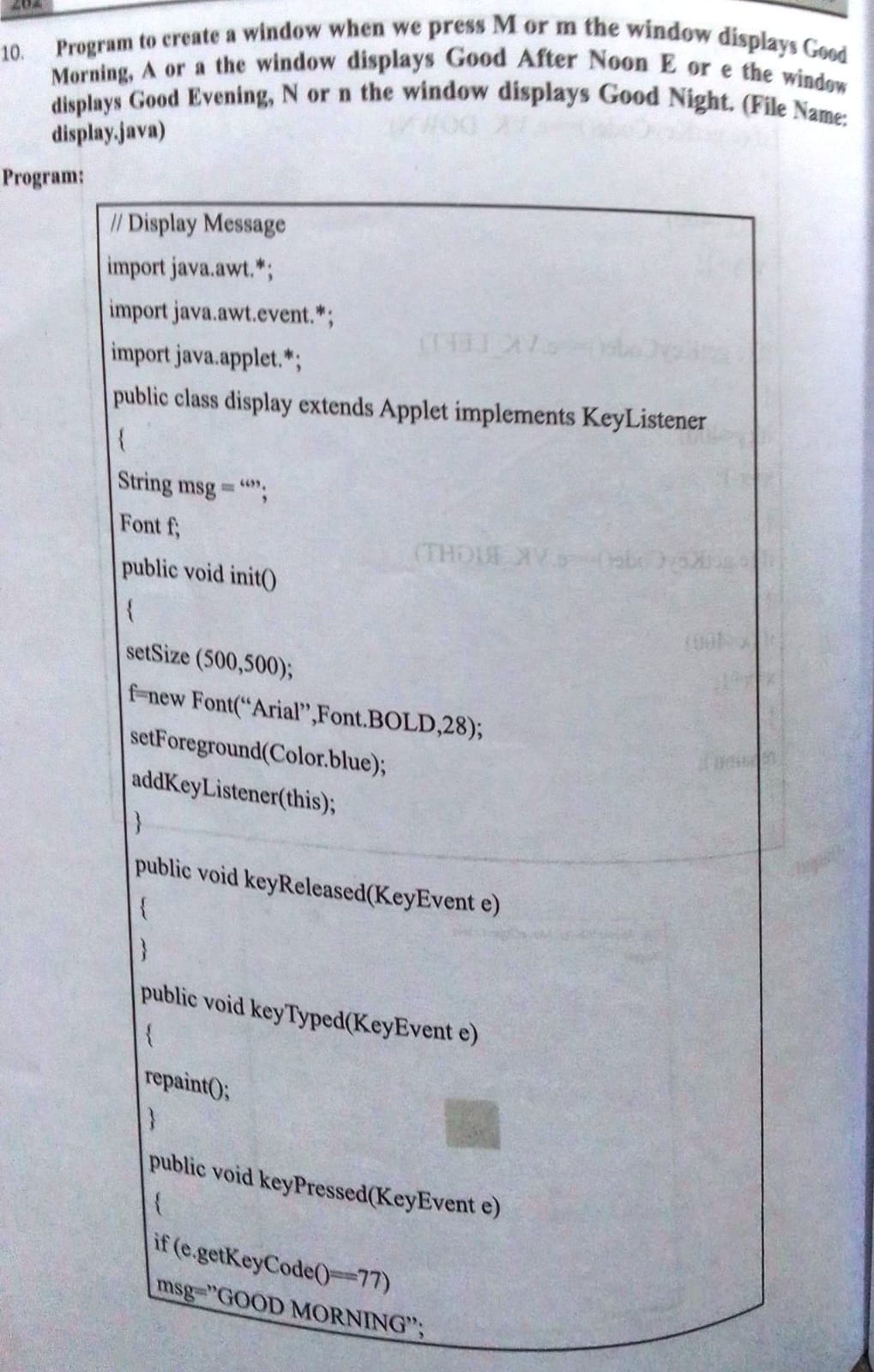


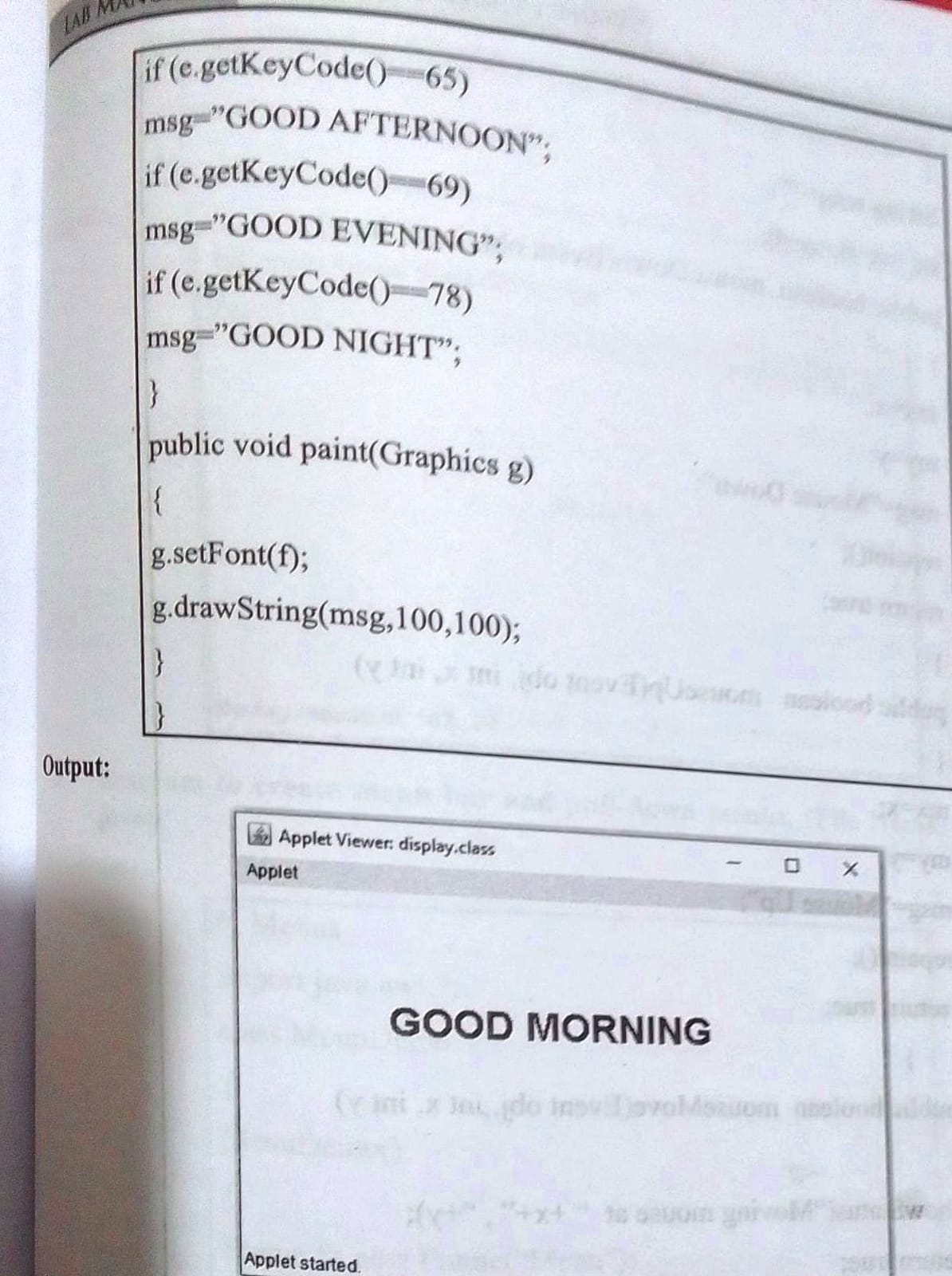
**9./\* Program 9- Move shapes Keyboard events\*/**





**10./\* Program 10-To create a window\*/**

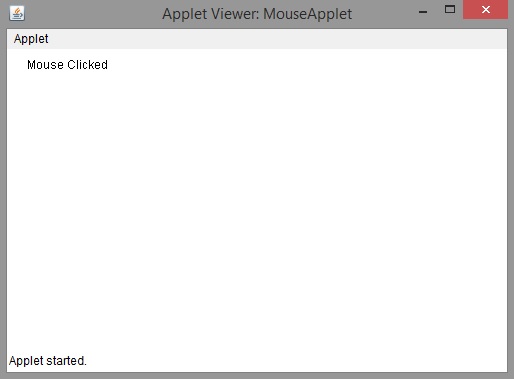




11. Demonstrate the various mouse handling events using suitable example.

|  |
| --- |
| import java.applet.\*; import java.awt.\*; import java.awt.event.\*;  public class B11MouseEvents extends Applet implements MouseListener{  String msg = "Initial Message";  public void init(){  addMouseListener(this);  }  public void mouseClicked(MouseEvent m){  msg = "Mouse Clicked";  repaint();  }  public void mousePressed(MouseEvent m){  msg = "Mouse Pressed";  repaint();  }  public void mouseReleased(MouseEvent m){  msg = "Mouse Released";  repaint();  }  public void mouseEntered(MouseEvent m){  msg = "Mouse Entered";  repaint();  }  public void mouseExited(MouseEvent m){  msg = "Mouse Exited";  repaint();  }  public void paint(Graphics g){  g.drawString(msg, 30, 30);  }  } |

**Output -**



12./\* Program-Menu bar\*/

