Object Oriented Programming and Practice using Java

Assignment [ My Mini Zoo ]

<Report>

Class : 다

Submit Date : 13th June, 2014

|  |  |  |
| --- | --- | --- |
|  | Student Name | Student ID |
| A | 이진우(Lee Jin Woo) | 20132431 |
| B | 지윤호(Ji Yun Ho) | 20132463 |
| C | 강준호(Kang Jun Ho) | 20112315 |

a) Program Structure

BullClass

LionClass

<<interface>>Animal

ElephantClass

BearClass

FrogClass

Gorilla Class

AssignmentPanel

SelectPanel

AssignmentViewer

StarClass

b) Source code

- 6 animal classes, 1 interface called Animal, 3 JPanel components and 1 Viewer class

- Animal.java

**import** java.awt.Graphics2D;

**public** **interface** Animal **extends** Runnable

{

**public** **void** HowToMove();

**public** **void** LetsMove(**int** n);

**public** **boolean** isAnimal(**int** ms\_x,**int** ms\_y);

**public** **void** changePosition(**int** x,**int** y);

**public** **void** sound();

**public** **void** Draw(Graphics2D g2);

**public** **void** setState(**boolean** input);

}

- LionClass.java

**import** javax.swing.JPanel;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Ellipse2D;

**import** java.awt.Color;

**import** java.awt.geom.Line2D;

**import** java.awt.BasicStroke;

**import** java.io.File;

**import** java.util.Random;

**import** javax.sound.sampled.\*;

**import** java.awt.RenderingHints;

**public** **class** LionClass **extends** JPanel **implements** Animal

{

**private** **int** x;

**private** **int** y;

**private** String CurrentStateUD;

**private** String CurrentStateLR;

**private** **boolean** NormalMoving;

**private** **static** **final** **int** *FRAME\_WIDTH*=1100;

**private** **static** **final** **int** *FRAME\_HEIGHT*=700;

**private** **int** randomGenerator;

**private** **static** **final** **int** *LEFT\_SIDE\_LION*=50;

**private** **static** **final** **int** *RIGHT\_SIDE\_LION*=160;

**private** **static** **final** **int** *UPPER\_SIDE\_LION*=70;

**private** **static** **final** **int** *LOWER\_SIDE\_LION*=100;

**public** LionClass(**int** x,**int** y)

{

**try**

{

File file=**new** File("LionSetup.wav");

AudioInputStream audioStream=AudioSystem.*getAudioInputStream*(file);

Clip player=AudioSystem.*getClip*();

player.open(audioStream);

player.start();

}

**catch**(Exception exception)

{

exception.printStackTrace();

}

**this**.x=x;

**this**.y=y;

NormalMoving=**true**; //if (NormalMoving==false) -> FeverTime

CurrentStateUD="up";

CurrentStateLR="right";

randomGenerator=0;

}

**public** **void** run()

{

**while** (**true**)

{

**try**

{

Thread.*sleep*(20);

randomGenerator++; //if randomGenerator become 1000 == randomly move

**if** (randomGenerator==1000)

randomMove();

HowToMove();

**if** (NormalMoving)

{

LetsMove(1);

}

**else**

{

LetsMove(3);

}

}

**catch** (Exception exception)

{

exception.printStackTrace();

}

}

}

**public** **boolean** isAnimal(**int** ms\_x,**int** ms\_y)

{

**boolean** LR=**true**;

**boolean** UD=**true**;

**if** (x-*LEFT\_SIDE\_LION*<ms\_x && ms\_x<x+*RIGHT\_SIDE\_LION*)

LR=**true**;

**else**

LR=**false**;

**if** (y-*UPPER\_SIDE\_LION*<ms\_y && ms\_y<y+*LOWER\_SIDE\_LION*)

UD=**true**;

**else**

UD=**false**;

**return** (LR&&UD);

}

**public** **void** setState(**boolean** input)

{

NormalMoving=input;

}

**public** **void** randomMove()

{

randomGenerator=0;

Random obj=**new** Random();

**int** pos=obj.nextInt(4);

**if** (pos==0)

{

CurrentStateLR="left";

CurrentStateUD="up";

}

**else** **if** (pos==1)

{

CurrentStateLR="right";

CurrentStateUD="up";

}

**else** **if** (pos==2)

{

CurrentStateLR="right";

CurrentStateUD="down";

}

**else**

{

CurrentStateLR="left";

CurrentStateUD="down";

}

}

**public** **void** HowToMove()

{

**if** (x-*LEFT\_SIDE\_LION*<0)

CurrentStateLR="right";

**if** (x+*RIGHT\_SIDE\_LION*+20>*FRAME\_WIDTH*)

CurrentStateLR="left";

**if** (y-*UPPER\_SIDE\_LION*<70)

CurrentStateUD="down";

**if** (y+*LOWER\_SIDE\_LION*+20>*FRAME\_HEIGHT*)

CurrentStateUD="up";

}

**public** **void** LetsMove(**int** n)

{

**if** (CurrentStateLR.equals("right"))

x+=n;

**if** (CurrentStateLR.equals("left"))

x-=n;

**if** (CurrentStateUD.equals("up"))

y-=n;

**if** (CurrentStateUD.equals("down"))

y+=n;

}

**public** **void** changePosition(**int** x,**int** y)

{

**this**.x=x;

**this**.y=y;

}

**public** **void** sound()

{

**try**

{

File file1=**new** File("MovingLionSound.wav");

AudioInputStream movingStream=AudioSystem.*getAudioInputStream*(file1);

Clip movingSound=AudioSystem.*getClip*();

movingSound.open(movingStream);

movingSound.start();

}

**catch** (Exception e)

{

e.printStackTrace();

}

}

**public** **void** Draw(Graphics2D g2)

{

g2.setRenderingHint(RenderingHints.*KEY\_ANTIALIASING*,RenderingHints.*VALUE\_ANTIALIAS\_ON*);

**if** (CurrentStateLR.equals("left"))

{

//body

**int** LbodyX[]={x+25,x+50,x+70,x+90,x+100,x+105,x+110,x+110,x+90,x+75,x+58,x+42,x+27,x+13,x+7,x+3,x};

**int** LbodyY[]={y,y+8,y+10,y+12,y+14,y+16,y+25,y+50,y+50,y+50,y+55,y+50,y+50,y+50,y+34,y+22,y+10};

g2.setColor(**new** Color(212,161,80));

g2.fillPolygon(LbodyX,LbodyY,17);

g2.drawPolygon(LbodyX,LbodyY,17);

//head

**int** LFur1x[]={x+11,x+33,x+28};

**int** LFur1y[]={y-38,y-43,y-21};

**int** LFur2x[]={x+28,x+48,x+28};

**int** LFur2y[]={y-21,y-10,y+1};

**int** LFur3x[]={x+28,x+33,x+11};

**int** LFur3y[]={y+1,y+23,y+18};

**int** LFur4x[]={x+11,x,x-11};

**int** LFur4y[]={y+18,y+38,y+18};

**int** LFur5x[]={x-11,x-33,x-28};

**int** LFur5y[]={y+18,y+23,y+1};

**int** LFur6x[]={x-28,x-48,x-28};

**int** LFur6y[]={y+1,y-10,y-21};

**int** LFur7x[]={x-28,x-33,x-11};

**int** LFur7y[]={y-21,y-43,y-38};

**int** LFur8x[]={x-11,x,x+11};

**int** LFur8y[]={y-38,y-58,y-38};

g2.setColor(**new** Color(170,66,37));

g2.fillPolygon(LFur1x,LFur1y,3);

g2.fillPolygon(LFur2x,LFur2y,3);

g2.fillPolygon(LFur3x,LFur3y,3);

g2.fillPolygon(LFur4x,LFur4y,3);

g2.fillPolygon(LFur5x,LFur5y,3);

g2.fillPolygon(LFur6x,LFur6y,3);

g2.fillPolygon(LFur7x,LFur7y,3);

g2.fillPolygon(LFur8x,LFur8y,3);

g2.drawPolygon(LFur1x,LFur1y,3);

g2.drawPolygon(LFur2x,LFur2y,3);

g2.drawPolygon(LFur3x,LFur3y,3);

g2.drawPolygon(LFur4x,LFur4y,3);

g2.drawPolygon(LFur5x,LFur5y,3);

g2.drawPolygon(LFur6x,LFur6y,3);

g2.drawPolygon(LFur7x,LFur7y,3);

g2.drawPolygon(LFur8x,LFur8y,3);

**int** LFurin1x[]={x+28,x+30,x+46,x+38};

**int** LFurin1y[]={y-21,y-31,y-29,y-15};

**int** LFurin2x[]={x+28,x+38,x+46,x+30};

**int** LFurin2y[]={y+1,y-4,y+9,y+12};

**int** LFurin3x[]={x+11,x+22,x+19,x+5};

**int** LFurin3y[]={y+18,y+20,y+38,y+28};

**int** LFurin4x[]={x-11,x-5,x-19,x-22};

**int** LFurin4y[]={y+18,y+28,y+38,y+20};

**int** LFurin5x[]={x-28,x-30,x-46,x-38};

**int** LFurin5y[]={y+1,y+12,y+9,y-4};

**int** LFurin6x[]={x-28,x-38,x-46,x-30};

**int** LFurin6y[]={y-21,y-15,y-29,y-31};

**int** LFurin7x[]={x-11,x-22,x-19,x-5};

**int** LFurin7y[]={y-38,y-40,y-58,y-48};

**int** LFurin8x[]={x+11,x+5,x+19,x+22};

**int** LFurin8y[]={y-38,y-48,y-58,y-40};

g2.setColor(**new** Color(140,30,43));

g2.fillPolygon(LFurin1x,LFurin1y,4);

g2.fillPolygon(LFurin2x,LFurin2y,4);

g2.fillPolygon(LFurin3x,LFurin3y,4);

g2.fillPolygon(LFurin4x,LFurin4y,4);

g2.fillPolygon(LFurin5x,LFurin5y,4);

g2.fillPolygon(LFurin6x,LFurin6y,4);

g2.fillPolygon(LFurin7x,LFurin7y,4);

g2.fillPolygon(LFurin8x,LFurin8y,4);

g2.drawPolygon(LFurin1x,LFurin1y,4);

g2.drawPolygon(LFurin2x,LFurin2y,4);

g2.drawPolygon(LFurin3x,LFurin3y,4);

g2.drawPolygon(LFurin4x,LFurin4y,4);

g2.drawPolygon(LFurin5x,LFurin5y,4);

g2.drawPolygon(LFurin6x,LFurin6y,4);

g2.drawPolygon(LFurin7x,LFurin7y,4);

g2.drawPolygon(LFurin8x,LFurin8y,4);

//head

Ellipse2D.Double Lhead=**new** Ellipse2D.Double(x-30,y-40,60,60);

g2.setColor(**new** Color(212,160,80));

g2.fill(Lhead);

g2.draw(Lhead);

//eye

Ellipse2D.Double LEye1=**new** Ellipse2D.Double(x-17,y-27,10,10);

Ellipse2D.Double LEye2=**new** Ellipse2D.Double(x+7,y-27,10,10);

g2.setColor(Color.*black*);

g2.fill(LEye1);

g2.fill(LEye2);

g2.draw(LEye1);

g2.draw(LEye2);

//tail

**int** Ltail1X[]={x+105,x+120,x+130,x+135,x+125,x+120,x+110};

**int** Ltail1Y[]={y+16,y+24,y+35,y+50,y+40,y+30,y+25};

g2.setColor(**new** Color(212,160,80));

g2.fillPolygon(Ltail1X,Ltail1Y,7);

g2.drawPolygon(Ltail1X,Ltail1Y,7);

**int** Ltail2X[]={x+135,x+139,x+140,x+137,x+128,x+127,x+135};

**int** Ltail2Y[]={y+50,y+56,y+61,y+70,y+60,y+55,y+50};

g2.setColor(**new** Color(140,30,43));

g2.fillPolygon(Ltail2X,Ltail2Y,7);

g2.drawPolygon(Ltail2X,Ltail2Y,7);

//leg

**int** Lleg1X[]={x+13,x+15,x+13,x+10,x+24,x+26,x+27};

**int** Lleg1Y[]={y+50,y+60,y+70,y+80,y+80,y+65,y+50};

**int** Lleg2X[]={x+27,x+30,x+28,x+41,x+45,x+42};

**int** Lleg2Y[]={y+50,y+65,y+80,y+80,y+65,y+50};

**int** Lleg3X[]={x+75,x+77,x+69,x+79,x+89,x+95};

**int** Lleg3Y[]={y+50,y+65,y+80,y+80,y+65,y+50};

**int** Lleg4X[]={x+95,x+93,x+90,x+105,x+110,x+110};

**int** Lleg4Y[]={y+50,y+65,y+80,y+80,y+65,y+50};

g2.setColor(**new** Color(210,160,80));

g2.fillPolygon(Lleg1X,Lleg1Y,7);

g2.fillPolygon(Lleg2X,Lleg2Y,6);

g2.fillPolygon(Lleg3X,Lleg3Y,6);

g2.fillPolygon(Lleg4X,Lleg4Y,6);

g2.drawPolygon(Lleg1X,Lleg1Y,7);

g2.drawPolygon(Lleg2X,Lleg2Y,6);

g2.drawPolygon(Lleg3X,Lleg3Y,6);

g2.drawPolygon(Lleg4X,Lleg4Y,6);

//foot

**int** Lfoot1X[]={x+10,x+8,x+6,x+25,x+24};

**int** Lfoot1Y[]={y+80,y+85,y+90,y+90,y+80};

**int** Lfoot2X[]={x+28,x+27,x+26,x+42,x+41};

**int** Lfoot2Y[]={y+80,y+85,y+90,y+90,y+80};

**int** Lfoot3X[]={x+69,x+67,x+65,x+78,x+79};

**int** Lfoot3Y[]={y+80,y+85,y+90,y+90,y+80};

**int** Lfoot4X[]={x+86,x+84,x+82,x+102,x+105};

**int** Lfoot4Y[]={y+80,y+85,y+90,y+90,y+80};

g2.setColor(**new** Color(210,201,146));

g2.fillPolygon(Lfoot1X,Lfoot1Y,5);

g2.fillPolygon(Lfoot2X,Lfoot2Y,5);

g2.fillPolygon(Lfoot3X,Lfoot3Y,5);

g2.fillPolygon(Lfoot4X,Lfoot4Y,5);

g2.drawPolygon(Lfoot1X,Lfoot1Y,5);

g2.drawPolygon(Lfoot2X,Lfoot2Y,5);

g2.drawPolygon(Lfoot3X,Lfoot3Y,5);

g2.drawPolygon(Lfoot4X,Lfoot4Y,5);

//mouth

Line2D.Double Lmouth1=**new** Line2D.Double(x,y-10,x,y+2);

Line2D.Double Lmouth2=**new** Line2D.Double(x,y+2,x+10,y+8);

Line2D.Double Lmouth3=**new** Line2D.Double(x,y+2,x-10,y+8);

g2.setColor(Color.*black*);

g2.setStroke(**new** BasicStroke(3));

g2.draw(Lmouth1);

g2.draw(Lmouth2);

g2.draw(Lmouth3);

//nose

Ellipse2D.Double Lnose=**new** Ellipse2D.Double(x-3,y-10,6,6);

g2.fill(Lnose);

g2.draw(Lnose);

//change to default stroke

g2.setStroke(**new** BasicStroke(1));

}

**else**

{

//body

**int** RbodyX[]={x+110,x+107,x+103,x+97,x+83,x+68,x+52,x+35,x+20,x,x,x+5,x+10,x+20,x+40,x+60,x+85};

**int** RbodyY[]={y+10,y+22,y+34,y+50,y+50,y+50,y+55,y+50,y+50,y+50,y+25,y+16,y+14,y+12,y+10,y+8,y};

g2.setColor(**new** Color(212,161,80));

g2.fillPolygon(RbodyX,RbodyY,17);

g2.drawPolygon(RbodyX,RbodyY,17);

//fur,head

**int** Rfur1x[]={x+121,x+143,x+138};

**int** Rfur1y[]={y-38,y-43,y-21};

**int** Rfur2x[]={x+138,x+158,x+138};

**int** Rfur2y[]={y-21,y-10,y+1};

**int** Rfur3x[]={x+138,x+143,x+121};

**int** Rfur3y[]={y+1,y+23,y+18};

**int** Rfur4x[]={x+121,x+110,x+99};

**int** Rfur4y[]={y+18,y+38,y+18};

**int** Rfur5x[]={x+99,x+77,x+82};

**int** Rfur5y[]={y+18,y+23,y+1};

**int** Rfur6x[]={x+82,x+62,x+82};

**int** Rfur6y[]={y+1,y-10,y-21};

**int** Rfur7x[]={x+82,x+77,x+99};

**int** Rfur7y[]={y-21,y-43,y-38};

**int** Rfur8x[]={x+99,x+110,x+121};

**int** Rfur8y[]={y-38,y-58,y-38};

g2.setColor(**new** Color(170,66,37));

g2.fillPolygon(Rfur1x,Rfur1y,3);

g2.fillPolygon(Rfur2x,Rfur2y,3);

g2.fillPolygon(Rfur3x,Rfur3y,3);

g2.fillPolygon(Rfur4x,Rfur4y,3);

g2.fillPolygon(Rfur5x,Rfur5y,3);

g2.fillPolygon(Rfur6x,Rfur6y,3);

g2.fillPolygon(Rfur7x,Rfur7y,3);

g2.fillPolygon(Rfur8x,Rfur8y,3);

g2.drawPolygon(Rfur1x,Rfur1y,3);

g2.drawPolygon(Rfur2x,Rfur2y,3);

g2.drawPolygon(Rfur3x,Rfur3y,3);

g2.drawPolygon(Rfur4x,Rfur4y,3);

g2.drawPolygon(Rfur5x,Rfur5y,3);

g2.drawPolygon(Rfur6x,Rfur6y,3);

g2.drawPolygon(Rfur7x,Rfur7y,3);

g2.drawPolygon(Rfur8x,Rfur8y,3);

**int** Rfurin1x[]={x+140,x+156,x+148,x+138};

**int** Rfurin1y[]={y-32,y-29,y-15,y-21};

**int** Rfurin2x[]={x+148,x+156,x+140,x+138};

**int** Rfurin2y[]={y-4,y+9,y+12,y+1};

**int** Rfurin3x[]={x+132,x+129,x+115,x+121};

**int** Rfurin3y[]={y+20,y+38,y+28,y+18};

**int** Rfurin4x[]={x+104,x+91,x+88,x+99};

**int** Rfurin4y[]={y+28,y+38,y+20,y+18};

**int** Rfurin5x[]={x+79,x+64,x+72,x+82};

**int** Rfurin5y[]={y+12,y+9,y-4,y+1};

**int** Rfurin6x[]={x+72,x+64,x+79,x+82};

**int** Rfurin6y[]={y-15,y-29,y-32,y-21};

**int** Rfurin7x[]={x+88,x+91,x+104,x+99};

**int** Rfurin7y[]={y-40,y-58,y-48,y-38};

**int** Rfurin8x[]={x+115,x+129,x+132,x+121};

**int** Rfurin8y[]={y-48,y-58,y-40,y-38};

g2.setColor(**new** Color(140,30,43));

g2.fillPolygon(Rfurin1x,Rfurin1y,4);

g2.fillPolygon(Rfurin2x,Rfurin2y,4);

g2.fillPolygon(Rfurin3x,Rfurin3y,4);

g2.fillPolygon(Rfurin4x,Rfurin4y,4);

g2.fillPolygon(Rfurin5x,Rfurin5y,4);

g2.fillPolygon(Rfurin6x,Rfurin6y,4);

g2.fillPolygon(Rfurin7x,Rfurin7y,4);

g2.fillPolygon(Rfurin8x,Rfurin8y,4);

g2.drawPolygon(Rfurin1x,Rfurin1y,4);

g2.drawPolygon(Rfurin2x,Rfurin2y,4);

g2.drawPolygon(Rfurin3x,Rfurin3y,4);

g2.drawPolygon(Rfurin4x,Rfurin4y,4);

g2.drawPolygon(Rfurin5x,Rfurin5y,4);

g2.drawPolygon(Rfurin6x,Rfurin6y,4);

g2.drawPolygon(Rfurin7x,Rfurin7y,4);

g2.drawPolygon(Rfurin8x,Rfurin8y,4);

//head

Ellipse2D.Double Rhead=**new** Ellipse2D.Double(x+80,y-40,60,60);

g2.setColor(**new** Color(212,160,80));

g2.fill(Rhead);

g2.draw(Rhead);

//eyes

Ellipse2D.Double REye1=**new** Ellipse2D.Double(x+93,y-27,10,10);

Ellipse2D.Double REye2=**new** Ellipse2D.Double(x+117,y-27,10,10);

g2.setColor(Color.*black*);

g2.fill(REye1);

g2.fill(REye2);

g2.draw(REye1);

g2.draw(REye2);

//leg

**int** Rleg1X[]={x+97,x+95,x+97,x+100,x+86,x+84,x+83};

**int** Rleg1Y[]={y+50,y+60,y+70,y+80,y+80,y+65,y+50};

**int** Rleg2X[]={x+83,x+80,x+82,x+69,x+65,x+68};

**int** Rleg2Y[]={y+50,y+65,y+80,y+80,y+65,y+50};

**int** Rleg3X[]={x+35,x+31,x+33,x+22,x+20,x+20};

**int** Rleg3Y[]={y+50,y+65,y+80,y+80,y+65,y+50};

**int** Rleg4X[]={x+20,x+15,x+16,x+5,x,x};

**int** Rleg4Y[]={y+50,y+65,y+80,y+80,y+65,y+50};

g2.setColor(**new** Color(212,160,80));

g2.fillPolygon(Rleg1X,Rleg1Y,7);

g2.fillPolygon(Rleg2X,Rleg2Y,6);

g2.fillPolygon(Rleg3X,Rleg3Y,6);

g2.fillPolygon(Rleg4X,Rleg4Y,6);

g2.drawPolygon(Rleg1X,Rleg1Y,7);

g2.drawPolygon(Rleg2X,Rleg2Y,6);

g2.drawPolygon(Rleg3X,Rleg3Y,6);

g2.drawPolygon(Rleg4X,Rleg4Y,6);

//foot

**int** Rfoot1X[]={x+100,x+102,x+104,x+85,x+86};

**int** Rfoot1Y[]={y+80,y+85,y+90,y+90,y+80};

**int** Rfoot2X[]={x+82,x+83,x+84,x+68,x+69};

**int** Rfoot2Y[]={y+80,y+85,y+90,y+90,y+80};

**int** Rfoot3X[]={x+33,x+35,x+37,x+22,x+22};

**int** Rfoot3Y[]={y+80,y+85,y+90,y+90,y+80};

**int** Rfoot4X[]={x+16,x+18,x+20,x+5,x+5};

**int** Rfoot4Y[]={y+80,y+85,y+90,y+90,y+80};

g2.setColor(**new** Color(210,201,146));

g2.fillPolygon(Rfoot1X,Rfoot1Y,5);

g2.fillPolygon(Rfoot2X,Rfoot2Y,5);

g2.fillPolygon(Rfoot3X,Rfoot3Y,5);

g2.fillPolygon(Rfoot4X,Rfoot4Y,5);

g2.drawPolygon(Rfoot1X,Rfoot1Y,5);

g2.drawPolygon(Rfoot2X,Rfoot2Y,5);

g2.drawPolygon(Rfoot3X,Rfoot3Y,5);

g2.drawPolygon(Rfoot4X,Rfoot4Y,5);

//mouth

Line2D.Double Rmouth1=**new** Line2D.Double(x+110,y-10,x+110,y+2);

Line2D.Double Rmouth2=**new** Line2D.Double(x+110,y+2,x+120,y+8);

Line2D.Double Rmouth3=**new** Line2D.Double(x+110,y+2,x+100,y+8);

//tail1

**int** Rtail1X[]={x+5,x,x-10,x-15,x-25,x-20,x-10};

**int** Rtail1Y[]={y+16,y+25,y+30,y+40,y+50,y+35,y+24};

g2.setColor(**new** Color(212,160,80));

g2.fillPolygon(Rtail1X,Rtail1Y,7);

g2.drawPolygon(Rtail1X,Rtail1Y,7);

**int** Rtail2X[]={x-25,x-17,x-18,x-27,x-30,x-29};

**int** Rtail2Y[]={y+50,y+55,y+60,y+70,y+61,y+56};

g2.setColor(**new** Color(140,30,43));

g2.fillPolygon(Rtail2X,Rtail2Y,6);

g2.drawPolygon(Rtail2X,Rtail2Y,6);

g2.setColor(Color.*black*);

g2.setStroke(**new** BasicStroke(3));

//mouth

g2.draw(Rmouth1);

g2.draw(Rmouth2);

g2.draw(Rmouth3);

//nose

Ellipse2D.Double Rnose=**new** Ellipse2D.Double(x+107,y-10,6,6);

g2.draw(Rnose);

//change to default stroke

g2.setStroke(**new** BasicStroke(1));

}

}

}

- BullClass.java

**import** java.awt.Color;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Ellipse2D;

**import** java.awt.geom.Line2D;

**import** java.awt.BasicStroke;

**import** javax.swing.JPanel;

**import** java.awt.RenderingHints;

**import** java.util.Random;

**import** java.io.File;

**import** javax.sound.sampled.\*;

**public** **class** BullClass **extends** JPanel **implements** Animal

{

**private** **int** x;

**private** **int** y;

**private** String CurrentStateLR;

**private** String CurrentStateUD;

**private** **boolean** NormalMoving;

**private** **static** **final** **int** *FRAME\_WIDTH*=1100;

**private** **static** **final** **int** *FRAME\_HEIGHT*=700;

**private** **int** randomGenerator;

**private** **static** **final** **int** *LEFT\_SIDE\_COW*=50;

**private** **static** **final** **int** *RIGHT\_SIDE\_COW*=160;

**private** **static** **final** **int** *UPPER\_SIDE\_COW*=50;

**private** **static** **final** **int** *LOWER\_SIDE\_COW*=130;

**public** BullClass(**int** xInput,**int** yInput)

{

**try**

{

File file=**new** File("BullSetup.wav");

AudioInputStream audioStream=AudioSystem.*getAudioInputStream*(file);

Clip player=AudioSystem.*getClip*();

player.open(audioStream);

player.start();

}

**catch**(Exception exception)

{

exception.printStackTrace();

}

NormalMoving=**true**;

CurrentStateLR="right";

CurrentStateUD="down";

x=xInput;

y=yInput;

randomGenerator=0;

}

**public** **boolean** isAnimal(**int** ms\_x,**int** ms\_y)

{

**boolean** LR=**true**;

**boolean** UD=**true**;

**if** (x-*LEFT\_SIDE\_COW*<ms\_x && ms\_x<x+*RIGHT\_SIDE\_COW*)

LR=**true**;

**else**

LR=**false**;

**if** (y-*UPPER\_SIDE\_COW*<ms\_y && ms\_y<y+*LOWER\_SIDE\_COW*)

UD=**true**;

**else**

UD=**false**;

**return** (LR&&UD);

}

**public** **void** changePosition(**int** x,**int** y)

{

**this**.x=x;

**this**.y=y;

}

**public** **void** randomMove()

{

randomGenerator=0;

Random obj=**new** Random();

**int** pos=obj.nextInt(4);

**if** (pos==0)

{

CurrentStateLR="left";

CurrentStateUD="up";

}

**else** **if** (pos==1)

{

CurrentStateLR="right";

CurrentStateUD="up";

}

**else** **if** (pos==2)

{

CurrentStateLR="right";

CurrentStateUD="down";

}

**else**

{

CurrentStateLR="left";

CurrentStateUD="down";

}

}

**public** **void** run()

{

**while** (**true**)

{

**try**

{

Thread.*sleep*(10);

randomGenerator++;

**if** (randomGenerator==1000)

randomMove();

HowToMove();

**if** (NormalMoving)

{

LetsMove(1);

}

**else**

{

LetsMove(3);

}

}

**catch** (Exception exception)

{

exception.printStackTrace();

}

}

}

**public** **void** setState(**boolean** input)

{

NormalMoving=input;

}

**public** **void** HowToMove()

{

**if** (x-*LEFT\_SIDE\_COW*<0)

CurrentStateLR="right";

**if** (x+*RIGHT\_SIDE\_COW*+20>*FRAME\_WIDTH*)

CurrentStateLR="left";

**if** (y-*UPPER\_SIDE\_COW*<70)

CurrentStateUD="down";

**if** (y+*LOWER\_SIDE\_COW*+20>*FRAME\_HEIGHT*)

CurrentStateUD="up";

}

**public** **void** LetsMove(**int** n)

{

**if** (CurrentStateLR.equals("left"))

x-=n;

**if** (CurrentStateLR.equals("right"))

x+=n;

**if** (CurrentStateUD.equals("up"))

y-=n;

**if** (CurrentStateUD.equals("down"))

y+=n;

}

**public** **void** sound()

{

**try**

{

File file1=**new** File("MovingBullSound.wav");

AudioInputStream movingStream=AudioSystem.*getAudioInputStream*(file1);

Clip movingSound=AudioSystem.*getClip*();

movingSound.open(movingStream);

movingSound.start();

}

**catch** (Exception e)

{

e.printStackTrace();

}

}

**public** **void** Draw(Graphics2D g2)

{

g2.setRenderingHint(RenderingHints.*KEY\_ANTIALIASING*,RenderingHints.*VALUE\_ANTIALIAS\_ON*);

//if bull towards to the left side

**if** (CurrentStateLR.equals("left"))

{

//body

**int** LbodyX[]={x+20,x+50,x+70,x+110,x+120,x+125,x+125,x+120,x+110,x+100,x+90,x+70,x+60,x+45,x+27,x+10,x+5,x};

**int** LbodyY[]={y-10,y-5,y,y-5,y,y+10,y+15,y+30,y+60,y+60,y+60,y+65,y+65,y+60,y+60,y+60,y+40,y+25};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(LbodyX,LbodyY,18);

g2.drawPolygon(LbodyX,LbodyY,18);

//leg

**int** Lleg1X[]={x+10,x+11,x+12,x+13,x+13,x+12,x+25,x+25,x+27};

**int** Lleg1Y[]={y+60,y+68,y+76,y+84,y+92,y+100,y+100,y+80,y+60};

**int** Lleg2X[]={x+27,x+38,x+35,x+49,x+47,x+45};

**int** Lleg2Y[]={y+60,y+75,y+100,y+100,y+80,y+60};

**int** Lleg3X[]={x+90,x+87,x+90,x+87,x+96,x+98,x+96,x+100};

**int** Lleg3Y[]={y+60,y+76,y+85,y+100,y+100,y+80,y+70,y+60};

**int** Lleg4X[]={x+100,x+105,x+103,x+115,x+110};

**int** Lleg4Y[]={y+60,y+80,y+100,y+95,y+60};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(Lleg1X,Lleg1Y,9);

g2.fillPolygon(Lleg2X,Lleg2Y,6);

g2.fillPolygon(Lleg3X,Lleg3Y,8);

g2.fillPolygon(Lleg4X,Lleg4Y,5);

g2.drawPolygon(Lleg1X,Lleg1Y,9);

g2.drawPolygon(Lleg2X,Lleg2Y,6);

g2.drawPolygon(Lleg3X,Lleg3Y,8);

g2.drawPolygon(Lleg4X,Lleg4Y,5);

//foot

**int** Lfoot1X[]={x+12,x+11,x+10,x+25,x+25};

**int** Lfoot1Y[]={y+100,y+105,y+110,y+110,y+100};

**int** Lfoot2X[]={x+35,x+34,x+33,x+49,x+49};

**int** Lfoot2Y[]={y+100,y+105,y+110,y+110,y+100};

**int** Lfoot3X[]={x+87,x+86,x+85,x+97,x+96};

**int** Lfoot3Y[]={y+100,y+105,y+110,y+110,y+100};

**int** Lfoot4X[]={x+103,x+100,x+101,x+117,x+115};

**int** Lfoot4Y[]={y+100,y+105,y+110,y+103,y+95};

g2.setColor(Color.*BLACK*);

g2.fillPolygon(Lfoot1X,Lfoot1Y,5);

g2.fillPolygon(Lfoot2X,Lfoot2Y,5);

g2.fillPolygon(Lfoot3X,Lfoot3Y,5);

g2.fillPolygon(Lfoot4X,Lfoot4Y,5);

g2.drawPolygon(Lfoot1X,Lfoot1Y,5);

g2.drawPolygon(Lfoot2X,Lfoot2Y,5);

g2.drawPolygon(Lfoot3X,Lfoot3Y,5);

g2.drawPolygon(Lfoot4X,Lfoot4Y,5);

//head

**int** hx=x-5;

**int** hy=y+10;

**int** headX[]={hx+12,hx+24,hx+27,hx+27,hx+27,hx+26,hx+19,hx,hx-19,hx-26,hx-27,hx-27,hx-27,hx-24,hx-12};

**int** headY[]={hy-30,hy-28,hy-25,hy-20,hy,hy+20,hy+18,hy+17,hy+18,hy+20,hy,hy-20,hy-25,hy-28,hy-30};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(headX,headY,15);

g2.drawPolygon(headX,headY,15);

//ear

**int** RearX[]={hx+27,hx+42,hx+47,hx+42,hx+27};

**int** RearY[]={hy-20,hy-17,hy-15,hy-12,hy};

g2.setColor(**new** Color(210,201,146));

g2.fillPolygon(RearX,RearY,5);

g2.drawPolygon(RearX,RearY,5);

**int** LearX[]={hx-27,hx-42,hx-47,hx-42,hx-27};

**int** LearY[]={hy-20,hy-17,hy-15,hy-12,hy};

g2.fillPolygon(LearX,LearY,5);

g2.drawPolygon(LearX,LearY,5);

//mouth

**int** mouthX[]={hx+26,hx+25,hx+23,hx+20,hx,hx-20,hx-23,hx-25,hx-26,hx-19,hx,hx+19};

**int** mouthY[]={hy+20,hy+33,hy+37,hy+39,hy+40,hy+39,hy+37,hy+33,hy+20,hy+18,hy+17,hy+18};

g2.setColor(**new** Color(160,82,45));

g2.fillPolygon(mouthX,mouthY,12);

g2.drawPolygon(mouthX,mouthY,12);

//bull

**int** RbullX[]={hx+24,hx+35,hx+38,hx+42,hx+40,hx+36,hx+27};

**int** RbullY[]={hy-28,hy-34,hy-37,hy-42,hy-24,hy-21,hy-20};

g2.setColor(Color.*WHITE*);

g2.fillPolygon(RbullX,RbullY,7);

g2.drawPolygon(RbullX,RbullY,7);

**int** LbullX[]={hx-24,hx-35,hx-38,hx-42,hx-40,hx-36,hx-27};

**int** LbullY[]={hy-28,hy-34,hy-37,hy-42,hy-24,hy-21,hy-20};

g2.fillPolygon(LbullX,LbullY,7);

g2.drawPolygon(LbullX,LbullY,7);

//eyes

Ellipse2D.Double Leye=**new** Ellipse2D.Double(hx-15,hy-15,7,7);

g2.setColor(Color.*BLACK*);

g2.fill(Leye);

g2.draw(Leye);

Ellipse2D.Double Reye=**new** Ellipse2D.Double(hx+5,hy-15,7,7);

g2.fill(Reye);

g2.draw(Reye);

//mouthline

g2.setStroke(**new** BasicStroke(2));

Line2D.Double ml1=**new** Line2D.Double(hx+17,hy+37,hx+12,hy+36);

Line2D.Double ml2=**new** Line2D.Double(hx+12,hy+36,hx+6,hy+35);

Line2D.Double ml3=**new** Line2D.Double(hx+6,hy+35,hx,hy+34);

Line2D.Double ml4=**new** Line2D.Double(hx,hy+34,hx-6,hy+35);

Line2D.Double ml5=**new** Line2D.Double(hx-6,hy+35,hx-12,hy+36);

Line2D.Double ml6=**new** Line2D.Double(hx-12,hy+36,hx-17,hy+37);

g2.setColor(Color.*BLACK*);

g2.draw(ml1);

g2.draw(ml2);

g2.draw(ml3);

g2.draw(ml4);

g2.draw(ml5);

g2.draw(ml6);

//set default stroke

g2.setStroke(**new** BasicStroke(1));

//nose

**int** RnoseX[]={hx+7,hx+15,hx+14,hx+6};

**int** RnoseY[]={hy+22,hy+21,hy+23,hy+25};

g2.setColor(Color.*BLACK*);

g2.fillPolygon(RnoseX,RnoseY,4);

g2.drawPolygon(RnoseX,RnoseY,4);

**int** LnoseX[]={hx-7,hx-15,hx-14,hx-6};

**int** LnoseY[]={hy+22,hy+21,hy+23,hy+25};

g2.fillPolygon(LnoseX,LnoseY,4);

g2.drawPolygon(LnoseX,LnoseY,4);

//tail

**int** Ltail1X[]={x+125,x+130,x+134,x+138,x+139,x+132,x+134,x+132,x+129,x+124};

**int** Ltail1Y[]={y+10,y+13,y+24,y+30,y+57,y+56,y+48,y+30,y+24,y+20};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(Ltail1X,Ltail1Y,10);

g2.drawPolygon(Ltail1X,Ltail1Y,10);

**int** Ltail2X[]={x+139,x+142,x+143,x+141,x+133,x+130,x+128,x+129,x+132};

**int** Ltail2Y[]={y+57,y+61,y+66,y+71,y+76,y+71,y+66,y+61,y+56};

g2.setColor(Color.*BLACK*);

g2.fillPolygon(Ltail2X,Ltail2Y,9);

g2.drawPolygon(Ltail2X,Ltail2Y,9);

}

//if bull towards to the right side

**else**

{

//body

**int** RbodyX[]={x+120,x+115,x+110,x+93,x+75,x+60,x+50,x+30,x+20,x+10,x,x-5,x-5,x,x+10,x+50,x+70,x+100};

**int** RbodyY[]={y+25,y+40,y+60,y+60,y+60,y+65,y+65,y+60,y+60,y+60,y+30,y+15,y+10,y,y-5,y,y-5,y-10};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(RbodyX,RbodyY,18);

g2.drawPolygon(RbodyX,RbodyY,18);

//leg

**int** Rleg1X[]={x+110,x+109,x+108,x+107,x+107,x+108,x+95,x+95,x+93};

**int** Rleg1Y[]={y+60,y+68,y+76,y+84,y+92,y+100,y+100,y+80,y+60};

**int** Rleg2X[]={x+93,x+82,x+85,x+71,x+73,x+75};

**int** Rleg2Y[]={y+60,y+75,y+100,y+100,y+80,y+60};

**int** Rleg3X[]={x+30,x+33,x+30,x+33,x+24,x+22,x+24,x+20};

**int** Rleg3Y[]={y+60,y+76,y+85,y+100,y+100,y+80,y+70,y+60};

**int** Rleg4X[]={x+20,x+15,x+17,x+5,x+10};

**int** Rleg4Y[]={y+60,y+80,y+100,y+95,y+60};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(Rleg1X,Rleg1Y,9);

g2.fillPolygon(Rleg2X,Rleg2Y,6);

g2.fillPolygon(Rleg3X,Rleg3Y,8);

g2.fillPolygon(Rleg4X,Rleg4Y,5);

g2.drawPolygon(Rleg1X,Rleg1Y,9);

g2.drawPolygon(Rleg2X,Rleg2Y,6);

g2.drawPolygon(Rleg3X,Rleg3Y,8);

g2.drawPolygon(Rleg4X,Rleg4Y,5);

//foot

**int** Rfoot1X[]={x+108,x+109,x+110,x+95,x+95};

**int** Rfoot1Y[]={y+100,y+105,y+110,y+110,y+100};

**int** Rfoot2X[]={x+85,x+86,x+87,x+71,x+71};

**int** Rfoot2Y[]={y+100,y+105,y+110,y+110,y+100};

**int** Rfoot3X[]={x+33,x+34,x+35,x+23,x+24};

**int** Rfoot3Y[]={y+100,y+105,y+110,y+110,y+100};

**int** Rfoot4X[]={x+17,x+20,x+19,x+3,x+5};

**int** Rfoot4Y[]={y+100,y+105,y+110,y+103,y+95};

g2.setColor(Color.*BLACK*);

g2.fillPolygon(Rfoot1X,Rfoot1Y,5);

g2.fillPolygon(Rfoot2X,Rfoot2Y,5);

g2.fillPolygon(Rfoot3X,Rfoot3Y,5);

g2.fillPolygon(Rfoot4X,Rfoot4Y,5);

g2.drawPolygon(Rfoot1X,Rfoot1Y,5);

g2.drawPolygon(Rfoot2X,Rfoot2Y,5);

g2.drawPolygon(Rfoot3X,Rfoot3Y,5);

g2.drawPolygon(Rfoot4X,Rfoot4Y,5);

//head

**int** hx=x+125;

**int** hy=y+10;

**int** headX[]={hx+12,hx+24,hx+27,hx+27,hx+27,hx+26,hx+19,hx,hx-19,hx-26,hx-27,hx-27,hx-27,hx-24,hx-12};

**int** headY[]={hy-30,hy-28,hy-25,hy-20,hy,hy+20,hy+18,hy+17,hy+18,hy+20,hy,hy-20,hy-25,hy-28,hy-30};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(headX,headY,15);

g2.drawPolygon(headX,headY,15);

//ear

**int** RearX[]={hx+27,hx+42,hx+47,hx+42,hx+27};

**int** RearY[]={hy-20,hy-17,hy-15,hy-12,hy};

g2.setColor(**new** Color(210,201,146));

g2.fillPolygon(RearX,RearY,5);

g2.drawPolygon(RearX,RearY,5);

**int** LearX[]={hx-27,hx-42,hx-47,hx-42,hx-27};

**int** LearY[]={hy-20,hy-17,hy-15,hy-12,hy};

g2.fillPolygon(LearX,LearY,5);

g2.drawPolygon(LearX,LearY,5);

//mouth

**int** mouthX[]={hx+26,hx+25,hx+23,hx+20,hx,hx-20,hx-23,hx-25,hx-26,hx-19,hx,hx+19};

**int** mouthY[]={hy+20,hy+33,hy+37,hy+39,hy+40,hy+39,hy+37,hy+33,hy+20,hy+18,hy+17,hy+18};

g2.setColor(**new** Color(160,82,45));

g2.fillPolygon(mouthX,mouthY,12);

g2.drawPolygon(mouthX,mouthY,12);

//bull

**int** RbullX[]={hx+24,hx+35,hx+38,hx+42,hx+40,hx+36,hx+27};

**int** RbullY[]={hy-28,hy-34,hy-37,hy-42,hy-24,hy-21,hy-20};

g2.setColor(Color.*WHITE*);

g2.fillPolygon(RbullX,RbullY,7);

g2.drawPolygon(RbullX,RbullY,7);

**int** LbullX[]={hx-24,hx-35,hx-38,hx-42,hx-40,hx-36,hx-27};

**int** LbullY[]={hy-28,hy-34,hy-37,hy-42,hy-24,hy-21,hy-20};

g2.fillPolygon(LbullX,LbullY,7);

g2.drawPolygon(LbullX,LbullY,7);

//eyes

Ellipse2D.Double Leye=**new** Ellipse2D.Double(hx-15,hy-15,7,7);

g2.setColor(Color.*BLACK*);

g2.fill(Leye);

g2.draw(Leye);

Ellipse2D.Double Reye=**new** Ellipse2D.Double(hx+5,hy-15,7,7);

g2.fill(Reye);

g2.draw(Reye);

//mouthline

g2.setStroke(**new** BasicStroke(2));

Line2D.Double ml1=**new** Line2D.Double(hx+17,hy+37,hx+12,hy+36);

Line2D.Double ml2=**new** Line2D.Double(hx+12,hy+36,hx+6,hy+35);

Line2D.Double ml3=**new** Line2D.Double(hx+6,hy+35,hx,hy+34);

Line2D.Double ml4=**new** Line2D.Double(hx,hy+34,hx-6,hy+35);

Line2D.Double ml5=**new** Line2D.Double(hx-6,hy+35,hx-12,hy+36);

Line2D.Double ml6=**new** Line2D.Double(hx-12,hy+36,hx-17,hy+37);

g2.setColor(Color.*BLACK*);

g2.draw(ml1);

g2.draw(ml2);

g2.draw(ml3);

g2.draw(ml4);

g2.draw(ml5);

g2.draw(ml6);

//set default stroke

g2.setStroke(**new** BasicStroke(1));

//nose

**int** RnoseX[]={hx+7,hx+15,hx+14,hx+6};

**int** RnoseY[]={hy+22,hy+21,hy+23,hy+25};

g2.setColor(Color.*BLACK*);

g2.fillPolygon(RnoseX,RnoseY,4);

g2.drawPolygon(RnoseX,RnoseY,4);

**int** LnoseX[]={hx-7,hx-15,hx-14,hx-6};

**int** LnoseY[]={hy+22,hy+21,hy+23,hy+25};

g2.fillPolygon(LnoseX,LnoseY,4);

g2.drawPolygon(LnoseX,LnoseY,4);

//tail

**int** Rtail1X[]={x-4,x-9,x-12,x-14,x-12,x-19,x-18,x-14,x-10,x-5};

**int** Rtail1Y[]={y+20,y+24,y+30,y+48,y+56,y+57,y+30,y+24,y+13,y+10};

g2.setColor(**new** Color(139,69,19));

g2.fillPolygon(Rtail1X,Rtail1Y,10);

g2.drawPolygon(Rtail1X,Rtail1Y,10);

**int** Rtail2X[]={x-12,x-9,x-8,x-10,x-13,x-21,x-23,x-22,x-19};

**int** Rtail2Y[]={y+56,y+61,y+66,y+71,y+76,y+71,y+66,y+61,y+57};

g2.setColor(Color.*BLACK*);

g2.fillPolygon(Rtail2X,Rtail2Y,9);

g2.drawPolygon(Rtail2X,Rtail2Y,9);

}

}

}

- BearClass.java

- ElephantClass.java

- FrogClass.java

- GorillaClass.java

- StarClass.java

**import** java.awt.Graphics2D;

**import** javax.swing.JPanel;

**import** java.awt.Color;

**import** java.util.Random;

**import** java.awt.RenderingHints;

**public** **class** StarClass **extends** JPanel

{

**private** **boolean** currentStateUD; //false==up, true==down

**private** **boolean** currentStateLR; //false==left, true==right

**private** **int** x;

**private** **int** y;

**private** **int** changeColor;

**private** String currentColor;

**private** **static** **final** **int** *WIDTH*=1100;

**private** **static** **final** **int** *HEIGHT*=700;

**private** Random rand;

**public** StarClass(**int** xInput, **int** yInput)

{

x=xInput;

y=yInput;

rand=**new** Random();

changeColor=0;

**if** ((rand.nextInt())%2==0)

currentColor="Red";

**else**

currentColor="Yellow";

currentStateUD=**true**;

currentStateLR=**true**; //right,down

}

**public** **void** starDirection()

{

**if** (x>*WIDTH*-75)

currentStateLR=**false**;

**if** (x<75)

currentStateLR=**true**;

**if** (y<120)

currentStateUD=**true**;

**if** (y>*HEIGHT*-75)

currentStateUD=**false**;

changeColor++;

}

**public** **void** starMove()

{

**if** (currentStateLR)

x+=3;

**else**

x-=3;

**if** (currentStateUD)

y+=3;

**else**

y-=3;

changeColor++;

**if** (changeColor==100)

{

changeColor=0;

swapColor();

}

}

**public** **void** swapColor()

{

**if** (currentColor.equals("Red"))

currentColor="Yellow";

**else**

currentColor="Red";

}

**public** **void** draw(Graphics2D g2)

{

g2.setRenderingHint(RenderingHints.*KEY\_ANTIALIASING*,RenderingHints.*VALUE\_ANTIALIAS\_ON*);

**int** starX[]={x,x+20,x+75,x+25,x+37,x,x-37,x-25,x-75,x-20};

**int** starY[]={y-75,y-20,y-20,y+20,y+75,y+25,y+75,y+20,y-20,y-20};

**if** (currentColor.equals("Red"))

g2.setColor(Color.*RED*);

**else**

g2.setColor(Color.*YELLOW*);

g2.fillPolygon(starX,starY,10);

g2.drawPolygon(starX,starY,10);

}

}

- SelectPanel.java

- AssignmentPanel.java

- AssignmentViewer.java

**import** javax.swing.JFrame;

**public** **class** AssignmentViewer

{

**public** **static** **void** main(String[] args)

{

JFrame frame=**new** JFrame();

frame.setSize(1100,700);

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

frame.setTitle("Object Oriented Programming and Practical using Java:::Assignment[My Mini Zoo]");

AssignmentPanel panel=**new** AssignmentPanel();

frame.add(panel);

frame.setVisible(**true**);

}

}

c) Sample screenshots