|  |  |
| --- | --- |
| NAME: \_\_\_\_\_\_\_\_\_\_\_\_\_ | **Computer Programming** |
| Lego Classes | **DATE: Monday, December 05, 2016** |

/\*

 \* This class represents a lego piece.

 \* It has an x,y (its location on grid - the upper left stud is 0,0

 \* It has a color - if not chosen it is anything

 \* The width is the horizontal distance, length - vertical

 \* Direction is either horizontal or vertical. By default it is horizontal. \*

 \*/

public class LegoPiece

{

 int x;

 int y;

 int width;

 int length;

 Color color;

 String direction="horizontal";

 int level=1;

 public LegoPiece (int theWidth, int theLength)

 {

 width=theWidth;

 length=theLength;

 }

 public LegoPiece (int theWidth, int theLength, Color theColor)

 {

 width=theWidth;

 length=theLength;

 color=theColor;

 }

 public void setDirection(String newDirection)

 {

 direction=newDirection;

 }

 public void setLocation(int newX, int newY)

 {

 x=newX;

 y=newY;

 }

 //This method will put this lego piece on top other lego piece with same upper left coordinate.

 //note lego's coordinates start at 0,0.

 public void snapTo(LegoPiece otherPiece, int legoX, int legoY)

 {

 x=otherPiece.x+legoX;

 y=otherPiece.y+legoY;

 level=otherPiece.level++;

 }

}

The class to the right represents a lego piece.

Try to follow the instructions to create my shape:

public void myInstructions()

 {

 LegoPiece rect1=new LegoPiece(4,2);

 rect1.setDirection("vertical");

 rect1.setLocation(2,0);

 LegoPiece square1=new LegoPiece(2,2);

 square1.setLocation(0,1);

 LegoPiece square2=new LegoPiece(2,2);

 square2.setLocation(4,1);

 LegoPiece rect2=new LegoPiece(2,1);

 rect2.snapTo(square1,1,0);

 LegoPiece rect3=new LegoPiece(2,1);

 rect3.snapTo(rect1,1,1);

 LegoPiece rect4=new LegoPiece(2,1);

 rect4.snapTo(rect2,1,0);

 LegoPiece square3=new LegoPiece(2,2,Color.red);

 square3.snapTo(rect4,0,0);

 }



Create your own instruction sheet. You can use up to 10 bricks.