**GraphicsPrac7**

**Directions:**

1. Create the Shot Class below
	1. The Shot class does not need to extend Applet
2. In GraphicsPrac7, create a new ArrayList *shots* that will hold Shot objects, and instantiate the ArrayList in the init() method.
3. Add a for-statement inside of the *paint(Graphics g)* method that loops through each Shot in *shots* and calls the *draw* method of the Shot class.
4. Remove the if-else statements in the *paint(Graphics g)* method that calculate the endpoint of the turret and place them in a helper method called *updateTurret().* The *updateTurret()* method should also ensure that the *angle* variable never becomes greater than 359 and less than 0.
5. Make a call to the *updateTurret()* method in the *paint(Graphics g)* method
6. Add a for-statement inside of the *run()* method that loops through each Shot in *shots* and calls the *move()* method of the Shot class.
7. Add a new if-statement to the *keyPressed(KeyEvent e)* method that instantiates a new Shot when the SPACE bar is pressed. The new Shot object should be added to the ArrayList.

**Shot Class**

**Directions:** Create the Shot class with the constructors, methods, and variables defined below.

1. Variables
	1. doubles x and y for position coordinate position of the Shot
	2. doubles xVel and yVel for the velocity of the new Shot
	3. final double SHOT\_SPEED that represents the speed at which the shot will be fired
2. Constructors
	1. Shot(double x, double y, double angle, double shipXVel, double shipYVel)
		1. Use *angle, shipXVel,* and *shipYVel* to calculate *xVel* and *yVel* for the shot
		2. The parameters *x* and *y* should be assigned to the instance variables *x* and *y*
3. Methods
	1. public void move()
		1. Alters the x and y position of the shot
		2. Ensures that the shot stays on the screen
	2. public void draw(Graphics g)
		1. Uses *fillOval* to draw a shot at position *x*, *y*