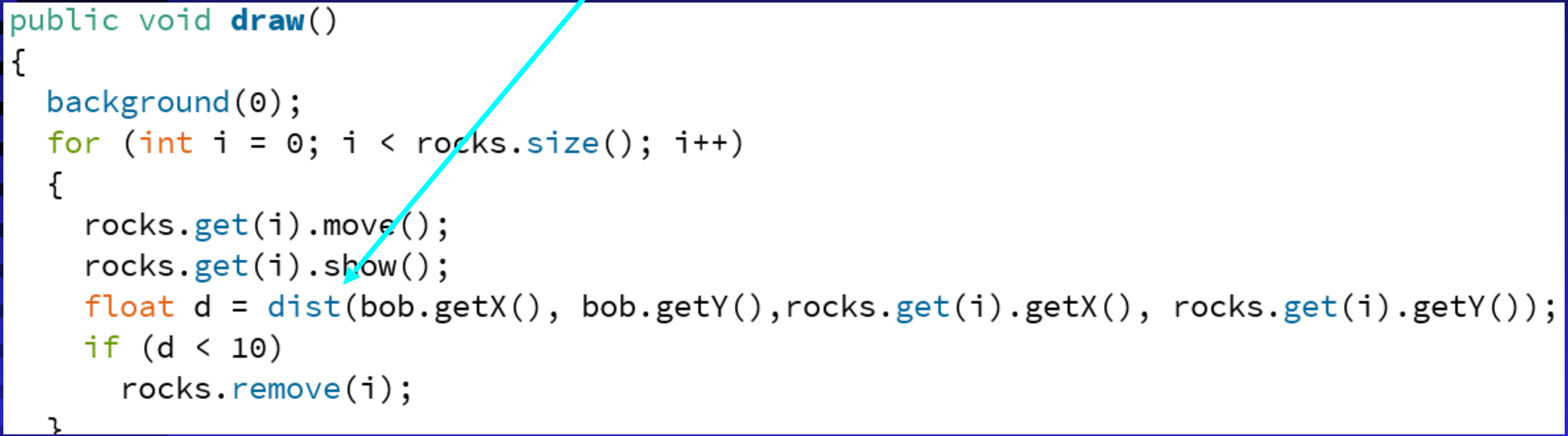
Asteroids Part 3: Adding Bullets

- Add collision detection
- Add Bullets to Spaceship

For collision detection in Asteroids you can use Processing’s **`dist()`** function

- Everytime you move an Asteroid, check to see if it is close enough to crash into the ship
- One way is with **`dist()`**

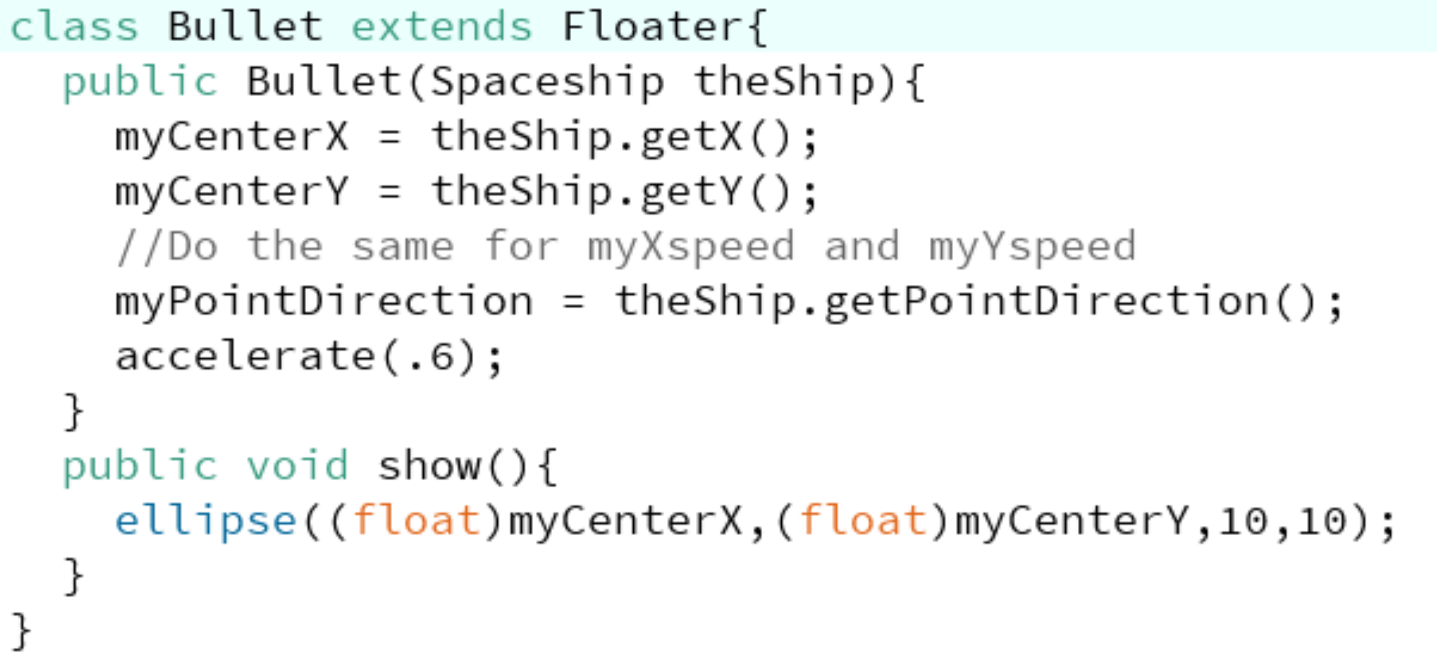


```
public void draw()
{
  background(0);
  for (int i = 0; i < rocks.size(); i++)
  {
    rocks.get(i).move();
    rocks.get(i).show();
    float d = dist(bob.getX(), bob.getY(),rocks.get(i).getX(), rocks.get(i).getY());
    if (d < 10)
      rocks.remove(i);
  }
```

- **`dist()`** takes 4 arguments, the x & y of two points, and returns the distance between the points

Part 3: Adding Bullets

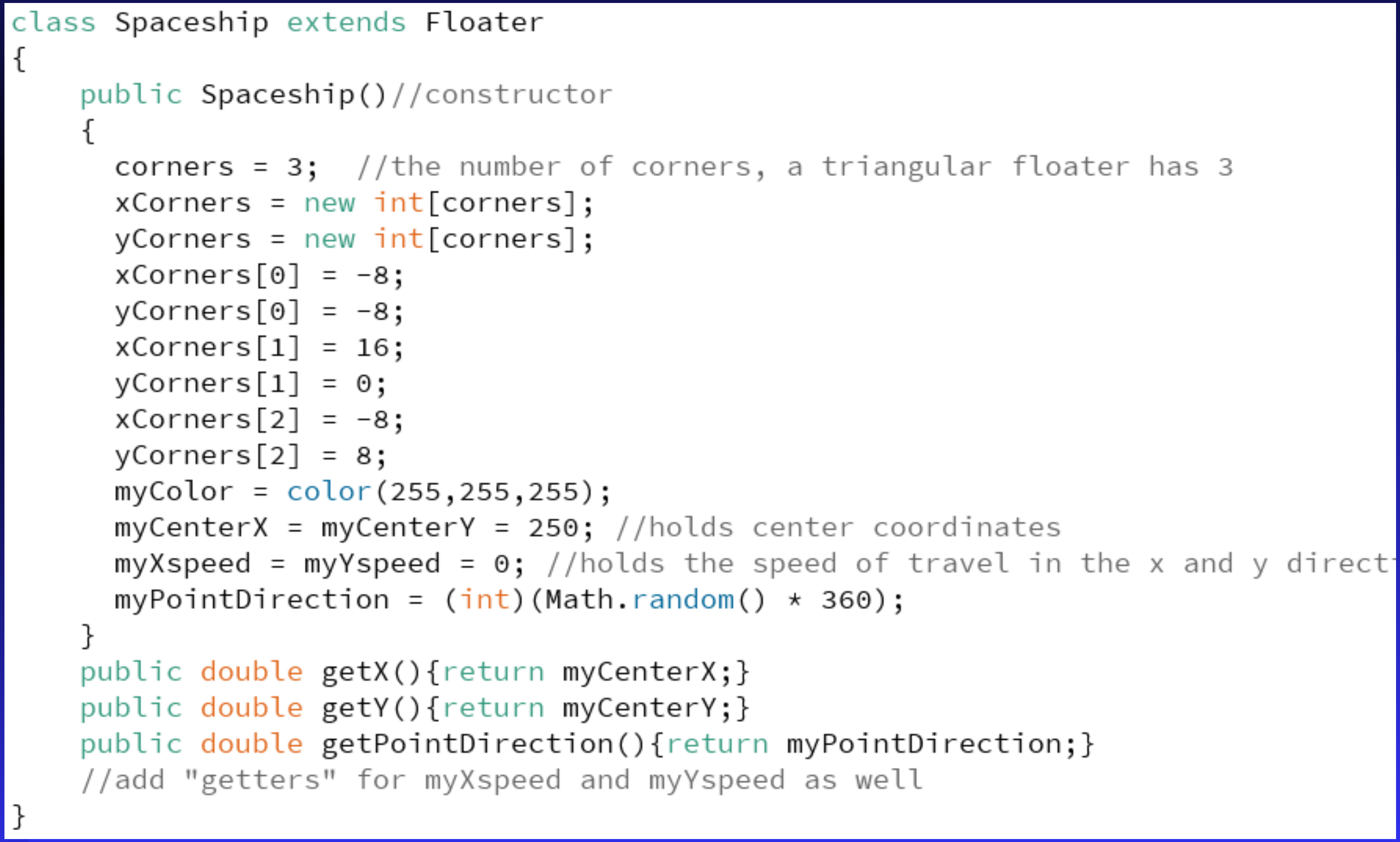
- Add **Bullet.pde** with:
  - a constructor
  - *Override* the **show()** method of the **Floater** class so that you can use circular bullets



```
class Bullet extends Floater{
  public Bullet(Spaceship theShip){
    myCenterX = theShip.getX();
    myCenterY = theShip.getY();
    //Do the same for myXspeed and myYspeed
    myPointDirection = theShip.getPointDirection();
    accelerate(.6);
  }
  public void show(){
    ellipse((float)myCenterX,(float)myCenterY,10,10);
  }
}
```

Part 3: Adding Bullets

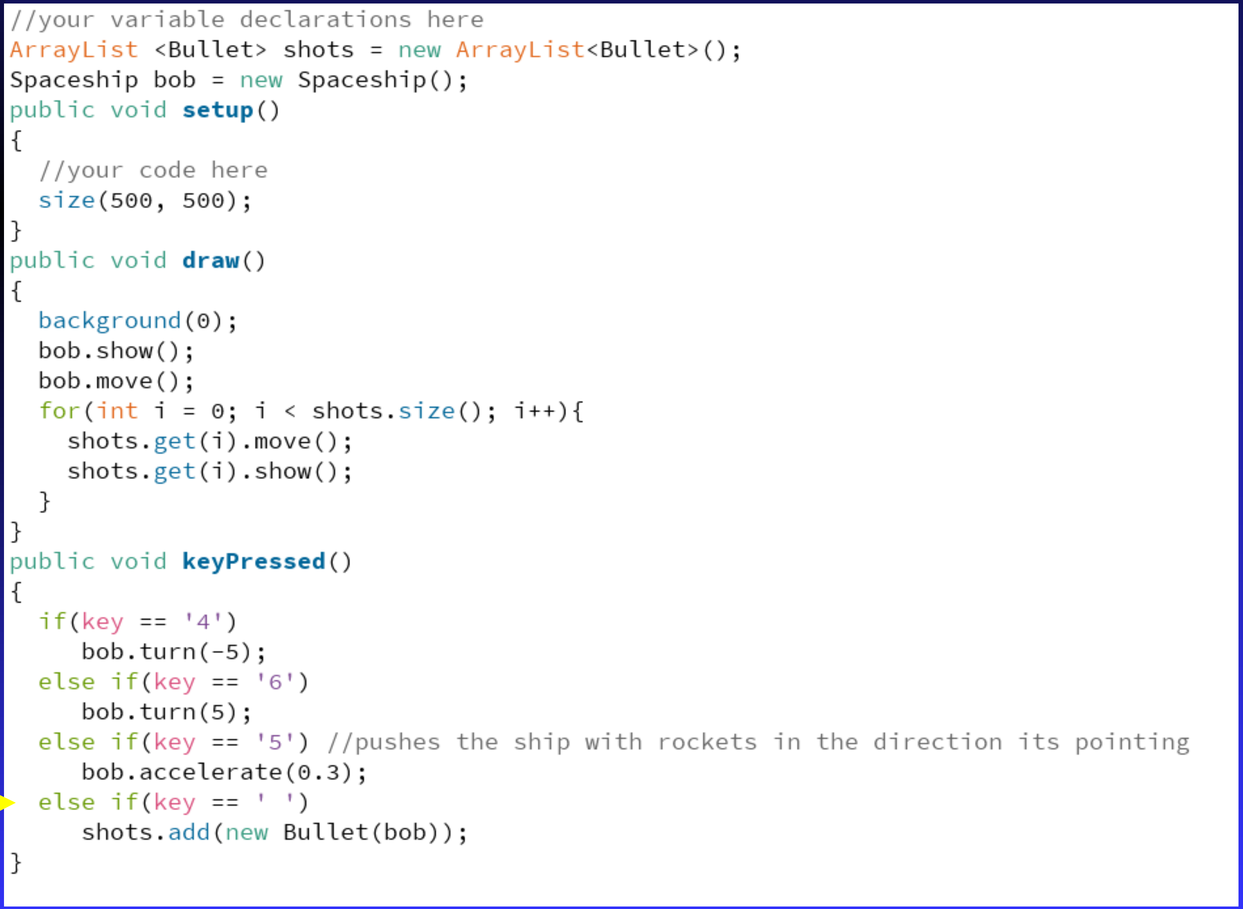
- If you haven’t written them already, you’ll need 5 “getter” functions in the the **`Spaceship`** class



```
class Spaceship extends Floater
{
    public Spaceship()//constructor
    {
      corners = 3;  //the number of corners, a triangular floater has 3
      xCorners = new int[corners];
      yCorners = new int[corners];
      xCorners[0] = -8;
      yCorners[0] = -8;
      xCorners[1] = 16;
      yCorners[1] = 0;
      xCorners[2] = -8;
      yCorners[2] = 8;
      myColor = color(255,255,255);
      myCenterX = myCenterY = 250; //holds center coordinates
      myXspeed = myYspeed = 0; //holds the speed of travel in the x and y direct
      myPointDirection = (int)(Math.random() * 360);
    }
    public double getX(){return myCenterX;}
    public double getY(){return myCenterY;}
    public double getPointDirection(){return myPointDirection;}
    //add "getters" for myXspeed and myYspeed as well
}
```

Part 3: Adding Bullets

- Add an **`if`** statement in **`keyPressed()`** that adds a **`new Bullet`** to the **`ArrayList`** when you press a particular key (I’m using the spacebar)



```
//your variable declarations here
ArrayList <Bullet> shots = new ArrayList<Bullet>();
Spaceship bob = new Spaceship();
public void setup()
{
  //your code here
  size(500, 500);
}
public void draw()
{
  background(0);
  bob.show();
  bob.move();
  for(int i = 0; i < shots.size(); i++){
    shots.get(i).move();
    shots.get(i).show();
  }
}
public void keyPressed()
{
  if(key == '4')
     bob.turn(-5);
  else if(key == '6')
     bob.turn(5);
  else if(key == '5') //pushes the ship with rockets in the direction its pointing
     bob.accelerate(0.3);
  else if(key == ' ')
     shots.add(new Bullet(bob));
}
```