```
//  Austin Matel
// 12/3/19
//  This is a comment
//  The setup function function is called once when your program
begins
var ship;
var planet, disToPlanet;
function setup() {
  var cnv = createCanvas(800, 800);
  cnv.position((windowWidth-width)/2, 30);
  background(5, 5, 5);

  loadObjects(100);
}

//  The draw function is called @  30 fps
function draw() {
  background(5,5,5,20);
  runShips();
}
function loadObjects(n){
  ship = new Ship(random(width), random(height), random(-3,3),
random(-3,3), 1);
  planet = new Planet(width/2, height/2, random(-0.5,0.5),
random(-2,2), 0);
}
function runShips(){
  ship.run();
  planet.run();
}

//Austin Matel
//12/3/19
class Ship{
  constructor(x, y, dx, dy, id){
    this.loc = createVector(x, y);
    this.vel = createVector(dx, dy);
    this.clr = color(random(225), random(225), random(225));
    this.angle = 0;
    this.acc = createVector(0,0.1);
    this.id = id;
  }
  run(){
    this.checkEdges();
    this.update();
    this.render();
  }

  checkEdges(){
    if(this.loc.x < 0){
      this.vel.x = -this.vel.x;
    }
    if(this.loc.x > width){
      this.vel.x = -this.vel.x;
    }
    if(this.loc.y < 0){
      this.vel.y = -this.vel.y;
    }
    if(this.loc.y > height){
      this.vel.y = -this.vel.y;
    }
  }
  update(){
    disToPlanet = this.loc.dist(planet.loc);
    if(disToPlanet > 100){
      this.acc = p5.Vector.sub(planet.loc, this.loc);
      this.acc.normalize();
      this.acc.mult(0.1);
    }
    this.vel.limit(4);
    this.vel.add(this.acc);
    this.loc.add(this.vel);
    }

  render(){
    fill(this.clr);
    this.angle = this.vel.heading() + 360;
    this.angle = this.angle - 0.1;
    push();
      translate(this.loc.x,this.loc.y);
      rotate(this.angle);
      triangle(-10,16,10,16,0,-16);
    pop();
    }
}
```

```
//Austin Matel
//12/3/19
class Planet{
  constructor(x, y, dx, dy, id){
    this.loc = createVector(x, y);
    this.vel = createVector(dx, dy);
    this.clr = color(random(225), random(225), random(225));
    this.w = 10;
    this.acc = createVector(0,0);
    this.id = id;
  }

  run(){
    this.checkEdges();
    this.update();
    this.render();
  }

  checkEdges(){
    if (this.loc.x < 0){
      this.vel.x = -(this.vel.x);
    }
    if (this.loc.x > width){
      this.vel.x = -(this.vel.x);
    }
    if (this.loc.y < 0){
      this.vel.y = -(this.vel.y);
    }
    if (this.loc.y > height){
        this.vel.y = -(this.vel.y);
    }
  }
  update(){
    this.vel.add(this.acc);
    this.vel.limit(5);
    this.loc.add(this.vel);
    if(disToPlanet < 70){
      this.loc.x = random(10,790);
      this.loc.y = random(10,790);
      this.vel.x = random(-.5,.5);
      this.vel.y = random(-2,2);
    }
  }
  render(){
    fill(this.clr);
    if(this.id == 0){
      this.w = 50;
    }
    ellipse(this.loc.x, this.loc.y, this.w, this.w)
  }
}
```