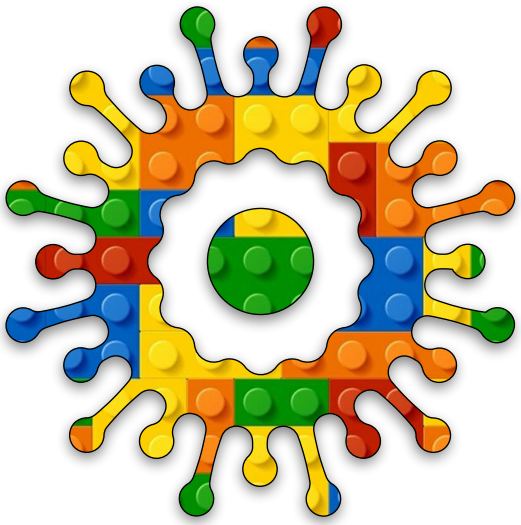
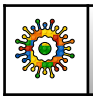
Making Games Module A Unit #5 asteroids game





Content

Module A Unit #5 Asteroids Game

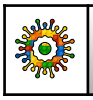
Sketch A5.1 drawing the ship
Sketch A5.2 class Ship
Sketch A5.3 in the middle
Sketch A5.4 rotate the ship
Sketch A5.5 turning
Sketch A5.6 rotating the ship
Sketch A5.7 stopping
Sketch A5.8 rotating the ship
Sketch A5.9 applying a force
Sketch A5.10 on the move
Sketch A5.11 accuracy and dampening
Sketch A5.12 holding the key down
Sketch A5.13 boolean boost
Sketch A5.14 arriving at the edge
Sketch A5.15 push and pop

Sketch A5.16 Asteroid class
Sketch A5.17 random circle
Sketch A5.18 more like an asteroid
Sketch A5.19 regular asteroids
Sketch A5.20 more asteroids
Sketch A5.21 smaller asteroids
Sketch A5.22 irregular asteroids
Sketch A5.23 floating asteroids
Sketch A5.24 I like to move it, move it
Sketch A5.25 reaching the canvas edges
Sketch A5.26 reappears as if by magic

Sketch A5.27 adding the lasers
Sketch A5.28 an array of lasers
Sketch A5.29 laser as a particle
Sketch A5.30 spacebar laser
Sketch A5.31 directing the laser
Sketch A5.32 firing the laser cannon
Sketch A5.33 a tweaking of the laser

Sketch A5.34 testing a hit
Sketch A5.35 it's a hit
Sketch A5.36 working well, hopefully
Sketch A5.37 breaking the asteroid
Sketch A5.38 the breakup
Sketch A5.39 deleting the particle
Sketch A5.40 another asteroid
Sketch A5.41 half pint asteroids
Sketch A5.42 I've been hit response
Sketch A5.43 watch out for the asteroid

Sketch A5.44 the laser array problem
Sketch A5.45 off the screen solution
Sketch A5.46 creating a ship png
Sketch A5.47 uploading the ship png
Sketch A5.48 adding the ship PNG



Before we begin...

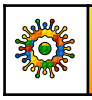
Step 1: You will create three more files:

ship.js
laser.js
asteroid.js

Step 2: You will add them in just the same way as for the previous games; remember to add them to the index.html file.

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/1.11.10/p5.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/1.11.10/addons/p5.sound.min.js"></script>
    <link rel="stylesheet" type="text/css" href="style.css">
    <meta charset="utf-8" />
  </head>
  <body>
    <main>
    </main>
    <script src="sketch.js"></script>
    <script src="laser.js"></script>
    <script src="asteroid.js"></script>
    <script src="ship.js"></script>
  </body>
</html>
```



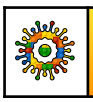
Sketch A5.1 drawing the ship

❗ Starting with sketch.js

This draws a triangle (the ship) in not a very good place, but it is a start. Next, we will translate the co-ordinates.

sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
}
```



Sketch A5.2 class Ship

❗ ship.js

Create a Ship class with a `constructor()` function.

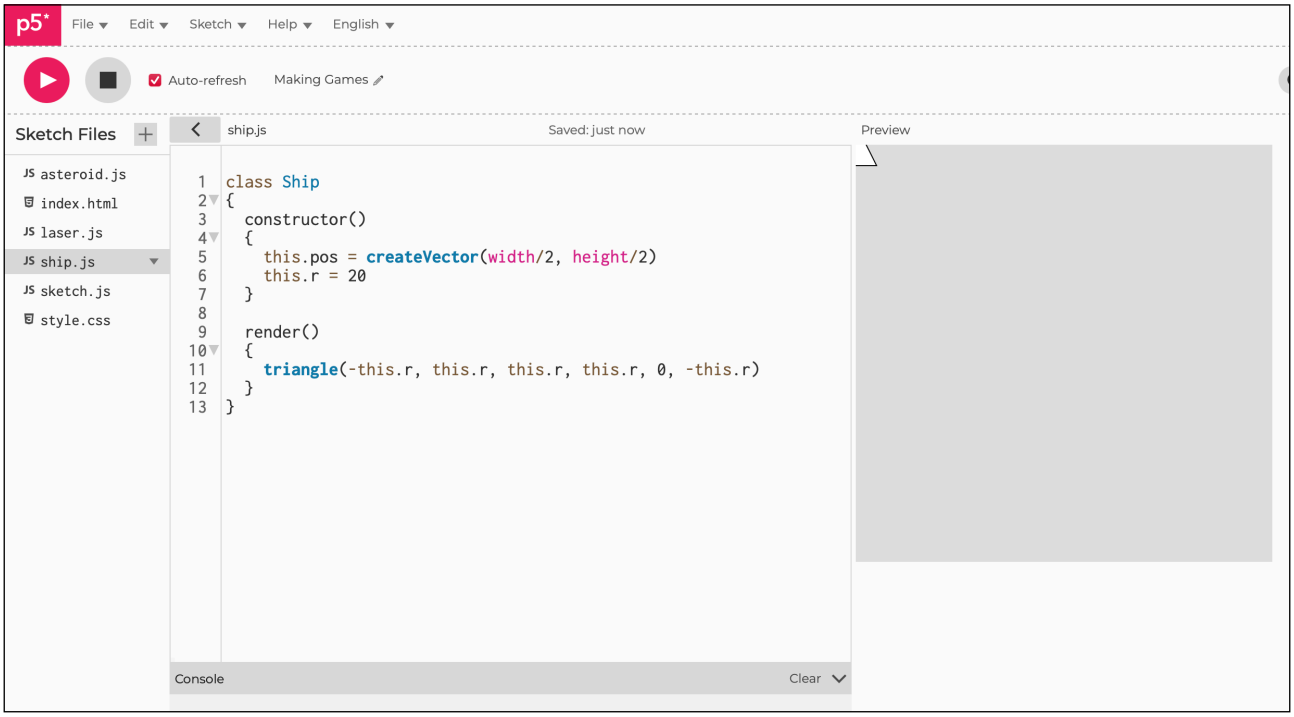
ship.js

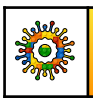
```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
  }

  show()
  {
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }
}
```

📝 Notes

The ship is at co-ordinates (0, 0).

Figure A5.2


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
ship.js
Saved: just now
Preview
1 class Ship
2 {
3 constructor()
4 {
5 this.pos = createVector(width/2, height/2)
6 this.r = 20
7 }
8
9 render()
10 {
11 triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
12 }
13 }
Console
Clear



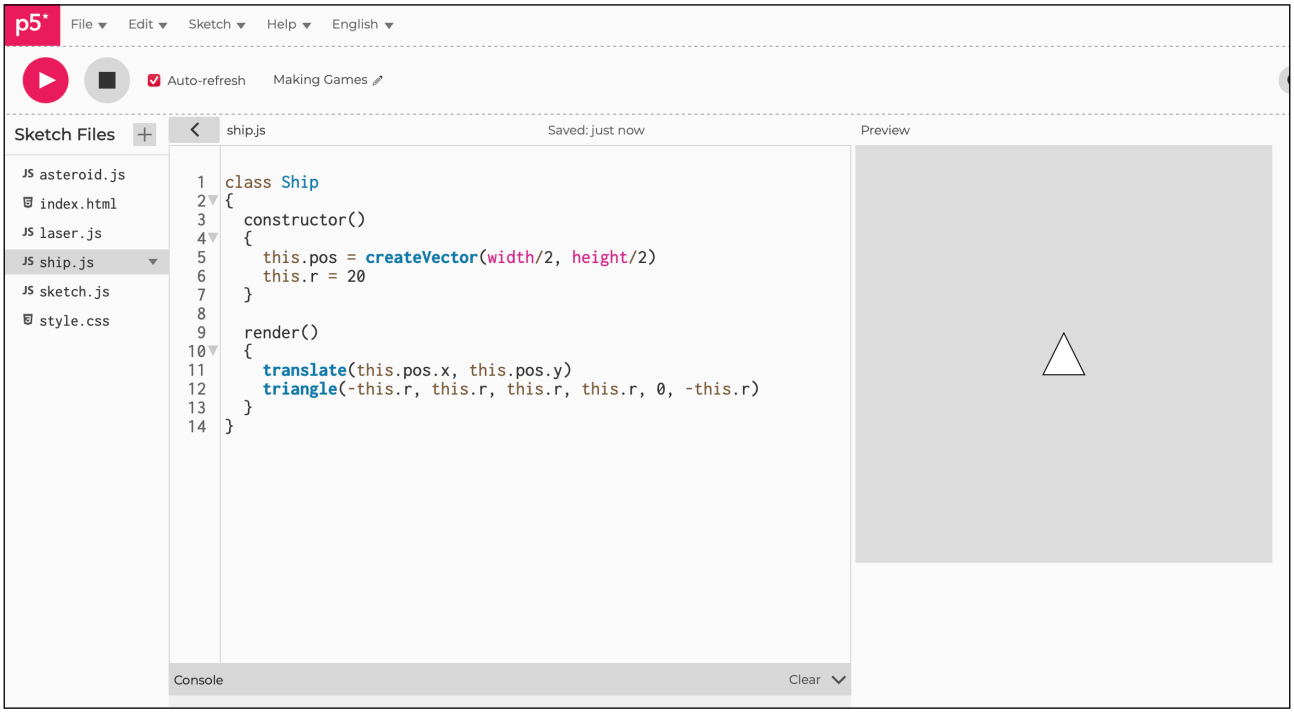
Sketch A5.3 in the middle

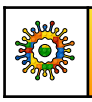
We will translate to the centre of the canvas.

ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
  }

  show()
  {
    translate(this.pos.x, this.pos.y)
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }
}
```

Figure A5.3


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
ship.js
Saved: just now
Preview
1 class Ship
2 {
3 constructor()
4 {
5 this.pos = createVector(width/2, height/2)
6 this.r = 20
7 }
8
9 render()
10 {
11 translate(this.pos.x, this.pos.y)
12 triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
13 }
14 }
Console
Clear



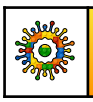
Sketch A5.4 rotate the ship

❗ sketch.js

We now want to be able to rotate the ship. Remember, unless you specify degrees, the angles are measured in radians (90° = π/2). This simply rotates the ship to show it is working.

sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
  ship.turn(0.01)
}
```



Sketch A5.5 turning

❗ ship.js

Creating the `turn()` function, the angle is give the value `0.01` in the main sketch.

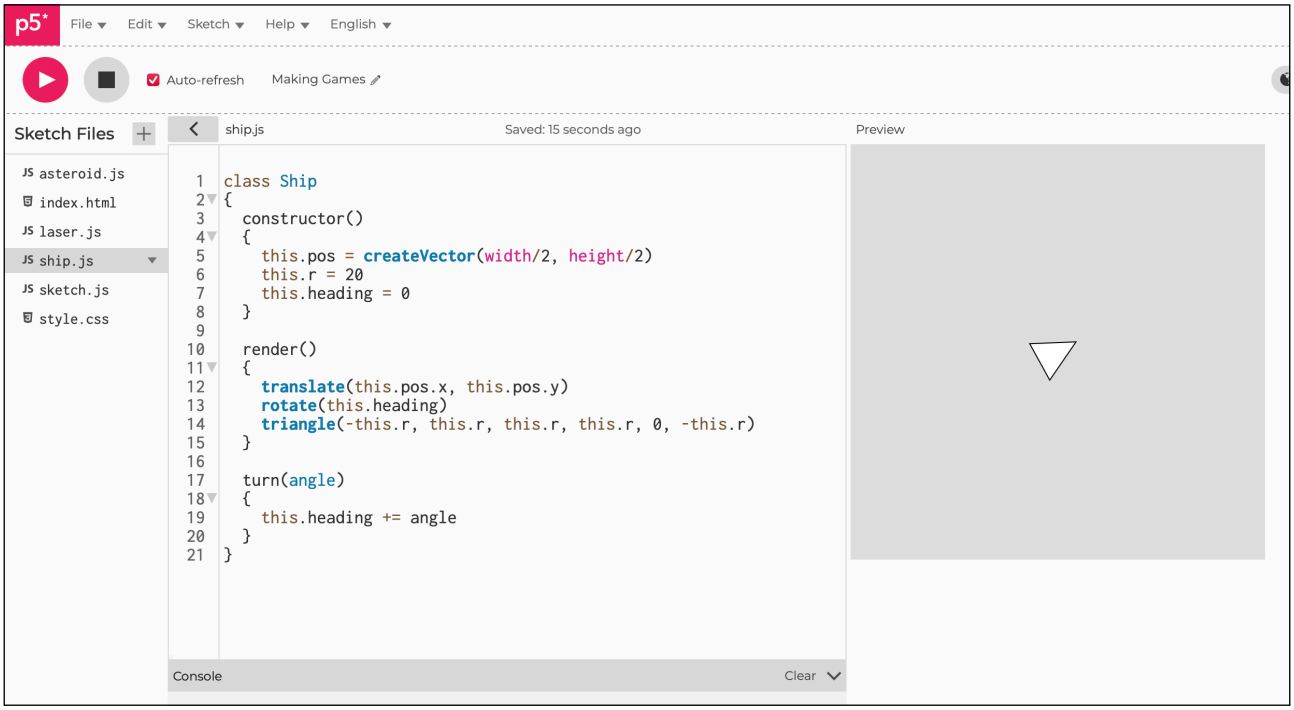
ship.js

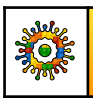
```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
  }

  show()
  {
    translate(this.pos.x, this.pos.y)
    rotate(this.heading)
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }

  turn(angle)
  {
    this.heading += angle
  }
}
```

Notes

Your ship should be slowly rotating clockwise.

Figure A5.5


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
ship.js
Saved: 15 seconds ago
Preview
class Ship
{
constructor()
{
this.pos = createVector(width/2, height/2)
this.r = 20
this.heading = 0
}
render()
{
translate(this.pos.x, this.pos.y)
rotate(this.heading)
triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
}
turn(angle)
{
this.heading += angle
}
}
Console
Clear



Sketch A5.6 rotating the ship

❗ sketch.js

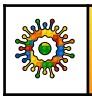
Now to gain control of the ship with a `keyPressed()` function. Tapping the arrow keys moves it round.

❗ remove the `ship.turn(0.01)`.

sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.turn(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.turn(-0.1)
  }
```

```
}
```



Sketch A5.7 stopping

Instead of tapping the arrow keys, now the ship will rotate when you hold down the key and stop when you release the key.

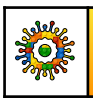
sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
}

function keyReleased()
{
  ship.setRotation(0)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
```

```
  {
    ship.setRotation(-0.1)
  }
}
```

Notes

You will get an error message until we create the functions.



Sketch A5.8 rotating the ship

❗ ship.js

Now you should be able to turn using the arrow keys (remember to click on the canvas first).

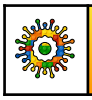
ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
    this.rotation = 0
  }

  show()
  {
    translate(this.pos.x, this.pos.y)
    rotate(this.heading)
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
```

```
}
```

Notes

You should be able to rotate the ship with the left and right arrow keys.



Sketch A5.9 applying a force

❗ sketch.js

Now to move it by applying a force to the ship. This will get it moving in a particular direction, but it is still not what we want.

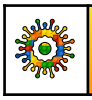
sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
}

function keyReleased()
{
  ship.setRotation(0)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
```

```
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boost()
  }
}
```

🗒️ Notes

Now need to add the `move()` and `boost()` functions in `ship.js`.



Sketch A5.10 on the move

❗ ship.js

In ship.js, we create the vectors we need. It gets it moving in one direction; it's a start...

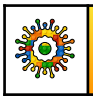
ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
    this.rotation = 0
    this.vel = createVector(0, 0)
  }

  move()
  {
    this.pos.add(this.vel)
  }

  boost()
  {
    let force = p5.Vector.fromAngle(this.heading)
    this.vel.add(force)
  }

  show()
  {
    translate(this.pos.x, this.pos.y)
    rotate(this.heading)
```

```
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
}
```

🗒️ Notes

You will now be able to rotate with the left and right arrow keys, but when you press the up arrow key, it glides gently in a direction. This will be fixed shortly, but at least it should be able to move in a direction, just not a very useful direction.



Sketch A5.11 accuracy and dampening

Adding some accuracy to the heading and some dampening to the force, which makes it much more controllable and manoeuvrable.

ship.js

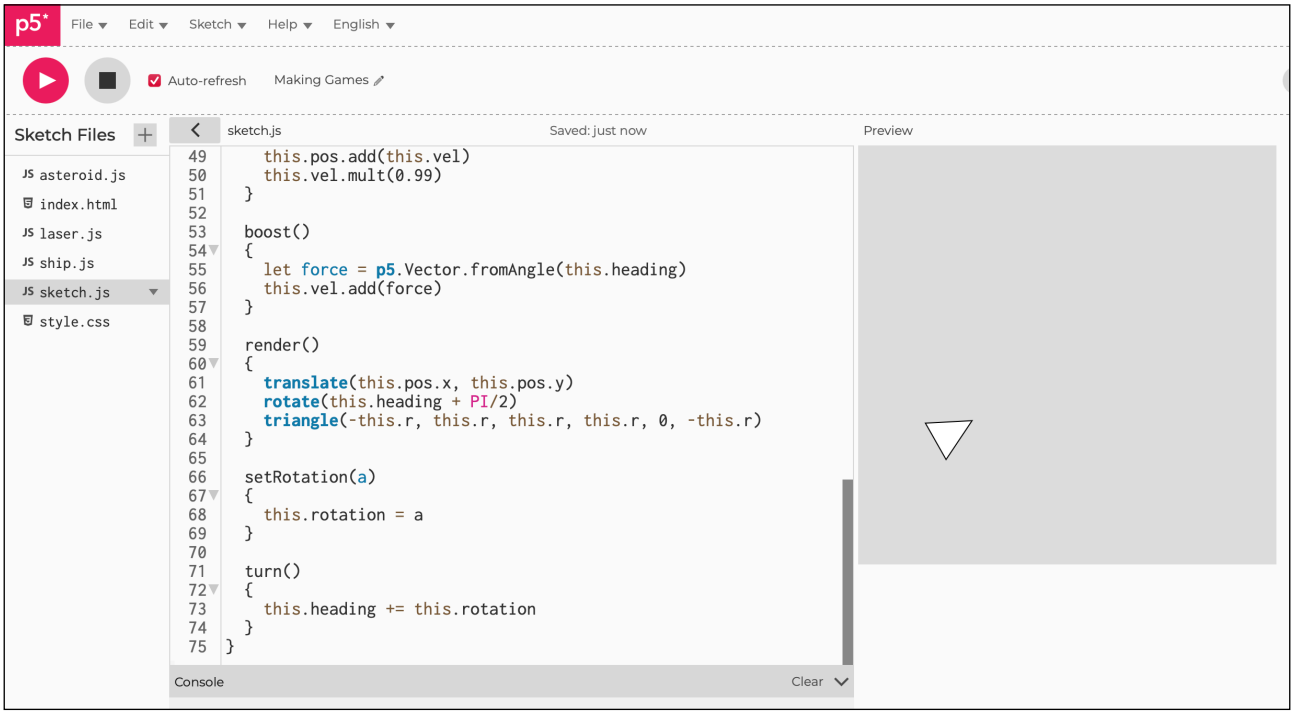
```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
    this.rotation = 0
    this.vel = createVector(0, 0)
  }

  move()
  {
    this.pos.add(this.vel)
    this.vel.mult(0.99)
  }

  boost()
  {
    let force = p5.Vector.fromAngle(this.heading)
    this.vel.add(force)
  }

  show()
  {
    translate(this.pos.x, this.pos.y)
    rotate(this.heading + PI/2)
```

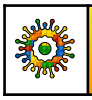
```
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
}
```

🗒️ Notes

Now it will move in a sensible direction when you press the up arrow key and slowly draw to a stop.

Figure A5.11


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: just now
Preview
49 this.pos.add(this.vel)
50 this.vel.mult(0.99)
51 }
52
53 boost()
54 {
55 let force = p5.Vector.fromAngle(this.heading)
56 this.vel.add(force)
57 }
58
59 render()
60 {
61 translate(this.pos.x, this.pos.y)
62 rotate(this.heading + PI/2)
63 triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
64 }
65
66 setRotation(a)
67 {
68 this.rotation = a
69 }
70
71 turn()
72 {
73 this.heading += this.rotation
74 }
75 }
Console
Clear



Sketch A5.12 holding the key down

❗ sketch.js

Now to tackle the issue of holding the key down to move the ship.

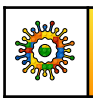
sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
```

```
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

Notes

Notice that it changes from `ship.boost()` to `ship.boosting(false)`, don't forget to add the `ing` to `boost`.



Sketch A5.13 boolean boost

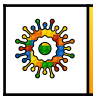
❗ ship.js

Adding `boosting()` to `ship.js`.

ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
    this.rotation = 0
    this.vel = createVector(0, 0)
    this.isBoosting = false
  }

  boosting(b)
  {
    this.isBoosting = b
  }

  move()
  {
    if (this.isBoosting)
    {
      this.boost()
    }
    this.pos.add(this.vel)
    this.vel.mult(0.99)
  }
```

```
  boost()
  {
    let force = p5.Vector.fromAngle(this.heading)
    force.mult(0.1)
    this.vel.add(force)
  }

  show()
  {
    translate(this.pos.x, this.pos.y)
    rotate(this.heading + PI/2)
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
}
```



Sketch A5.14 arriving at the edge

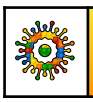
❗ sketch.js

Creating an `edge()` function so if it goes off the screen, it returns on the other side. To help understand this, the `pos.x`, `pos.y` change as the ship moves around the screen. They are the centre co-ordinates of the ship.

sketch.js

```
let ship

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
```

```
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```



Sketch A5.15 push and pop

❗ ship.js

We add `push()` and `pop()` to isolate the ship from the coming asteroids. It is important; otherwise, the translation of the asteroid is built on top of the translation of the ship, and it gets messy!

ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
    this.rotation = 0
    this.vel = createVector(0, 0)
    this.isBoosting = false
  }

  boosting(b)
  {
    this.isBoosting = b
  }

  move()
  {
    if (this.isBoosting)
    {
      this.boost()
    }
    this.pos.add(this.vel)
    this.vel.mult(0.99)
```

```
  }

  boost()
  {
    let force = p5.Vector.fromAngle(this.heading)
    force.mult(0.1)
    this.vel.add(force)
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    rotate(this.heading + PI/2)
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
    pop()
  }

  edges()
  {
    if (this.pos.x > width + this.r)
    {
      this.pos.x = -this.r
    }
    else if (this.pos.x < -this.r)
    {
      this.pos.x = width + this.r
    }
    if (this.pos.y > height + this.r)
    {
      this.pos.y = -this.r
    }
    else if (this.pos.y < -this.r)
```

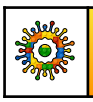
```
    {
      this.pos.y = height + this.r
    }
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
}
```

🗒️ Notes

When you move the ship off the edge of the canvas, it will reappear at the corresponding edge. Looking a bit more playable.

Making the asteroids

Now we add the asteroids, making them irregular. You could do it simpler by just having circles.



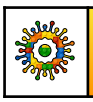
Sketch A5.16 Asteroid class

❗ asteroid.js

Add another file called `asteroid.js` and now for an asteroid. To start with, we will draw a simple circle to represent an asteroid. Draw one at a random location. Also, use `push()` and `pop()` because there will be many asteroids roaming around independently.

asteroid.js

```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.r = 50
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    circle(0, 0, this.r)
    pop()
  }
}
```



Sketch A5.17 random circle

❗ sketch.js

And in sketch.js, we get ready to include the asteroids; this will create a circle at a random point on the canvas.

sketch.js

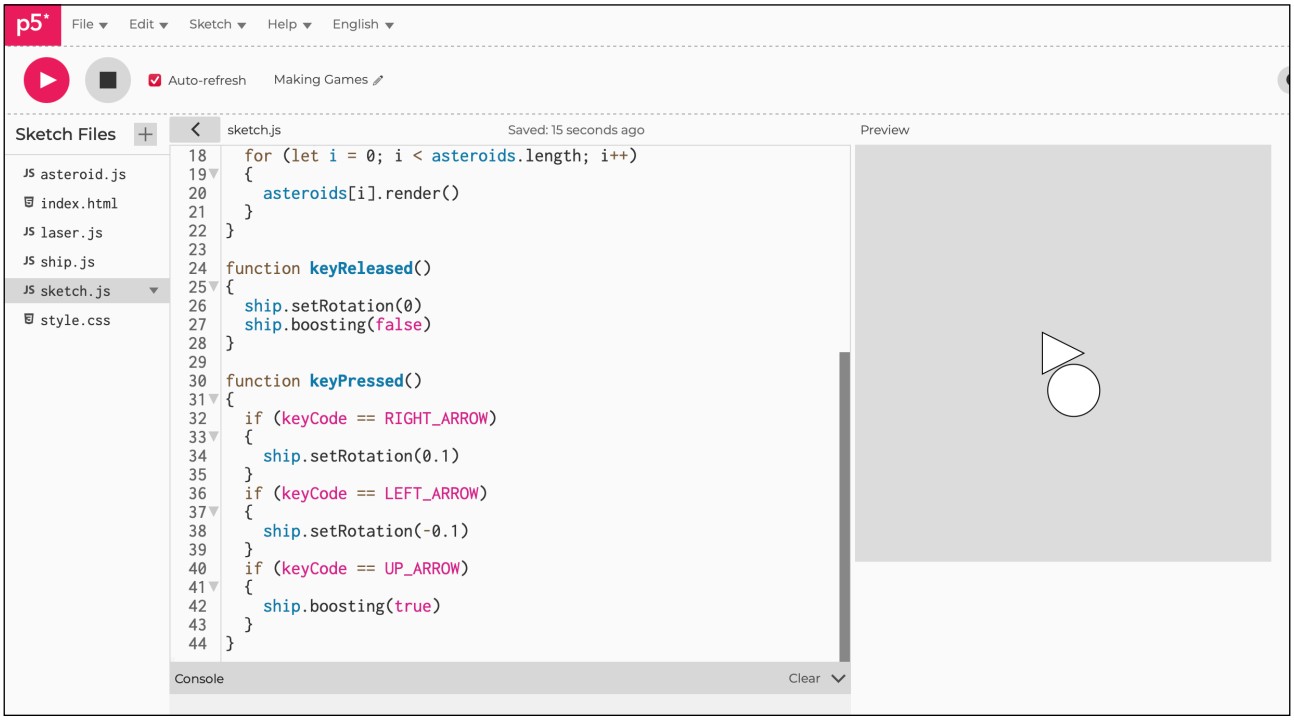
```
let ship
let asteroids = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  asteroids.push(new Asteroid())
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
  }
}

function keyReleased()
{
  ship.setRotation(0)
```

```
  ship.boosting(false)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

The asteroid, for now, is just a single, plain circle.

Figure A5.17


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: 15 seconds ago
Preview
18 for (let i = 0; i < asteroids.length; i++)
19 {
20 asteroids[i].render()
21 }
22 }
23
24 function keyReleased()
25 {
26 ship.setRotation(0)
27 ship.boosting(false)
28 }
29
30 function keyPressed()
31 {
32 if (keyCode == RIGHT_ARROW)
33 {
34 ship.setRotation(0.1)
35 }
36 if (keyCode == LEFT_ARROW)
37 {
38 ship.setRotation(-0.1)
39 }
40 if (keyCode == UP_ARROW)
41 {
42 ship.boosting(true)
43 }
44 }
Console
Clear

Sketch A5.18 more like an asteroid

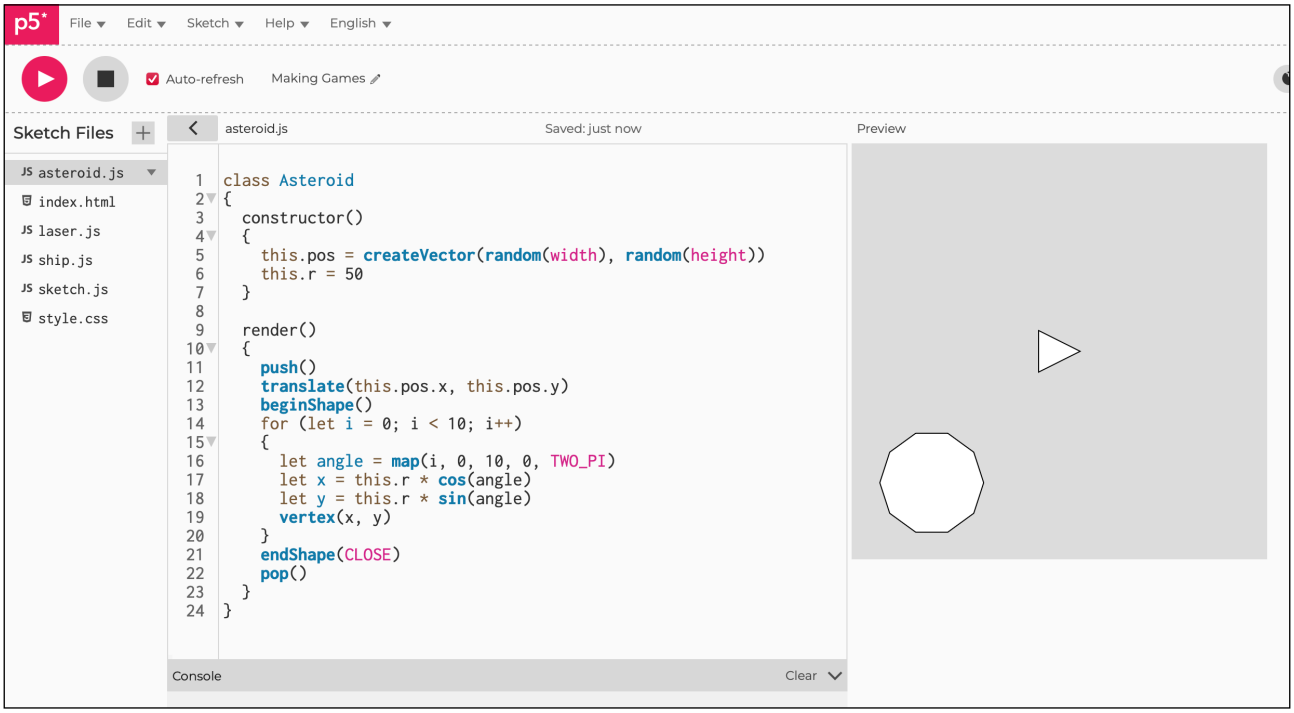
❗ asteroid.js

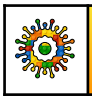
The circle in the asteroid.js is going to be replaced with a set of vertices because we want to make an object that is not regular a bit more asteroidy in shape. This gives us a regular 10-sided shape.

asteroid.js

```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.r = 50
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    beginShape()
    for (let i = 0; i < 10; i++)
    {
      let angle = map(i, 0, 10, 0, TWO_PI)
      let x = this.r * cos(angle)
      let y = this.r * sin(angle)
      vertex(x, y)
    }
    endShape(CLOSE)
    pop()
  }
}
```

Figure A5.18


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
asteroid.js
Saved: just now
Preview
class Asteroid
{
constructor()
{
this.pos = createVector(random(width), random(height))
this.r = 50
}
render()
{
push()
translate(this.pos.x, this.pos.y)
beginShape()
for (let i = 0; i < 10; i++)
{
let angle = map(i, 0, 10, 0, TWO_PI)
let x = this.r * cos(angle)
let y = this.r * sin(angle)
vertex(x, y)
}
endShape(CLOSE)
pop()
}
}
Console
Clear



Sketch A5.19 regular asteroids

Now create different-sided shapes (still regular for now); `floor` means making it a whole number. If you keep refreshing, you get different-sided shapes at different locations (positions).

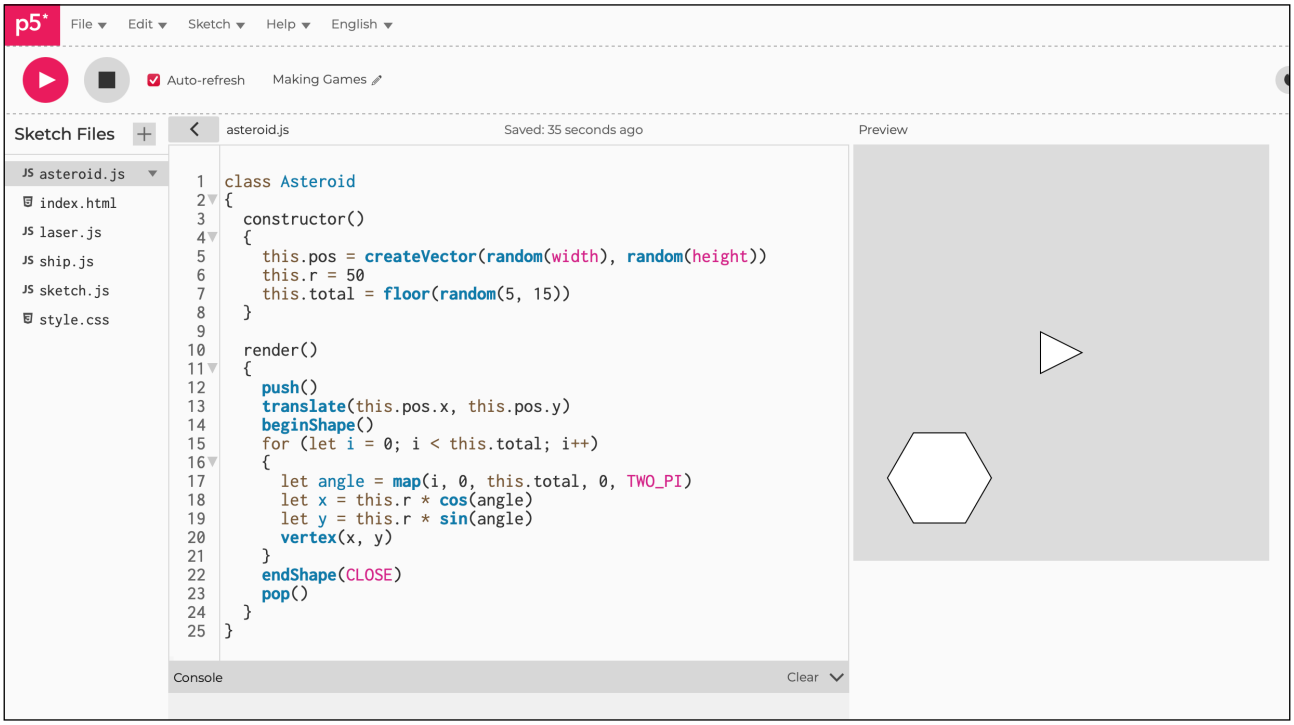
asteroid.js

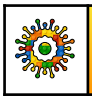
```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.r = 50
    this.total = floor(random(5, 15))
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    beginShape()
    for (let i = 0; i < this.total; i++)
    {
      let angle = map(i, 0, this.total, 0, TWO_PI)
      let x = this.r * cos(angle)
      let y = this.r * sin(angle)
      vertex(x, y)
    }
    endShape(CLOSE)
    pop()
  }
}
```

Notes

The asteroid will have a random position and a random number of sides between five and fifteen.

Figure A5.19


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
asteroid.js
Saved: 35 seconds ago
Preview
class Asteroid
{
constructor()
{
this.pos = createVector(random(width), random(height))
this.r = 50
this.total = floor(random(5, 15))
}
render()
{
push()
translate(this.pos.x, this.pos.y)
beginShape()
for (let i = 0; i < this.total; i++)
{
let angle = map(i, 0, this.total, 0, TWO_PI)
let x = this.r * cos(angle)
let y = this.r * sin(angle)
vertex(x, y)
}
endShape(CLOSE)
pop()
}
}
Console
Clear



Sketch A5.20 more asteroids

❗ sketch.js

Creating more of them in the sketch.js.

sketch.js

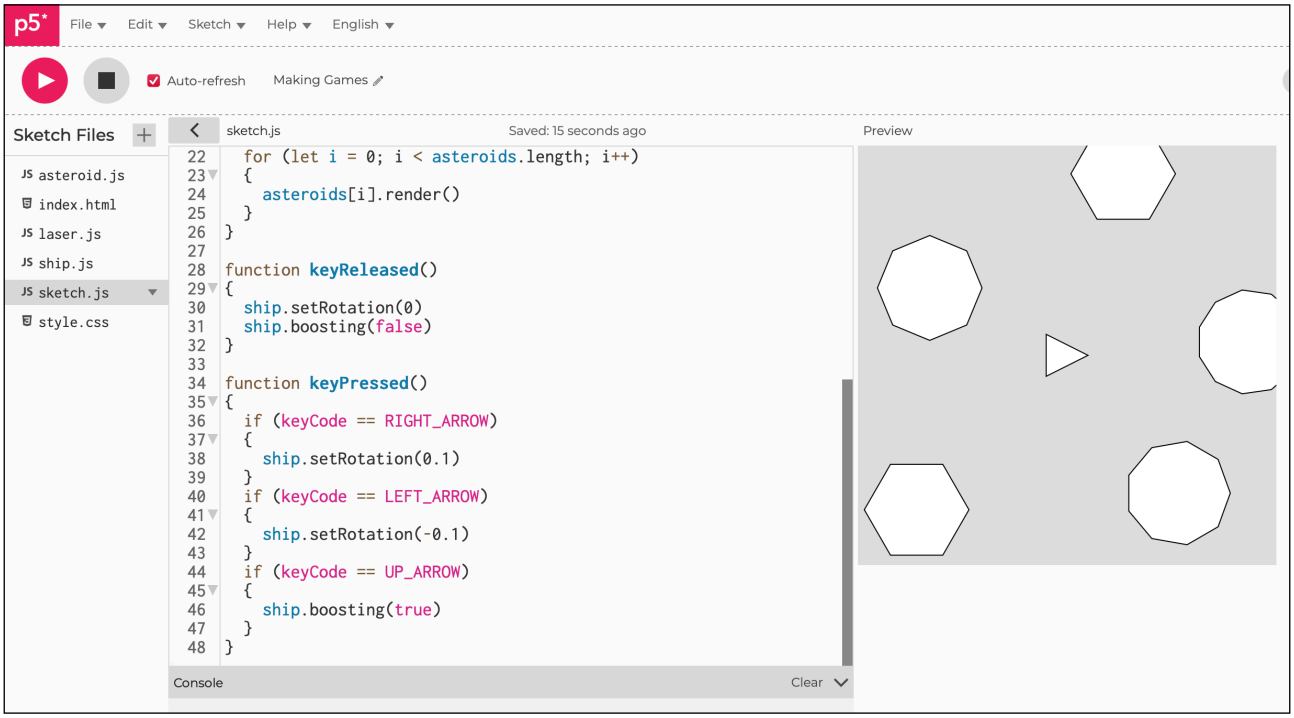
```
let ship
let asteroids = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()

  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
  }
}
```

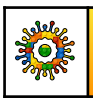
```
function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

Creating five new asteroids.

Figure A5.20


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: 15 seconds ago
Preview
for (let i = 0; i < asteroids.length; i++)
{
asteroids[i].render()
}
function keyReleased()
{
ship.setRotation(0)
ship.boosting(false)
}
function keyPressed()
{
if (keyCode == RIGHT_ARROW)
{
ship.setRotation(0.1)
}
if (keyCode == LEFT_ARROW)
{
ship.setRotation(-0.1)
}
if (keyCode == UP_ARROW)
{
ship.boosting(true)
}
}
Console
Clear



Sketch A5.21 smaller asteroids

❗ asteroid.js

Make them a bit smaller and random sizes in asteroid.js.

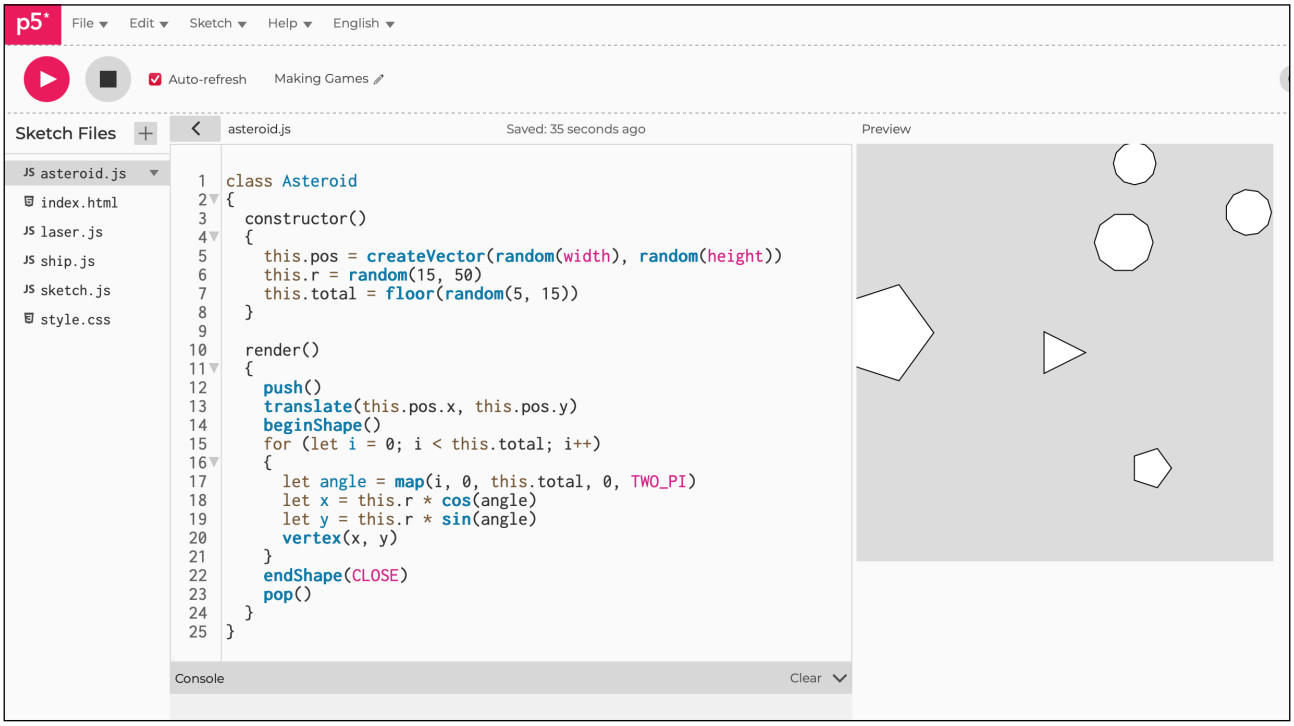
asteroid.js

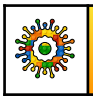
```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.r = random(15, 50)
    this.total = floor(random(5, 15))
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    beginShape()
    for (let i = 0; i < this.total; i++)
    {
      let angle = map(i, 0, this.total, 0, TWO_PI)
      let x = this.r * cos(angle)
      let y = this.r * sin(angle)
      vertex(x, y)
    }
    endShape(CLOSE)
    pop()
  }
}
```

Notes

Making them a bit smaller.

Figure A5.21


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
asteroid.js
Saved: 35 seconds ago
Preview
class Asteroid
{
constructor()
{
this.pos = createVector(random(width), random(height))
this.r = random(15, 50)
this.total = floor(random(5, 15))
}
render()
{
push()
translate(this.pos.x, this.pos.y)
beginShape()
for (let i = 0; i < this.total; i++)
{
let angle = map(i, 0, this.total, 0, TWO_PI)
let x = this.r * cos(angle)
let y = this.r * sin(angle)
vertex(x, y)
}
endShape(CLOSE)
pop()
}
}
Console
Clear



Sketch A5.22 irregular asteroids

Making them irregular in asteroid.js.

asteroid.js

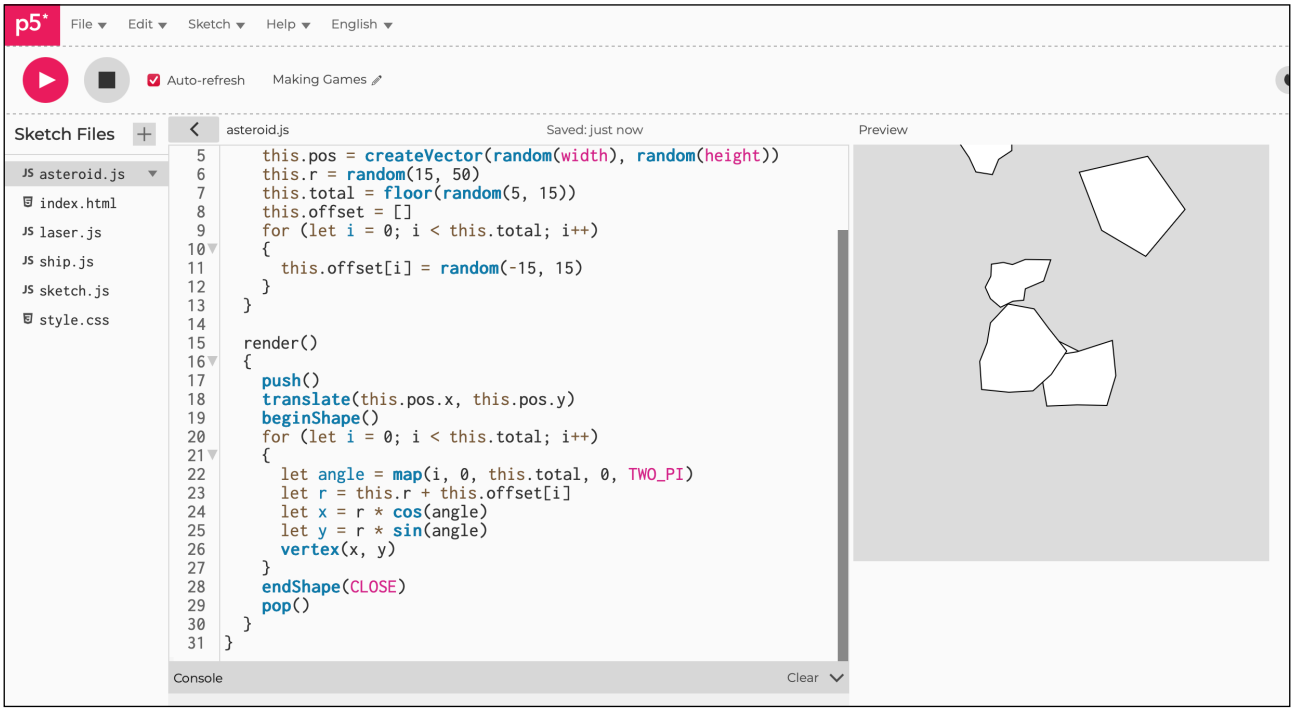
```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.r = random(15, 50)
    this.total = floor(random(5, 15))
    this.offset = []
    for (let i = 0; i < this.total; i++)
    {
      this.offset[i] = random(-15, 15)
    }
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    beginShape()
    for (let i = 0; i < this.total; i++)
    {
      let angle = map(i, 0, this.total, 0, TWO_PI)
      let r = this.r + this.offset[i]
      let x = r * cos(angle)
      let y = r * sin(angle)
      vertex(x, y)
    }
```

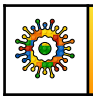
```
    endShape(CLOSE)
    pop()
  }
}
```

Notes

Now they are more irregular.

Figure A5.22


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
asteroid.js
Saved: just now
Preview
5 this.pos = createVector(random(width), random(height))
6 this.r = random(15, 50)
7 this.total = floor(random(5, 15))
8 this.offset = []
9 for (let i = 0; i < this.total; i++)
10 {
11 this.offset[i] = random(-15, 15)
12 }
13 }
14
15 render()
16 {
17 push()
18 translate(this.pos.x, this.pos.y)
19 beginShape()
20 for (let i = 0; i < this.total; i++)
21 {
22 let angle = map(i, 0, this.total, 0, TWO_PI)
23 let r = this.r + this.offset[i]
24 let x = r * cos(angle)
25 let y = r * sin(angle)
26 vertex(x, y)
27 }
28 endShape(CLOSE)
29 pop()
30 }
31 }
Console
Clear



Sketch A5.23 floating asteroids

We now want them to float around like asteroids! With a `move()` function in `asteroid.js`. We give velocity (this.vel) a random value from `p5.Vector.random2D()`.

asteroid.js

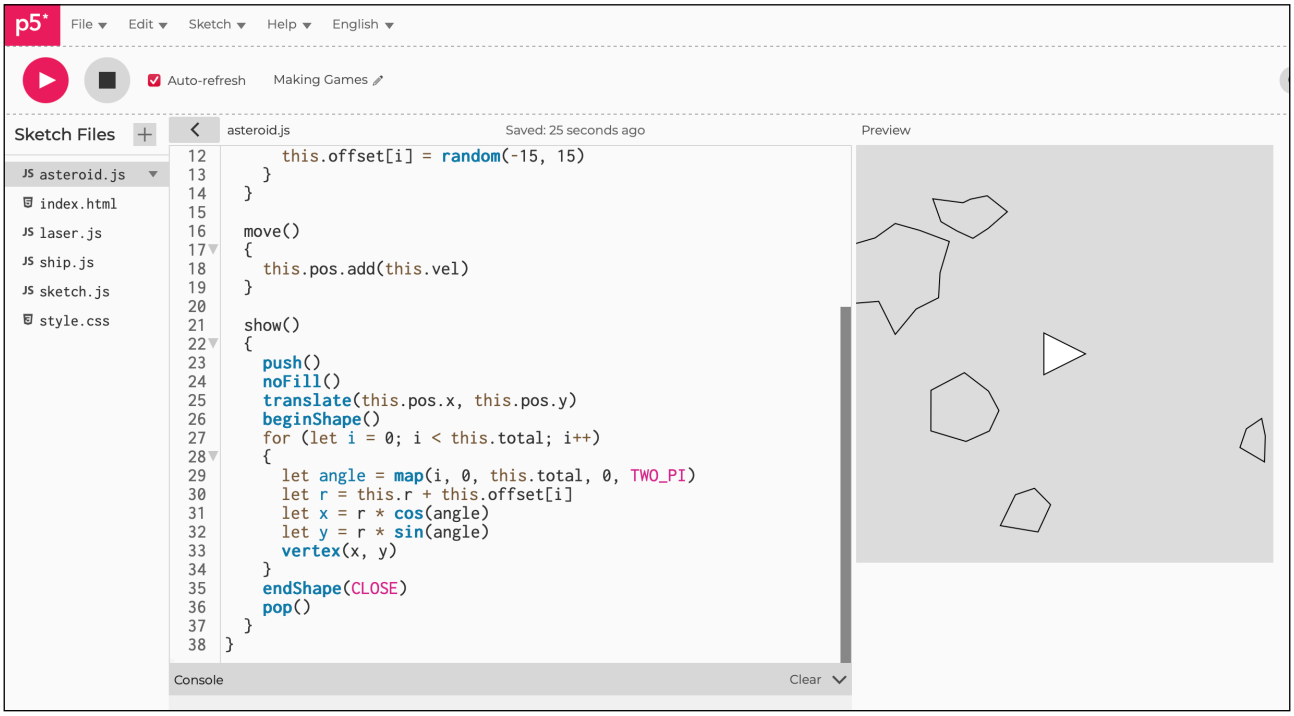
```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.vel = p5.Vector.random2D()
    this.r = random(15, 50)
    this.total = floor(random(5, 15))
    this.offset = []
    for (let i = 0; i < this.total; i++)
    {
      this.offset[i] = random(-15, 15)
    }
  }


  move()
  {
    this.pos.add(this.vel)
  }


  show()
  {
    push()
    noFill()
    translate(this.pos.x, this.pos.y)
    beginShape()
```

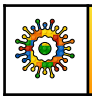
```
    for (let i = 0; i < this.total; i++)
    {
      let angle = map(i, 0, this.total, 0, TWO_PI)
      let r = this.r + this.offset[i]
      let x = r * cos(angle)
      let y = r * sin(angle)
      vertex(x, y)
    }
    endShape(CLOSE)
    pop()
  }
}
```

🗒️ Notes

They won't float just yet!

Figure A5.23


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
asteroid.js
Saved: 25 seconds ago
Preview
this.offset[i] = random(-15, 15)
}
}
move()
{
this.pos.add(this.vel)
}
show()
{
push()
noFill()
translate(this.pos.x, this.pos.y)
beginShape()
for (let i = 0; i < this.total; i++)
{
let angle = map(i, 0, this.total, 0, TWO_PI)
let r = this.r + this.offset[i]
let x = r * cos(angle)
let y = r * sin(angle)
vertex(x, y)
}
endShape(CLOSE)
pop()
}
}
Console
Clear



Sketch A5.24 I like to move it, move it

❗ sketch.js

In `sketch.js`, we add the `move()` function.

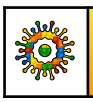
sketch.js

```
let ship
let asteroids = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
  }
}
```

```
function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

They should be floating around and disappearing off the canvas. We can get them back.



Sketch A5.25 reaching the canvas edges

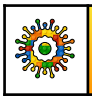
❗ asteroid.js

We need the asteroids to stay on the screen, so we will use the code from the `ship.js` edges function and adapt it.

asteroid.js

```
class Asteroid
{
  constructor()
  {
    this.pos = createVector(random(width), random(height))
    this.vel = p5.Vector.random2D()
    this.r = random(15, 50)
    this.total = floor(random(5, 15))
    this.offset = []
    for (let i = 0; i < this.total; i++)
    {
      this.offset[i] = random(-15, 15)
    }
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    noFill()
    translate(this.pos.x, this.pos.y)
    beginShape()
```

```
    for (let i = 0; i < this.total; i++)
    {
      let angle = map(i, 0, this.total, 0, TWO_PI)
      let r = this.r + this.offset[i]
      let x = r * cos(angle)
      let y = r * sin(angle)
      vertex(x, y)
    }
    endShape(CLOSE)
    pop()
  }

  edges()
  {
    if (this.pos.x > width + this.r)
    {
      this.pos.x = -this.r
    }
    else if (this.pos.x < -this.r)
    {
      this.pos.x = width + this.r
    }
    if (this.pos.y > height + this.r)
    {
      this.pos.y = -this.r
    }
    else if (this.pos.y < -this.r)
    {
      this.pos.y = height + this.r
    }
  }
}
```



Sketch A5.26 reappears as if by magic

❗ sketch.js

Now adding the edges() function to sketch.js.

sketch.js

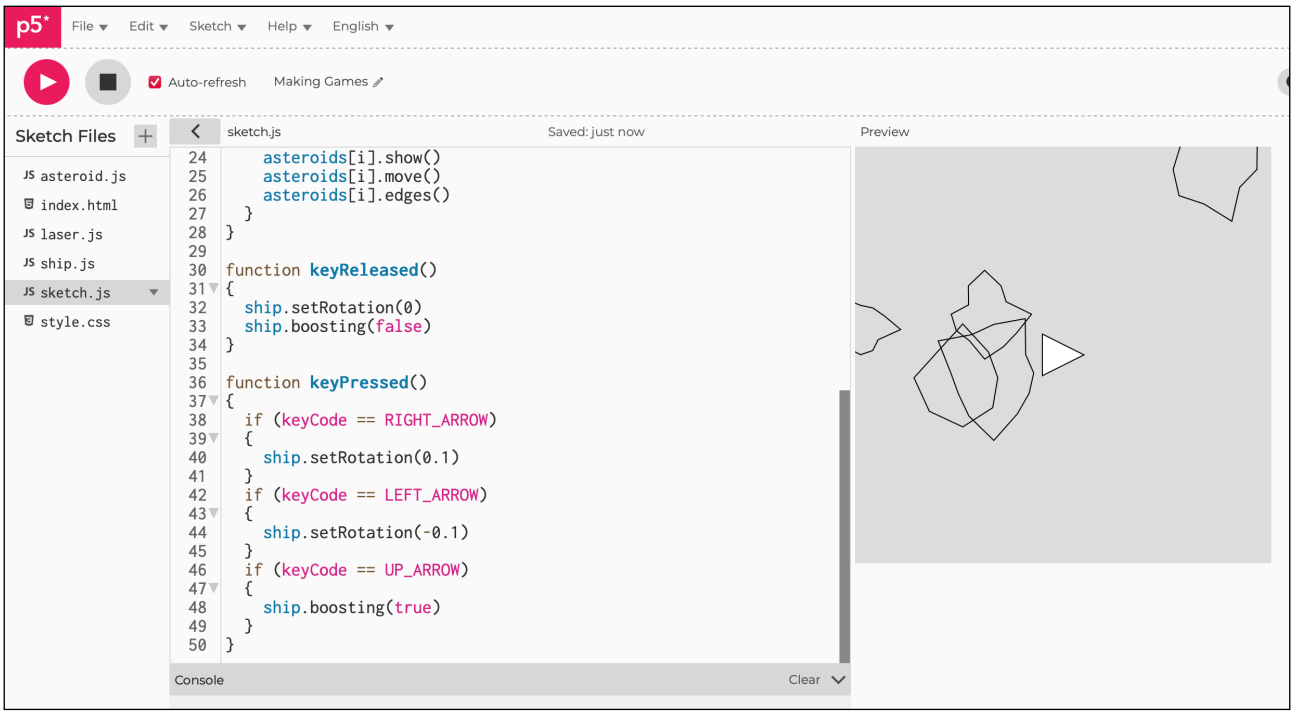
```
let ship
let asteroids = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()

  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
```

```
  }
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

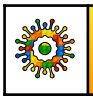
Floating in space and reappearing after disappearing off the edge of the canvas. We have a small canvas and a large ship and large asteroids; this makes the game quite a challenge, but this is a demo introducing key components of coding, not making a full-blown playable commercial game.

Figure A5.26


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: just now
Preview
24 asteroids[i].show()
25 asteroids[i].move()
26 asteroids[i].edges()
27 }
28 }
29
30 function keyReleased()
31 {
32 ship.setRotation(0)
33 ship.boosting(false)
34 }
35
36 function keyPressed()
37 {
38 if (keyCode == RIGHT_ARROW)
39 {
40 ship.setRotation(0.1)
41 }
42 if (keyCode == LEFT_ARROW)
43 {
44 ship.setRotation(-0.1)
45 }
46 if (keyCode == UP_ARROW)
47 {
48 ship.boosting(true)
49 }
50 }
Console
Clear

Making lasers

What would this game be without something to fire to destroy the asteroids?



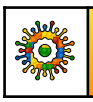
Sketch A5.27 adding the lasers

❗ laser.js

Now to add the lasers, but first add a new file called `laser.js` if not already done (remember `index.html`).

laser.js

```
class Laser
{
  constructor()
  {
    this.pos = createVector()
    this.vel = createVector()
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    strokeWeight(5)
    point(this.pos.x, this.pos.y)
    pop()
  }
}
```



Sketch A5.28 an array of lasers

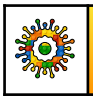
❗ sketch.js

Create an array for the laser particles.

sketch.js

```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
  for (let i = 0; i < lasers.length; i++)
  {
    lasers[i].show()
    lasers[i].move()
  }
```

```
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```



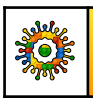
Sketch A5.29 laser as a particle

❗ laser.js

Create the laser particles when pressing the spacebar at the position of the ship.

laser.js

```
class Laser
{
  constructor(shipPosition)
  {
    this.pos = createVector(shipPosition.x, shipPosition.y)
    this.vel = createVector()
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    strokeWeight(5)
    point(this.pos.x, this.pos.y)
    pop()
  }
}
```



Sketch A5.30 spacebar laser

❗ sketch.js

In sketch.js, we use the keyPressed() function we already have. Now, when you press the arrow keys, you move the ship around, and when you press the spacebar, it deposits a laser particle!

sketch.js

```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()

  for (let i = 0; i < lasers.length; i++)
  {
    lasers[i].show()
```

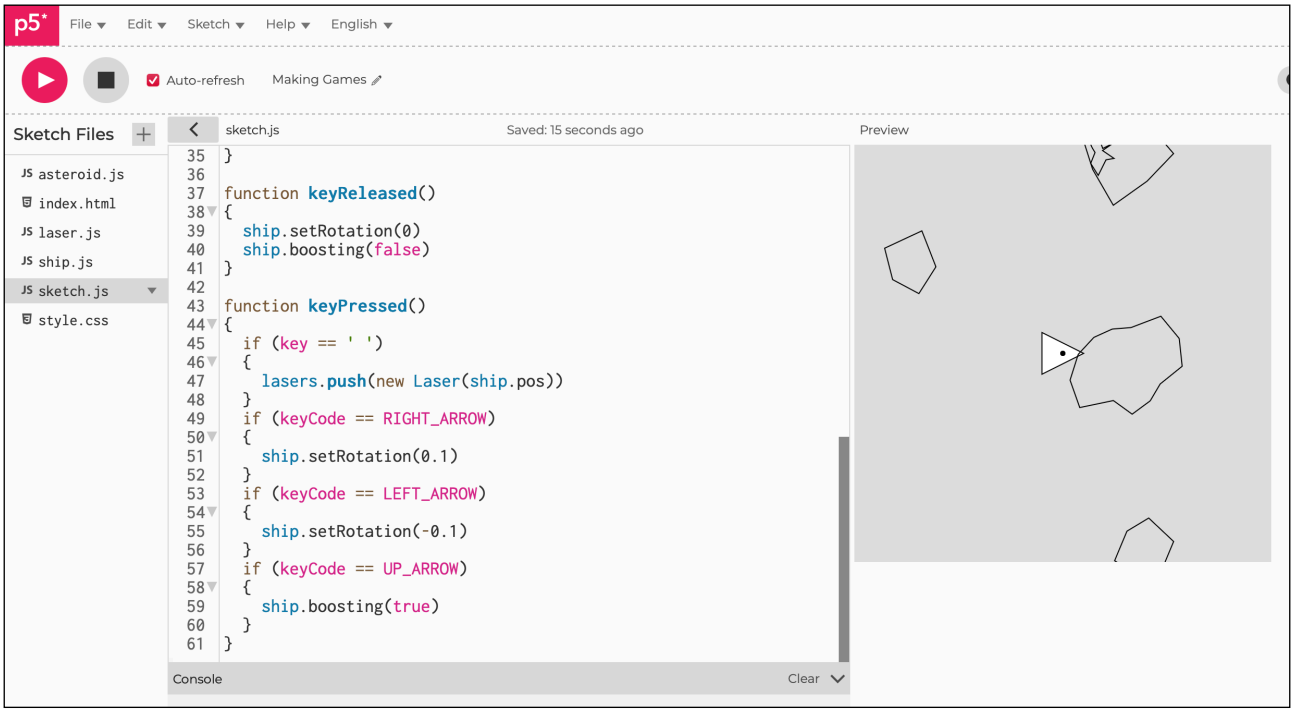
```
    lasers[i].move()
  }

  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
```

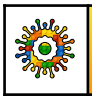
```
    ship.boosting(true)
  }
}
```

🗒️ Notes

You should get a little dot to appear when you press the space bar; the laser isn't going anywhere and there is only one laser particle.

Figure A5.30


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: 15 seconds ago
Preview
35 }
36
37 function keyReleased()
38 {
39 ship.setRotation(0)
40 ship.boosting(false)
41 }
42
43 function keyPressed()
44 {
45 if (key == ' ')
46 {
47 lasers.push(new Laser(ship.pos))
48 }
49 if (keyCode == RIGHT_ARROW)
50 {
51 ship.setRotation(0.1)
52 }
53 if (keyCode == LEFT_ARROW)
54 {
55 ship.setRotation(-0.1)
56 }
57 if (keyCode == UP_ARROW)
58 {
59 ship.boosting(true)
60 }
61 }
Console
Clear



Sketch A5.31 directing the laser

We move the particles in the direction that the ship is pointing by adding the `ship.heading` argument into the new laser particle.

sketch.js

```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()

  for (let i = 0; i < lasers.length; i++)
  {
    lasers[i].show()
    lasers[i].move()
  }
```

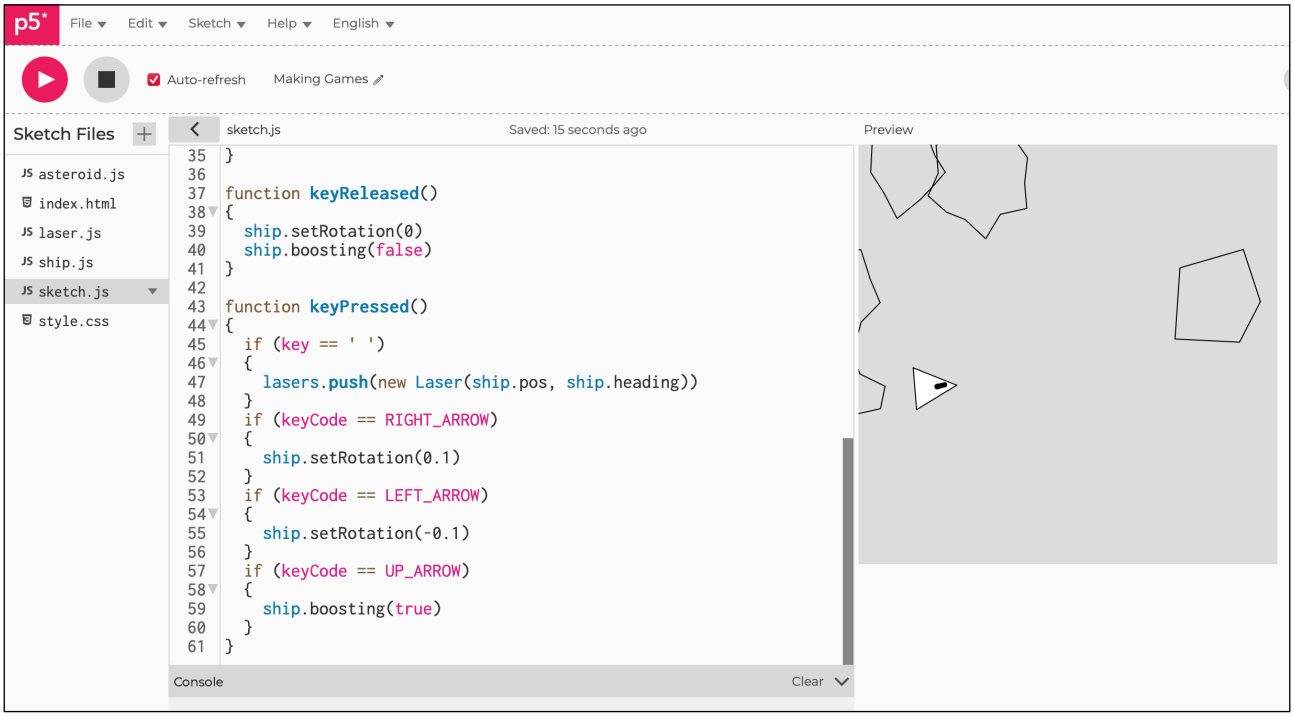
```
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
```

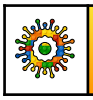
```
}
```

Notes

Don't worry if nothing much happens to the lasers; that will soon change. If you press the space bar a few times, you'll see the particles line up.

Figure A5.31


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: 15 seconds ago
Preview
35 }
36
37 function keyReleased()
38 {
39 ship.setRotation(0)
40 ship.boosting(false)
41 }
42
43 function keyPressed()
44 {
45 if (key == ' ')
46 {
47 lasers.push(new Laser(ship.pos, ship.heading))
48 }
49 if (keyCode == RIGHT_ARROW)
50 {
51 ship.setRotation(0.1)
52 }
53 if (keyCode == LEFT_ARROW)
54 {
55 ship.setRotation(-0.1)
56 }
57 if (keyCode == UP_ARROW)
58 {
59 ship.boosting(true)
60 }
61 }
Console
Clear



Sketch A5.32 firing the laser cannon

❗ laser.js

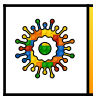
While over in `laser.js`, we add angle as an argument to the laser particle and use the `p5.Vector`. Now when you press the `spacebar`, it shoots the particle laser.

laser.js

```
class Laser
{
  constructor(shipPosition, angle)
  {
    this.pos = createVector(shipPosition.x, shipPosition.y)
    this.vel = p5.Vector.fromAngle(angle)
    this.vel.mult(5)
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    strokeWeight(5)
    point(this.pos.x, this.pos.y)
    pop()
  }
}
```

Notes

The ship now fires its laser cannons in whichever direction it is facing. It moves very fast (too fast to get a screen image).



Sketch A5.33 a tweaking of the laser

❗ sketch.js

Move the ship in `sketch.js` after the laser so that it draws it afterwards, thus hiding the fact that the laser particles are coming from the centre.

Just cut and paste:

```
ship.show()
ship.turn()
ship.move()
ship.edges()
```

And put them after the asteroids loop.

sketch.js

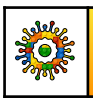
```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < lasers.length; i++)
  {
```

```
    lasers[i].show()
    lasers[i].move()
  }
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
```

```
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

Starting to look like a real game.



Sketch A5.34 testing a hit

Testing to see if the laser particles hit the asteroids. A slight rearranging of `sketch.js`.

Step 1: Move the asteroids loop above the lasers loop.
Step 2: Create another loop (asteroids with a variable j) inside the laser loop.

sketch.js

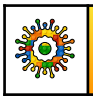
```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
```

```
  for (let i = 0; i < lasers.length; i++)
  {
    lasers[i].show()
    lasers[i].move()
    for (let j = 0; j < asteroids.length; j++)
    {
      if (lasers[i].hits(asteroids[j]))
      {
        return
      }
    }
  }
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
```

```
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

Don't run this just yet, not until you put the `hits()` function in `laser.js`. Also, it is an empty loop for the moment.



Sketch A5.35 it's a hit

❗ laser.js

Adding the `hits()` function onto the laser.js, you should get a `hit` in the console if everything is working fine.

laser.js

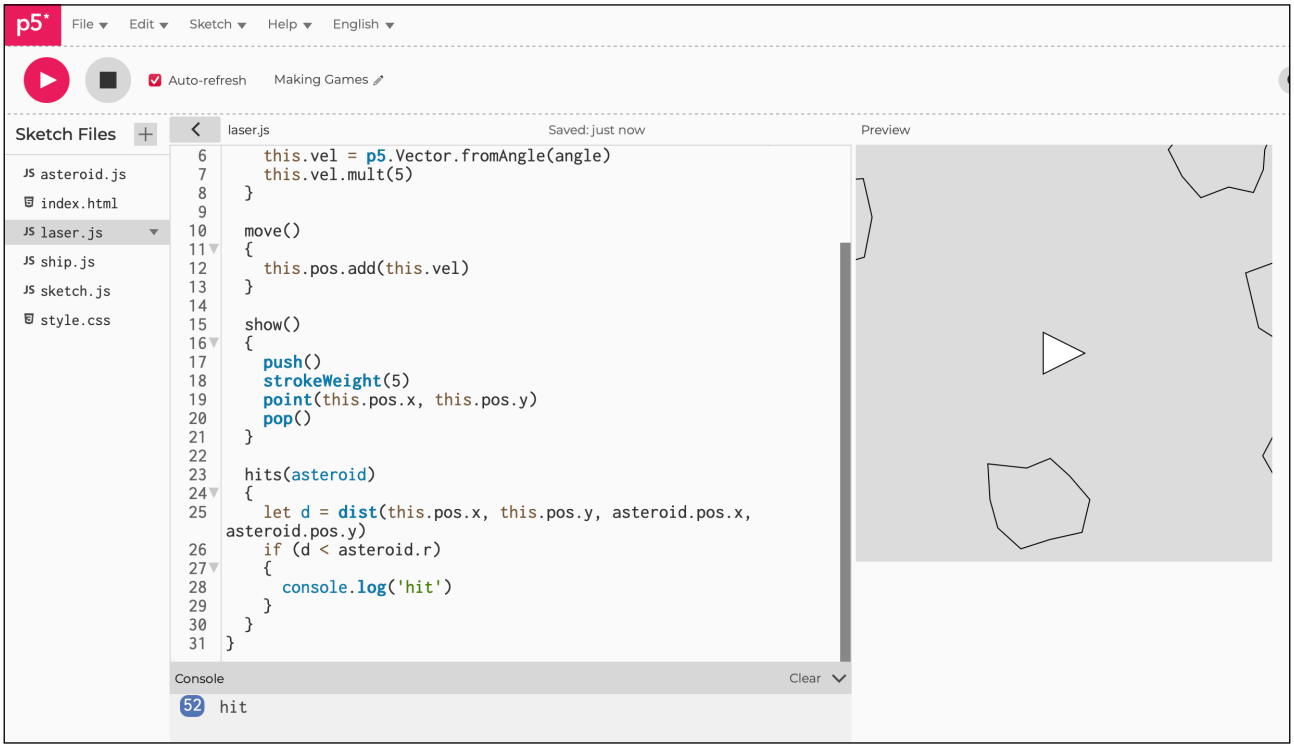
```
class Laser
{
  constructor(shipPosition, angle)
  {
    this.pos = createVector(shipPosition.x, shipPosition.y)
    this.vel = p5.Vector.fromAngle(angle)
    this.vel.mult(5)
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    strokeWeight(5)
    point(this.pos.x, this.pos.y)
    pop()
  }

  hits(asteroid)
  {
    let d = dist(this.pos.x, this.pos.y, asteroid.pos.x,
asteroid.pos.y)
```

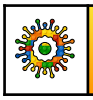
```
    if (d < asteroid.r)
    {
      console.log('hit')
    }
  }
}
```

Notes

You should be getting hits in the console every time you fire a laser and it hits an asteroid.

Figure A5.35


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
laser.js
Saved: just now
Preview
6 this.vel = p5.Vector.fromAngle(angle)
7 this.vel.mult(5)
8 }
9
10 move()
11 {
12 this.pos.add(this.vel)
13 }
14
15 show()
16 {
17 push()
18 strokeWeight(5)
19 point(this.pos.x, this.pos.y)
20 pop()
21 }
22
23 hits(asteroid)
24 {
25 let d = dist(this.pos.x, this.pos.y, asteroid.pos.x, asteroid.pos.y)
26 if (d < asteroid.r)
27 {
28 console.log('hit')
29 }
30 }
31 }
Console
Clear
52 hit



Sketch A5.36 working well, hopefully

If everything is working fine, then remove the `console.log()` and return `true`.

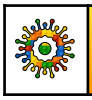
laser.js

```
class Laser
{
  constructor(shipPosition, angle)
  {
    this.pos = createVector(shipPosition.x, shipPosition.y)
    this.vel = p5.Vector.fromAngle(angle)
    this.vel.mult(5)
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    strokeWeight(5)
    point(this.pos.x, this.pos.y)
    pop()
  }

  hits(asteroid)
  {
    let d = dist(this.pos.x, this.pos.y, asteroid.pos.x,
asteroid.pos.y)
    if (d < asteroid.r)
```

```
    {
      return true
    }
    else
    {
      return false
    }
  }
}
```

🗒️ Notes

You may get some flicker from the ship when firing the lasers.



Sketch A5.37 breaking the asteroid

❗ sketch.js

Breaking up the asteroids when the laser particle returns true. Then we need to create a function in asteroid.js to break up the asteroid into two new asteroids. The first step will just delete the asteroid altogether.

sketch.js

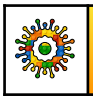
```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }

  for (let i = 0; i < lasers.length; i++)
```

```
  {
    lasers[i].show()
    lasers[i].move()
    for (let j = asteroids.length - 1; j >= 0; j--)
    {
      if (lasers[i].hits(asteroids[j]))
      {
        let newA = asteroids[j].breakup()
        asteroids.splice(j, 1)
      }
    }
  }

  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
```

```
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

Notes

At this stage, it will just delete the asteroid.



Sketch A5.38 the breakup

❗ asteroid.js

Creating the `breakup()` function in `asteroid.js`.

asteroid.js

```
class Asteroid
{
  constructor(pos)
  {
    if (pos)
    {
      this.pos = pos.copy()
    }
    else
    {
      this.pos = createVector(random(width), random(height))
    }
    this.vel = p5.Vector.random2D()
    this.r = random(15, 50)
    this.total = floor(random(5, 15))
    this.offset = []
    for (let i = 0; i < this.total; i++)
    {
      this.offset[i] = random(-15, 15)
    }
  }

  move()
  {
    this.pos.add(this.vel)
  }
```

```
show()
{
  push()
  noFill()
  translate(this.pos.x, this.pos.y)
  beginShape()
  for (let i = 0; i < this.total; i++)
  {
    let angle = map(i, 0, this.total, 0, TWO_PI)
    let r = this.r + this.offset[i]
    let x = r * cos(angle)
    let y = r * sin(angle)
    vertex(x, y)
  }
  endShape(CLOSE)
  pop()
}
```

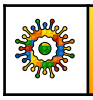
```
breakup()
{
  let newA = []
  newA[0] = new Asteroid(this.pos)
  newA[1] = new Asteroid(this.pos)
  return newA
}
```

```
edges()
{
  if (this.pos.x > width + this.r)
  {
    this.pos.x = -this.r
  }
```

```
    else if (this.pos.x < -this.r)
    {
      this.pos.x = width + this.r
    }
    if (this.pos.y > height + this.r)
    {
      this.pos.y = -this.r
    }
    else if (this.pos.y < -this.r)
    {
      this.pos.y = height + this.r
    }
  }
}
```

🗒️ Notes

This deletes the asteroid completely; we want two asteroids for the price of one.



Sketch A5.39 deleting the particle

❗ sketch.js

The problem is that the laser particle will still be there when the new two asteroids are created in that space. The challenge is to delete the laser particle that is there.

sketch.js

```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
  for (let i = lasers.length - 1; i >= 0; i--)
  {
```

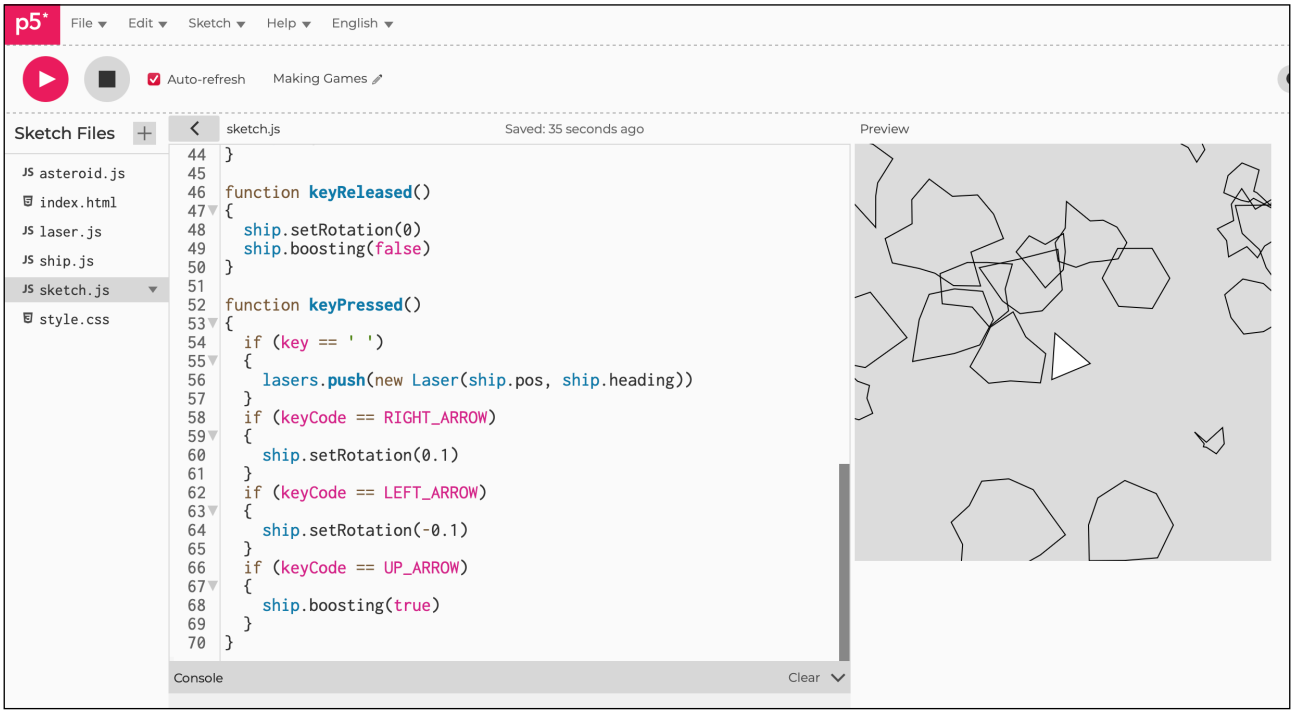
```
    lasers[i].show()
    lasers[i].move()
    for (let j = asteroids.length - 1; j >= 0; j--)
    {
      if (lasers[i].hits(asteroids[j]))
      {
        let newA = asteroids[j].breakup()
        asteroids = asteroids.concat(newA)
        asteroids.splice(j, 1)
        lasers.splice(i, 1)
        break
      }
    }
  }
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
```

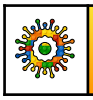
```
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

Notes

We split the asteroid into two, but we have a problem as you can see. We have lots and lots of asteroids; also, they generate their own size. We want to halve the size; also, we want to eventually destroy them when they get below a certain size.

Figure A5.39


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
sketch.js
Saved: 35 seconds ago
Preview
44 }
45
46 function keyReleased()
47 {
48 ship.setRotation(0)
49 ship.boosting(false)
50 }
51
52 function keyPressed()
53 {
54 if (key == ' ')
55 {
56 lasers.push(new Laser(ship.pos, ship.heading))
57 }
58 if (keyCode == RIGHT_ARROW)
59 {
60 ship.setRotation(0.1)
61 }
62 if (keyCode == LEFT_ARROW)
63 {
64 ship.setRotation(-0.1)
65 }
66 if (keyCode == UP_ARROW)
67 {
68 ship.boosting(true)
69 }
70 }
Console
Clear



Sketch A5.40 another asteroid

Adding a feature so that while the diameter r is more than 10, it will draw another asteroid; otherwise, not.

sketch.js

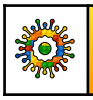
```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }
  for (let i = lasers.length - 1; i >= 0; i--)
  {
    lasers[i].show()
    lasers[i].move()
```

```
    for (let j = asteroids.length - 1; j >= 0; j--)
    {
      if (lasers[i].hits(asteroids[j]))
      {
        if (asteroids[j].r > 10)
        {
          let newA = asteroids[j].breakup()
          asteroids = asteroids.concat(newA)
        }
        asteroids.splice(j, 1)
        lasers.splice(i, 1)
        break
      }
    }
  }
  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
```

```
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

🗒️ Notes

Although this kind of works, we are still getting big asteroids.



Sketch A5.41 half pint asteroids

❗ asteroid.js

To make them half the size when hit. You are adding an argument `r` into the asteroid object.

asteroid.js

```
class Asteroid
{
  constructor(pos, r)
  {
    if (pos)
    {
      this.pos = pos.copy()
    }
    else
    {
      this.pos = createVector(random(width), random(height))
    }
    if (r)
    {
      this.r = r * 0.5
    }
    else
    {
      this.r = random(15, 50)
    }
    this.vel = p5.Vector.random2D()
    this.total = floor(random(5, 15))
    this.offset = []
    for (let i = 0; i < this.total; i++)
    {
      this.offset[i] = random(-this.r * 0.5, this.r * 0.5)
```

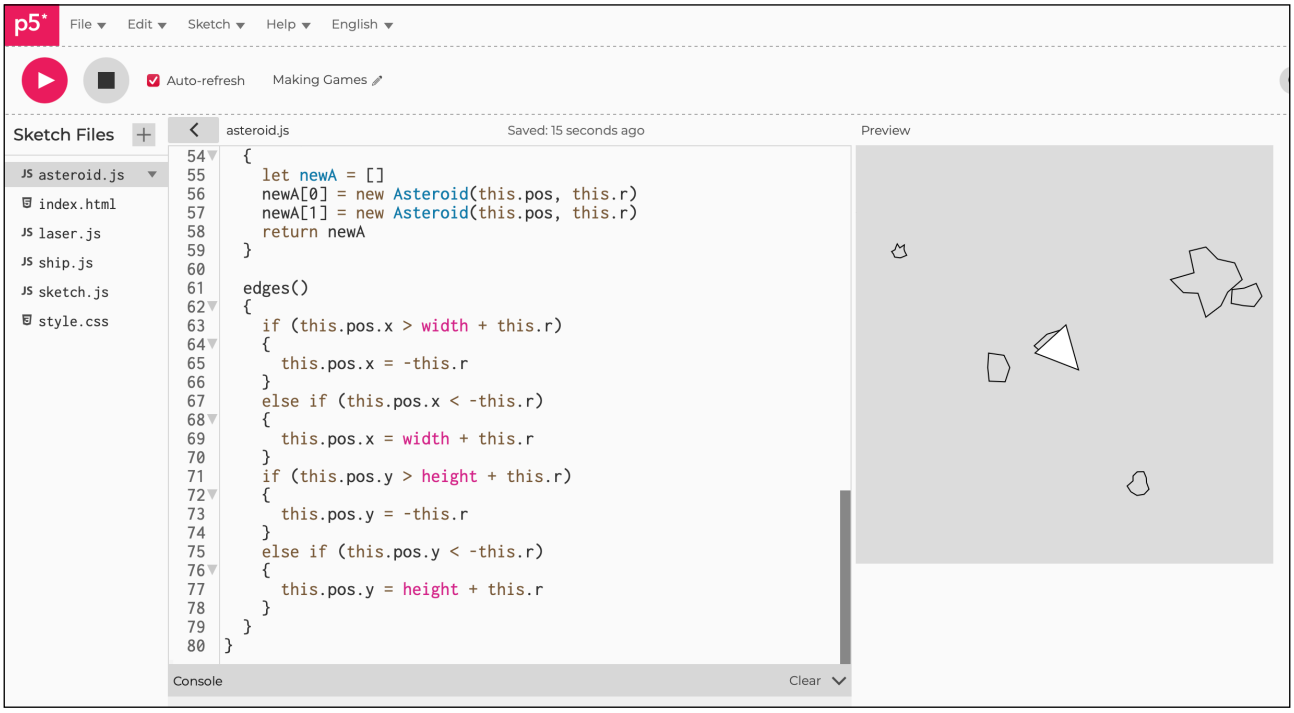
```
    }
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    noFill()
    translate(this.pos.x, this.pos.y)
    beginShape()
    for (let i = 0; i < this.total; i++)
    {
      let angle = map(i, 0, this.total, 0, TWO_PI)
      let r = this.r + this.offset[i]
      let x = r * cos(angle)
      let y = r * sin(angle)
      vertex(x, y)
    }
    endShape(CLOSE)
    pop()
  }

  breakup()
  {
    let newA = []
    newA[0] = new Asteroid(this.pos, this.r)
    newA[1] = new Asteroid(this.pos, this.r)
    return newA
  }
```

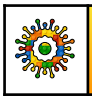
```
  edges()
  {
    if (this.pos.x > width + this.r)
    {
      this.pos.x = -this.r
    }
    else if (this.pos.x < -this.r)
    {
      this.pos.x = width + this.r
    }
    if (this.pos.y > height + this.r)
    {
      this.pos.y = -this.r
    }
    else if (this.pos.y < -this.r)
    {
      this.pos.y = height + this.r
    }
  }
}
```

Notes

That's better.

Figure A5.41


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
asteroid.js
Saved: 15 seconds ago
Preview
54 {
55 let newA = []
56 newA[0] = new Asteroid(this.pos, this.r)
57 newA[1] = new Asteroid(this.pos, this.r)
58 return newA
59 }
60
61 edges()
62 {
63 if (this.pos.x > width + this.r)
64 {
65 this.pos.x = -this.r
66 }
67 else if (this.pos.x < -this.r)
68 {
69 this.pos.x = width + this.r
70 }
71 if (this.pos.y > height + this.r)
72 {
73 this.pos.y = -this.r
74 }
75 else if (this.pos.y < -this.r)
76 {
77 this.pos.y = height + this.r
78 }
79 }
80 }
Console
Clear



Sketch A5.42 I've been hit response

❗ sketch.js

Final push to get a response when the asteroid hits the ship. Just use a console log to check if something is happening.

sketch.js

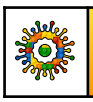
```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
    if (ship.hits(asteroids[i]))
    {
      console.log('oops')
    }
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
```

```
  }

  for (let i = lasers.length - 1; i >= 0; i--)
  {
    lasers[i].show()
    lasers[i].move()
    for (let j = asteroids.length - 1; j >= 0; j--)
    {
      if (lasers[i].hits(asteroids[j]))
      {
        if (asteroids[j].r > 10)
        {
        let newA = asteroids[j].breakup()
        asteroids = asteroids.concat(newA)
        }
        asteroids.splice(j, 1)
        lasers.splice(i, 1)
        break
      }
    }
  }

  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}
```

```
function keyPressed()
{
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```

Notes

We need the ship to tell us.



Sketch A5.43 watch out for the asteroid

❗ ship.js

And in the ship, we add a `hits()` function for when the asteroid hits the ship.

ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 20
    this.heading = 0
    this.rotation = 0
    this.vel = createVector(0, 0)
    this.isBoosting = false
  }

  boosting(b)
  {
    this.isBoosting = b
  }

  move()
  {
    if (this.isBoosting)
    {
      this.boost()
    }
    this.pos.add(this.vel)
    this.vel.mult(0.99)
  }
```

```
  boost()
  {
    let force = p5.Vector.fromAngle(this.heading)
    force.mult(0.1)
    this.vel.add(force)
  }

  hits(asteroid)
  {
    let d = dist(this.pos.x, this.pos.y, asteroid.pos.x,
asteroid.pos.y)
    if (d < this.r + asteroid.r)
    {
      return true
    }
    else
    {
      return false
    }
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    rotate(this.heading + PI/2)
    triangle(-this.r, this.r, this.r, this.r, 0, -this.r)
    pop()
  }

  edges()
  {
```

```
    if (this.pos.x > width + this.r)
    {
      this.pos.x = -this.r
    }
    else if (this.pos.x < -this.r)
    {
      this.pos.x = width + this.r
    }
    if (this.pos.y > height + this.r)
    {
      this.pos.y = -this.r
    }
    else if (this.pos.y < -this.r)
    {
      this.pos.y = height + this.r
    }
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
}
```

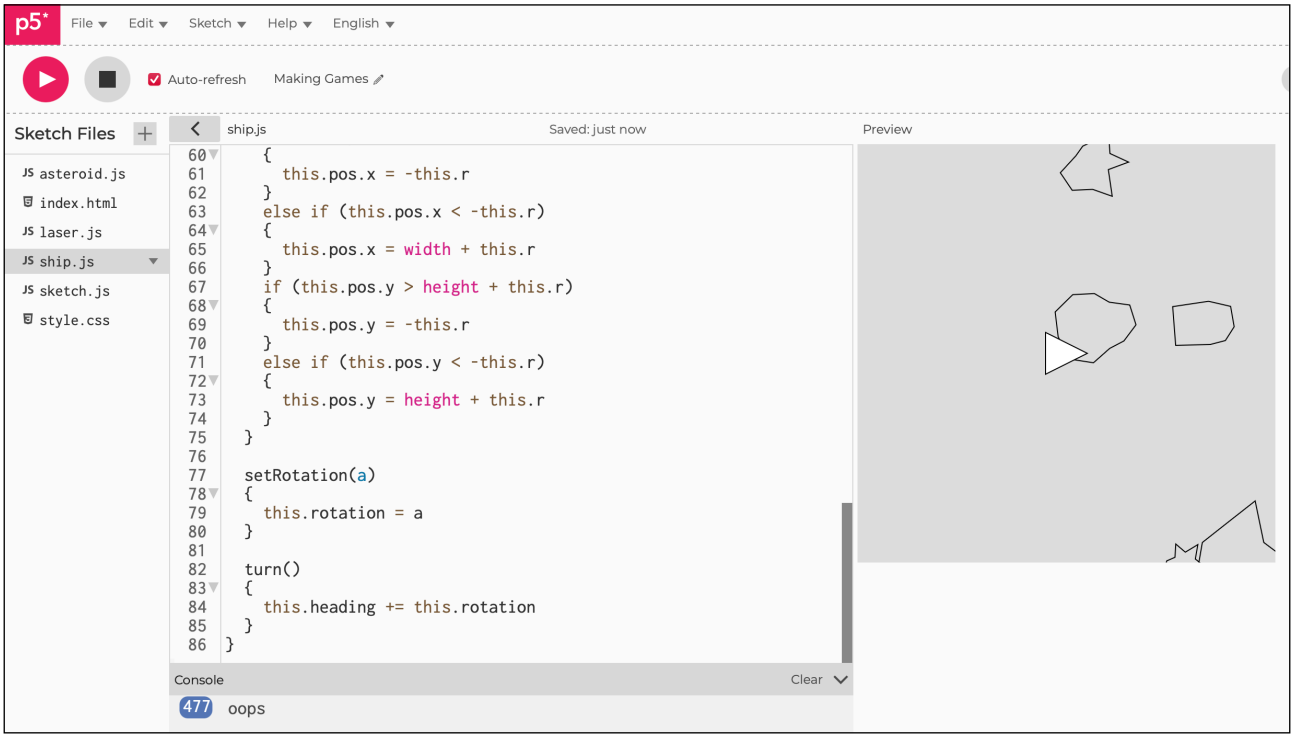
🗒️ Notes

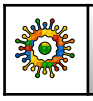
We get an oops every time an asteroid hits the ship. This is far from finished, but the essence of the game is here.

🌻 Challenges

1. Make a counter for how many asteroids have been destroyed.
2. Make a counter for how many times you have been hit.
3. Change colour when hit.

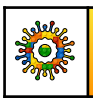
Figure A5.43


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
index.html
laser.js
ship.js
sketch.js
style.css
ship.js
Saved: just now
Preview
60 {
61 this.pos.x = -this.r
62 }
63 else if (this.pos.x < -this.r)
64 {
65 this.pos.x = width + this.r
66 }
67 if (this.pos.y > height + this.r)
68 {
69 this.pos.y = -this.r
70 }
71 else if (this.pos.y < -this.r)
72 {
73 this.pos.y = height + this.r
74 }
75 }
76
77 setRotation(a)
78 {
79 this.rotation = a
80 }
81
82 turn()
83 {
84 this.heading += this.rotation
85 }
86 }
Console
Clear
477
oops



Finishing touches

Elements to make the game more playable and more visually interesting include making a PNG of the ship rather than using the triangle.



Sketch A5.44 the laser array problem

❗ sketch.js

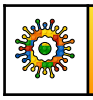
Although the game works fine, there is one problem outstanding. The laser array just gets bigger and bigger, which is never a good thing in a game. So now, to finish off, we will allow the laser particles to be deleted from the array. Also, `I have been hit` is in the message.

sketch.js

```
let ship
let asteroids = []
let lasers = []

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
    if (ship.hits(asteroids[i]))
    {
      console.log('I have been hit')
    }
    asteroids[i].show()
```

```
    asteroids[i].move()
    asteroids[i].edges()
  }

  for (let i = lasers.length - 1; i >= 0; i--)
  {
    lasers[i].show()
    lasers[i].move()
    if (lasers[i].offscreen())
    {
      lasers.splice(i, 1)
    }
    else
    {
      for (let j = asteroids.length - 1; j >= 0; j--)
      {
        if (lasers[i].hits(asteroids[j]))
        {
          if (asteroids[j].r > 10)
          {
            let newA = asteroids[j].breakup()
            asteroids = asteroids.concat(newA)
          }
          asteroids.splice(j, 1)
          lasers.splice(i, 1)
          break
        }
      }
    }
  }

  ship.show()
  ship.turn()
```

```
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
  {
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
}
```



Sketch A5.45 off the screen solution

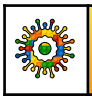
❗ laser.js

In the `laser.js` sketch, we add the `offscreen()` function.

laser.js

```
class Laser
{
  constructor(shipPosition, angle)
  {
    this.pos = createVector(shipPosition.x, shipPosition.y)
    this.vel = p5.Vector.fromAngle(angle)
    this.vel.mult(5)
  }

  move()
  {
    this.pos.add(this.vel)
  }

  show()
  {
    push()
    strokeWeight(5)
    point(this.pos.x, this.pos.y)
    pop()
  }

  hits(asteroid)
  {
    let d = dist(this.pos.x, this.pos.y, asteroid.pos.x,
asteroid.pos.y)
    if (d < asteroid.r)
```

```
    {
      return true
    }
    else
    {
      return false
    }
  }

  offscreen()
  {
    if (this.pos.x > width || this.pos.x < 0)
    {
      return true
    }
    if (this.pos.y > height || this.pos.y < 0)
    {
      return true
    }
    else
    {
      return false
    }
  }
}
```



Sketch A5.46 creating a ship png

❗ save your main sketch and start a fresh one for the PNG.

To make it more like a spaceship than a triangle. We will make a simple spaceship. Feel free to design your own.

ship PNG sketch

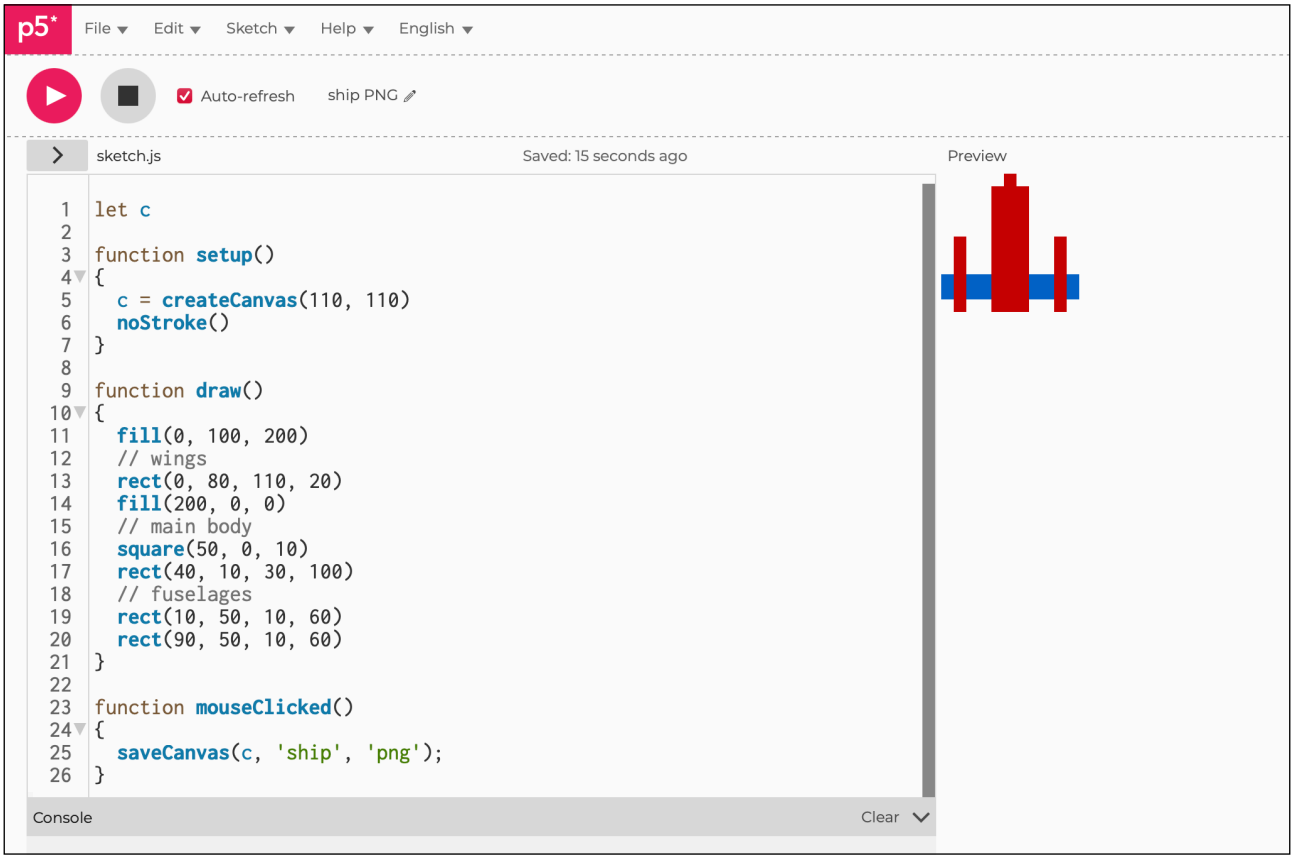
```
let c

function setup()
{
  c = createCanvas(110, 110)
  noStroke()
}

function draw()
{
  fill(0, 100, 200)
  // wings
  rect(0, 80, 110, 20)
  fill(200, 0, 0)
  // main body
  square(50, 0, 10)
  rect(40, 10, 30, 100)
  // fuselages
  rect(10, 50, 10, 60)
  rect(90, 50, 10, 60)
}

function mouseClicked()
{
  saveCanvas(c, 'ship', 'png');
```

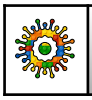
```
}
```

Notes

When you download the image, you may get a suffix number because of the Space Invaders game, where we created a ship for that game. We can rename it in the next sketch.

Figure 1: ship PNG





Adding the ship to the sketch

Exactly the same as before.

Step 1: Click on the sketch and add a folder.
Step 2: Give it the name `images` and add a folder.
Step 3: In the sketch files (left-hand side, you will see a folder called `images` and a drop-down arrow; click on that and choose `upload file`).
Step 4: Drag and drop the file into that box.
Step 5: Rename it (if necessary) to `ship.png` (from `ship-2.png`).

Figure 2: uploading image

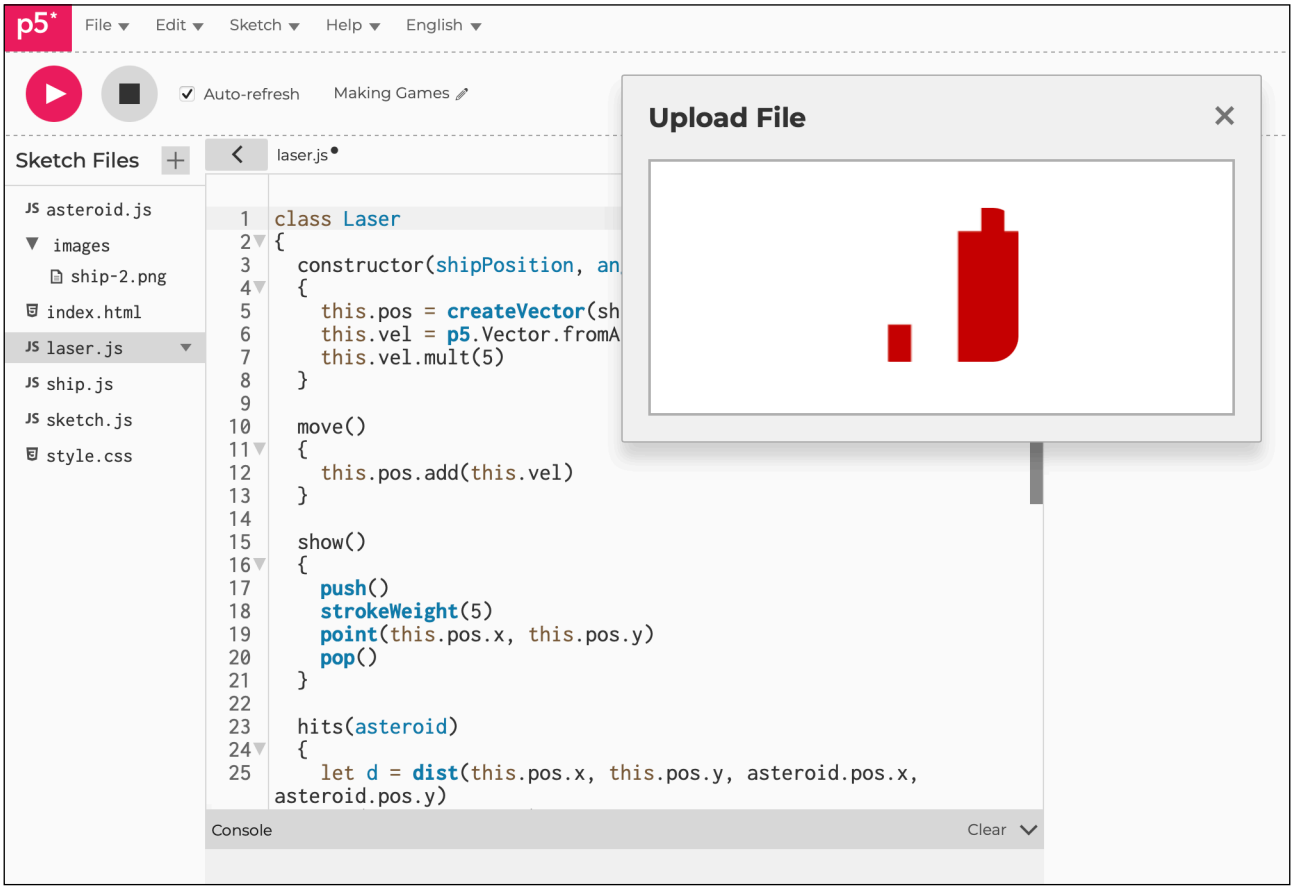
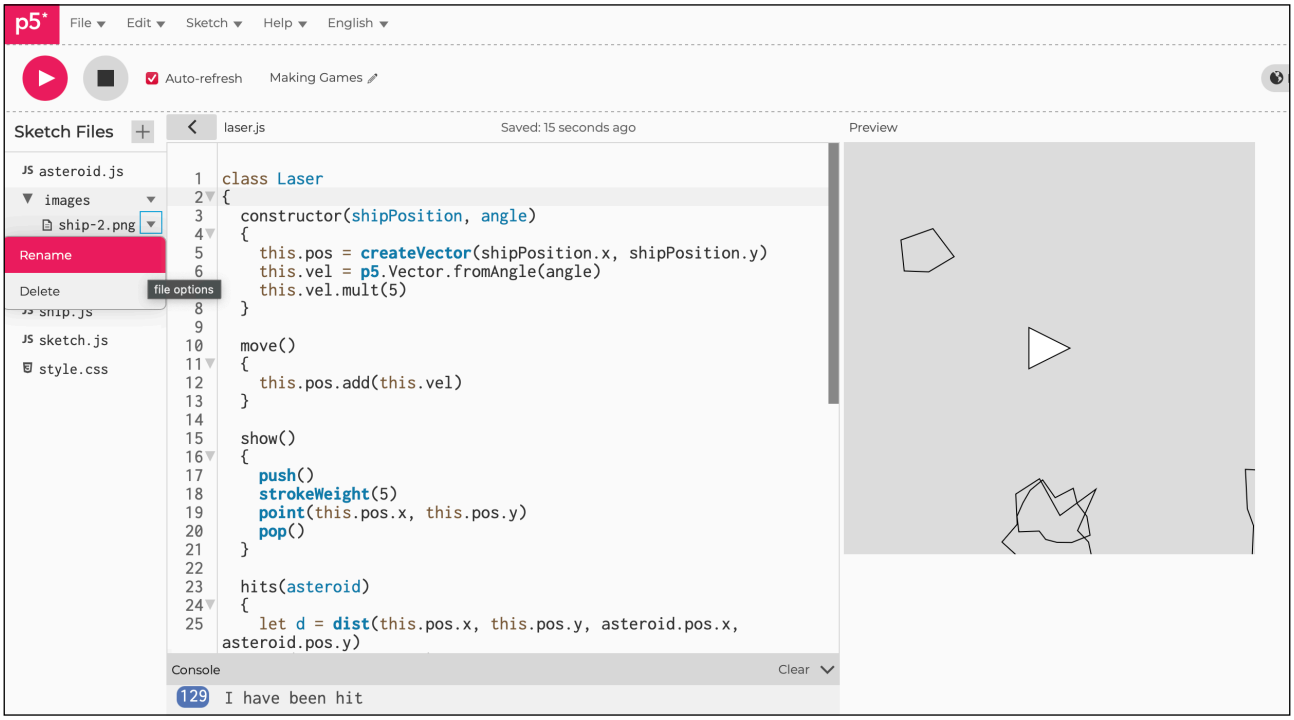
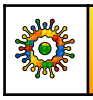


Figure 3: renaming





Sketch A5.47 uploading the ship png

❗ sketch.js

Inserting the image into the main `sketch.js`. It is preloaded as well so that it doesn't start playing the game before it has loaded. We will need to go to the `ship.js` next to replace the triangle.

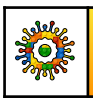
sketch.js

```
let ship
let asteroids = []
let lasers = []
let img

function preload()
{
  img = loadImage('images/ship.png')
}

function setup()
{
  createCanvas(400, 400)
  ship = new Ship()
  for (let i = 0; i < 5; i++)
  {
    asteroids.push(new Asteroid())
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < asteroids.length; i++)
  {
```

```
    if (ship.hits(asteroids[i]))
    {
      console.log('I have been hit')
    }
    asteroids[i].show()
    asteroids[i].move()
    asteroids[i].edges()
  }

  for (let i = lasers.length - 1; i >= 0; i--)
  {
    lasers[i].show()
    lasers[i].move()
    if (lasers[i].offscreen())
    {
      lasers.splice(i, 1)
    }
    else
    {
      for (let j = asteroids.length - 1; j >= 0; j--)
      {
        if (lasers[i].hits(asteroids[j]))
        {
          if (asteroids[j].r > 10)
          {
            let newA = asteroids[j].breakup()
            asteroids = asteroids.concat(newA)
          }
          asteroids.splice(j, 1)
          lasers.splice(i, 1)
          break
        }
      }
```

```
    }
  }

  ship.show()
  ship.turn()
  ship.move()
  ship.edges()
}

function keyReleased()
{
  ship.setRotation(0)
  ship.boosting(false)
}

function keyPressed()
  {
  if (key == ' ')
  {
    lasers.push(new Laser(ship.pos, ship.heading))
  }
  if (keyCode == RIGHT_ARROW)
  {
    ship.setRotation(0.1)
  }
  if (keyCode == LEFT_ARROW)
  {
    ship.setRotation(-0.1)
  }
  if (keyCode == UP_ARROW)
  {
    ship.boosting(true)
  }
```

```
}
```



Sketch A5.48 adding the ship PNG

❗ ship.js

And then in ship.js.

ship.js

```
class Ship
{
  constructor()
  {
    this.pos = createVector(width/2, height/2)
    this.r = 50
    this.heading = 0
    this.rotation = 0
    this.vel = createVector(0, 0)
    this.isBoosting = false
  }

  boosting(b)
  {
    this.isBoosting = b
  }

  move()
  {
    if (this.isBoosting)
    {
      this.boost()
    }
    this.pos.add(this.vel)
    this.vel.mult(0.99)
  }
```

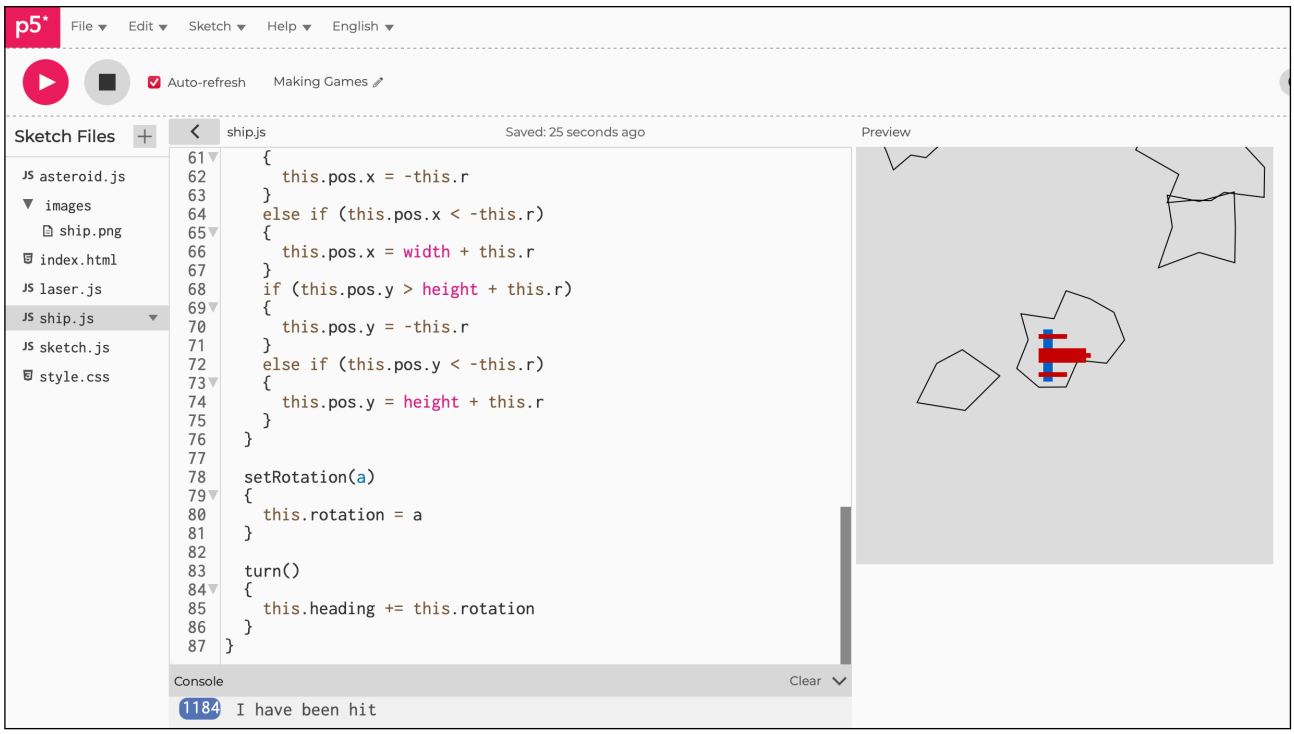
```
  boost()
  {
    let force = p5.Vector.fromAngle(this.heading)
    force.mult(0.1)
    this.vel.add(force)
  }

  hits(asteroid)
  {
    let d = dist(this.pos.x, this.pos.y, asteroid.pos.x,
asteroid.pos.y)
    if (d < this.r + asteroid.r)
    {
      return true
    }
    else
    {
      return false
    }
  }

  show()
  {
    push()
    translate(this.pos.x, this.pos.y)
    rotate(this.heading + PI/2)
    imageMode(CENTER)
    image(img, 0, 0, this.r, this.r)
    pop()
  }

  edges()
```

```
  {
    if (this.pos.x > width + this.r)
    {
      this.pos.x = -this.r
    }
    else if (this.pos.x < -this.r)
    {
      this.pos.x = width + this.r
    }
    if (this.pos.y > height + this.r)
    {
      this.pos.y = -this.r
    }
    else if (this.pos.y < -this.r)
    {
      this.pos.y = height + this.r
    }
  }

  setRotation(a)
  {
    this.rotation = a
  }

  turn()
  {
    this.heading += this.rotation
  }
}
```

🗒️ Notes

The results are seen below.

Figure A5.48


p5*
File
Edit
Sketch
Help
English
Auto-refresh
Making Games
Sketch Files
asteroid.js
images
ship.png
index.html
laser.js
ship.js
sketch.js
style.css
ship.js
Saved: 25 seconds ago
Preview
61 {
62 this.pos.x = -this.r
63 }
64 else if (this.pos.x < -this.r)
65 {
66 this.pos.x = width + this.r
67 }
68 if (this.pos.y > height + this.r)
69 {
70 this.pos.y = -this.r
71 }
72 else if (this.pos.y < -this.r)
73 {
74 this.pos.y = height + this.r
75 }
76 }
77
78 setRotation(a)
79 {
80 this.rotation = a
81 }
82
83 turn()
84 {
85 this.heading += this.rotation
86 }
87 }
Console
Clear
1184 I have been hit

What next?

That is a basic game which needs some more work. This unit will get very long if I added everything; I will leave that to you.

Suggestions:

1. Space background with twinkling stars.
2. Larger canvas (windowWidth and windowHeight).
3. Have a health bar (for when hit).
4. Have a `won/lost` screen.
5. Click to restart.
6. Add sounds?