

Through a square shape with a round face and a square shape with arms and legs, I am going to draw the character of the game and express it as a human.

The log where the letters will stand is drawn in a long rectangle.

The swords the characters will hold will overlap two rectangles in the shape of a cross, and then increase the pentagon to make it look like a medieval sword.

The letters will be distinguished by painting the whole body blue on one side, red on the other, brown under the log and blue under the log to make it look like the sea. The background, like sunset, will create an atmosphere in which orange and red colors are properly combined. The knife will use light gray to give the feeling of iron.

With knives and characters as variables, we will use the function set-up to call them up and use the function keypress to make the character swing or move the knife. And I will implement the song and background sound using function preload.

The stabbing ,response to enemy attacks, and the most important fall in logs are expressed in conditional statements.

I will express the waves and background sounds under the cloud log floating in the background through the loop.

I will make two characters and knives using the class to make the same characteristics using the characters and knives.

I will use vectors to make gravity to the bottom so that it can fall from the log.

Ex)

```
let sword;
```

```
let human;
```

```
function preload() {
```

```
  bg = loadSound('song.mp3')
```

```
}
```

```
function setup() {
```

```
  createCanvas(400, 400)
```

```
  bg.loop()
```

```
  sword = new Sword()
```

```
  human = new Human()
```

```
  bottom = new particle()
```

```
}
```

```
function draw() {
```

```
  background('orange');
```

```
  push()
```

```
  //sun
```

```
  circle(~,~,~)
```

```
  fill('red')
```

```
  human.gravitateTo(bottom)
```

```
pop()
```

```
}
```

```
}
```

```
class Sword{
```

```
constructor(){
```

```
    this.~
```

```
    this.~~
```

```
    ~~
```

```
}
```

```
draw(){
```

```
    push();
```

```
    fill('gray')
```

```
    rect(~)
```

```
    rect(~)
```

```
    pop();
```

```
}
```

```
update(){
```

```
    if(~~){
```

```
        if(~){
```

```
            ~
```

```
        }
```

```
    }else if(keyIsDown(RIGHT_ARROW)){
```

```
        if(~){
```

```
            ~
```

```
        }
```

```
    }
```

```
}
```

```
}
```

```
class Human{
```

```
    constructor(){
```

```
        this.~~
```

```
        this.~`
```

```

    }
    draw(){
        push();
        fill('red')
    circle(~~)
    rect(~~)
        pop();
    }
    update(){
        if(~~){
            if(~~){
                ~~}
            }else if(~~){
                if(~~){
                    ~~~
                }
            }
        }
    }
}

```

```

setAngle(angle_in_radians)

```

```

{
    var length = this.getLength()

    this.x = cos(angle_in_radians)*length

    this.y = sin(angle_in_radians)*length
}

```

```

getLength(){

    return sqrt((this.x)*(this.x)+(this.y)*(this.y))
}

```

```
}
```

```
setLength(length)
```

```
{
```

```
    var angle_in_radians = this.getAngle()
```

```
    this.x = cos(angle_in_radians)*length
```

```
    this.y = sin(angle_in_radians)*length
```

```
}
```

```
multiply(scalar) {
```

```
    let v3 = new Vec2(this.x*scalar, this.y*scalar)
```

```
    return v3
```

```
}
```

```
multiplyBy(scalar) {
```

```
    this.x = this.x*scalar
```

```
    this.y = this.y*scalar
```

```
}
```

```
add(v2)
```

```
{
```

```
    let v3 = new Vec2(this.x+v2.x, this.y+v2.y)
```

```
    return v3
```

```
}
```

