

```
1 package juego;
2
3 import java.awt.Color;
4
5
6
7
8
9 public class Mono {
10     private double x;
11     private double y;
12     private int ancho;
13     private int largo;
14     private int angulo;
15     private Color color;
16     private Image imagen;
17     private int timer;
18     private int salto;
19     private int disparosDisp;
20
21
22     // prueba
23
24     public Mono(int x, int y) {
25         this.x = x + this.ancho + 80;
26         this.y = y;
27         this.ancho = 50;
28         this.largo = 60;
29         this.angulo = 0;
30         this.color = Color.white;
31         this.imagen = Herramientas.cargarImagen("mono.gif");
32         this.timer = 0;
33         this.salto = 0;
34         this.disparosDisp = 3;
35
36
37     }
38
39     public int getDisparosDisp() {
40         return disparosDisp;
41     }
42
43     public void setDisparosDisp(int disparosDisp) {
44         this.disparosDisp = disparosDisp;
45     }
46
47     public void aumentarSalto() {
48         salto++;
49     }
50
51     public void aumentarTimer() {
52         timer++;
53     }
54
55     public boolean contadorSalto() {
56         //si timer es menor a 30 y salto es menor a 2 devuelve true
57         if (timer < 30 && salto < 2) {
58             return true;
59         }
60         return false;
61     }
62
63     public boolean chocaConSerpiente(Serpiente s) {
64         // si el mono choca con serpiente devuelve true
65         if (x + ancho / 2 > s.getX() - s.getAncho() / 2 && x - ancho / 2 < s.getX() + s.getAncho() / 2
66             && s.getY() - s.getAlto() / 2 < y + largo / 2 && y - largo / 2 < s.getY() + s.getAlto() / 2) {
67             return true;
68         }
69         return false;
70     }
71     // si el mono choca con aguilas devuelve true
72     public boolean chocaConAguila(Aguila a) {
73         if (x + ancho / 2 > a.getX() - a.getAncho() / 2 && x - ancho / 2 < a.getX() + a.getAncho() / 2
74             && a.getY() - a.getAlto() / 2 < y + largo / 2 && y - largo / 2 < a.getY() + a.getAlto() / 2) {
75             return true;
76         }
77         return false;
78     }
79     // si el mono choca con tigre devuelve true
80     public boolean chocaConTigre(Tigre t) {
81         if (x + ancho / 2 > t.getX() - t.getAncho() / 2 && x - ancho / 2 < t.getX() + t.getAncho() / 2
```

```
82         && t.getY() - t.getAlto() / 2 < y + largo / 2 && y - largo / 2 < t.getY() + t.getAlto() / 2) {
83             return true;
84         }
85         return false;
86     }
87
88     public boolean chocaConArbol(Arbol a) {
89         // si el mono choca con arbol devuelve true
90         if (x + ancho / 2 > a.getX() - a.getAncho() / 2 && x - ancho / 2 < a.getX() + a.getAncho() / 2
91             && a.getY() - a.getAlto() / 2 < y + largo / 2 && y + largo / 2 < a.getY() + a.getAlto() / 2) {
92             return true;
93         }
94         return false;
95     }
96
97     public boolean chocaConBanana(Items b) {
98         // si el mono choca con banana devuelve true
99         if (x + ancho / 2 > b.getBx() - b.getAncho() / 2 && x - ancho / 2 < b.getBx() + b.getAncho() / 2
100            && b.getBy() - b.getAlto() / 2 < y + largo / 2 && y - largo / 2 < b.getBy() + b.getAlto() / 2) {
101             return true;
102         }
103         return false;
104     }
105
106     public boolean chocaConPiedra(Items p) {
107         // si el mono choca con piedra devuelve true
108         if (x + ancho / 2 > p.getX() - p.getAncho() / 2 && x - ancho / 2 < p.getX() + p.getAncho() / 2
109            && p.getY() - p.getAlto() / 2 < y + largo / 2 && y - largo / 2 < p.getY() + p.getAlto() / 2) {
110             return true;
111         }
112         return false;
113     }
114
115
116     public void dibujarMono(Entorno e) {
117         // e.dibujarRectangulo(this.x, this.y, this.ancho, this.largo, 0, this.color);
118         e.dibujarImagen(imagen, x, y - 15, angulo, .3);
119     }
120
121     public void saltar(int s) {
122         this.angulo += 1;
123         this.y -= s;
124     } // saltar
125
126     public void gravedad() {
127         this.angulo = 0;
128         this.y += 5;
129     }
130
131     public boolean chocaConSuelo() {
132         // si la instancia y del mono es mayor a 500 devuelve true
133         if (this.y > 500) {
134             return true;
135         } else {
136             return false;
137         }
138     } // chocaConSuelo
139
140     public boolean chocaConArboles(Arbol[] a) {
141         for (int i = 0; i < a.length; i++) {
142             // si choca con con arbol es true devuelve true
143             if (chocaConArbol(a[i])) {
144                 return true;
145             }
146         }
147         return false;
148     }
149
150     public double getX() {
151         return x;
152     }
153
154     public void setX(int x) {
155         this.x = x;
156     }
157
158     public double getY() {
```

```
159         return y;
160     }
161
162     public void setY(int y) {
163         this.y = y;
164     }
165
166     public int getTimer() {
167         return timer;
168     }
169
170     public void setTimer(int timer) {
171         this.timer = timer;
172     }
173
174     public int getSalto() {
175         return salto;
176     }
177
178     public void setSalto(int salto) {
179         this.salto = salto;
180     }
181
182     public Piedra lanzarPiedra() {
183         this.disparosDisp--;
184         return new Piedra(x, y);
185     }
186 }
187
188
189 } // class Mono
```