

Barril.java

```
1 package juego;
2
3 import java.awt.Color;
9
10 public class Barril {
11
12     private double posX;
13     private double posY;
14     private int diametro;
15     private double escala;
16
17     private Image spin_izquierda;
18     private Image spin_derecha;
19     private String ultima;
20     private boolean saltado;
21
22     public Barril(Viga vigasuelo) {
23
24         // this.estado = "vivo";
25         this.diametro = 17;
26         this.escala = (double) this.diametro / 108;
27
28         // this.posy = 48;
29
30         if (vigasuelo.getPos() == 6) {
31             this.posy = (int) vigasuelo.dondeEmpiezaElSuelo() - 20;
32             this.ultima = "derecha";
33             this.posx = 120;
34         } else if (vigasuelo.getPos() == 4) {
35             this.posy = (int) vigasuelo.dondeEmpiezaElSuelo() - 190;
36             this.ultima = "izquierda";
37             this.posx = 120;
38         }
39
40         this.spin_izquierda =
41             Herramientas.cargarImagen("rsc/graficos/barriles/spin-izquierda.gif");
42         this.spin_derecha =
43             Herramientas.cargarImagen("rsc/graficos/barriles/spin-derecha.gif");
44
45         this.saltado = false;
46     }
47
48     public boolean deboDestruirme(Entorno entorno, Viga[] suelos) {
49         if (this.posx < 15 && this.pisando(entorno, suelos) == 0) {
50             return true;
51         } else {
52             return false;
53         }
54     }
55
56     public void dibujar(Entorno entorno, int contador, Viga[] suelos) {
57
58         // Image barril =
59         // Herramientas.cargarImagen("rsc/graficos/barriles/cayendo.png");
60         // entorno.dibujarCirculo(posx, posy, diametro, Color.blue);
61
62         // Si está rodando sobre el suelo
63         if (pisando(entorno, suelos) != -1) {
64
65             // En vigas con indice par desplazar a izquierda
66             if (this.posx >= 10 && pisando(entorno, suelos) % 2 == 0) {
```

Barril.java

```
66         this.posx = this.posx - 1.7;
67         entorno.dibujarImagen(spin_izquierda, this.posx, this.posy, 0,
        this.escala);
68         this.ultima = "izquierda";
69     }
70
71     // En vigas con indice impar desplazar a derecha
72     else if (this.posx <= 800 && pisando(entorno, suelos) % 2 == 1) {
73         this.posx = this.posx + 1.7;
74         entorno.dibujarImagen(spin_derecha, this.posx, this.posy, 0, this.escala);
75         this.ultima = "derecha";
76     }
77
78 }
79
80 // Si NO está rodando sobre el suelo
81 if (pisando(entorno, suelos) == -1) {
82
83     // cambia la posición con respecto al eje "y" hacia abajo
84     this.posy += 1;
85
86     // Si venia desplazandose a derecha pero está cayendo y hay espacio en el x,
87     // se sigue desplazando a derecha
88     if (this.posx <= 800 && this.ultima.equals("derecha")) {
89         this.posx = this.posx + 1.7;
90         entorno.dibujarImagen(spin_derecha, this.posx, this.posy, 0, this.escala);
91     }
92
93     // De lo contrario hay que indicarle que en el próximo tick se desplace a
94     // izquierda.
95     else {
96         this.ultima = "izquierda";
97     }
98
99     // Si venia desplazandose a izquierda pero está cayendo y hay espacio en el x,
100    // se sigue desplazando a izquierda
101    if (this.posx >= 10 && this.ultima.equals("izquierda")) {
102        this.posx = this.posx - 1.7;
103        entorno.dibujarImagen(spin_izquierda, this.posx, this.posy, 0,
        this.escala);
104    } else {
105        // De lo contrario hay que indicarle que en el próximo tick se desplace a
106        // derecha.
107        this.ultima = "derecha";
108    }
109
110 }
111
112 }
113
114 // Igual que pisando de Personaje
115 public int pisando(Entorno entorno, Viga[] suelos) {
116
117     for (int i = 0; i < suelos.length; i++) {
118
119         if (this.pies() == (int) suelos[i].dondeEmpiezaElSuelo()) {
120
121             if (this.lateralDerecho() < suelos[i].extremoIzquierdo()
122                 || this.lateralIzquierdo() > suelos[i].extremoDerecho()) {
123
124                 return -1;
125             }
126         }
127     }
128 }
```

Barril.java

```
126         } else {
127
128             return i;
129
130         }
131     }
132
133 }
134
135     return -1;
136
137 }
138
139 public int pies() {
140     return (int) this.posy + diametro / 2 - 2;
141 }
142
143 public int superior() {
144     return (int) this.posy - diametro / 2 + 2;
145 }
146
147 public int lateralDerecho() {
148     return (int) this.posx + diametro / 2;
149 }
150
151 public int lateralIzquierdo() {
152     return (int) this.posx - diametro / 2;
153 }
154
155 public int centroX() {
156     return (int) this.posx;
157 }
158
159 public void saltado() {
160     this.saltado = true;
161 }
162
163 public boolean fueSaltado() {
164     return this.saltado;
165 }
166
167 }
```