

Manual Técnico

Para los mensajes emergentes utilice el JOptionPane, así sea de advertencia o información.

```
JOptionPane.showMessageDialog(this, "Has perdido","Advertencia", JOptionPane.WARNING_MESSAGE);
```

Para pintar el fondo, la comida y la serpiente utilice este código:

```
@Override
public void paint(Graphics pintor){

    super.paint(pintor);
    pintor.setColor(colorserp);
    for (int i = 0; i < snake.size(); i++) {
        pintor.fillRect(res/2+snake.get(i)[0]*tam, res/2+snake.get(i)[1]*tam, tam-1, tam-1);
    }

    pintor.setColor(colorcomida);
    for (int i = 0; i < snake.size(); i++) {
        pintor.fillRect(res/2+comida[0]*tam, res/2+comida[1]*tam, tam-1, tam-1);
    }

}
```

Para que la serpiente se moviera y cambiara de direccion utilice este código:

```
int[] ultimo = snake.get(snake.size()-1);
int x=0;
int y=0;
boolean existe = false;
switch(direccion){
    case "r":
        x=1;
        break;
    case "l":
        x=-1;
        break;
    case "u":
        y=-1;
        break;
    case "d":
        y=1;
        break;
}
```

```

@Override
public void run() {

    while(this.bandera){
        panel.avanzar();
        panel.repaint();

        String cnt = Integer.toString(panel.count);
        vt.lbl_tamaño.setText(cnt);

        String vel = Integer.toString(velocidad);
        vt.lbl_velocidad.setText(vel);
        try {
            Thread.sleep(this.velocidad);
        } catch (InterruptedException e) {
            Logger.getLogger(Caminante.class.getName()).log(Level.SEVERE, null, e);
        }
    }

}

```

Para detectar si comía una fruta, y que al comerla aumentara la velocidad, o chocaba con una pared o el mismo utilice este código:

```

for (int i = 0; i < snake.size(); i++) {
    if(nuevo[0]==snake.get(i)[0]&&nuevo[1]==snake.get(i)[1]){
        existe=true;
        break;
    }
}

if(existe){
    df.t.stop();
    camino.parar();
    JOptionPane.showMessageDialog(this, "Has perdido", "Advertencia", JOptionPane.WARNING_MESSAGE);
    ReporteHTML html = new ReporteHTML();
    html.HTML();
}else{
    if(nuevo[0]==comida[0]&&nuevo[1]==comida[1]){
        snake.add(nuevo);
        count++;
        switch (df.dificultad) {
            case 0:
                camino.velocidad=(int) (camino.velocidad*(1-0.03));
                break;
            case 1:
                camino.velocidad=(int) (camino.velocidad*(1-0.06));
                break;
            case 2:
                camino.velocidad=(int) (camino.velocidad*(1-0.09));
                break;
        }

        if(snake.size()>25){
            df.t.stop();
            camino.parar();
            JOptionPane.showMessageDialog(this, "La serpiente ha alcanzado su tamaño máximo", "Informacion", JOptionPane.INFORMATION_MESSAGE);
            ReporteHTML html = new ReporteHTML();
            html.HTML();
        }
    }

    comida();
}else{
    snake.add(nuevo);
    snake.remove(0);
}
}

```

Para generar la comida utilice este código:

```
public void comida() {  
  
    boolean existe = false;  
    int a = (int) (Math.random() * can);  
    int b = (int) (Math.random() * can);  
    for (int i = 0; i < snake.size(); i++) {  
        if (snake.get(i)[0] == a && snake.get(i)[1] == b) {  
            existe = true;  
            comida();  
            break;  
        }  
    }  
  
    if (!existe) {  
        this.comida[0] = a;  
        this.comida[1] = b;  
    }  
  
}
```

Para crear el reporte en html utilice este código:

```

String cadenaHTML="<!DOCTYPE html>\n"
    + "<html lang=\"en\">\n"
    + "<head>\n"
    + "    <meta charset=\"utf-8\">\n"
    + "    <title>Reporte</title>\n"
    + "</head>\n"
    + "<table>\n"
    + "    <thead>\n"
    + "        <tr>\n"
    + "            <th>Angel Arreaga</th>\n"
    + "            <th>202004762</th>\n"
    + "        </tr>\n"
    + "    </thead>\n"
    + "    <tbody>\n"
    + "        <tr>\n"
    + "            <td>Dificultad</td>\n"
    + "            <td>"+ vt.dificultad2 + "</td>\n"
    + "        </tr>\n"
    + "        <tr>\n"
    + "            <td>Tiempo transcurrido</td>\n"
    + "            <td>"+ vt.m + " : " + vt.s + "</td>\n"
    + "        </tr>\n"
    + "        <tr>\n"
    + "            <td>Intervalo</td>\n"
    + "            <td>"+ vl.velocidad + "</td>\n"
    + "        </tr>\n"
    + "        <tr>\n"
    + "            <td>Tamaño de Serpiente</td>\n"
    + "            <td>"+ cnt.count + "</td>\n"
    + "        </tr>\n"
    + "        <tr>\n"
    + "            <td>Historial de movimiento</td>\n"
    + "            <td>"+ vt.historial + "</td>\n"
    + "        </tr>\n"
    ;

```

```

cadenaHTML+="</tbody>\n"
    + "</table>\n"
    + "</html>\n";

```

```

File archivo = new File("Reporte.html");
try{
    BufferedWriter bw = new BufferedWriter(new FileWriter(archivo));
    bw.write(cadenaHTML);
    bw.close();
}catch(Exception e){

```