

Integrantes del equipo:

Nombre: Reyno Sofía

Legajo: 44.173.232

Email: [sofireynoo30@gmail.com](mailto:sofireynoo30@gmail.com)

Nombre: Bravo Joaquín

Legajo: 46.698.653

Email: [joaquinnazabravo@gmail.com](mailto:joaquinnazabravo@gmail.com)

Nombre: Fernandez Jeremias

Legajo: 45.622.641

Email: [jerefernandez114@gmail.com](mailto:jerefernandez114@gmail.com)

Grupo：12

**Índice**

1. **BIENVENIDOS AL JUEGO**

1.1 *Presentación*

1.2 *Modo de juego*

1.3 *Reglas del juego*

1. **CLASES**

2.1 *Clase Juego*  
 2.2 *Clase Personaje*  
 2.3 *Clase Tortuga* 2.4 *Clase Gnomo*

2.5 *Clase Isla* 2.6 *Clase Poder* 2.7 *Clase Fondo*

2.8 *Clase Casa*

1. **PROBLEMAS Y SOLUCIONES**
2. **CONCLUSIÓN**

**Introducción**

1. **BIENVENIDOS AL JUEGO**
   1. **Presentación**

El juego "Al rescate de los gnomos" es una experiencia interactiva donde el jugador controla a Totoro, quien debe rescatar a los gnomos mientras evita los peligros, especialmente a las tortugas, que pueden matarlo. El objetivo principal es rescatar a todos los gnomos antes de que Totoro muera o se pierdan demasiados gnomos.

**1.2 Modo de juego**

El jugador puede mover a Totoro hacia los lados, hacer que salte y lanzar una bola para atacar a las tortugas. Cada gnomo rescatado otorga puntos y se debe evitar perder gnomos o permitir que Totoro caiga al vacío o sea alcanzado por las tortugas..

**1.3 Reglas del Juego**

El jugador utiliza las teclas de dirección o WAD para mover a Totoro y el botón izquierdo del mouse para lanzar una bola (si no hay otra en pantalla). Gana cuando rescata a todos los gnomos. Y pierde si Totoro cae, si se pierden tres gnomos o si Totoro es alcanzado por una tortuga.

1. **CLASES**

Aquí se presenta una explicación general de cada clase implementada, incluyendo las variables de instancia y una breve descripción de los métodos principales.

**2.1** **Clase Juego**

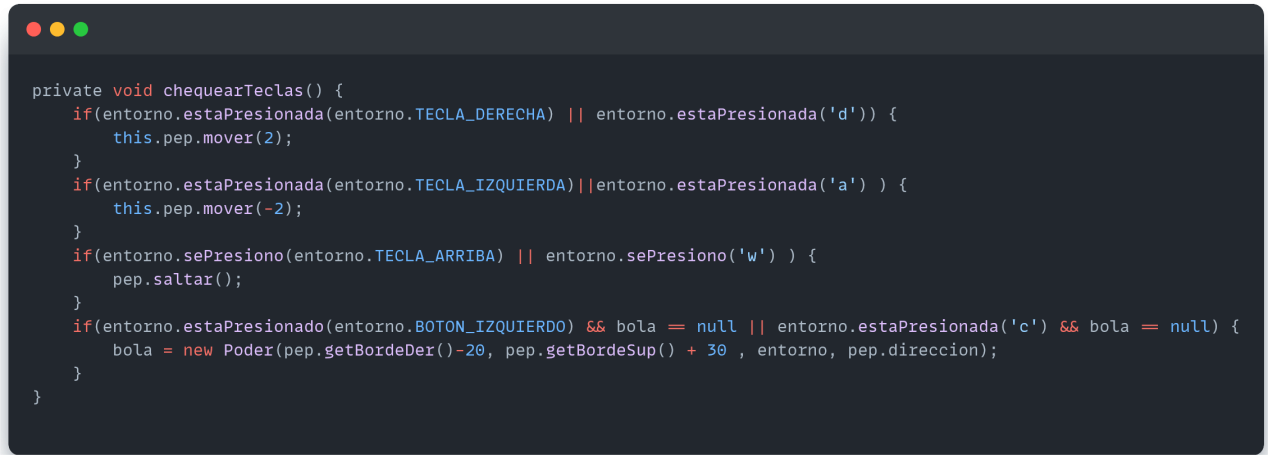
**Variables de instancia:**



En la ***clase Juego***, las variables de instancia incluyen el objeto *entorno*, que controla el entorno gráfico, y el *objeto pep*, una instancia de ***Personaje*** que representa al jugador. También están *islas* y *tortuga*, que son arrays de objetos ***Isla***y***Tortuga***, respectivamente, representando las plataformas y los enemigos. La variable *bola* es de tipo ***Poder*** y representa el proyectil que puede lanzar *Pep*. Los elementos visuales *fondo* y *casita* son instancias de ***Fondo*** y ***Casa***, y se utilizan para añadir detalles al entorno. Además, *gnomos* es un array de ***Gnomos*** que representan a los personajes que *Pep* debe rescatar. Los contadores *gnomosRescatados*, *gnomosPerdidos*, y *tortugasEliminadas* llevan un registro del progreso del jugador, mientras que *finJuego* es un booleano que indica si el juego ha terminado. Por último, *mensajeFinal* contiene el mensaje que se muestra al finalizar el juego.

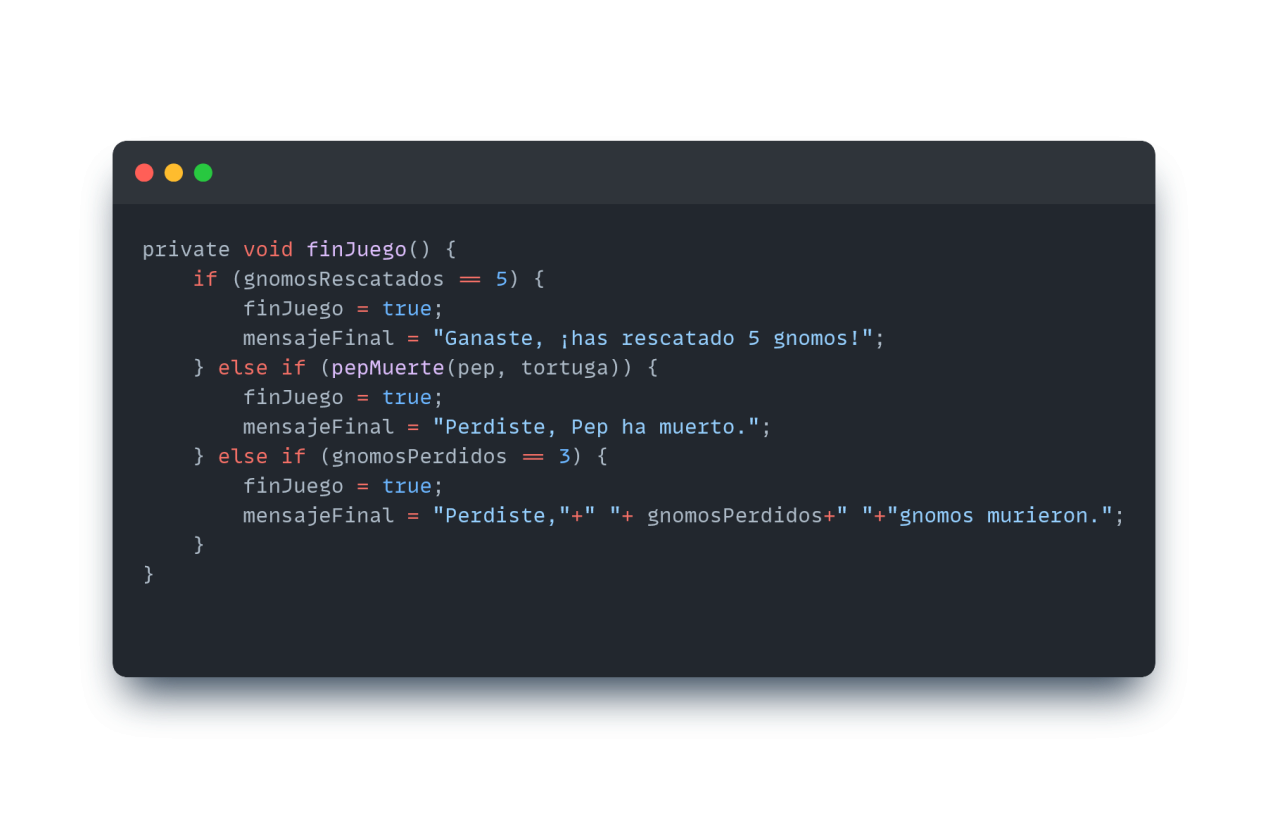
**Métodos principales:**

* **Tick():** Tick se encarga de actualizar el estado del juego en cada ciclo, revisando interacciones entre personajes y elementos. En cada ciclo, se procesan los movimientos de Totoro, el lanzamiento de la bola, y las colisiones.
* **chequearTeclas():**



Controla los movimientos de Totoro y permite al jugador lanzar una bola.

* **finJuego():**



Determina si el juego ha terminado revisando las condiciones de victoria o derrota. Si Totoro muere o se pierden suficientes gnomos, el juego se termina.

* **iniciarJuego():**

Reinicia el estado del juego, reposicionando los elementos en pantalla e inicializando los valores de las variables principales.

* **dibujarEstadisticas():**



Esta función muestra en la pantalla cosas como el tiempo de juego, gnomos rescatados y perdidos, y tortugas eliminadas.

* **pepMuerte():**

Verifica si Totoro ha colisionado con una tortuga, lo que resultaría en la muerte de Totoro y la finalización del juego.

* **verificarGnomosPisandoIsla():**

Comprueba si los gnomos están pisando una isla, permitiendo modificar su comportamiento y evitando que caigan.

**2.2** **Clase Personaje**

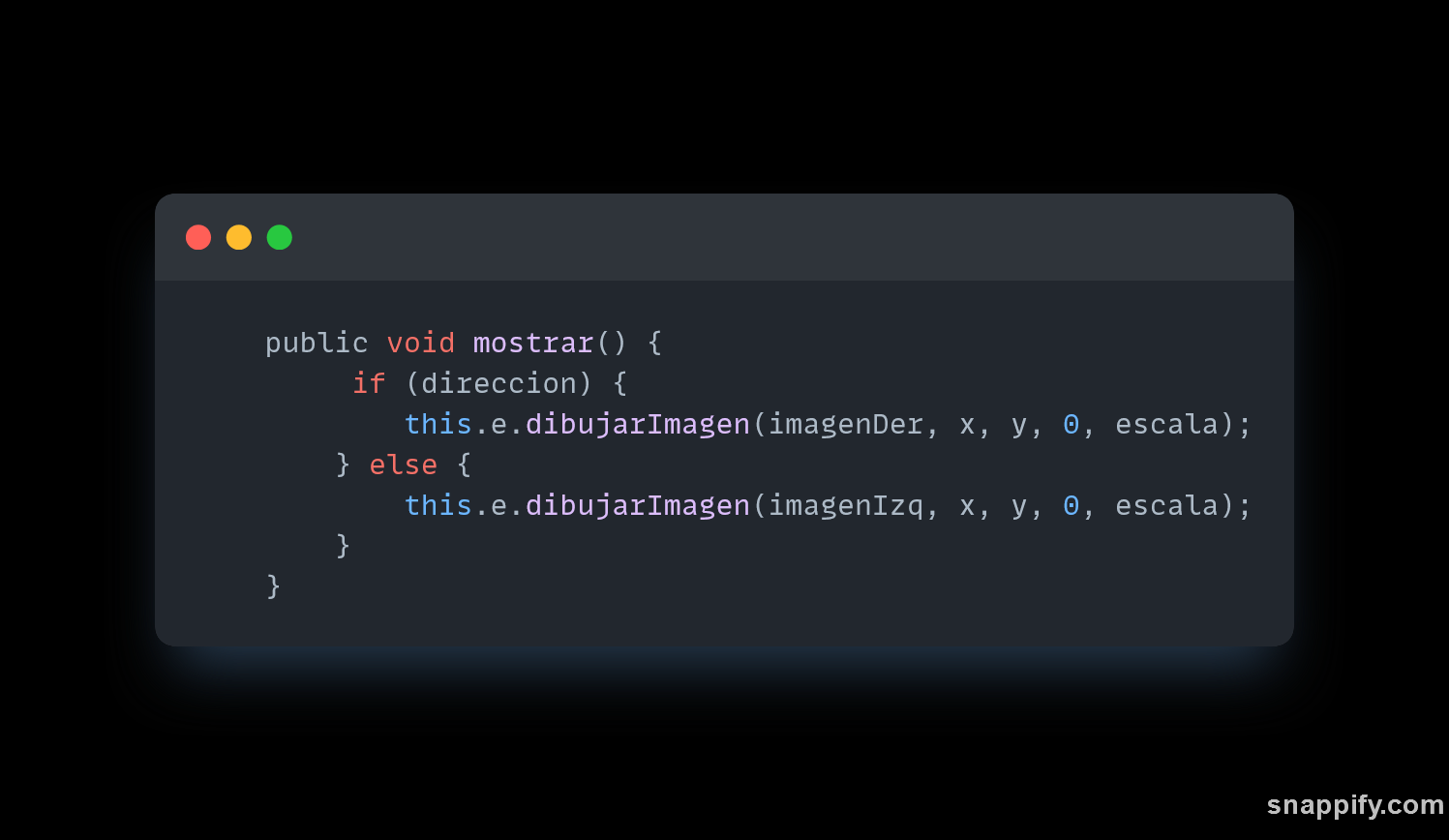
**Variables de instancia:**



La ***clase Personaje*** incluye las variables *x* y *y* para las coordenadas de *Pep*, y *ancho* y *alto* para su tamaño. La variable *direccion* indica si *Pep* mira hacia la derecha o la izquierda, y se usan las imágenes *imagenDer*, *imagenIzq* y *imagenHerido* para representar los diferentes estados visuales del personaje. *escala* y *velocidad* controlan el tamaño y velocidad de movimiento, mientras que *estaApoyado* y *estaSaltando* indican si *Pep* está en el suelo o en el aire. También está la variable *herido*, que señala si *Pep* está herido. Además, *contadorSalto* gestiona la duración de los saltos, y *posicionInicialX* y *posicionInicialY* guardan las posiciones iniciales de *Pep*.

**Métodos principales:**

* **mostrar():**



Se encarga de dibujar a Pep en la pantalla.

* **movVertical():**



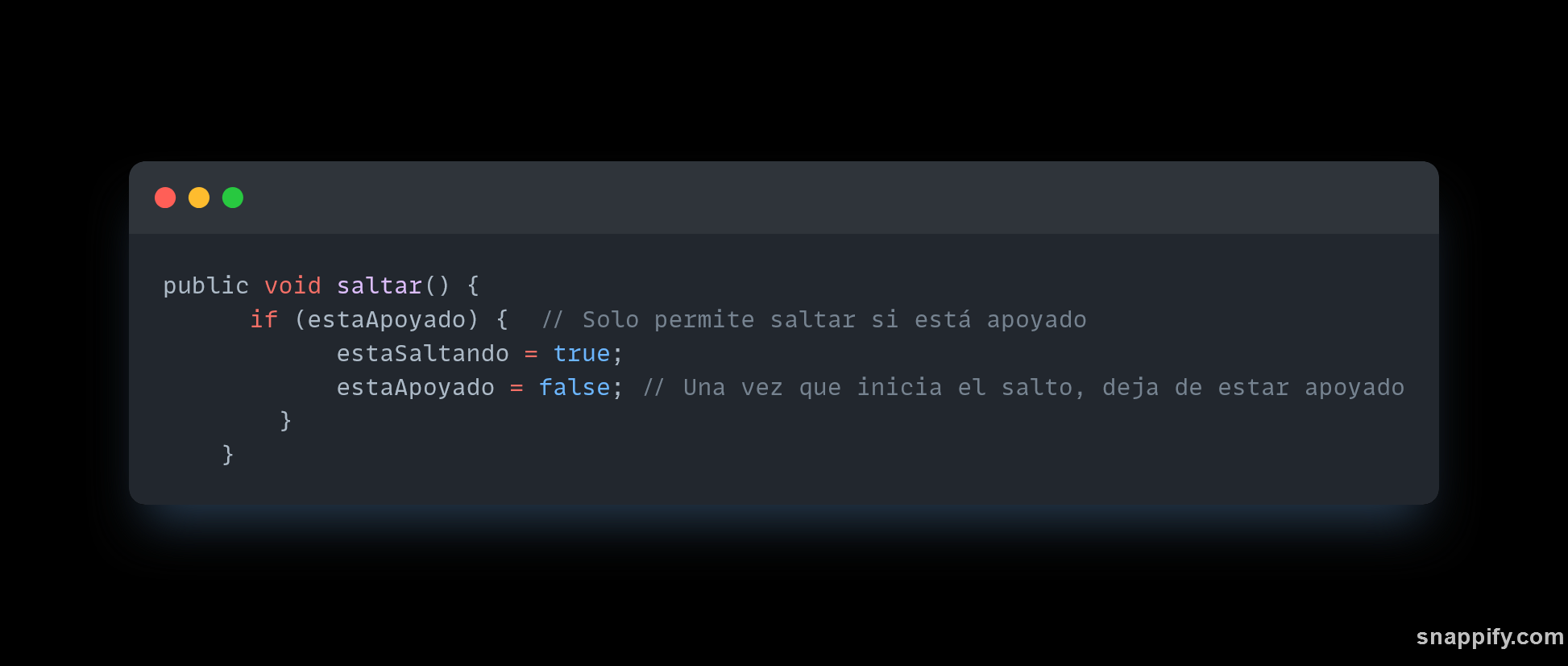
Gestiona el movimiento vertical (gravedad y salto).

* **mover(double v):**



Controla los movimientos de Pep horizontalmente.

* **saltar():**



Inicia el salto si Pep está en una plataforma.

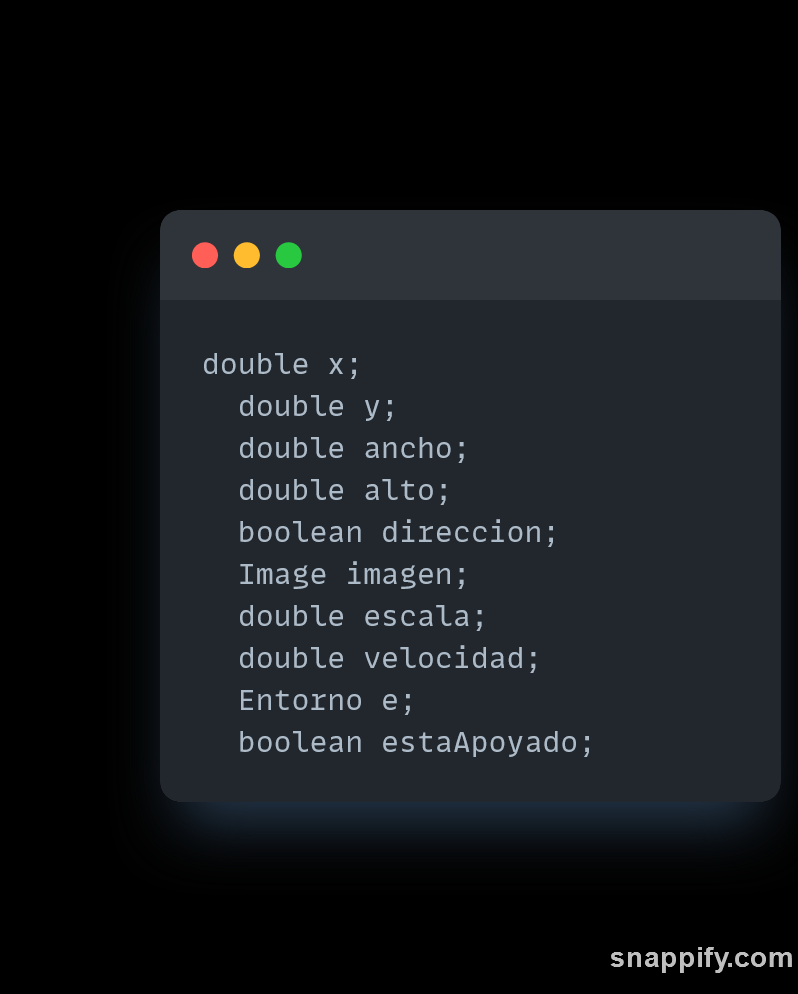
* **cancelarSalto():**



Finaliza el salto cuando Pep toca una superficie.

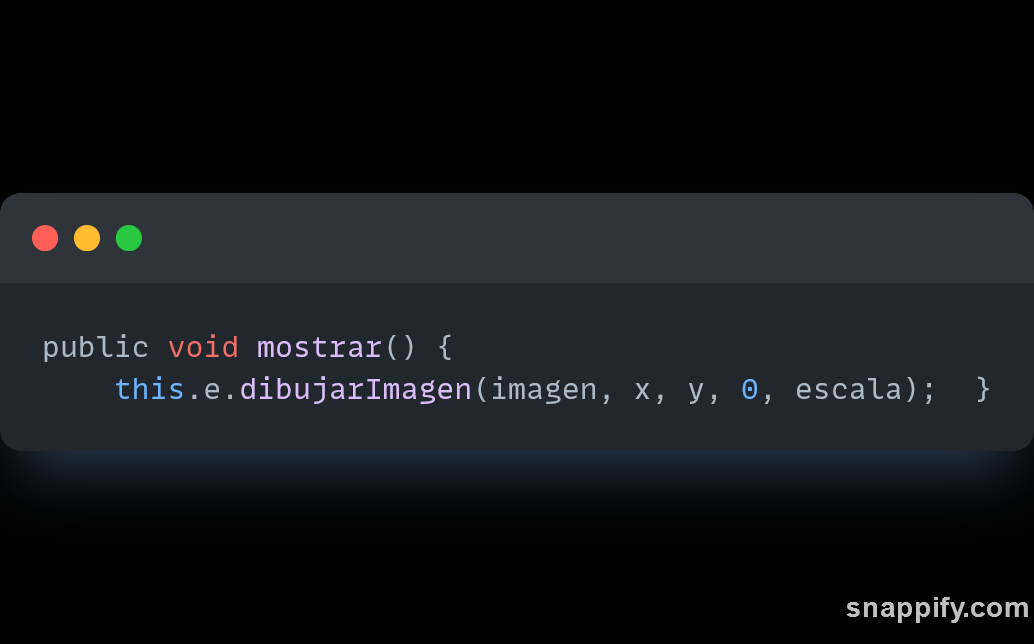
**2.3** **Clase Isla**

**Variables de instancia:**

En la ***clase Isla***, las variables *x* y *y* determinan la posición de la isla, mientras que *ancho* y *alto* definen su tamaño. *imagen* representa la imagen visual de la isla, y *escala* y *velocidad* permiten ajustar el tamaño y la velocidad de movimiento. La variable *estaApoyado* indica si la isla está en contacto con algún personaje.

**Métodos principales:**

* **mostrar():**



Dibuja las islas en la pantalla.

**2.4** **Clase Tortuga**

**Variables de instancia:**

La ***clase Tortuga*** contiene las variables *x* y *y* para la posición de la tortuga, así como *ancho* y *alto* para su tamaño. La dirección de movimiento está determinada por *direccion*, y las imágenes *imagenDer*, *imagenIzq* y *imagenHerida* representan distintos estados de la tortuga. *escala* y *velocidad* ajustan el tamaño y la velocidad, mientras que *herida* señala si la tortuga ha sido dañada.

**Métodos principales:**

* **mostrar():**
* **actualizar(Isla[] islas):**
* **herir():**
* **moverEnIsla(Isla isla):**

3.0 Conclusión