Clase JUEGO

Atributos:
- **niveles:** Lista de instancias de Nivel.
- **jugador:** Instancia de Jugador

Métodos:
- **iniciarJuego():** Inicia el juego.
- **cambiarNivel(int numeroNivel):** Cambia el nivel actual.
- **actualizarJuego():** Actualiza el estado general del juego.

Clase (abstracta) NIVEL

Atributos:
- **enemigos:** Lista de **Enemigo**.
- **objetos:** Lista de **ObjetoInteractivo**.

Métodos:
- i**niciarNivel():** Configura el nivel.
- **actualizarNivel():** Actualiza el nivel durante el juego.
- **verificarColisiones():** Revisa las colisiones entre jugador y objetos.

Clases NivelIniciacion, NivelAscenso, NivelRebelion

Cada clase representa un nivel específico y contiene lógica de juego única para dicho nivel (ejemplo: esquivar, lanzar objetos, etc.).

«use»

Clase ENEMIGO

Atributos:
- **posicion:** Posicion.
- **estado:** indica si está activo, derrotado, etc.

Métodos:
- **lanzarObjeto():** Lanza un objeto al jugador.

Clase JUGADOR

Atributos:
- **vidas:** int, cantidad de vidas del jugador.
- **posicion:** Posicion.

Métodos:
- **moverse():** Mueve al jugador.
- **saltar():** Inicia el salto del jugador.
- l**anzarObjeto():** Lanza un objeto.
- **recolectarCerveza():** Aumenta el contador de lanzamientos posibles.

Clase (abstracta) ObjetoInteractivo

Atributos Comunes:
- **posicion**: Posicion.
- **activo**: bool, indica si el objeto está activo o inactivo.

Métodos Comunes:
- **interactuarConJugador(Jugador jugador):** Define la interacción específica con el jugador.)

Clases PLATAFORMA, CERVEZA, OBSTACULO

Cada clase tiene una interacción específica con el jugador (ejemplo: las plataformas permiten el salto, las cervezas otorgan lanzamientos, y los obstáculos restan vidas).

«use»

«use»

«use»

«use»

Clase POSICION

Atributos:
- **x:** Coordenada horizontal.
- **y:** Coordenada vertical.