Explicación sobre la clase "Item"

# Introducción:

La clase 'Item' en este programa es una clase base que representa a los elementos dentro de una escena gráfica en un juego o simulador. Estos elementos pueden ser personajes, objetos interactivos u otros tipos de elementos visuales.

# Atributos:

- ID: Este atributo almacena un identificador único para cada ítem. Este ID se utiliza para referenciar y distinguir diferentes ítems dentro de la escena.

- posicion: Este atributo representa la posición del ítem dentro del espacio 2D, utilizando el tipo QPointF de Qt que guarda las coordenadas X e Y.

- geometria: Este atributo es un objeto QRectF que define el área rectangular de colisión del ítem. Es usado para determinar si un ítem ha colisionado con otro en la escena.

- masa: El atributo 'masa' almacena la masa del ítem, un valor numérico que puede ser utilizado en físicas o cálculos relacionados con la interacción entre objetos.

- sprite: Este atributo guarda la ruta del archivo de imagen (sprite) que representa visualmente al ítem. Es utilizado para mostrar el ítem en la pantalla en la posición determinada por 'posicion'.

- visible: El atributo 'visible' indica si el ítem es visible o no en la escena. Si es verdadero, el ítem será renderizado; si es falso, no será mostrado.

- destruir: Este atributo booleano se utiliza para marcar si el ítem debe ser destruido. El valor 'true' indica que el ítem debe ser eliminado de la escena.

- estadoColision: Este atributo define el estado de la colisión del ítem. Usa la enumeración 'EstadoColision' que puede tener tres valores: 'SinColision', 'ColisionDetectada' y 'ColisionResuelta'.

# Métodos:

- getID(): Este método devuelve el identificador único del ítem. Es un getter para el atributo 'ID'.

- getPosicion(): Este método devuelve la posición del ítem en la escena. Retorna un objeto QPointF.

- getGeometria(): Este método devuelve la geometría del ítem, un objeto QRectF, que se utiliza para las colisiones.

- getMasa(): Este método devuelve la masa del ítem, un valor de tipo float.

- getSprite(): Este método devuelve la ruta del sprite del ítem, es decir, el archivo de imagen asociado con el ítem.

- isVisible(): Este método devuelve un valor booleano que indica si el ítem es visible o no en la escena.

- isDestruir(): Este método devuelve un valor booleano que indica si el ítem está marcado para ser destruido.

- getEstadoColision(): Este método devuelve el estado de colisión del ítem, que es de tipo 'EstadoColision'.

- setID(): Este método establece un nuevo identificador para el ítem. Es un setter para el atributo 'ID'.

- setPosicion(): Este método establece la nueva posición del ítem. Recibe un objeto QPointF que define las nuevas coordenadas.

- setGeometria(): Este método establece la nueva geometría del ítem. Recibe un objeto QRectF que define el área de colisión.

- setMasa(): Este método establece la nueva masa del ítem.

- setSprite(): Este método establece la nueva ruta del sprite del ítem.

- setVisible(): Este método establece si el ítem será visible o no en la escena.

- setDestruir(): Este método establece si el ítem debe ser destruido o no.

- setEstadoColision(): Este método establece el estado de colisión del ítem, usando la enumeración 'EstadoColision'.

- interactuarCon(): Este es un método virtual puro que debe ser implementado por las clases derivadas. Permite que un ítem interactúe con otro, en este caso, con un objeto de la clase 'Bart'.

- boundingRect(): Este método sobrescribe el método de 'QGraphicsItem' y devuelve un rectángulo delimitador que representa el área de colisión del ítem. Es utilizado para la detección de colisiones.