

BrajanekDefence

Dokumentacja projektu PyGaming

Navigation

Related Topics

- [Documentation overview](#)

Quick search

Go

Welcome to Brajanek Defence’s documentation!

KLASY!

`class mySprites.BrajanekSprite`

Bases: `Sprite`

The main character of the game inherits from the `Sprite` class

`animate_running(left_leg, right_leg)`

Animates the sprite by changing the image to the left leg and then to the right leg :param left_leg: name of the image of the left leg :param right_leg: name of the image of the right leg :return: None

`change_image(name)`

Changes the image of the sprite to the one specified by the name :param name: name of the image to be changed to :return: None

`copy()`

Creates a copy of the sprite :return: copy of the sprite

`die()`

Changes the image of the sprite to the dead one :return: None

`move(direction)`

Moves the sprite in the specified direction :param direction: direction in which the sprite is to be moved :return: None

`update()`

Updates the position of the sprite :return: None

`class mySprites.Bullet(brajanek, speed, direction)`

Bases: `Sprite`

The bullet of the game inherits from the `Sprite` class. Used to kill ferals, has a limited range.

`kill()` → None

Kills the sprite :return: None

`class mySprites.Bushfence(isHorizontal)`

Bases: `Sprite`

The obstacle of the game inherits from the `Sprite` class. The obstacle can be horizontal or vertical

`check_collision(brajanek)`

Checks if the obstacle collides with the sprite :param brajanek: sprite to check collision with :return: True if the obstacle collides with the sprite, False otherwise

`get_direction(brajanek)`

Returns the direction in which the sprite is colliding with the obstacle :param brajanek: sprite to check collision with :return: direction in which the sprite is colliding with the obstacle

`set_location(x, y)`

Sets the location of the obstacle :param x: x coordinate of the obstacle :param y: y coordinate of the obstacle

`class mySprites.Cat(speed=0.5, spawn=(800, 300))`

Bases: `Sprite`

The cat of the game inherits from the `Sprite` class. A feral cat that wants to kill brajanek

`change_direction()`

Changes the direction of the sprite :return: None

`change_image(name)`

Changes the image of the sprite to the one specified by the name :param name: name of the image to be changed to :return: None

`check_collision(brajanek)`

Checks if the obstacle collides with the sprite :param brajanek: sprite to check collision with :return: True if the obstacle collides with the sprite, False otherwise

`get_direction(brajanek)`

Returns the direction in which the sprite is colliding with the obstacle :param brajanek: sprite to check collision with :return: direction in which the sprite is colliding with the obstacle

`move(brajanek)`

Moves the sprite in the direction of the sprite :param brajanek: sprite to move towards :return: None

`set_location(x, y)`

Sets the location of the obstacle :param x: x coordinate of the obstacle :param y: y coordinate of the obstacle

`update()`

Updates the location of the sprite :return: None

`class mySprites.Coin(catX, catY)`

Bases: `Sprite`

The coin of the game inherits from the `Sprite` class. Used as a game end mechanic.

`class mySprites.CoinUI(x, y)`

Bases: `Sprite`

The second coin of the game inherits from the `Sprite` class. Used as a visual representation of coins collected.

`class mySprites.Heart(x, y)`

Bases: `Sprite`

The heart of the game inherits from the `Sprite` class. Used as a visual representation of lives left.

`class mySprites.Player`

Bases: `object`

`decrease_lives()`

Decreases the amount of lives the player has by 1 :return: Players new amount of lives

`get_lives()`

Returns the amount of lives the player has :return: returns how many lives the player has left

`get_score()`

Returns the score of the player :return: how many cats the player had neutralized :)

`increase_score(amount)`

Increases the score of the player by the given amount :param amount: set in main :) :return: the new score of the player

Indices and tables

- [Index](#)
- [Module Index](#)
- [Search Page](#)