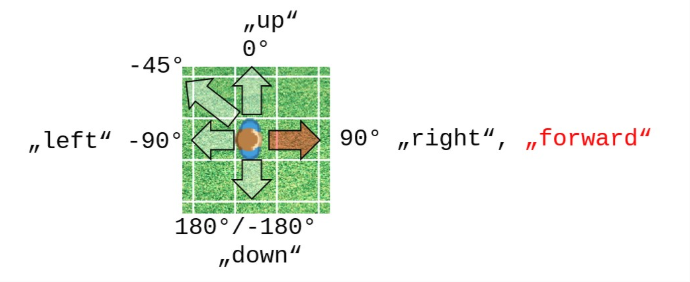
CHEAT-SHEET

miniworldmaker v.0.1

### Token

Tokens on board.

direction: direction of token



size: Size of token as tuple (e.g. (40, 40)

position: Position of tuple (e.g. (30, 20)

x; y: x and y coordinate

add\_image(path) – Adds an image

add\_costume(path) – Adds an costume with default image

switch\_costume(index) – Switch to next costume or costume with index

Actor (Subclass of Token):

Can use all attributes and methods of token

speed: The actor speed (Used in combination with move()

\_\_init\_\_(position) – Creates new actor

point\_in\_direction(direction) – Actor points in direction

point\_towards\_position(destination) – Actor points towards position

point\_towards\_token(destination) – Actor points towards token or actor

turn\_left(degrees) – Actor turns left by degrees degrees

turn\_right(degrees) – Actor turns left by degrees degrees

move\_in\_direction (direction) – Actor moves in direction

move\_to (position) – Actor moves to position

move (distance) – Actor moves distance (or self.speed) steps forward (distance can be negative)

flip\_x() – Flips the actor

bounce\_from\_border(border) – Actor bounces from border

bounce\_from\_token(token) – Actor bounces from token or actor

sensing\_tokens(distance, token, exact) – Actor senses tokens. **token**: The class of tokens to search for. **exact**: Should be the collision detection exact?

Sensing\_borders(distance) – Gets list of borders as string

sensing\_on\_board(distance) – True if sensing on\_board\_position

sensing\_color(color, distance) – Senses number of pixels of a given color

sensing\_colors(distance) – Senses all colors.

Board

Don’t use this. Use either PixelBoard or TiledBoard

speed – The animation speed (between 0 and 100

width, height: width and height of board

rows, columns: rows and columns of board

add\_image(path) – Adds an image

add\_to\_board(token) – Adds token or actor to board

get\_tokens\_by\_pixel(pixel) – Gets all tokens by pixel

remove\_from\_board(token) – Removes token or actor from board

play\_sound(path) – plays a sound

play\_music(path) – plays a music

get\_mouse\_position() – Gets the mouse\_position

### TiledBoard

\_\_init\_\_(columns, rows, tile\_size, tile\_margin) – Creates a new TilesBoard

### PixelBoard

\_\_init\_\_(columns, rows) – Creates a PixelBoard

### Costume (Subclass of Appearance)

Can be accessed via token.costume

is\_animated: Is token animated?

is\_rotatable: Is token rotatable?

is\_textured: Is token a texture and should be repeated?

is\_scaled: Is token scaled to actor.size remaining aspect ratio?

is\_scaled: Is token upscaled to actor.size not remaining aspect ratio?

orientation: Correct initial orientation (e.g. self.costume.orientation = -90)

add\_image(path) – Adds an image

colorize(color) – Colorizes a token with a color

### Background (Subclass of appearance)

Can be accessed via board.background

See costume for attributes, because they are sharing same attributes and methods.

### is\_scaled\_to\_tile: Is background scaled to tile\_size? Use this in combination with is\_textured

### BoardPosition

A board position. On TiledBoard this is the coordinate of the tile.

BoardPosition.from\_pixel(position) – Creates a BoardPosition from Pixel-Coordinates (e.g. mouse coordinates)

near(other, distance): Is Position near another position

to\_tuple() – Transforms position to tuple (x, y)

up(value), down(value),… – Gets the position value steps up

is\_on\_board() – Tests if position is on board