# 

### SquareToy%

x-pos: PosInt y-pos: PosInt speed: PosInt right? -> Boolean

On-tick -> SquareToy% Check-canvas-limit-right -> SquareToy% Check-canvas-limit-left ->

Check-carivas-iiiiit-left ->

SquareToy%

Add-to-scene -> Scene

Toy-x -> PosInt Toy-y-> PosInt

Toy-color -> ColorString

For-test:toy-speed-> PosInt

For-test:toy-speed -> PosInt For-test:toy-right? -> PosInt

For-test:toy-equal? -> Boolean

#### CircleToy%

x-pos : PosInt y-pos: PosInt

count -> NonNegInt

On-tick -> CircleToy% Add-to-scene -> Scene

Toy-x -> posInt Toy-y -> PosInt

Toy-color -> ColorString

For-test:toy-count ->

NonNegInt

For-test:toy-equal? ->

Boolean

## <<interface>> World%

On-tick On-mouse On-key On-draw

Target-x Target-y

Target-selected?

Get-toys

implements

#### World%

x-pos: Integer y-pos: Integer selected?: Boolean mouse-x: Integer mouse-y: Integer speed: PosInt toys: ListOf<Toys>

On-tick-> World% On-mouse-> World%

World-after-button-up ->World%

 $World-after-button-down \to World\%$ 

World-after-drag -> world%

In-circle? -> Boolean

On-key-> World%

World-after-s-key-> world%

On-draw -> scene Target-x -> Integer Target-y -> Integer

Target-selected? -> Boolean

Fortest:target-mx -> Integer

Fortest:target-my-> Integer

For-test:target-speed -> PosInt

Get-toys -> ListOf<Toys>

For-test:world-equal? -> boolean