POWER OF AR AND VR

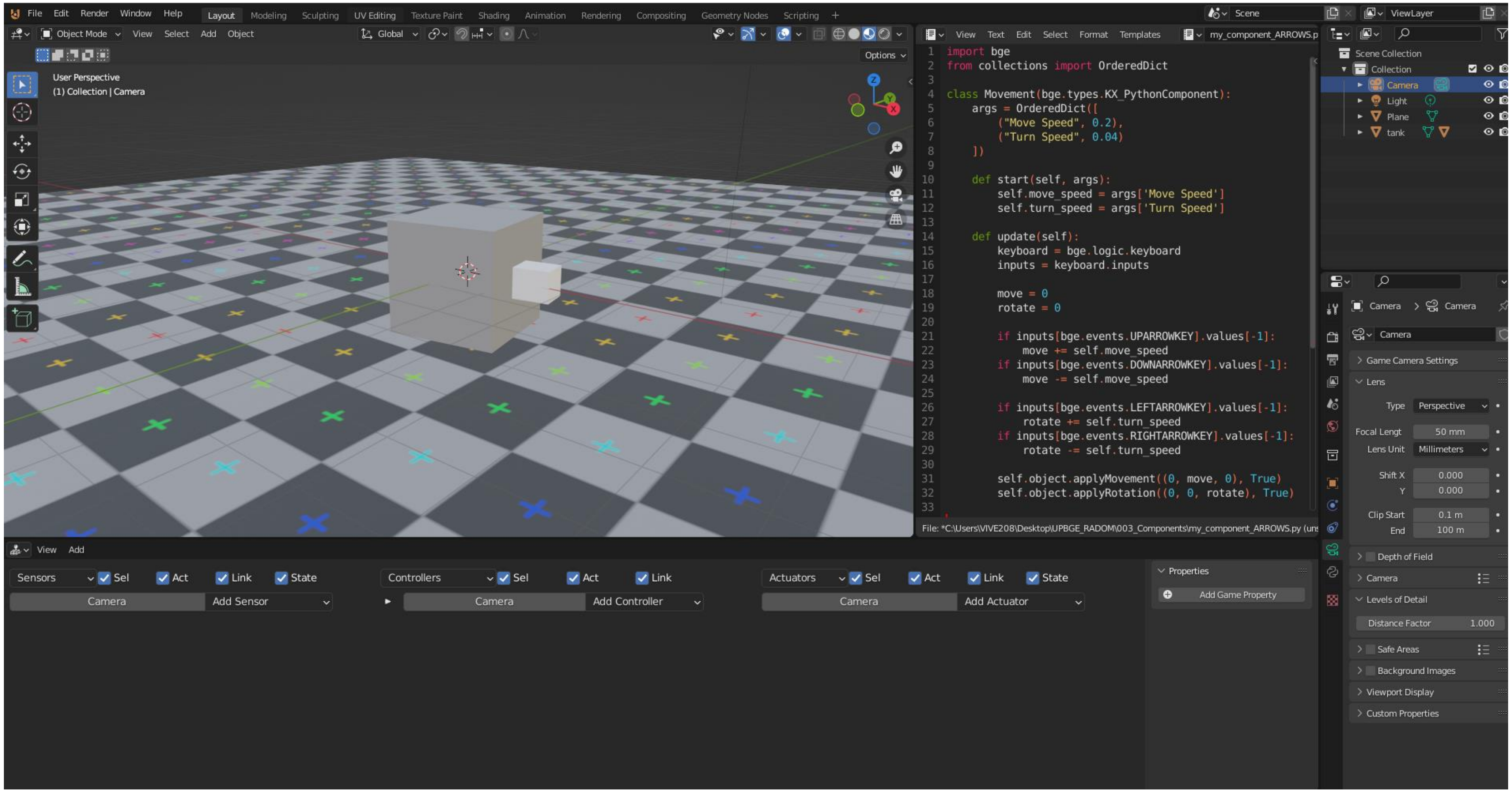
UPBGE

Camera follow

Co-funded by the European Union

JACEK KAWAŁEK

WE ARE OPENING AN PREVIOUS FILE

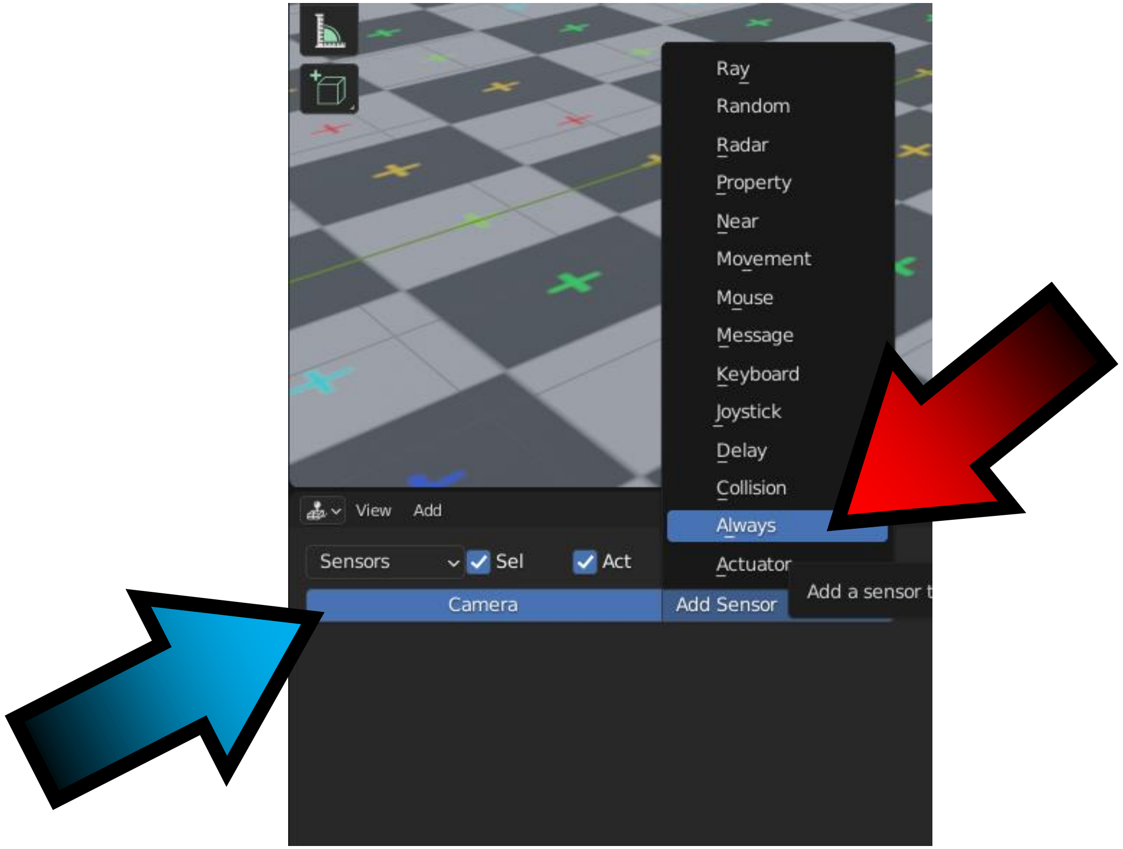


```python
import bge
from collections import OrderedDict

class Movement(bge.types.KX_PythonComponent):
    args = OrderedDict([
        ("Move Speed", 0.2),
        ("Turn Speed", 0.04)
    ])

    def start(self, args):
        self.move_speed = args['Move Speed']
        self.turn_speed = args['Turn Speed']

    def update(self):
        keyboard = bge.logic.keyboard
        inputs = keyboard.inputs

        move = 0
        rotate = 0

        if inputs[bge.events.UPARROWKEY].values[-1]:
            move += self.move_speed
        if inputs[bge.events.DOWNARROWKEY].values[-1]:
            move -= self.move_speed

        if inputs[bge.events.LEFTARROWKEY].values[-1]:
            rotate += self.turn_speed
        if inputs[bge.events.RIGHTARROWKEY].values[-1]:
            rotate -= self.turn_speed

        self.object.applyMovement((0, move, 0), True)
        self.object.applyRotation((0, 0, rotate), True)
```

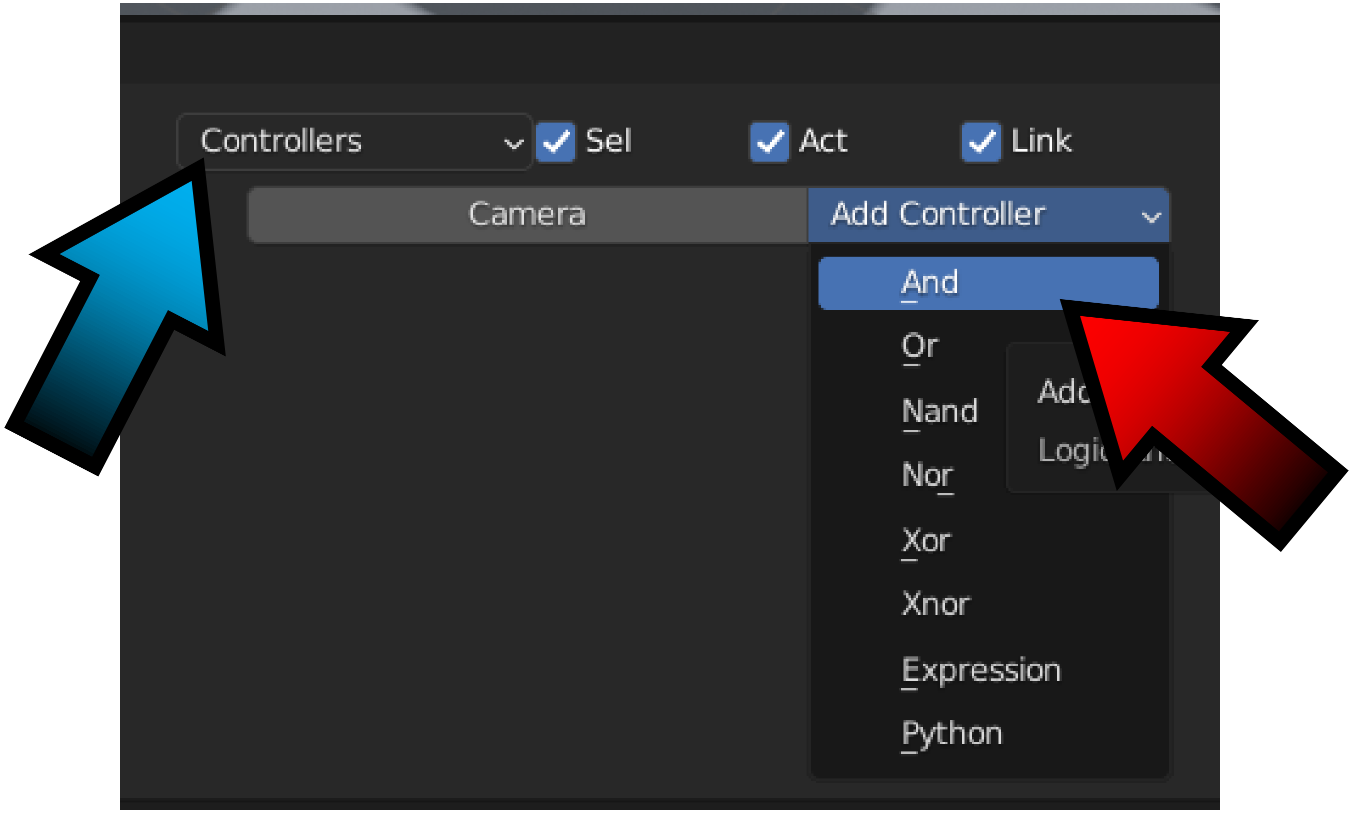
File: *C:\Users\VIVE208\Desktop\UPBGE_RADOM\003_Components\my_component_ARROWS.py (uns

Type: Perspective
Focal Lengt: 50 mm
Lens Unit: Millimeters
Shift X: 0.000
Y: 0.000
Clip Start: 0.1 m
End: 100 m
Distance Factor: 1.000

FOR THE SELECTED CAMERA WE CHOOSE SENSOR ALWAYS



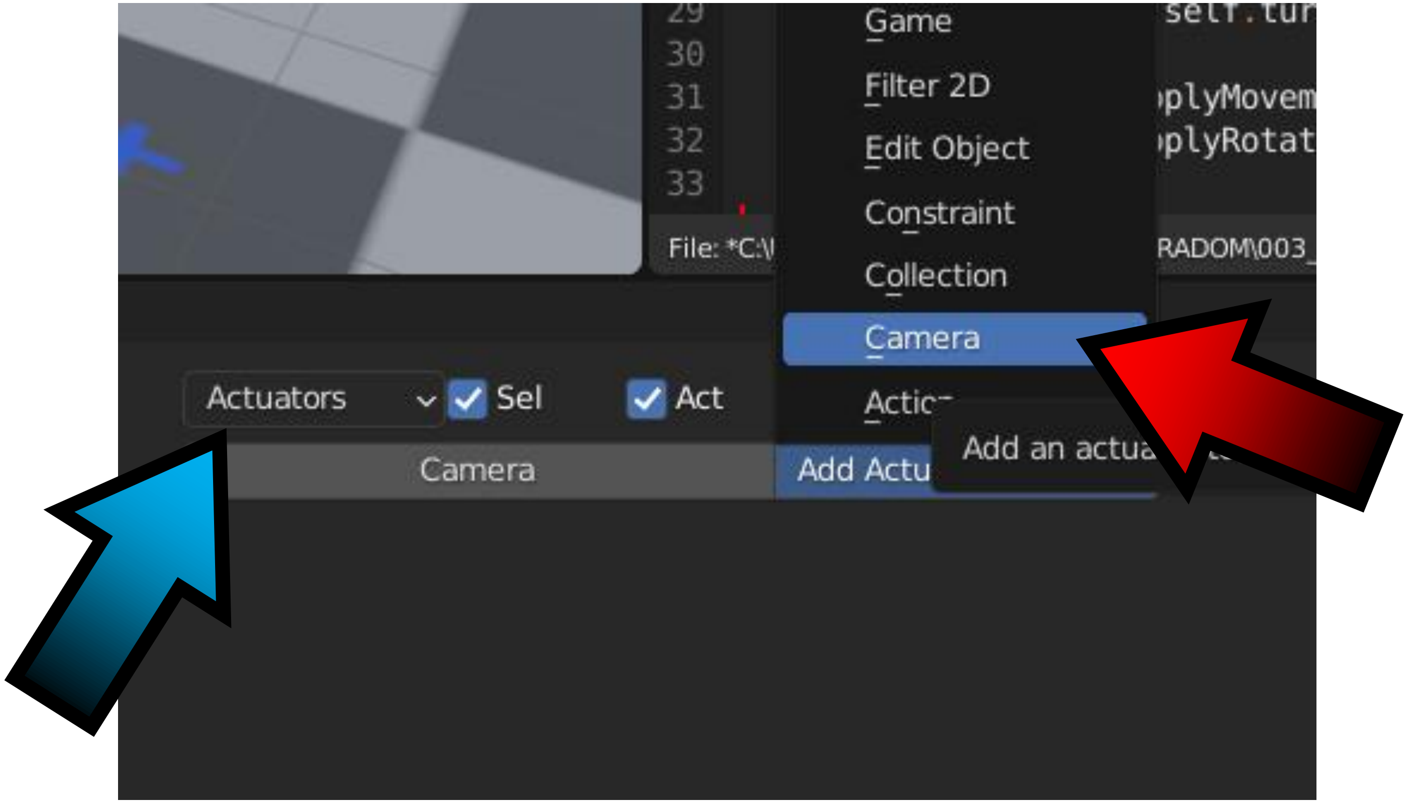
THEN FOR CONTROLLERS WE SELECT AND



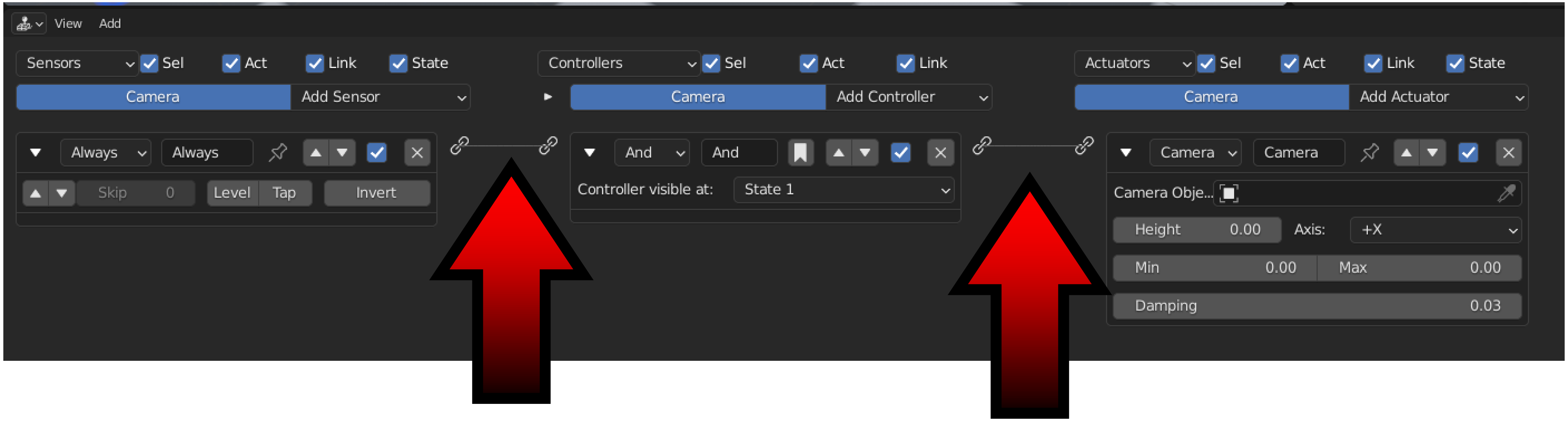
Controllers ☑ Sel ☑ Act ☑ Link

Camera | Add Controller

FOR ACTUATORS WE CHOOSE CAMERA



WE CONNECT AS BEFORE



View Add

Sensors ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Sensor

Always | Always

Skip 0 | Level | Tap | Invert

Controllers ☑ Sel ☑ Act ☑ Link

Camera | Add Controller

And | And

Controller visible at: State 1

Actuators ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Actuator

Camera | Camera

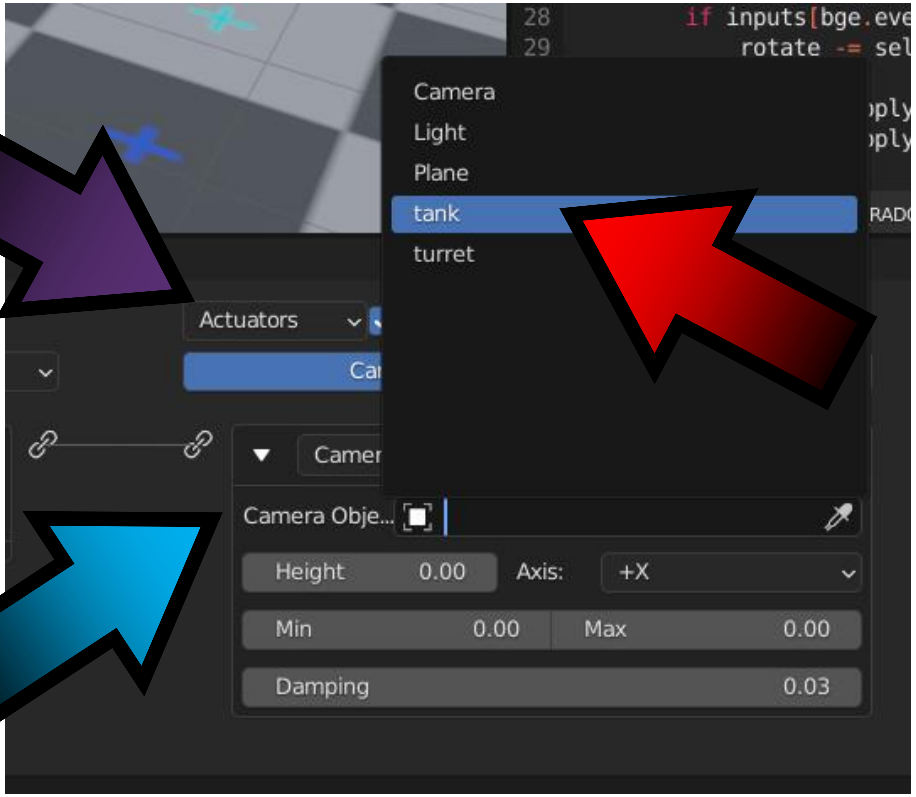
Camera Obje...

Height 0.00 | Axis: +X

Min 0.00 | Max 0.00

Damping 0.03

IN ACTUATORS FOR CAMERA OBJECT WE CHOOSE TANK



```
    if inputs[bge.eve
        rotate -= sel
```

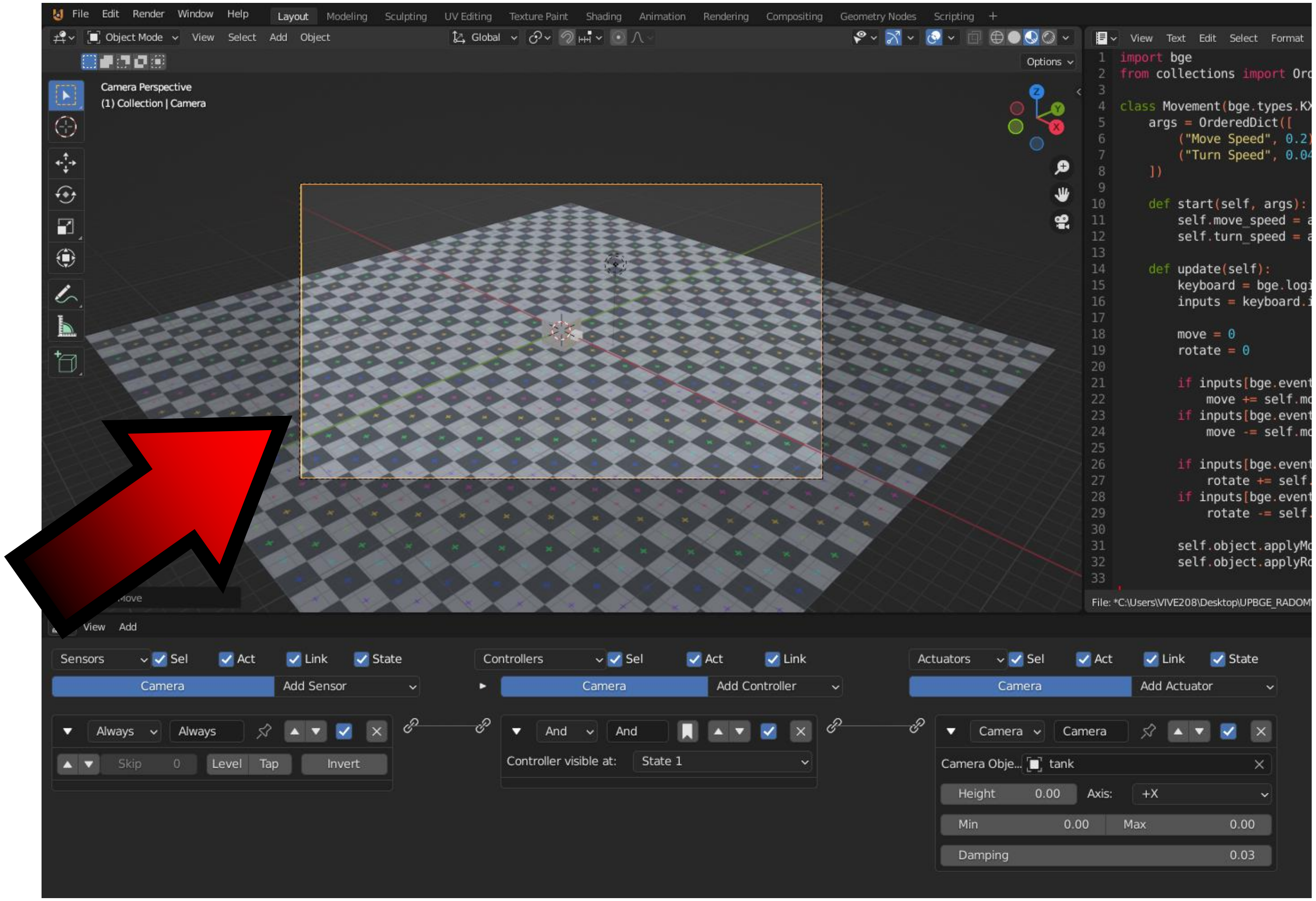
Camera Obje...

Height 0.00 Axis: +X

Min 0.00 Max 0.00

Damping 0.03

WE SET
THE CAMERA VIEW



File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Object Mode View Select Add Object
Global
Options
Camera Perspective
(1) Collection | Camera
Move
View Add
View Text Edit Select Format

```
import bge
from collections import Orc
 
class Movement(bge.types.KX
    args = OrderedDict([
        ("Move Speed", 0.2)
        ("Turn Speed", 0.04
    ])
 
    def start(self, args):
        self.move_speed = a
        self.turn_speed = a
 
    def update(self):
        keyboard = bge.logi
        inputs = keyboard.i
 
        move = 0
        rotate = 0
 
        if inputs[bge.event
            move += self.mo
        if inputs[bge.event
            move -= self.mo
 
        if inputs[bge.event
            rotate += self.
        if inputs[bge.event
            rotate -= self.
 
        self.object.applyMo
        self.object.applyRo
```

File: *C:\Users\VIVE208\Desktop\UPBGE_RADOM

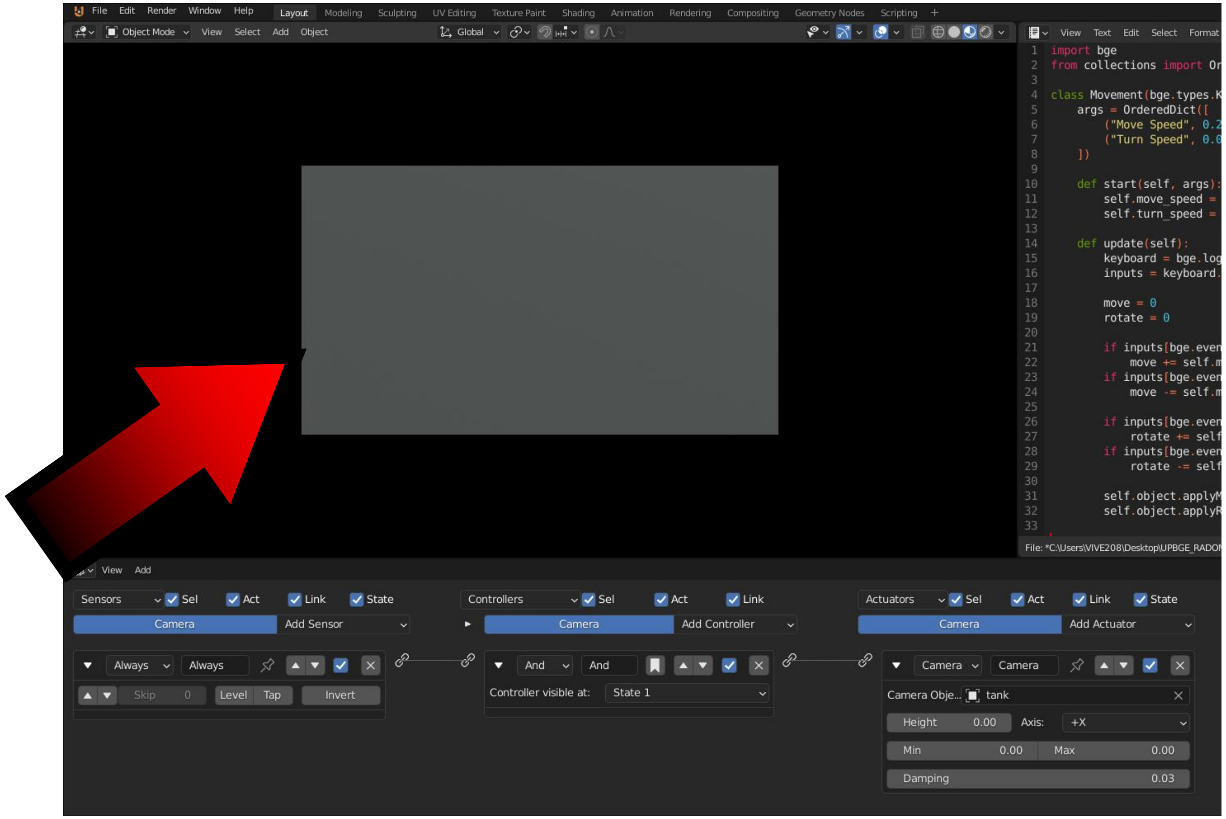
Sensors ☑ Sel ☑ Act ☑ Link ☑ State
Camera | Add Sensor
Always | Always ☑
Skip 0 | Level | Tap | Invert

Controllers ☑ Sel ☑ Act ☑ Link
Camera | Add Controller
And | And ☑
Controller visible at: State 1

Actuators ☑ Sel ☑ Act ☑ Link ☑ State
Camera | Add Actuator
Camera | Camera ☑
Camera Obje... tank
Height 0.00 Axis: +X
Min 0.00 Max 0.00
Damping 0.03

WE'RE STARTING THE GAME

WE'RE INSIDE THE TANK



```
import bge
from collections import Or

class Movement(bge.types.K
    args = OrderedDict([
        ("Move Speed", 0.2
        ("Turn Speed", 0.0
    ])

    def start(self, args):
        self.move_speed =
        self.turn_speed =

    def update(self):
        keyboard = bge.log
        inputs = keyboard.

        move = 0
        rotate = 0

        if inputs[bge.even
            move += self.m
        if inputs[bge.even
            move -= self.m

        if inputs[bge.even
            rotate += self
        if inputs[bge.even
            rotate -= self

        self.object.applyM
        self.object.applyR
```

File: *C:\Users\VIVE208\Desktop\UPBGE_RADOI

Sensors ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Sensor

Always | Always ☑

Skip 0 | Level | Tap | Invert

Controllers ☑ Sel ☑ Act ☑ Link

Camera | Add Controller

And | And ☑

Controller visible at: State 1

Actuators ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Actuator

Camera | Camera ☑

Camera Obje... tank

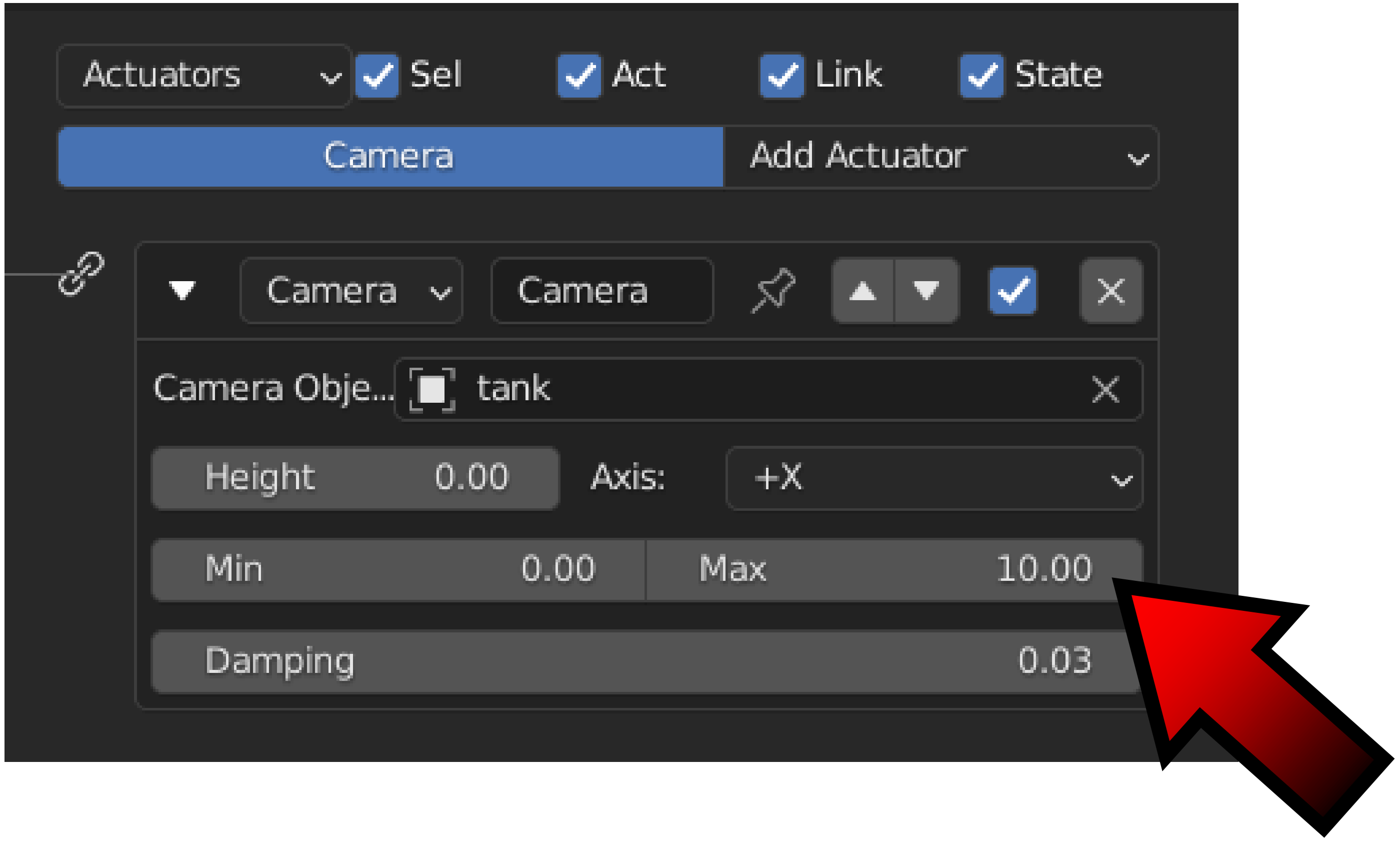
Height 0.00 Axis: +X

Min 0.00 Max 0.00

Damping 0.03

Co-funded by the European Union

LET'S CHANGE MAX TO 10



Actuators ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Actuator

Camera | Camera ☑

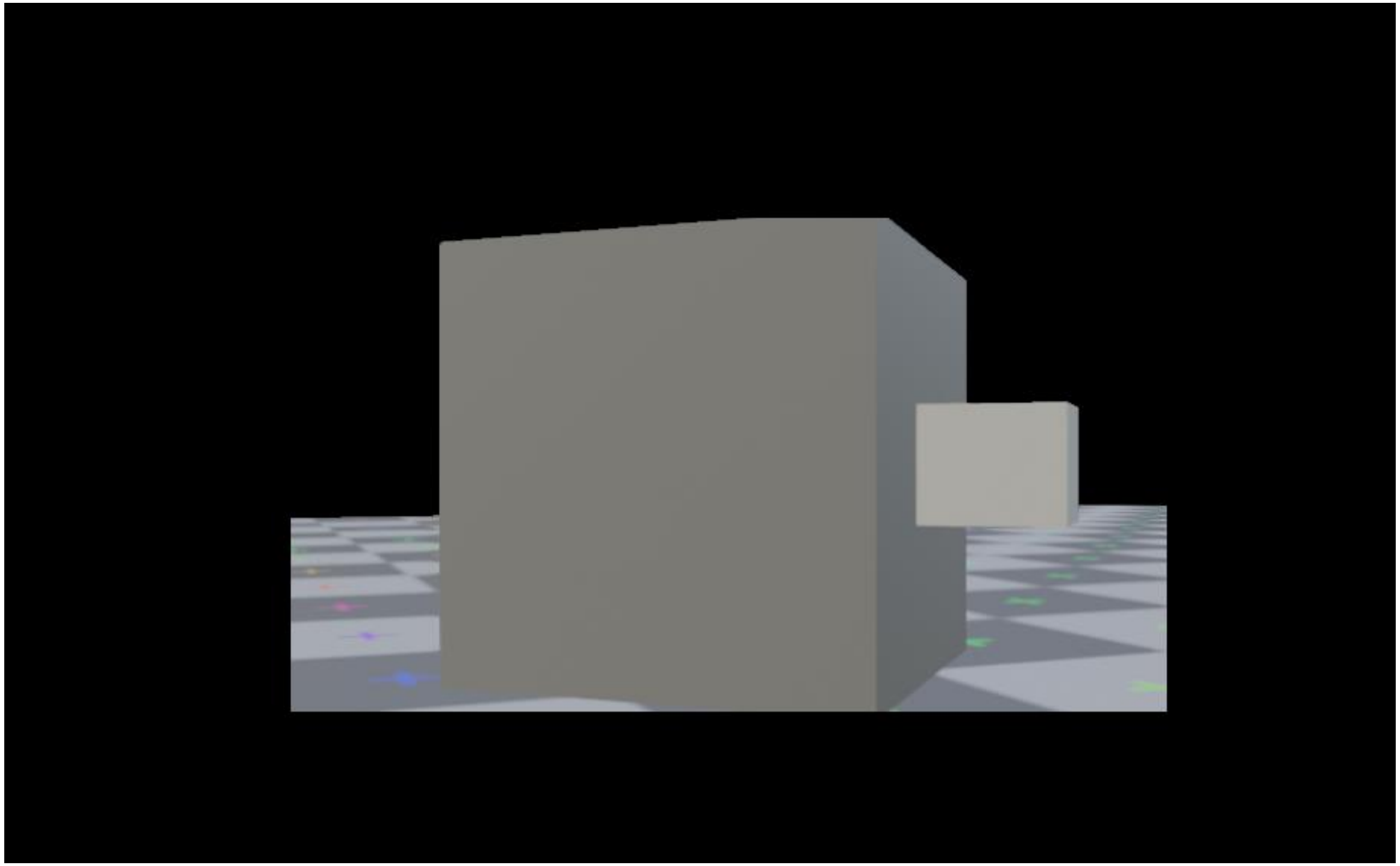
Camera Obje... tank

Height 0.00 Axis: +X

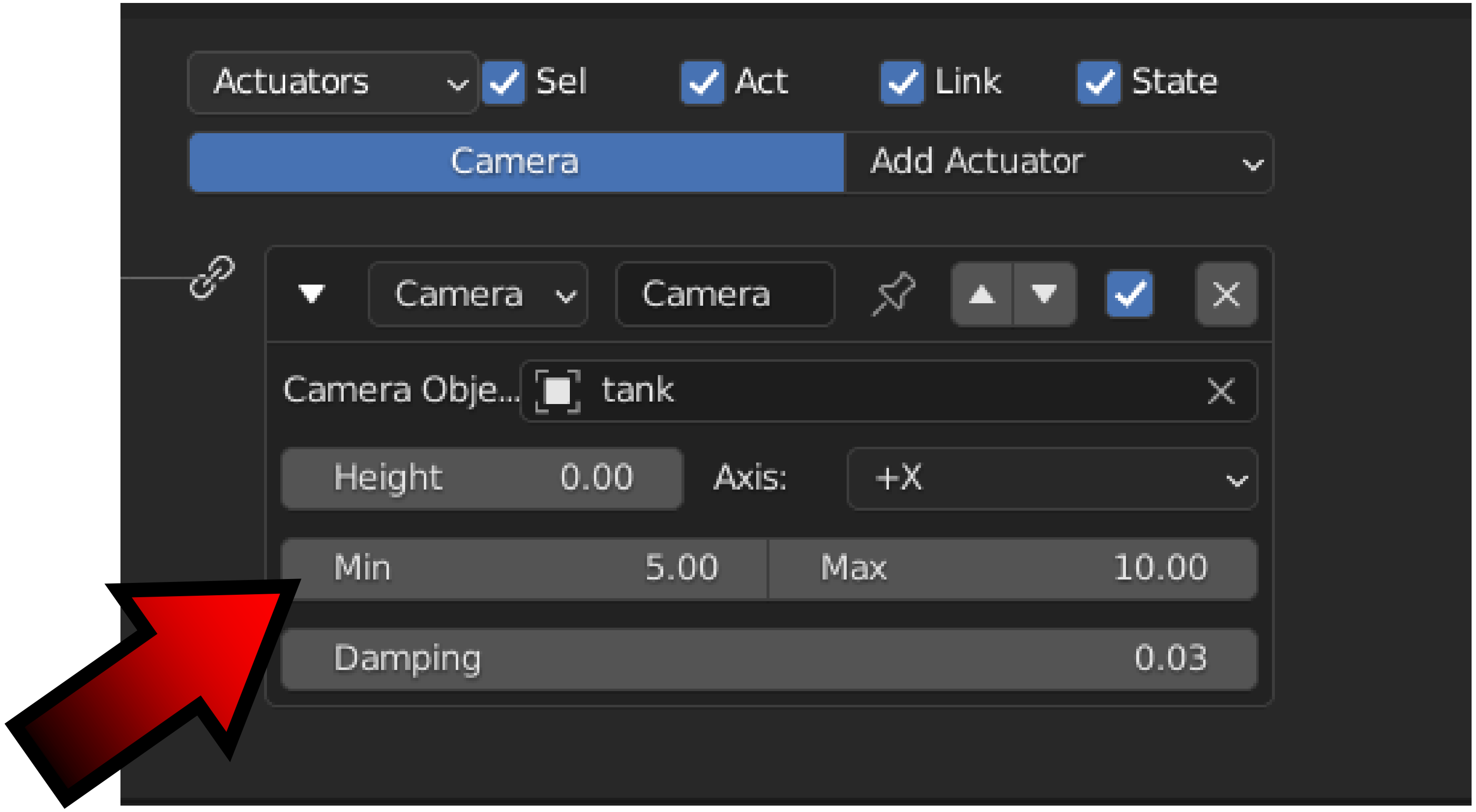
Min 0.00 Max 10.00

Damping 0.03

WE ARE STARTING THE GAME



SET MIN TO 5



Actuators ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Actuator

Camera | Camera ☑

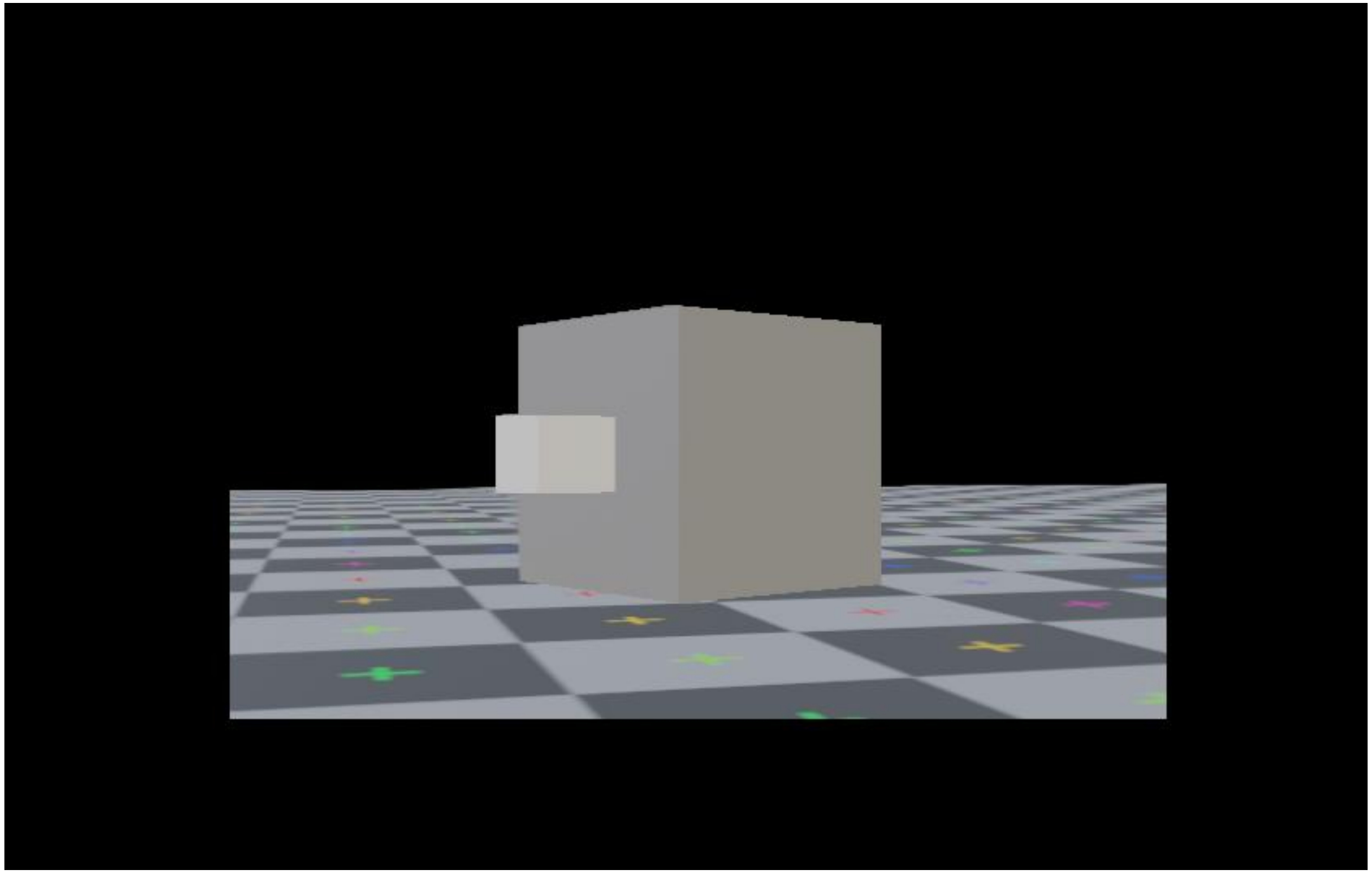
Camera Obje... tank

Height 0.00 Axis: +X

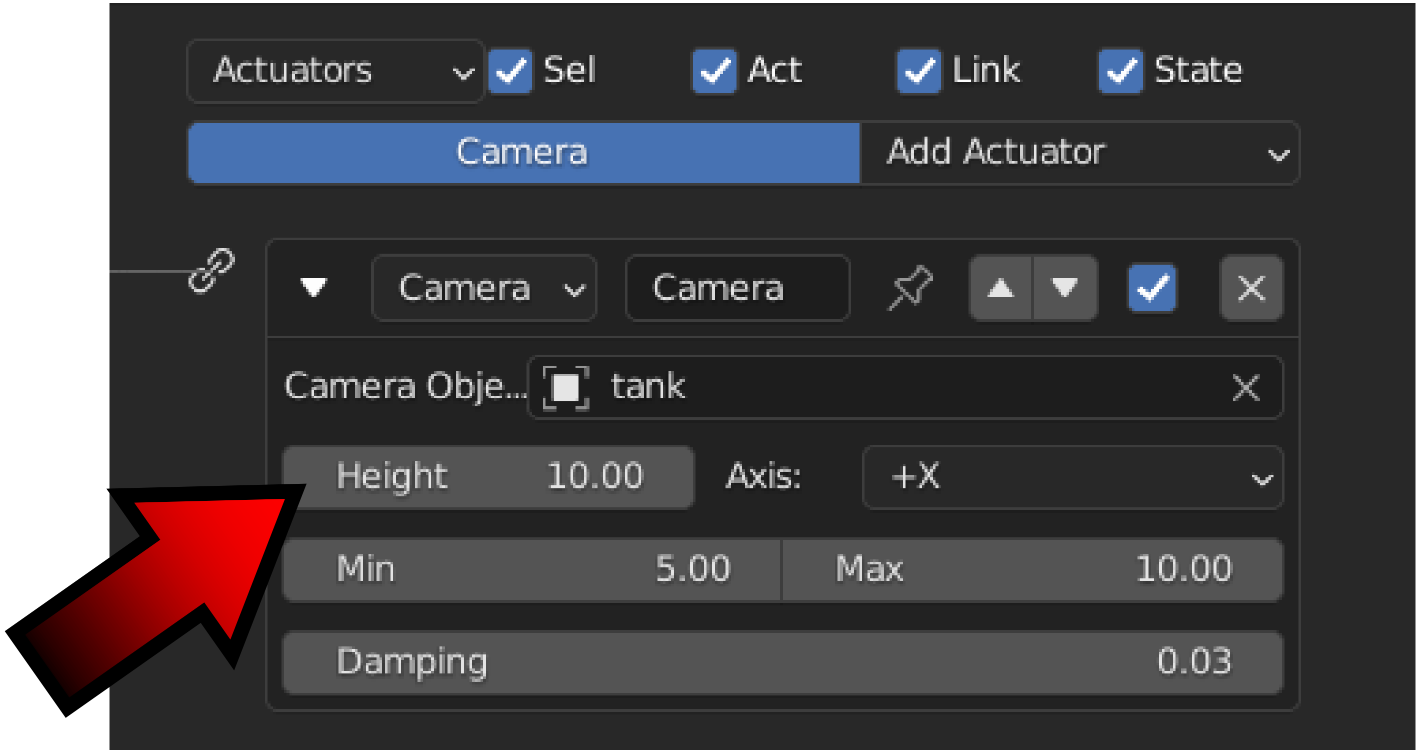
Min 5.00 Max 10.00

Damping 0.03

WE ARE STARTING THE GAME



SET HEIGHT TO 10



Actuators ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Actuator

Camera | Camera ☑

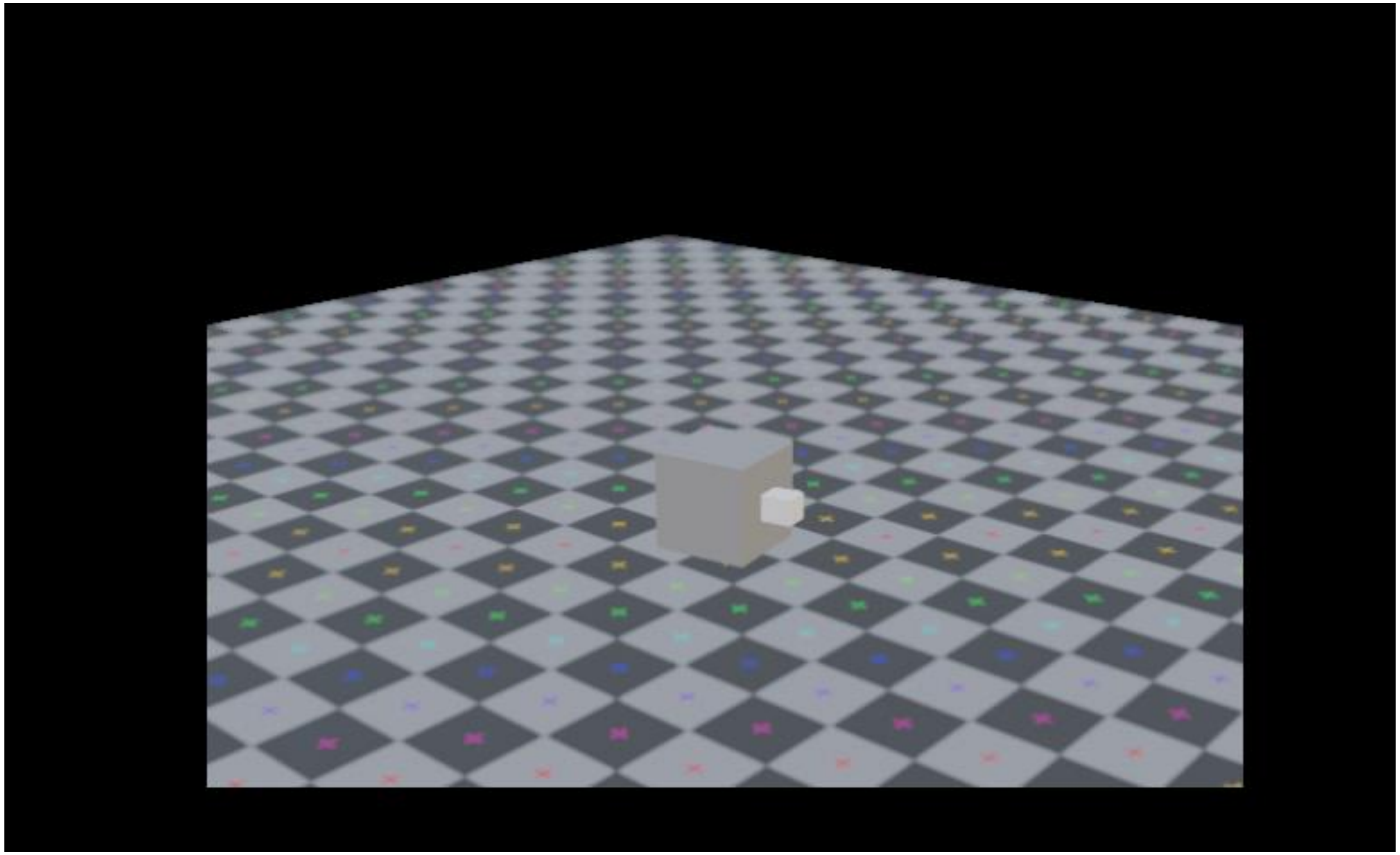
Camera Obje... tank

Height 10.00 Axis: +X

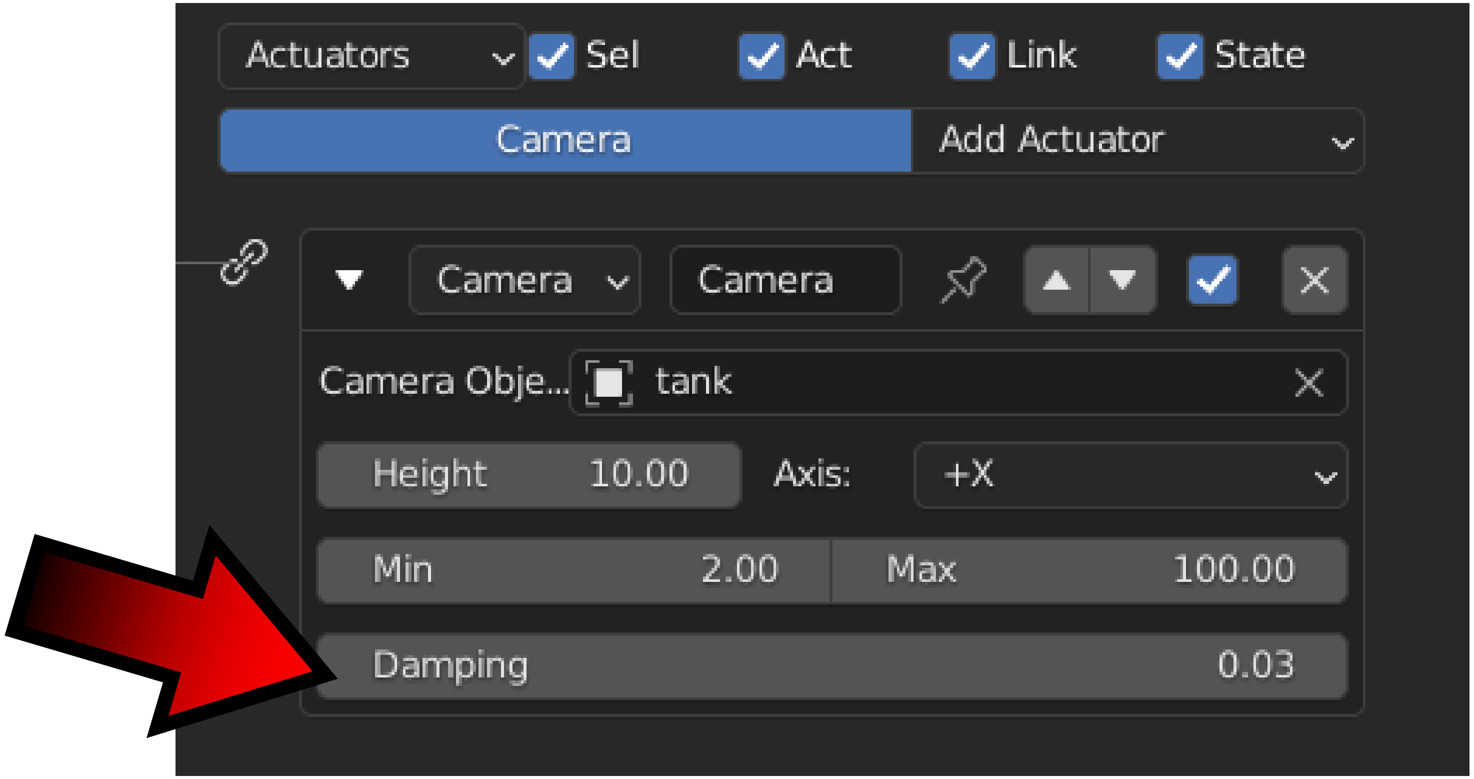
Min 5.00 Max 10.00

Damping 0.03

WE ARE STARTING THE GAME



PARAMETER DAMPING IS USED FOR DAMPING



Actuators ☑ Sel ☑ Act ☑ Link ☑ State

Camera | Add Actuator

Camera | Camera ☑

Camera Obje... tank

Height 10.00 Axis: +X

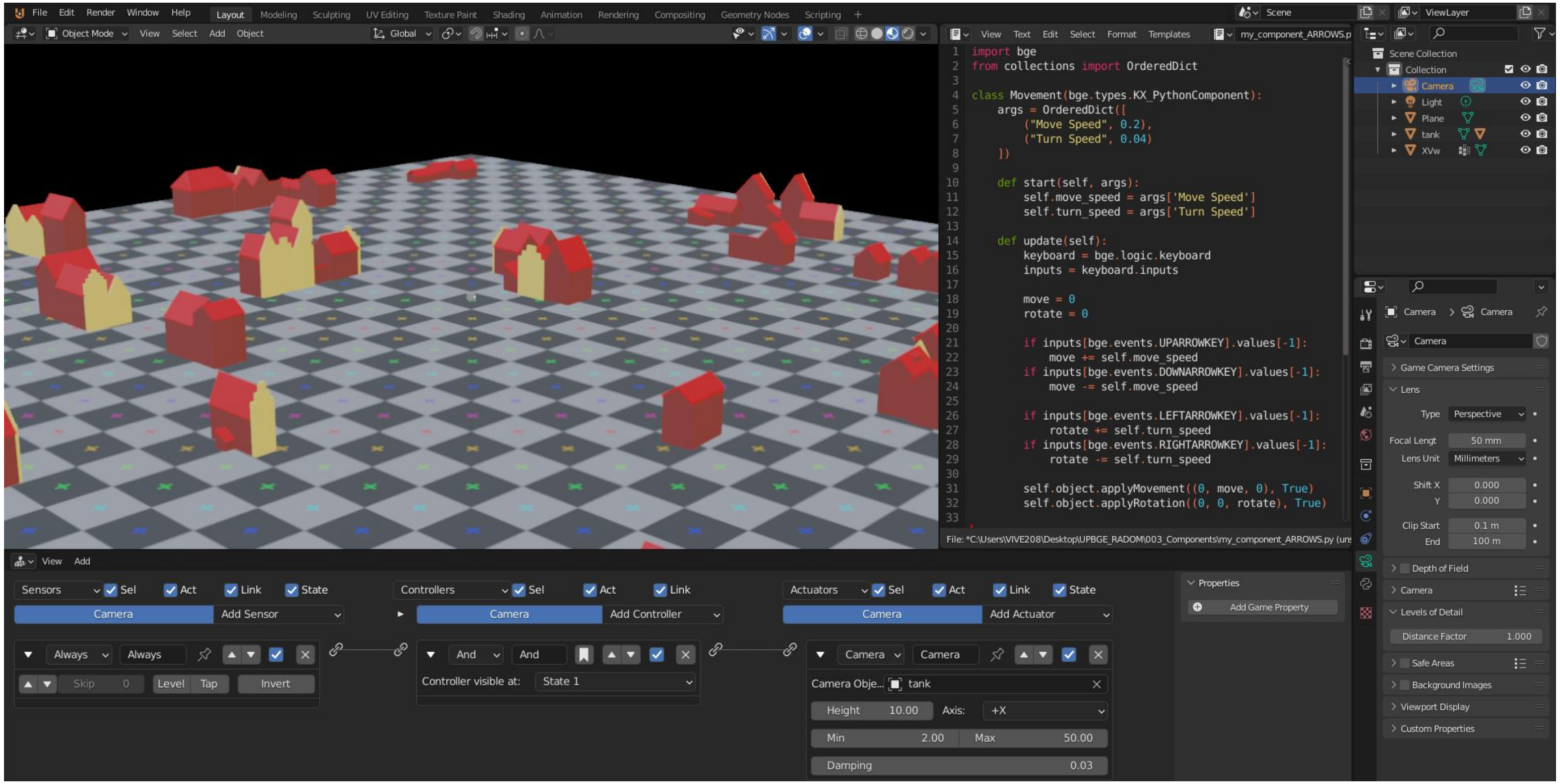
Min 2.00 Max 100.00

Damping 0.03

Co-funded by the European Union

POWER OF AR AND VR

THIS IS HOW YOU CAN TEST DIFFERENT PROJECTS



```python
import bge
from collections import OrderedDict

class Movement(bge.types.KX_PythonComponent):
    args = OrderedDict([
        ("Move Speed", 0.2),
        ("Turn Speed", 0.04)
    ])

    def start(self, args):
        self.move_speed = args['Move Speed']
        self.turn_speed = args['Turn Speed']

    def update(self):
        keyboard = bge.logic.keyboard
        inputs = keyboard.inputs

        move = 0
        rotate = 0

        if inputs[bge.events.UPARROWKEY].values[-1]:
            move += self.move_speed
        if inputs[bge.events.DOWNARROWKEY].values[-1]:
            move -= self.move_speed

        if inputs[bge.events.LEFTARROWKEY].values[-1]:
            rotate += self.turn_speed
        if inputs[bge.events.RIGHTARROWKEY].values[-1]:
            rotate -= self.turn_speed

        self.object.applyMovement((0, move, 0), True)
        self.object.applyRotation((0, 0, rotate), True)
```

File: *C:\Users\VIVE208\Desktop\UPBGE_RADOM\003_Components\my_component_ARROWS.py

Sensors: Camera — Always

Controllers: Camera — And — Controller visible at: State 1

Actuators: Camera — Camera — Camera Obje... tank — Height 10.00 — Axis: +X — Min 2.00 — Max 50.00 — Damping 0.03

Lens: Type Perspective — Focal Lengt 50 mm — Lens Unit Millimeters — Shift X 0.000 — Y 0.000 — Clip Start 0.1 m — End 100 m

Levels of Detail: Distance Factor 1.000

UPBGE

THANK YOU FOR YOUR ATTENTION

Co-funded by
the European Union