

POWER OF AR AND VR

UPBGE

Skrypty Python



**Co-funded by
the European Union**

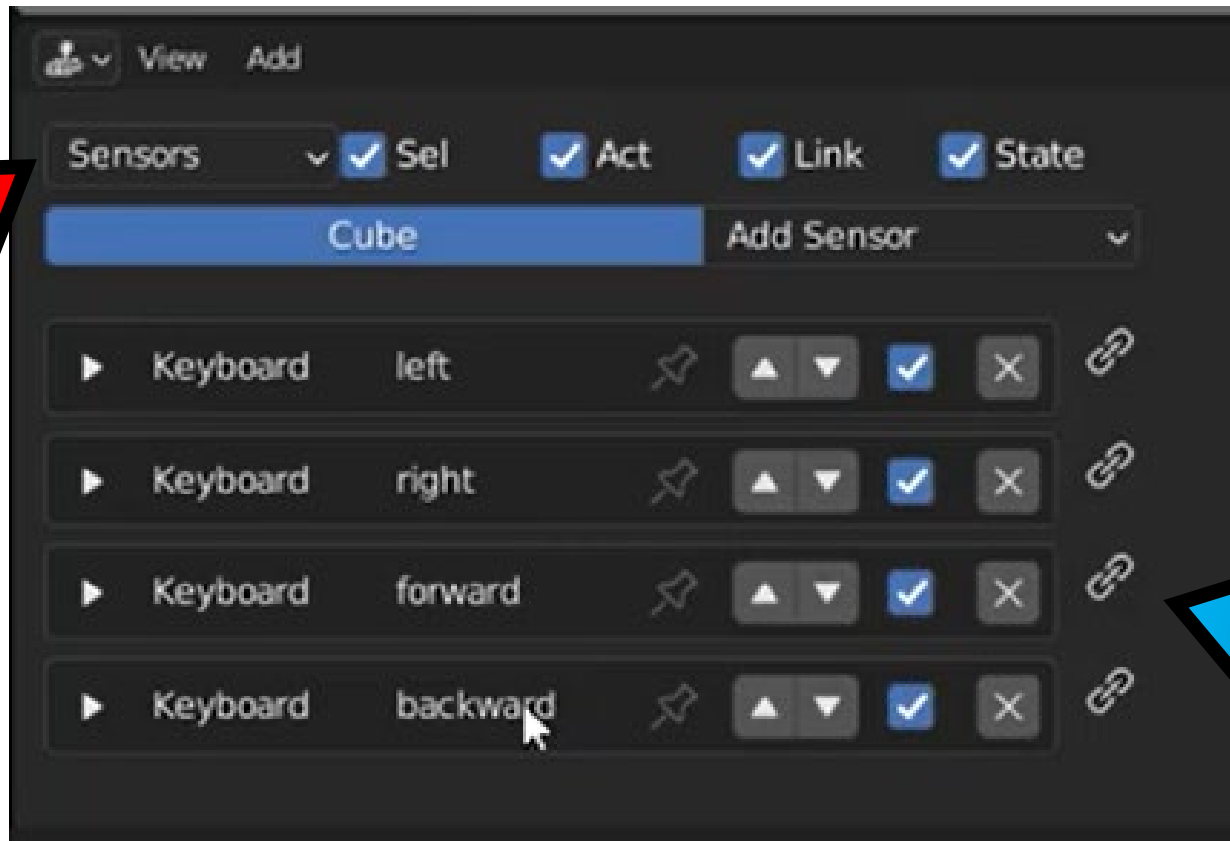


2024-1-PL01-KA220-VET-000243150

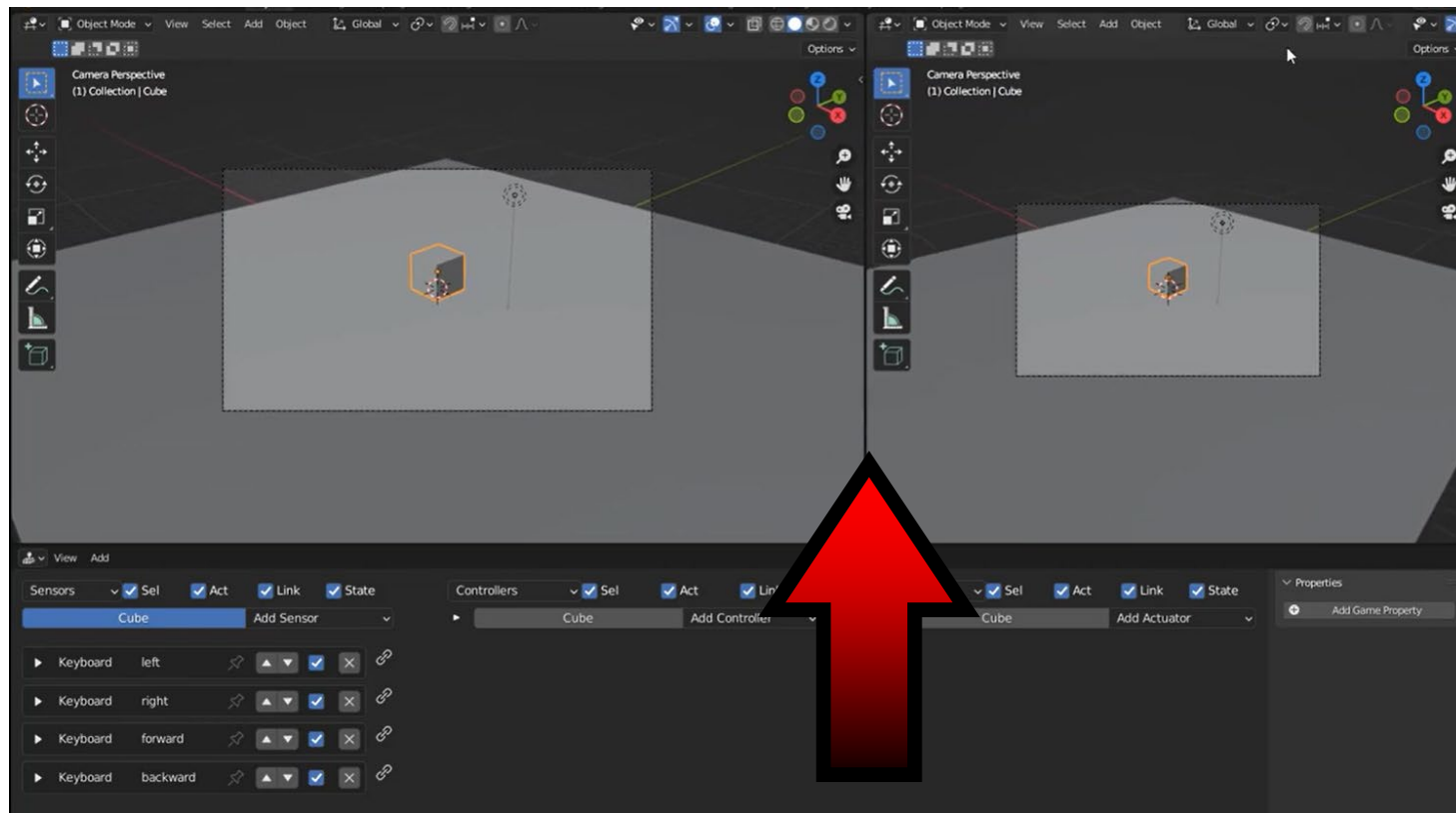
the European Union
co-funded by

2024-1-PL01-KA220-VET-000243150

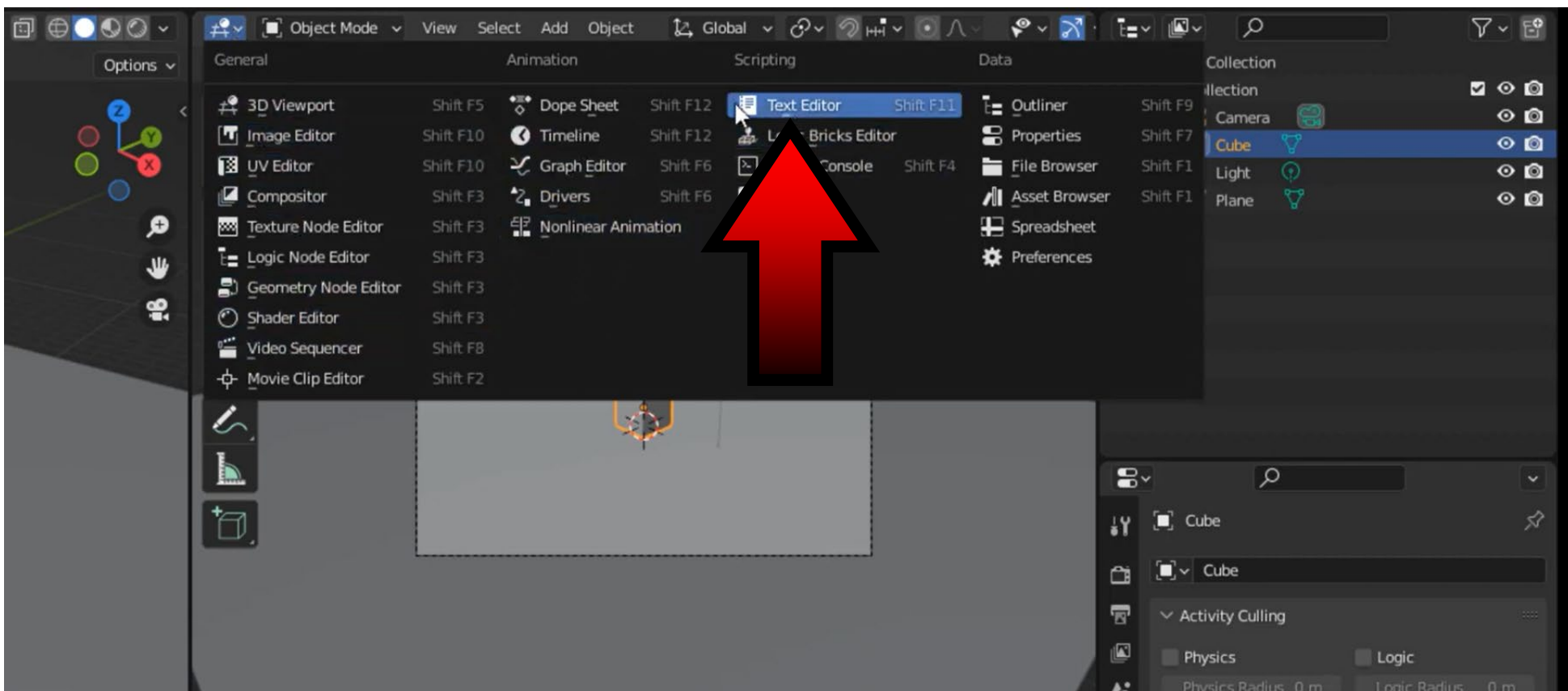
ZACZNIJ JAK WCZEŚNIEJ



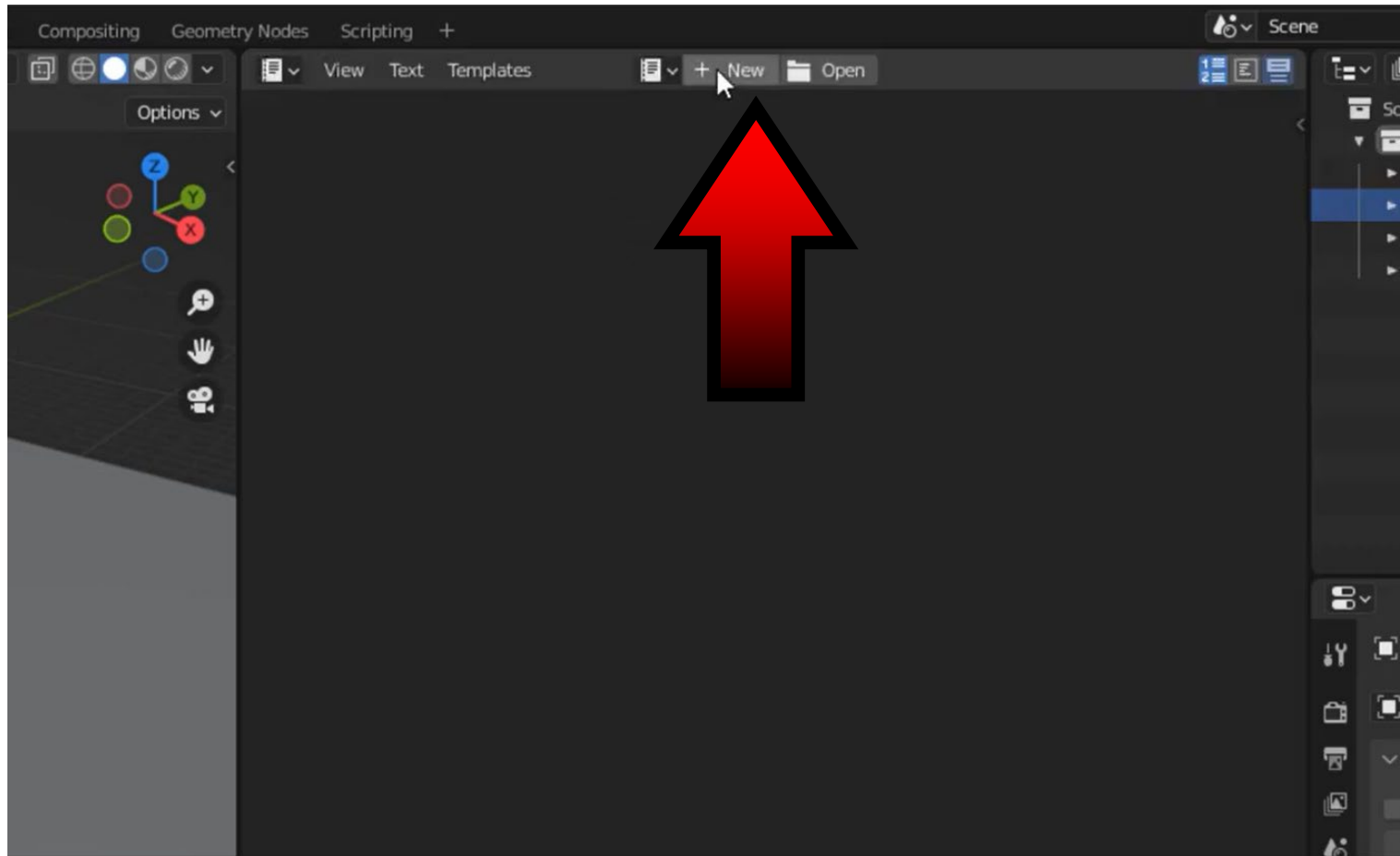
PODZIEL OKNO NA POŁOWĘ



W PRAWYM OKNIE WYBIERZ **TEXT EDITOR**

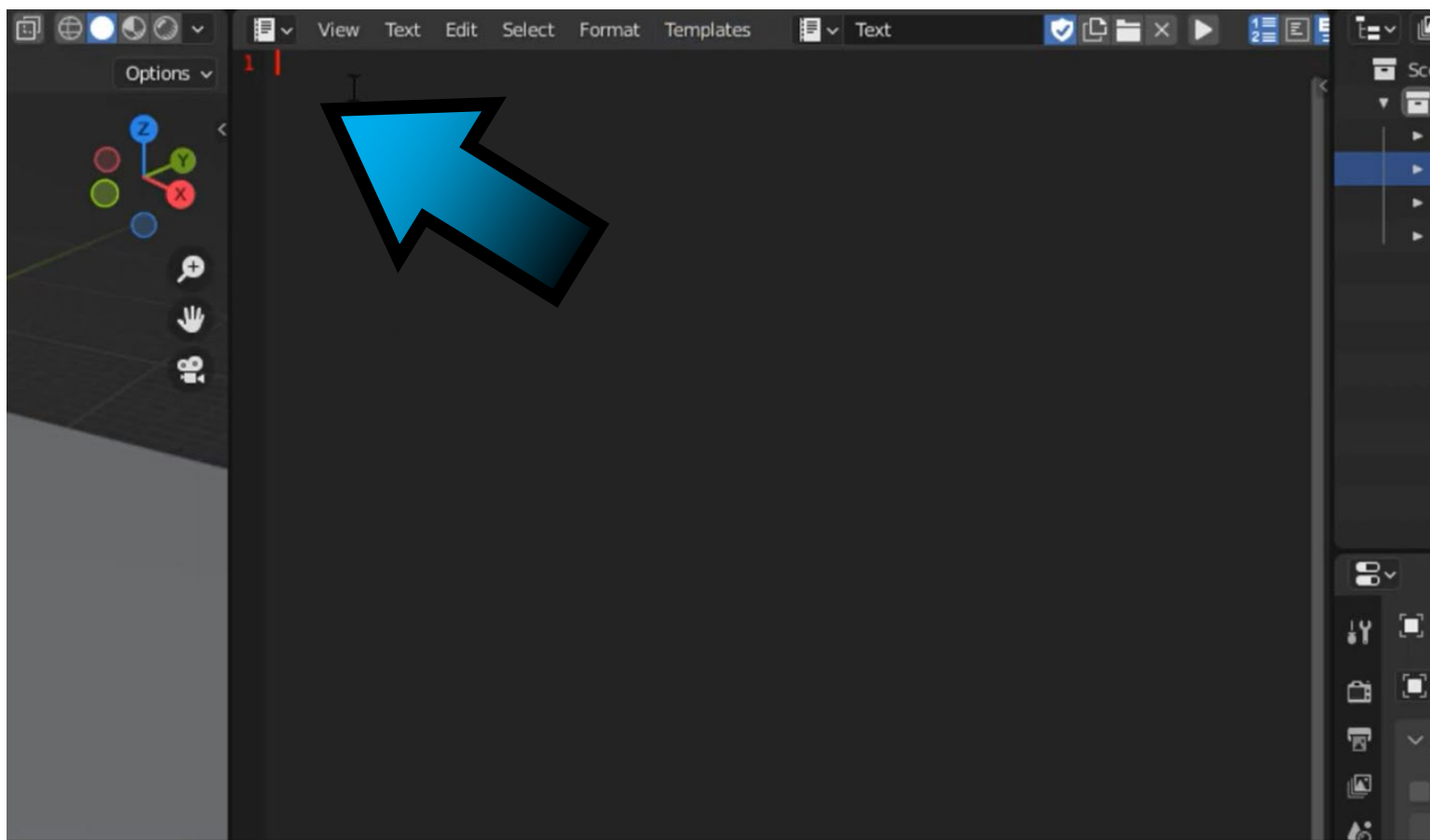


KLIKNIJ NA **NEW**

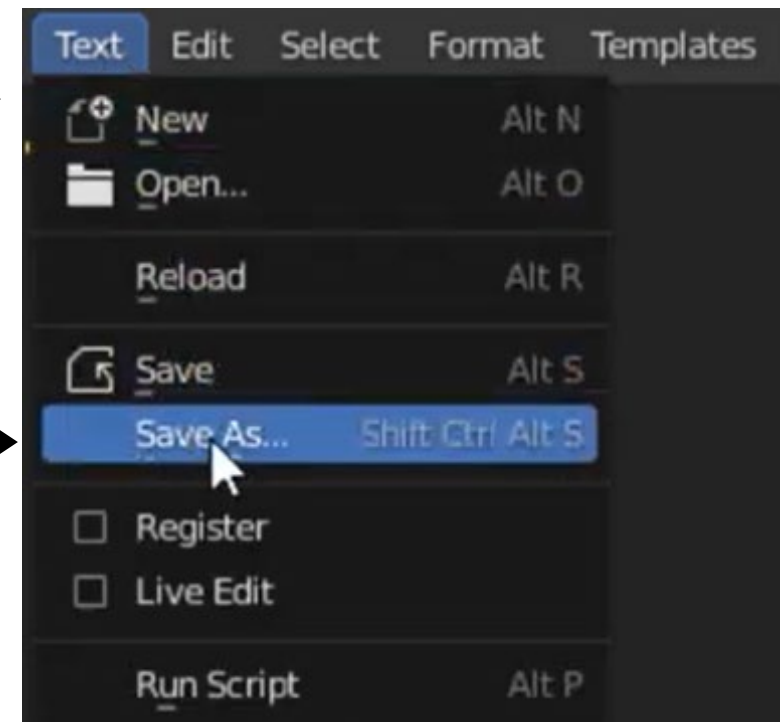
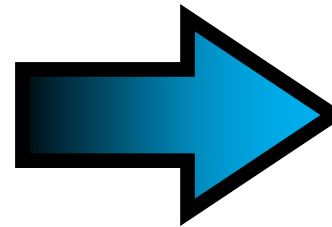
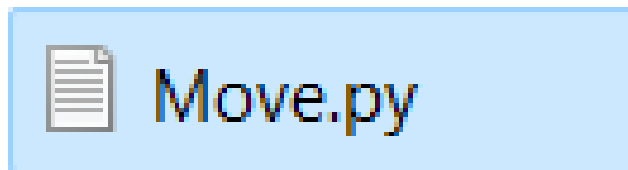
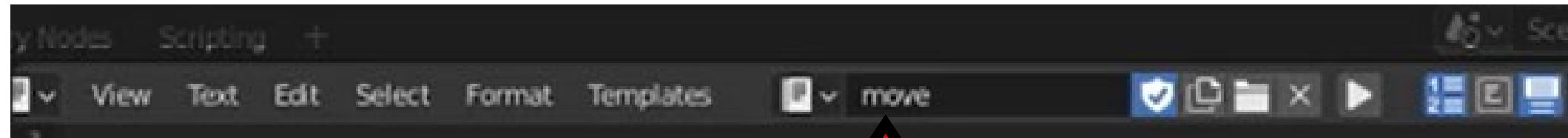


UPBGE

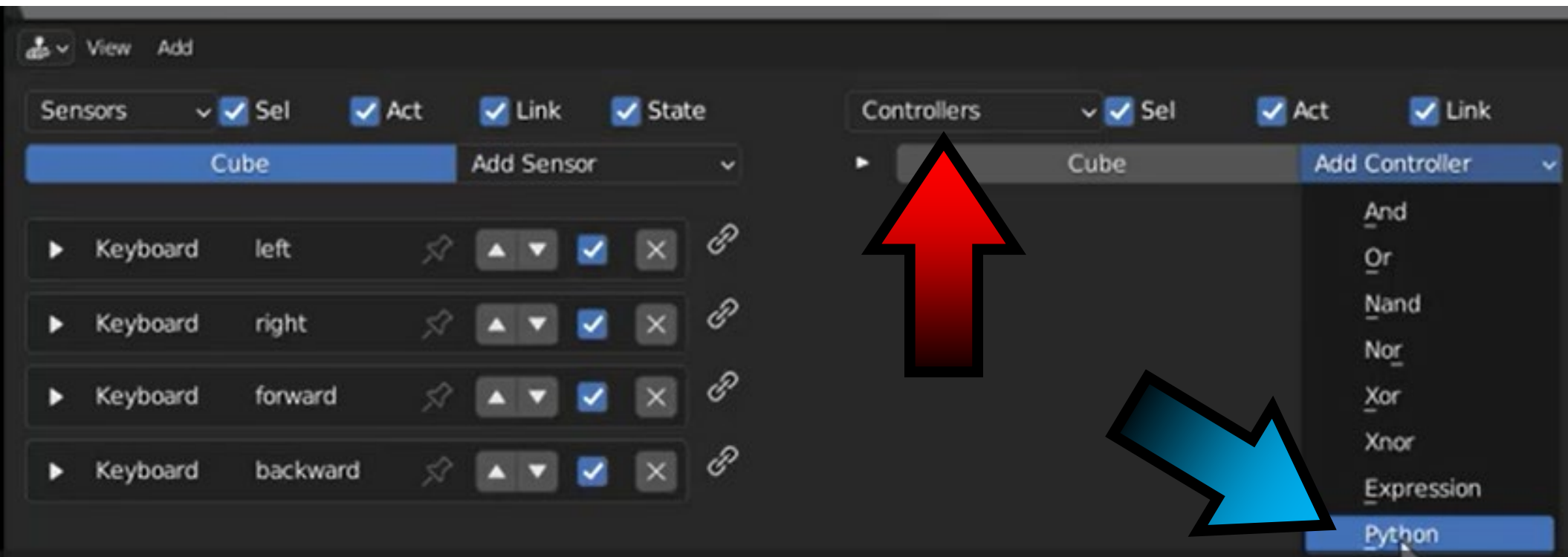
MAMY MOŻLIWOŚĆ **PISANIA SKRYPTÓW**



OKREŚL NAZWĘ I ZAPISZ SKRYPT

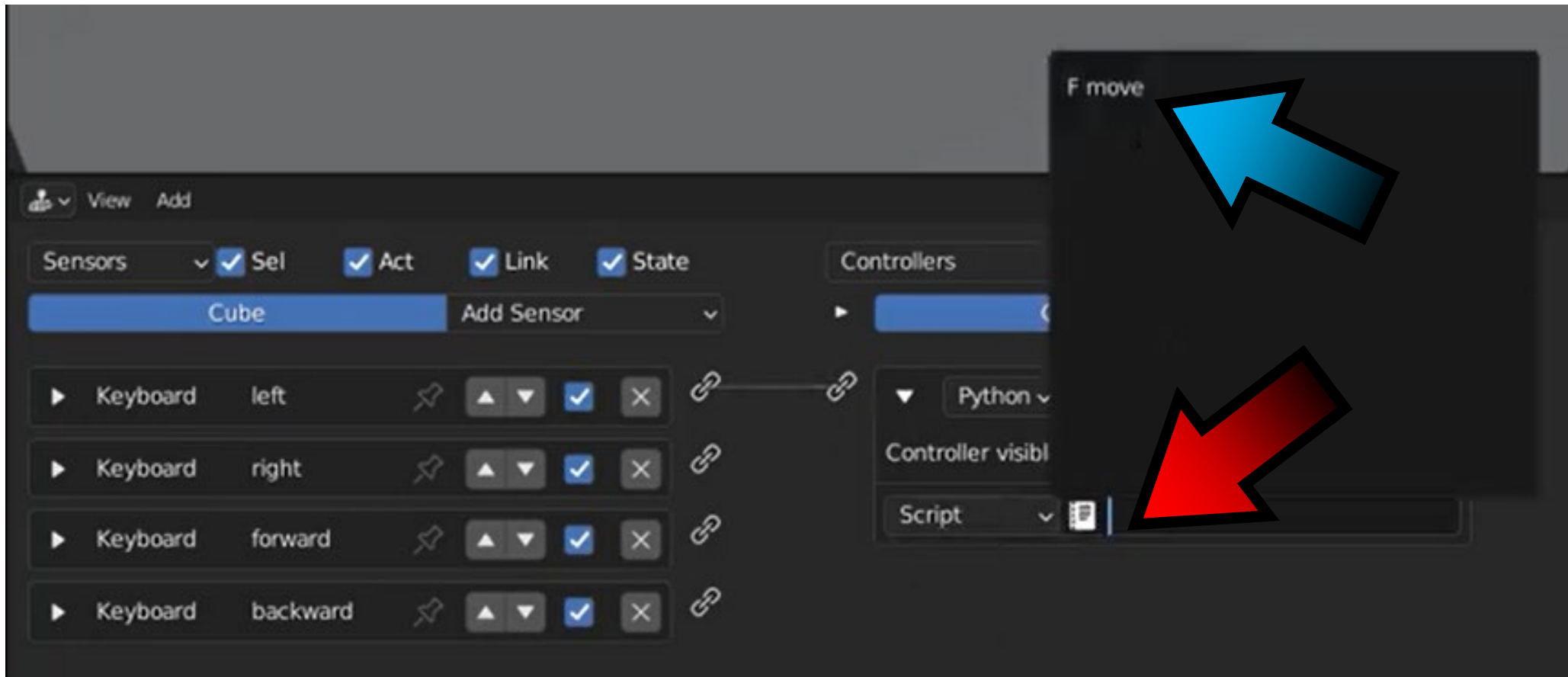


W **CONTROLLERS** WYBIERZ **PYTHON**

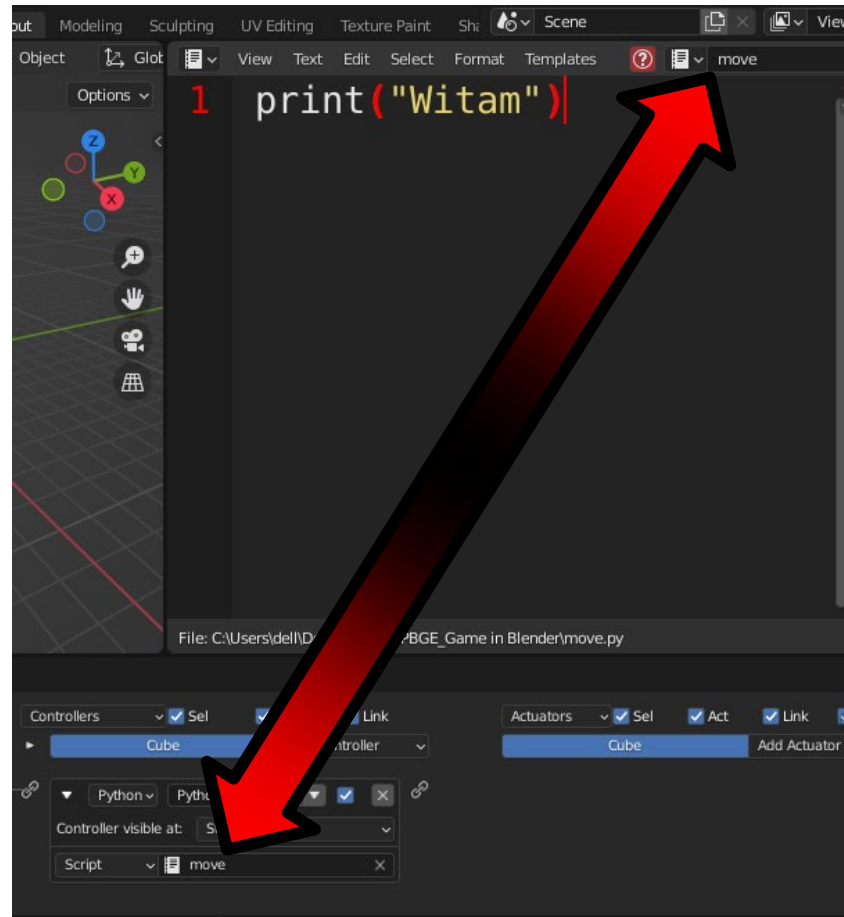


The screenshot shows a VR development interface with two main panels. The left panel is titled 'Sensors' and has a dropdown menu set to 'Cube'. It contains a list of sensors: 'Keyboard left', 'Keyboard right', 'Keyboard forward', and 'Keyboard backward'. Each sensor has a play button, a pin icon, a dropdown menu with up and down arrows, a checked checkbox, and an 'X' icon. The right panel is titled 'Controllers' and also has a dropdown menu set to 'Cube'. It contains a list of controller actions: '_And', '_Or', '_Nand', '_Nor', '_Xor', '_Xnor', '_Expression', and 'Python'. A red arrow points to the 'Cube' dropdown in the 'Controllers' panel, and a blue arrow points to the 'Python' option in the controller list.

ZAZNACZ LUB WPROWADŹ NAZWĘ



TE POZYCJE POWINNY BYĆ TAKIE SAME



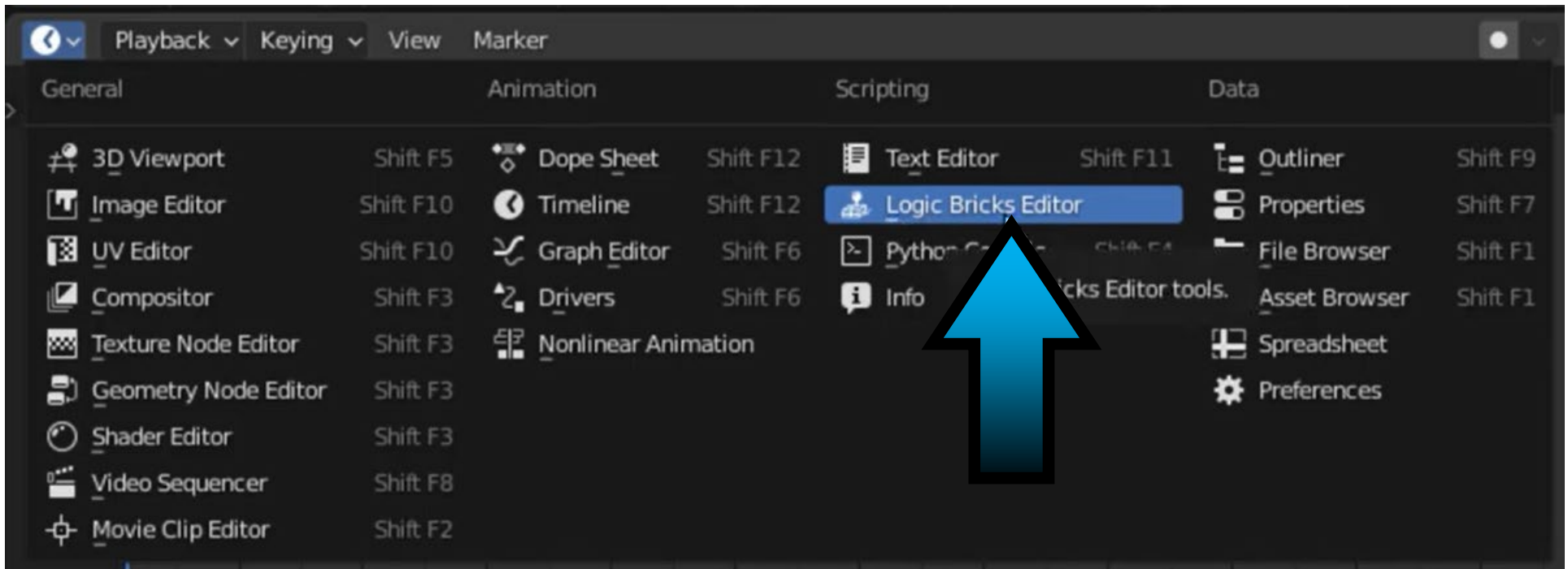


POWER OF AR AND VR



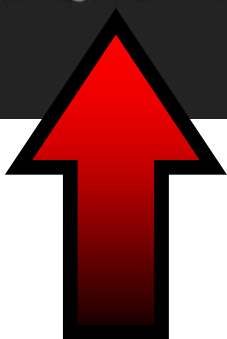
TO JUST IMPORT Z LOGIC BRICKS EDITOR

```
1  
2 from bge import logic
```

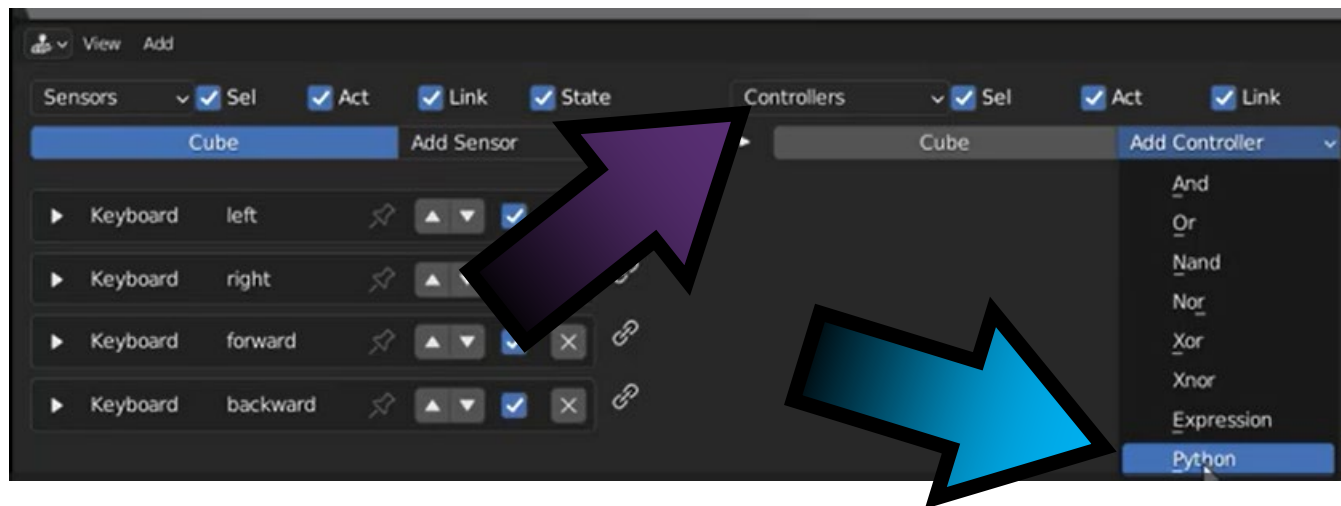


UPBGE

```
1  
2 from bge import logic  
3  
4 cont = logic.getCurrentController()
```



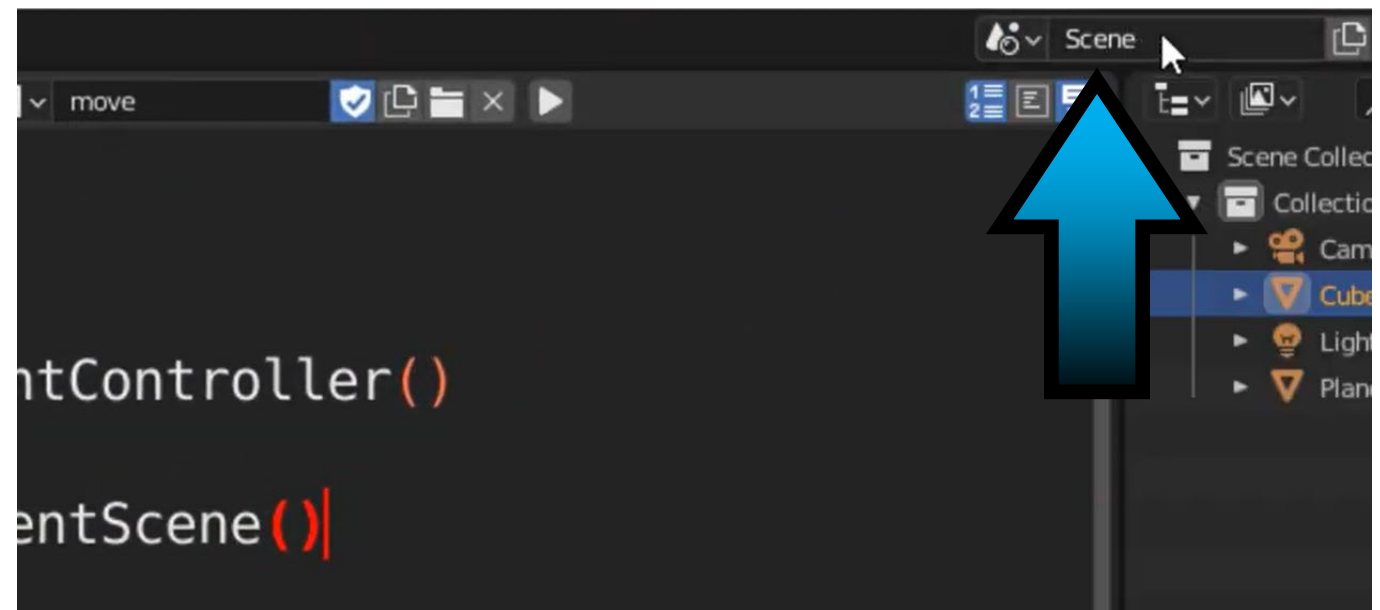
**TO JEST POBRANIE
USTAWIEŃ
Z CONTROLLER
KTÓRY JEST
SKRYPTEM PYTHON'A**



```
1  
2 from bge import logic  
3  
4 cont = logic.getCurrentController()  
5  
6 scene = logic.getCurrentScene()
```

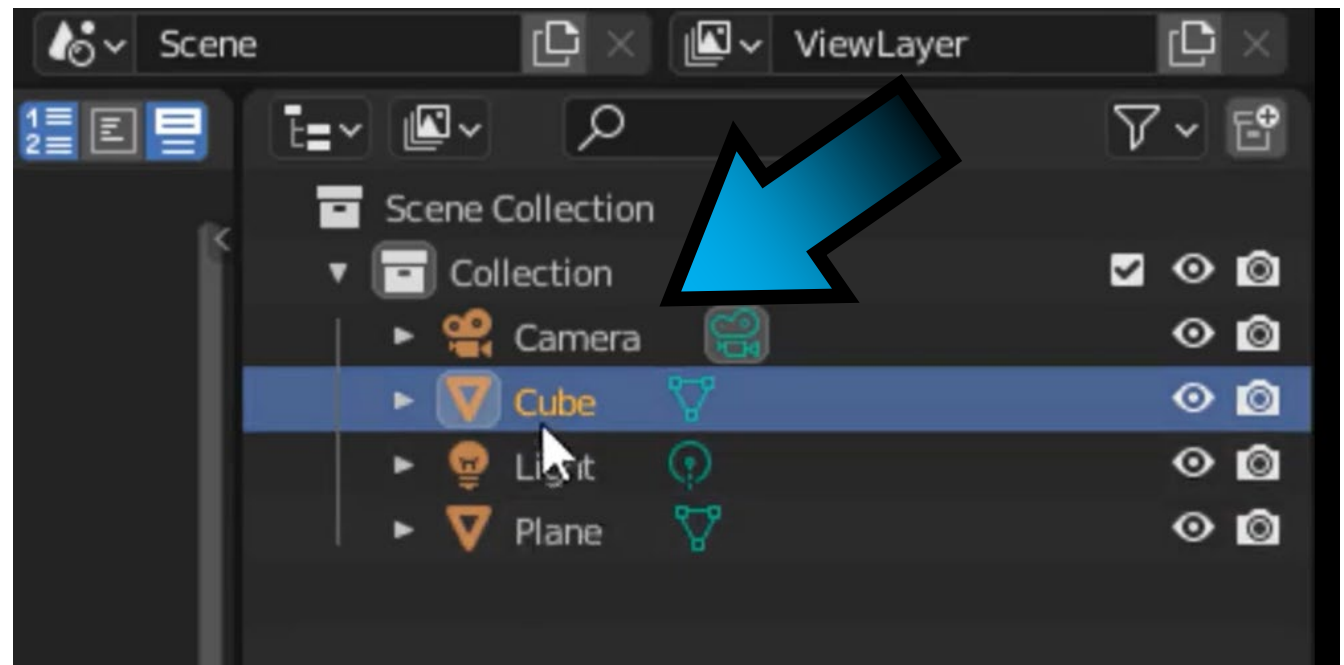
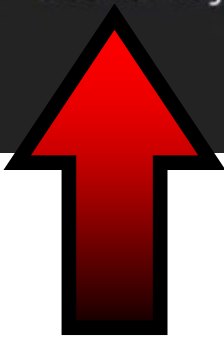


**TUTAJ
WSKAZUJEMY
SCENE**



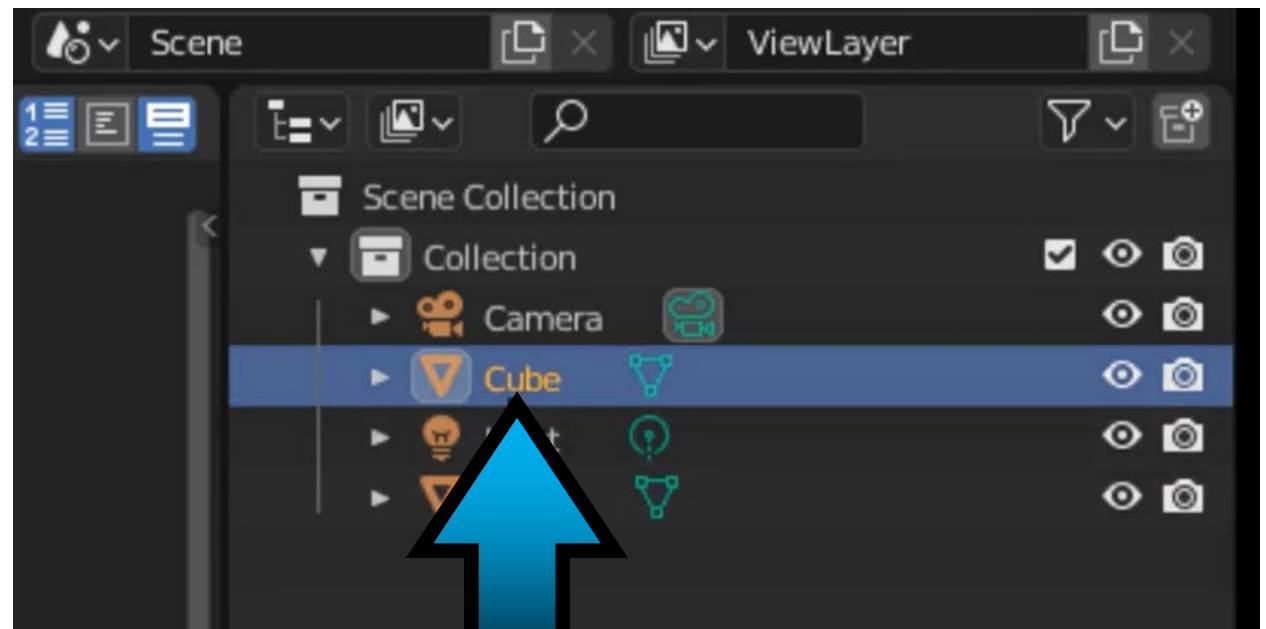
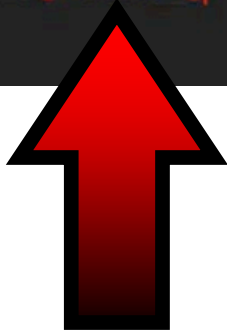
TUTAJ DEFINIUJEMY DOSTĘP DO OBJEKTÓW Z PRZYDZIELONEJ SCENY

```
1  
2 from bge import logic  
3  
4 cont = logic.getCurrentController()  
5  
6 scene = logic.getCurrentScene()  
7 objects = scene.objects  
8 |  
9
```



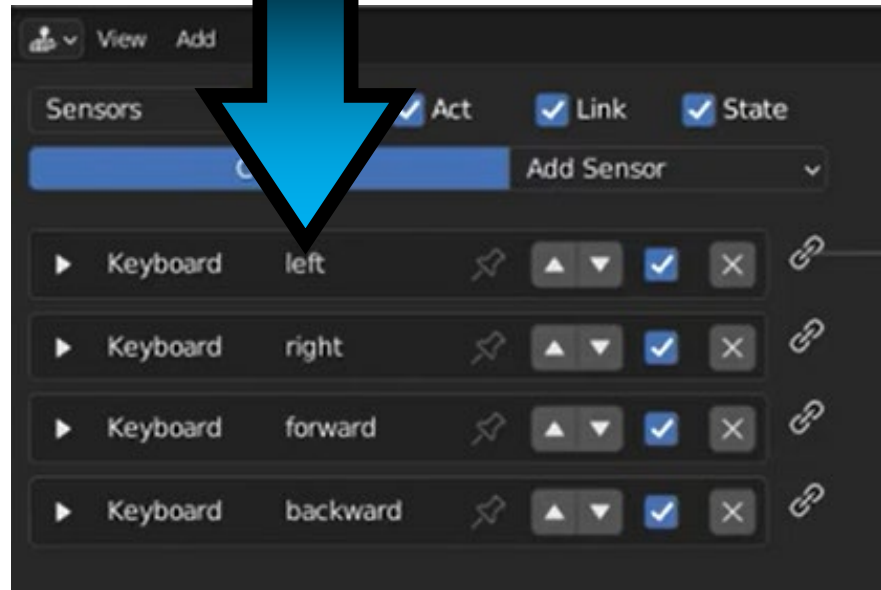
```
1  
2 from bge import logic  
3  
4 cont = logic.getCurrentController()  
5  
6 scene = logic.getCurrentScene()  
7 objects = scene.objects  
8  
9 cube = objects["Cube"]
```

PRZYDZIELENIE ZMIENNEJ DO NAZWY OBIEKTU

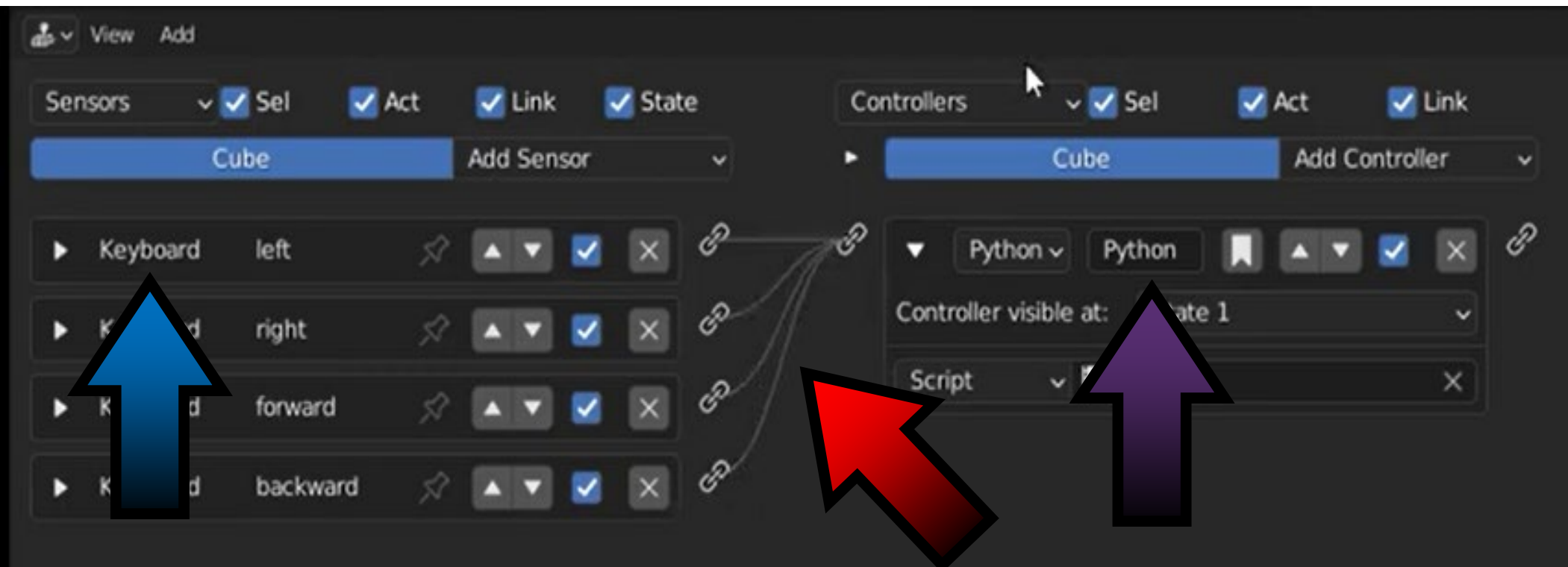


```
8  
9 cube = objects["Cube"]  
10  
11 left = cont.sensors['left']  
12 right = cont.sensors['right']  
13 forward = cont.sensors['forward']  
14 backward = cont.sensors['backward']
```

**PRZYDZIELENIE
ZMIENNEJ
DO
NAZWY
SENSORA**



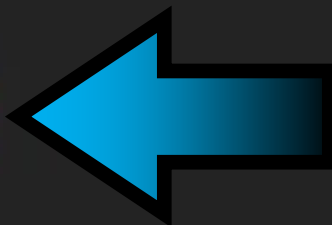

POŁĄCZENIE SENSORS Z CONTROLLERS



The screenshot displays a VR development interface with two main panels: 'Sensors' and 'Controllers'. Both panels are set to 'Cube' and have checkboxes for 'Sel', 'Act', and 'Link' checked. The 'Sensors' panel lists 'Keyboard' with sub-categories 'left', 'right', 'forward', and 'backward'. The 'Controllers' panel shows a 'Python' script. A red arrow points to the connection lines between the 'Keyboard' sensors and the 'Python' script. A blue arrow points to the 'Keyboard' sensor list, and a purple arrow points to the 'Python' script.



PRZESUNIĘCIE W LEWO

```
10
11 left = cont.sensors['left']
12 right = cont.sensors['right']
13 forward = cont.sensors['forward']
14 backward = cont.sensors['backward']
15
16 if left.positive:
17     cube.position.x -= 0.5
```





PRZESUNIĘCIE W PRAWO

```
15  
16 if left.positive:  
17     cube.position.x -= 0.5  
18  
19 if right.positive:  
20     cube.position.x += 0.5
```



PRZESUNIĘCIE DO PRZODU

```
15
16 if left.positive:
17     cube.position.x -= 0.5
18
19 if right.positive:
20     cube.position.x += 0.5
21
22 if forward.positive:
23     cube.position.y += 0.5
```

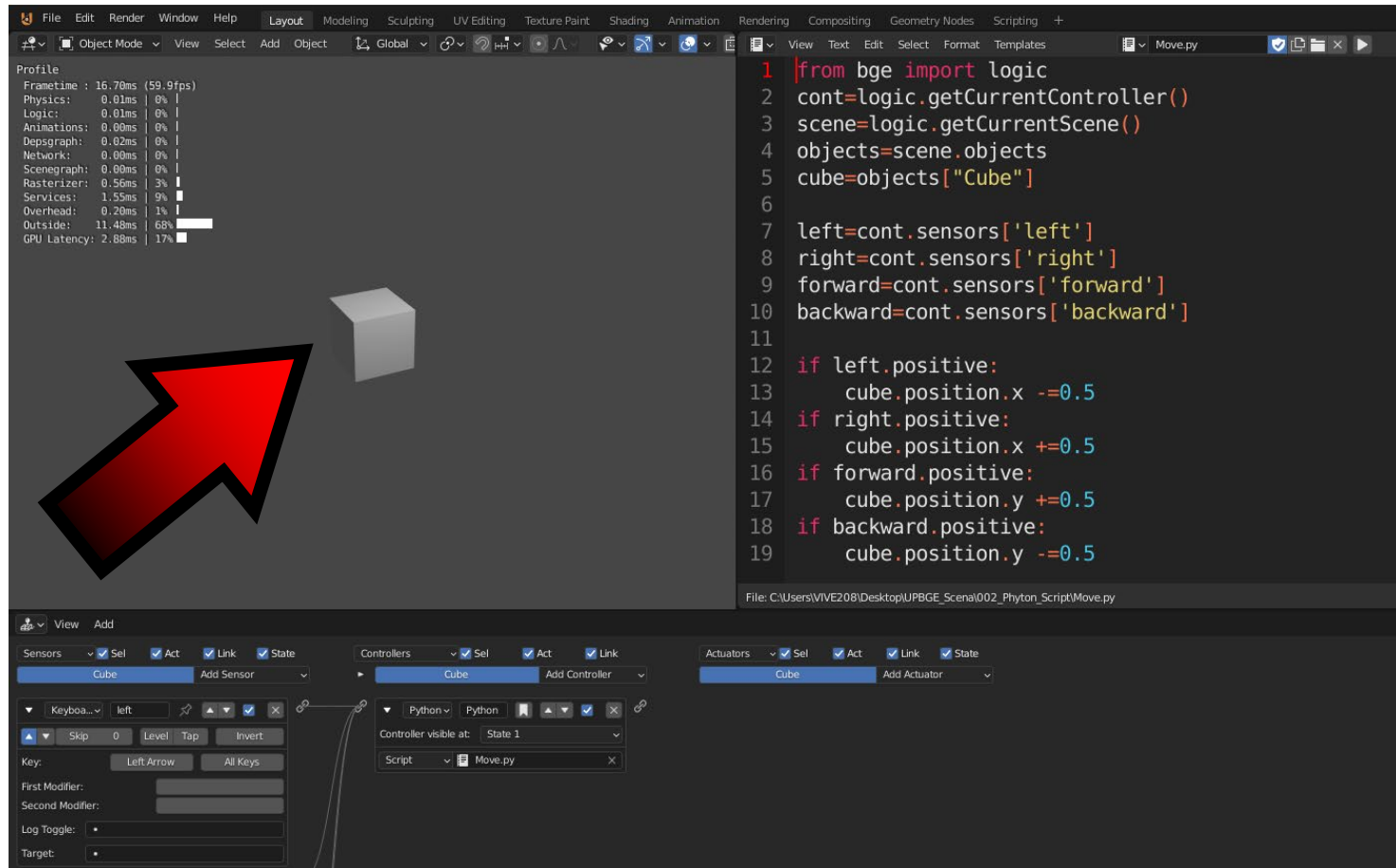


PRZESUNIĘCIE DO TYŁU

```
15  
16 if left.positive:  
17     cube.position.x -= 0.5  
18  
19 if right.positive:  
20     cube.position.x += 0.5  
21  
22 if forward.positive:  
23     cube.position.y += 0.5  
24  
25 if backward.positive:  
26     cube.position.y -= 0.5
```



SPRAWDŹ GRE URUCHAMIAJĄC KLAWISZEM P



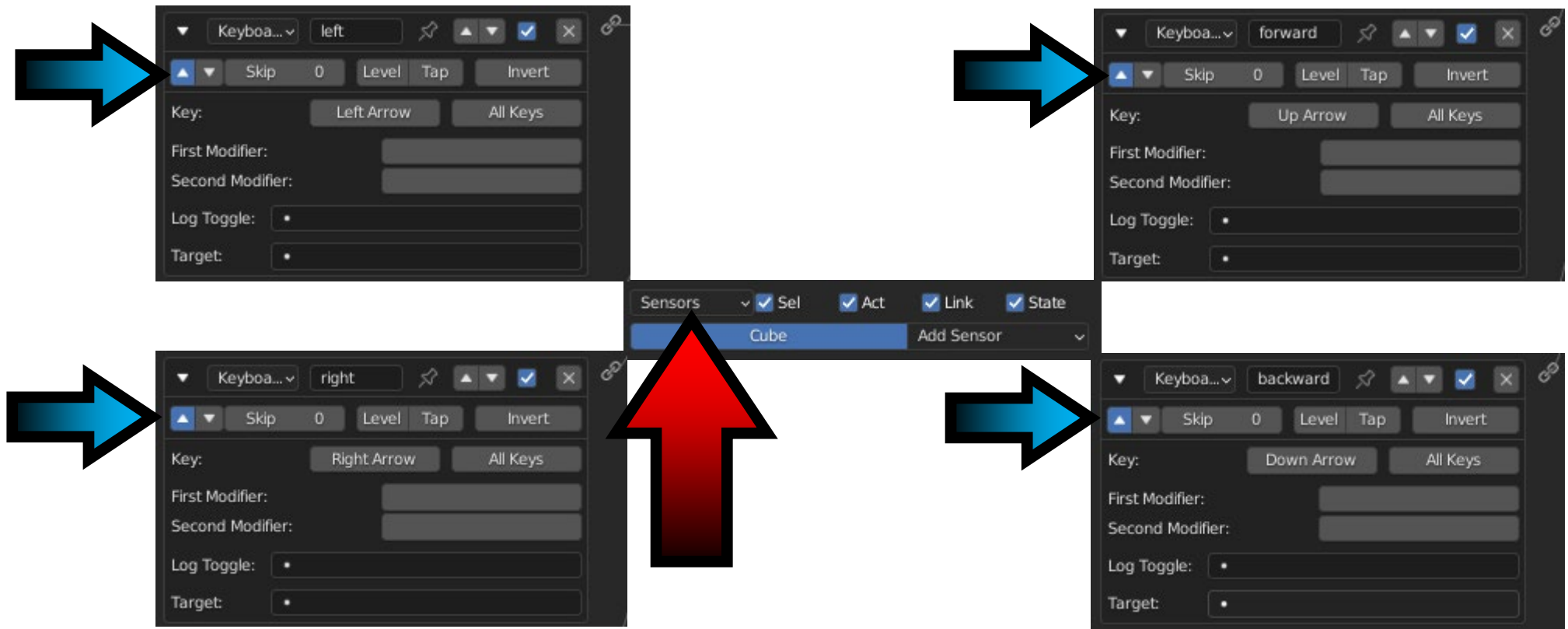
The screenshot displays the UPBGE engine interface. On the left, a 3D view shows a simple grey cube in a dark environment. A large, semi-transparent red arrow with a black outline points towards the cube. The top of the interface features a menu bar with options like File, Edit, Render, Window, Help, Layout, Modeling, Sculpting, UV Editing, Texture Paint, Shading, Animation, Rendering, Compositing, Geometry Nodes, and Scripting. Below the menu bar is a toolbar with various icons for navigation and editing. On the right side, a code editor window shows a Python script named 'Move.py' with the following code:

```
1 from bge import logic
2 cont=logic.getCurrentController()
3 scene=logic.getCurrentScene()
4 objects=scene.objects
5 cube=objects["Cube"]
6
7 left=cont.sensors['left']
8 right=cont.sensors['right']
9 forward=cont.sensors['forward']
10 backward=cont.sensors['backward']
11
12 if left.positive:
13     cube.position.x -=0.5
14 if right.positive:
15     cube.position.x +=0.5
16 if forward.positive:
17     cube.position.y +=0.5
18 if backward.positive:
19     cube.position.y -=0.5
```

Below the code editor, the logic editor is visible, showing a sensor (Keyboard) connected to a controller (Python) which is linked to an actuator (Cube). The sensor is configured with the key 'left' and the modifier 'Left Arrow'. The controller is set to 'Python' and the script 'Move.py'. The actuator is set to 'Cube'. The bottom of the interface shows a 'Profile' window with performance metrics:

Metric	Value	Percentage
Frametime	16.70ms	(59.9fps)
Physics	0.01ms	0%
Logic	0.01ms	0%
Animations	0.00ms	0%
Depsgraph	0.02ms	0%
Network	0.00ms	0%
Scenegraph	0.00ms	0%
Rasterizer	0.66ms	3%
Services	1.55ms	9%
Overhead	0.20ms	1%
Outside	11.48ms	68%
GPU Latency	2.88ms	17%

ABY WYKONAĆ CIĄGŁY RUCH DLA **SENSORS** WŁĄCZAMY **STRZAŁKI**



POWER OF AR AND VR

DZIĘKUJĘ ZA UWAGĘ



Co-funded by
the European Union



2024-1-PL01-KA220-VET-000243150

ημε Ενωθεν ηνιον
Co-funded by

2024-1-PL01-KA220-VET-000243150