

**PFE 10**

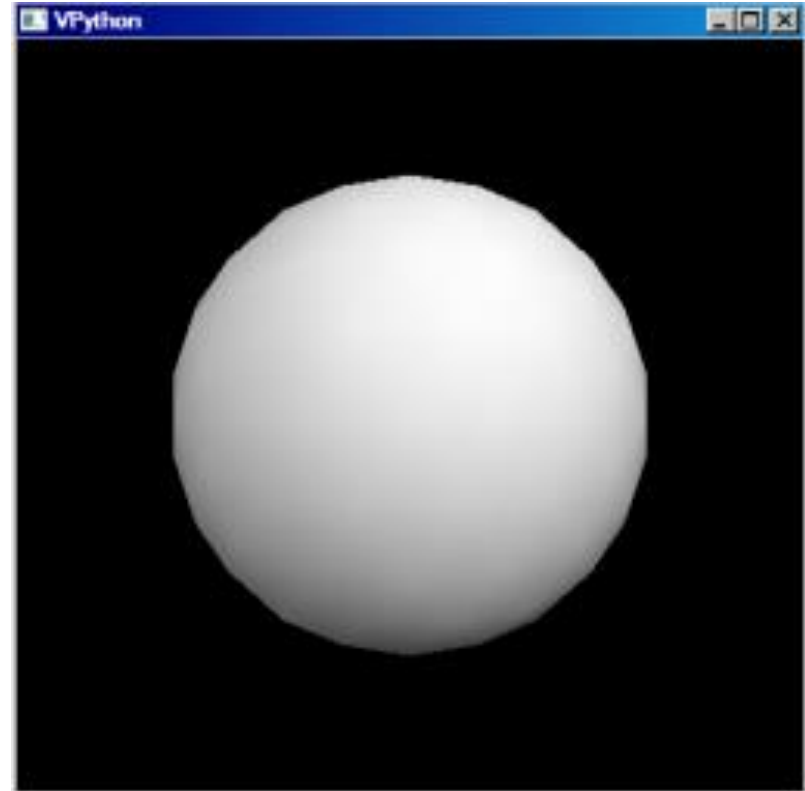
Palubicki

# VPython

- Python module for interactive 3D modeling
- Very easy to learn

# First steps

```
from visual import * #or...  
from vpython import *  
sphere()
```



# Changing sphere attributes

- Color
  - `sphere(color=color.red)`

# Changing sphere attributes

- Color
  - `sphere(color=color.red)`
- Radius
  - `sphere(radius=0.5,color=color.red)`

# Changing sphere attributes

- Color
  - `sphere(color=color.red)`
- Radius
  - `sphere(radius=0.5,color=color.red)`
- Name
  - `Sphere = sphere(radius=0.5,color=color.red)`

# Changing sphere attributes

- Color
  - `sphere(color=color.red)`
- Radius
  - `sphere(radius=0.5,color=color.red)`
- Name
  - `Sphere = sphere(radius=0.5,color=color.red)`
- Position
  - `Sphere = sphere(pos=(0,2,0),radius=0.5,color=color.red)`

# Changing sphere attributes

- Color
  - `sphere(color=color.red)`
- Radius
  - `sphere(radius=0.5,color=color.red)`
- Name
  - `Sphere = sphere(radius=0.5,color=color.red)`
- Position
  - `Sphere = sphere(pos=(0,2,0),radius=0.5,color=color.red)`
- Changing position
  - `Sphere.pos = (1,2,3)`



# Scene interaction

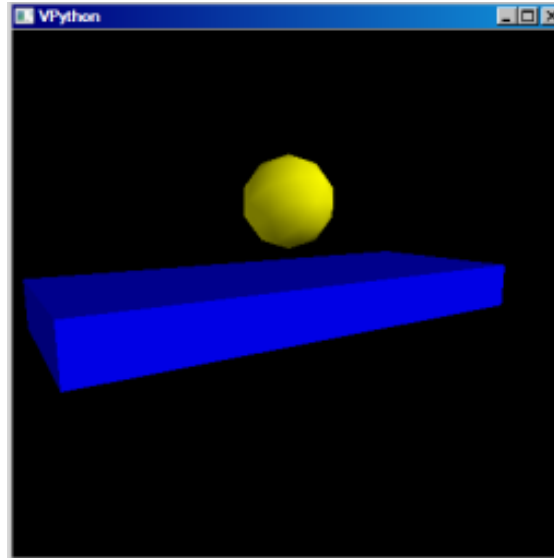
- Zoom
  - Hold middle mouse button
- Rotation
  - Hold left and right mouse button at once and move mouse

# Simple scene

```
from visual import *
```

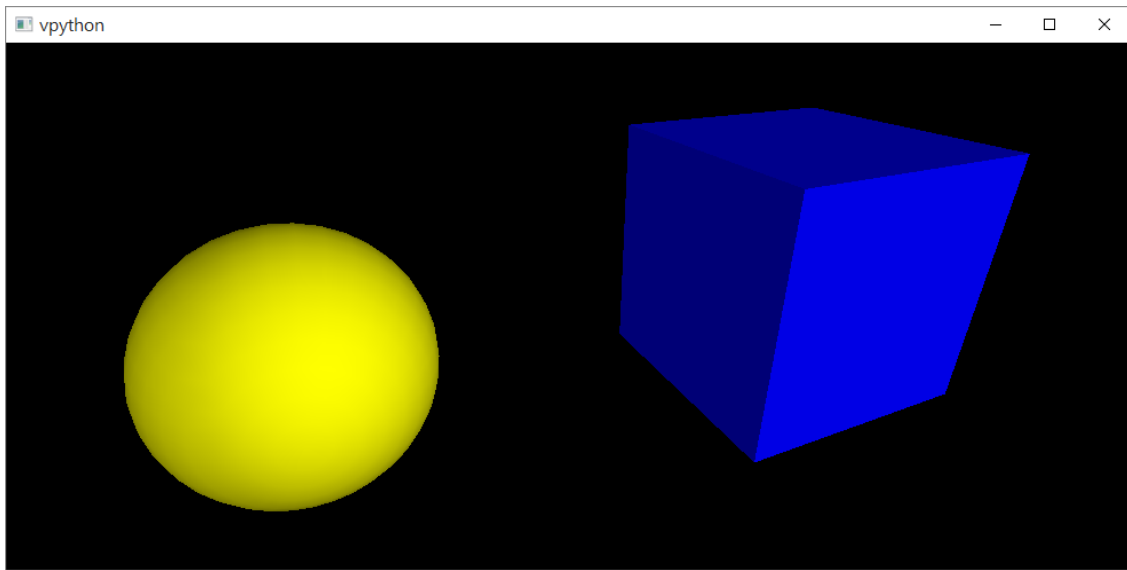
```
Sphere = sphere(pos=(0,2,0),color=color.yellow,radius=1)
```

```
Box = box(length=10, height=2, width=4,color=color.blue)
```

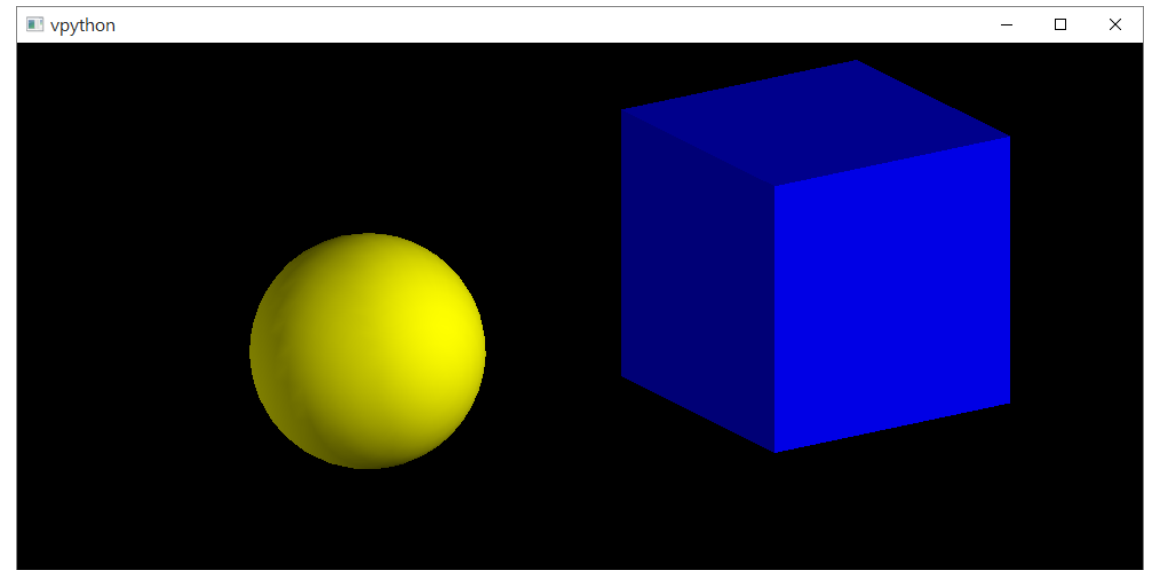


# Display()

- `scene = display(title='Scene', width=500, height=500, range=(5, 5, 5))`



`scene.fov(1.5)`



`scene.fov(0.001)`

# Simulation

Using a loop e.g.:

```
l=1
```

```
while l == 1:
```

```
...
```

```
    #update object positions for each time step
```

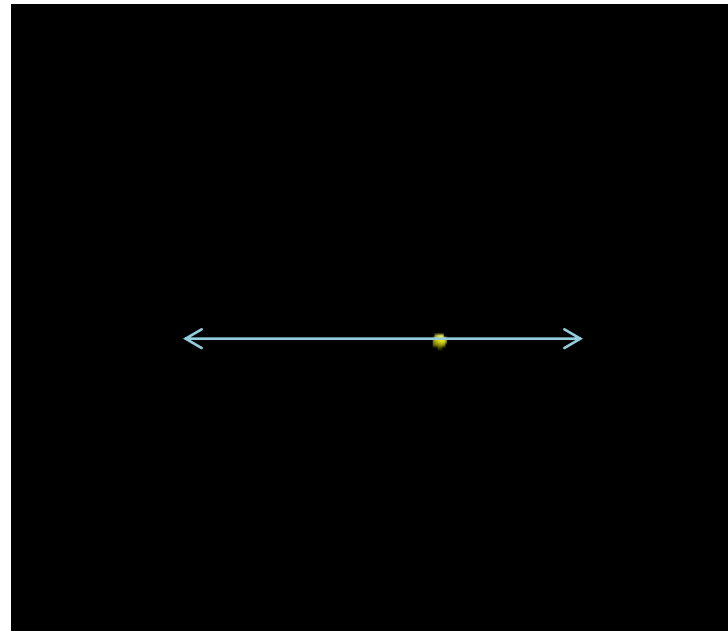


# Exercise

- Read the tutorial on simulating simple pendulum motion <http://techforcurious.website/simulation-of-pendulum-vpython-tutorial-visual-python/>.
- Plot the angle of the pendulum as a function of time by solving the differential equation with the Forward Euler Method. Test out different time steps, what do you observe?
- Reimplement the 3D pendulum simulation using vpython by following the tutorial.

# Exercise

- Create a temporal simulation of a ball moving on a horizontal line from one side of the screen to the other. Formulate appropriate differential equations describing the rate of change of position for the ball and solve them. Use the solution to update ball positions in each time step.



# Exercise

- Reimplement the activator-inhibitor model using vpython. Visualize the 2D grid as a planar arrangement of colored spheres or quads, where the color indicates the concentration of either activator or inhibitor. In contrast to your previous implementation without vpython visualize each time step.

