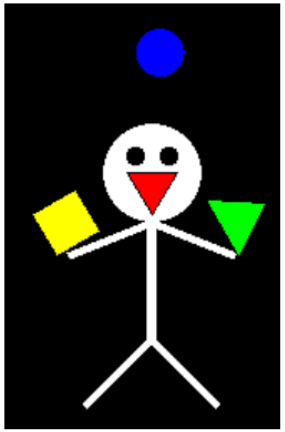
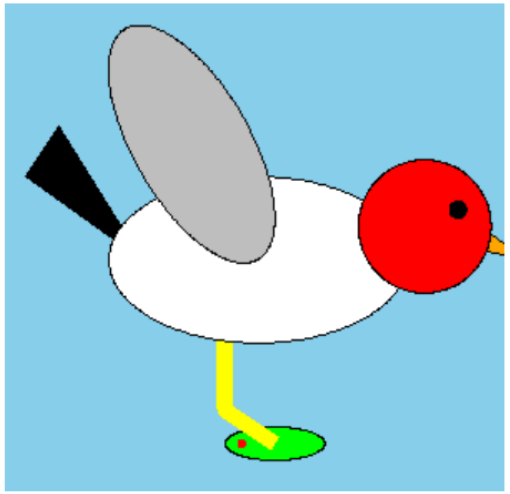
Week 10

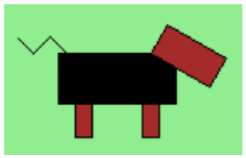
Objects – Animation

Upgrade your wonderful animal!

Recall the animal you created!







- But this time, make it much more elegant
- Use Python objects so that you can make it like **Hubo** in **cs1robots**.

Making the animal object

- It should have **parameters for the constructor**
  - E.g.) initial position, colors, size, options, etc.

```
mario=Mario('Blue', 'normal')
```

- It must have methods that can be used to **move it on the canvas**, and to **move its body parts**.
  - You must not use the graphical objects outside the object

- It must have at least one **event method** which can change the animal's shape or do the special action
  - E.g.) becoming super-mario, laying eggs, transforming etc.

Example – Supermario

```
from cs1graphics import *
from time import sleep

_scene = None
_world = None

def create_world():
    global _scene, _world
    if _scene:
        raise RuntimeError("A world already exists!")
    _world = _World(500, 300)
    _scene = Canvas(_world.width, _world.height)
    _scene.setTitle("Mario World")
    _world.draw_scene()

class _World(object):
    def __init__(self, width, height):
        self.width = width
        self.height = height

    def draw_scene(self):
        """
        draw background here
        Don't forget _scene.add(name)
        """
        pass
```

Name for the **Canvas** we will use

Background
Initialize your own background

Example – Supermario

```
COLOR = ['Red', 'Blue']
TYPE = ['super', 'normal']
class Mario (object):
    def __init__(self, color, type):
        assert type in TYPE and color in COLOR
        self.color = color
        self.type = type
        self.layer = Layer()
                :
                :
        _scene.add(self.layer)
```

Object drawing code should be here

Don't forget!

If you want to insert any **moving object** in the scene, you can make another **class** for that, e.g.

```
class Mushroom (object):
    def __init__(self, x, y):
        mushroom = Layer()
                :
                :
                :
                :
```

Example – Supermario

- Let's see how it works!

```
>>> create_world()
>>> mario = Mario('Blue', 'normal')
>>> mushroom = Mushroom(200, 92)
>>> mario.shoe_move()
>>> mario.shoe_move()
>>> mario.move(10,0)
>>> mario.move(-10,0)
```

- I can't see the mushroom. Where is it?

```
>>> mushroom = Mushroom(200,92)
>>> mushroom.arise()
```

- What if we want to cheat, starting as the super-mario?

```
>>> mario = Mario('Red', 'super')
```

Example – Supermario

- Let's make the animation using drawable objects

```
create_world()
mario = Mario('Blue', 'normal')
mushroom = Mushroom(200, 92)

t = 0.5

sleep(t)
mario.move(0, -50)
mushroom.arise()

sleep(t)
mario.move(0, 50)
mushroom.move(0,8)

sleep(t)
mushroom.move(20, 0)
mario.move(30,0)
mario.shoe_move()

sleep(t)
mushroom.move(20, 0)
mario.move(20, 0)
mario.shoe_move()

sleep(t)
mushroom.move(20, 0)
mario.move(20, 0)
mario.shoe_move()

sleep(t)
mushroom.move(20, 0)
mario.move(10, 0)
mario.shoe_move()

sleep(t)
mushroom.move(0,100)
...
```

Example – Supermario

- Make it interactive!
- We want to make kind of game using keyboard input:

```
def interactive_example():
        while True:
                e = _scene.wait()
                d = e.getDescription()
                if d == “keyboard”:
                        k = e.getKey()
                        if k == “q”:
                                _scene.close()
                                break
                        elif k == “w”:
                                mario.walk(20)
                        elif k == “r”:
                                mario.walk(40)
                        elif k == “j”:
                                mario.move(0, -50)
                                sleep(t)
                                mario.move(0, 50)
```

Summary

- Using a template in the elice platform
  - Make your own animal (or anything) object which provides following method
    - Moving in the canvas (like `mario.move(10, 20)`)
    - Moving the part of body (like `mario.shoe_move()`)
    - Special function (like `mario.supermario()`)
  - It should have parameters for the constructor and you must not use the graphical objects outside the object
  - Make your own background scene
    - (in '`draw_scene`' method of '`_World`' object)
  - Make animation scenario
  - Make interactive program

questions?

Be creative! Have fun!