

Python OOP Intro

A **module**, also known as a **namespace**, is a Python file that contains a collection of variable names, known as attributes. It's like a library of tools you can access by importing the .py file.

A **class** is a collection of related **objects**, including **attributes** (like characteristics) and **methods** (like actions).



```
class Wizard:
    name = "Defaultanor" # set attribute, 'name' for the class 'Wizard'
    creature = "Human Wizard"
    attire = ["Cape", "Robes", "Staff"]

    def change_name(self, new_name): # note that the first argument is self
        self.name = new_name # access the class attribute with the self keyword

player1 = Wizard() # instantiate the class

print(player1.name) # print the current object name

player1.change_name("Novador") # change the name using the change_name method
print(player1.name)
```

You can also define attributes at runtime using the `init` method

```
class Wizard:

    def __init__(self, name):
        self.name = name

    def change_name(self, new_name):
```

```

    self.name = new_name

# two variables are instantiated
naban = Wizard("Naban")
jodah = Wizard("Jodah")

# print the names of the two variables
print(naban.name)
print(jodah.name)

```

Maybe you want to define a number of attributes at once:

```

class Wizard:

    def __init__(self, name, talent, cost_white, cost_red, cost_black, cost_blue,
cost_green, cost_gen, power, toughness):
        self.name = name
        self.cost = [cost_white, cost_red, cost_black, cost_blue, cost_green,
cost_gen]
        self.talent = talent
        self.power = power
        self.toughness = toughness

# two variables are instantiated
naban = Wizard("Naban", "Looping", 0, 0, 0, 1, 0, 1, 1, 2)
jodah = Wizard("Jodah", "Flying", 1, 1, 0, 1, 0, 1, 4, 3)

print(naban.name, naban.talent, naban.cost)
print(jodah.name, jodah.talent, jodah.cost)

time.sleep(5) #going to need to import the time module to use this

print("FIGHT!")

time.sleep(2)
print("BIFF!      ", "\U0001F4A5")

time.sleep(2)
print("BAM!      ", "\U0001F4A5")

time.sleep(2)
print("\U0001F4A5", "      POW!")

time.sleep(5)

if (naban.toughness <= jodah.power):
    print("Jodah Wins")
else:
    print("Naban Wins")

```

Now add one of these wizards to the battle. How would you stage a wizard fight?



Your turn. Make another class for these characters:



Wizard Replica

3

**Artifact Creature — Wizard**

Flying

●, Sacrifice Wizard Replica: Counter target spell unless its controller pays 2.

It responds with unnatural precision.

1/3

Carl Critchlow

TM & © 1993–2003 Wizards of the Coast, Inc. 275/306