



Object Oriented Programming (OPP)



Diego Useche - dh.useche@uniandes.edu.co

Metodos Computacionales I

Physics Department, Universidad de los Andes, Bogotá



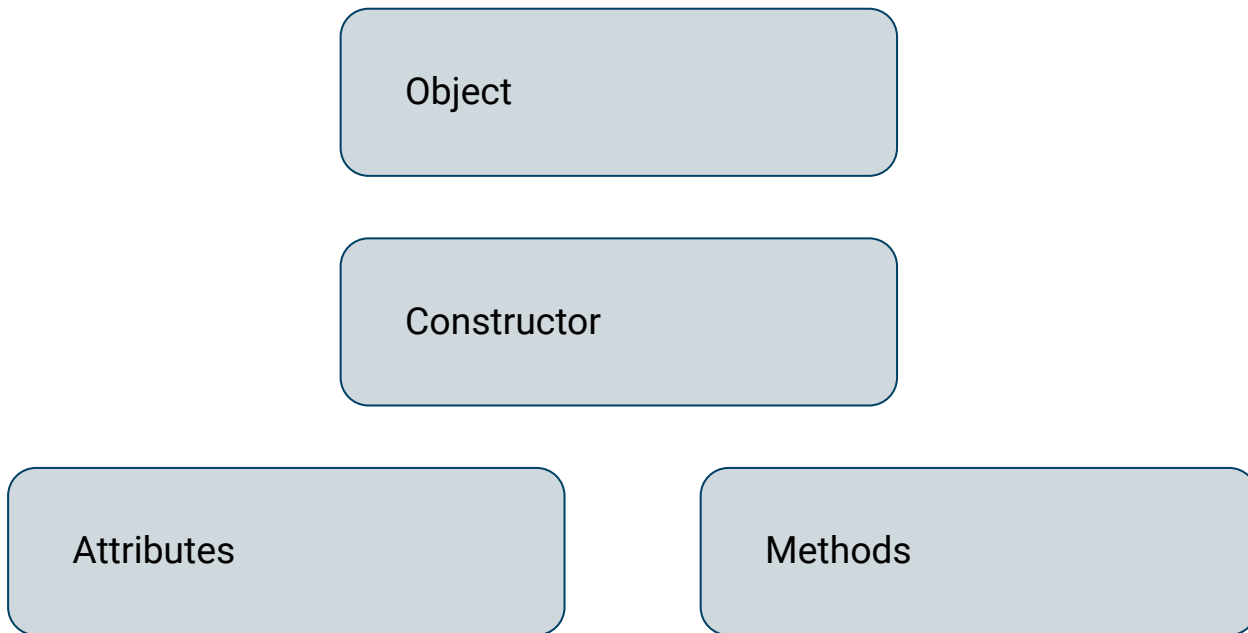
Paradigma OOP

- Python permite programar orientado a funciones y orientado a objetos.



OOP Paradigm

- Estructura General



Example (Constructor and Atributtes):

```
class Student:
    def __init__(self, name, major, id):
        self.name = name
        self.major = major
        self.id = id
```



Constructor, with
attributes

- Constructor Runs Automatically when the object is created
- self, indicates the object itself

Example (Methods, Getters, and Setters):

```
class Student:
    def __init__(self, name, major, id):
        self.name = name
        self.major = major
        self.id = id
```

```
def get_major(self):
    return self.major
```

```
def set_major(self, new_major):
    self.major = new_major
```



- Methods: Getters and Setters
- The constructor is a method

Example (Instances):

```
Maria = Student("Maria", "Filosofia", "1234")
```

```
Maria.get_major()
```

```
'Filosofia'
```



Instance 1

```
Dario = Student("Dario", "Administracion", "3456")
```

```
Dario.set_major("Literatura")
```

```
Dario.get_major()
```

```
'Literatura'
```



Instance 2

Paradigma OOP

- Instances of the class **Androide**



References

<https://www.cs.mtsu.edu/~xyang/2170/datatypes.html>

<https://www.sololearn.com/>

<https://www.hackerrank.com/>

<https://data-flair.training/blogs/c-tutorial/>

<https://www.astateofdata.com/python-programming/can-python-be-compiled/>