

# Object Oriented Programming

June Sallou, [june.benvegnu-sallou@irisa.fr](mailto:june.benvegnu-sallou@irisa.fr)

<https://github.com/Jnsl/ModelisationScientifique>



A blue line-art illustration of a rocket ship with a small flame at its base, positioned to the left of a planet with a prominent ring system. Two small five-pointed stars are scattered around the planet. The entire scene is set against a light blue grid background.

# Python is an OOP language

-

-

-

**How to respect those concepts  
technically ?**

# The class

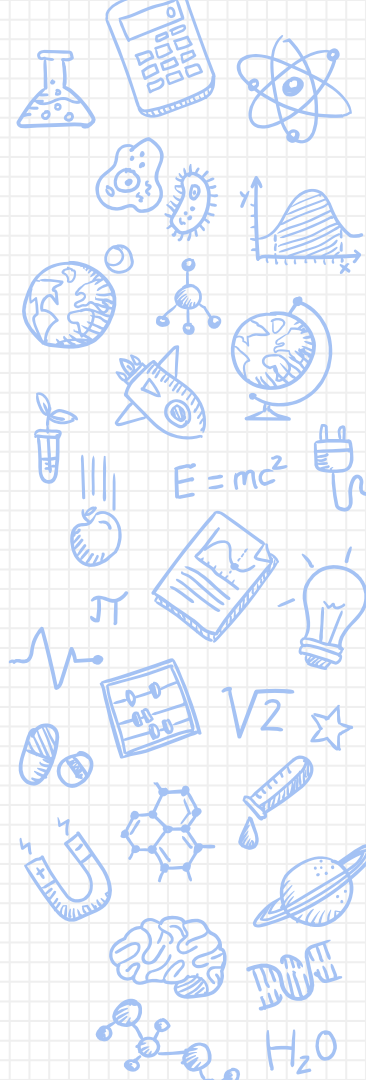
---

A class describes a set of objects that have the same data types and the same types of operations.

Objects of the same class have their own data.

```
class Person:
```

```
    pass
```



All classes have a function called `__init__()`, which is always executed when the class is being initiated.

Use the `__init__()` function to assign values to object properties, or other operations that are necessary to do when the object is being created.



## The class: `__Init__()`

```
def init (self, fname, lname):
```

```
self.firstname = fname
```

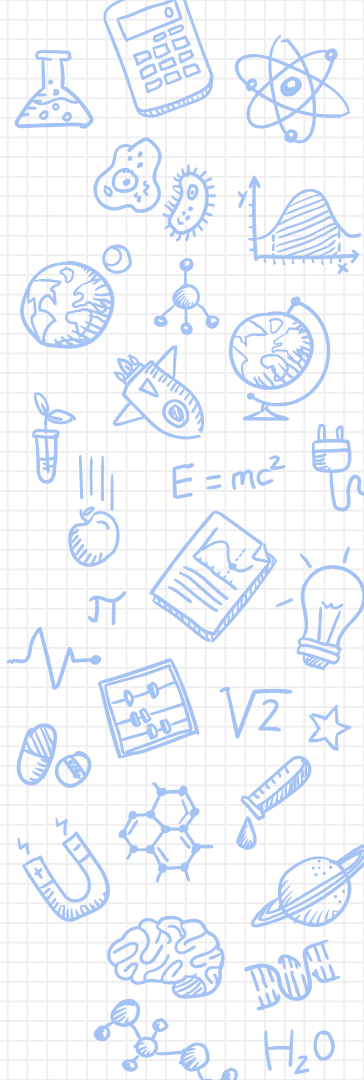
```
self.lastname = lname
```

# The class: methods

---

The methods correspond to the operations the objects belonging to the class can carry out.

```
def printname(self):  
    print("My name is",  
self.firstname, self.lastname +  
    ".")
```





# Heritage : super()

The `super()` function makes the child class inherit all the methods and properties from its parent

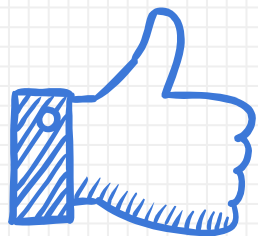
```
def init (self, fname, lname):
```

```
Person. init (self, fname, lname)
```

```
super().init(fname, lname)
```







# THANKS!

Any questions?