In python we use classes to create something called objects. Objects are made up of attributes and methods. Methods are the tasks that the objects can do.

Graphical user interface

Description automatically generated

Attributes are defined as shown below. 2 attributes are self.name and self.colour .

Self parameter is used , followed by the name of the attribute and we assigned the values “apple” and “red”

Text

Description automatically generated

In the above figure, we can call our fruit class and assign it to a variable.

* We can adjust the colour and name of the attributes as shown below. By doing so we are not restricting to the initial values which we have selected

Text

Description automatically generated

Now there is a problem with these attributes. We made the things limited as had hard coded the values – here in this case ‘apple’ ‘red’.

Text

Description automatically generated

* What can we do is create ***parameters*** as shown below and assign them to the attributes.

Graphical user interface

Description automatically generated with medium confidence

By doing so we can expand the functionality

Text

Description automatically generated

Let’s discuss ***METHODS***

It looks like a traditional function we must pass the self parameter to this function which will make it a method. We can think of methods as function related to objects.

Text

Description automatically generated

Now when we pass the parameter from method to method, here in this case, using self parameter something cool happens.

Text

Description automatically generated

We can then access the same attributes in the ***init***  inside the body of new details method. So even though these two different methods have completely different scopes the self keyword ensures all the attributes are accessible by all the methods.

Text

Description automatically generated

Now we can easily call these methods by specifying the name of the object followed by the name of the methods and then a set of brackets.

Text

Description automatically generated

* Let’s look at ***init*** method—why it is so significant. It doesn’t need to be called, we just need to define it and it runs automatically. It has a reserved name

Graphical user interface, application

Description automatically generated

* Let’s look at self parameter:

Which always represents the current instance of the class : so whenever we pass it from a method to method, we still refer to the same object.

Text

Description automatically generated

However, whenever we create a new instance of the class, the self keyword will have completely different meaning ex: self in case of apple represents the apple itself, in case of banana it no longer knows anything about the apple and represents itself about the banana itself.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

<https://www.programiz.com/python-programming/examples/power-anonymous>

https://www.programiz.com/python-programming/examples/number-divisible