Abstraction and Encapsulation in Python:

Abstraction: -

Abstraction is used to hide internal details and show only functionalities. Abstracting something means to give names to things, so that the name captures the basic idea of what a function or a whole program does.

Let's take an example.

> So let we will consider the motorcycle. A mechanic is attempting to repair your motorcycle or, say, a certain section of your motorcycle. You are the user, and you don't want to know your motorcycle specifics or what portion has actually broken. You don't care; you simply want your moto cycle in its original condition without worrying about the details. You told the mechanic what you wanted by separating the implementing part; this is called abstraction. The most important thing you did was fix your motorcycle instead of focusing on the particulars.

Encapsulation: -

Encapsulation is used to restrict access to methods and variables. In encapsulation, code and data are wrapped together within a single unit from being modified by accident.

Let's take an example.

> Only you need to know the device's behaviour without having to worry about implementing details like what sensors the Keyboard has and whether it is wireless or not etc. Every detail describes the mouse, but it is simply a keyboard, irrespective of the detail. You just have a keyboard interface, which is the mouse pointer in this situation, and this we called Encapsulation.

Comparison table of Abstraction vs Encapsulation:

Abstraction	Encapsulation
Abstraction works on the design level.	Encapsulation works on the application level.
Abstraction is implemented to hide unnecessary data and withdrawing relevant data.	Encapsulation is the mechanism of hiding the code and the data together from the outside world or misuse.
It highlights what the work of an object instead of how the object works is	It focuses on the inner details of how the object works. Modifications can be done later to the settings.
Abstraction focuses on outside viewing, for example, shifting the car.	Encapsulation focuses on internal working or inner viewing, for example, the production of the car.
Abstraction is supported in Java with the interface and the abstract class.	Encapsulation is supported using, e.g. public, private and secure access modification systems.
In a nutshell, abstraction is hiding implementation with the help of an interface and an abstract class.	In a nutshell, encapsulation is hiding the data with the help of getters and setters.