**Abstraction**

1] Abstraction is a feature of OOPs that hides the **unnecessary** detail but shows the essential information.

2] It solves an issue at the **design** level.

3] It focuses on the **external** lookout.

4] It can be implemented using **abstract classes** and **interfaces**.

5] It is the process of **gaining** information.

6] **Abstract classes** and **interfaces** to hide the code complexities.

7] The objects are **encapsulated** that helps to perform abstraction.

**Encapsulation**

1] Encapsulation is also a feature of OOPs. It hides the code and data into a **single** entity so that the data can be protected from the outside world.

2] Encapsulation solves an issue at **implementation** level.

3] It focuses on **internal** working.

4] It can be implemented by using the [**access modifiers**](https://www.javatpoint.com/access-modifiers) (private, public, protected).

5] It is the process of **containing** the information.

6] **Getters** and **Setters** methods to hide the data.

7] The object need not to **abstract** that result in encapsulation.