# Define Object Oriented Programming Language?

Object-oriented programming is a programming paradigm based on the concept of "objects", which can contain data, in the form of fields, and code, in the form of procedures. A feature of objects is an object's procedures that can access and often modify the data fields of the object with which they are associated.

Object Oriented programming is a programming style which is associated with the concepts like class, object, Inheritance, Encapsulation, Abstraction, Polymorphism. Most popular programming languages like Java, C++, C#, Ruby, etc. follow an object-oriented programming paradigm

# List down the Benefits of OOP?

* It provides a clear ***modular structure*** for programs which makes it good for defining abstract data types in which implementation details are hidden
* Objects can also be ***reused*** within an across applications. The reuse of software also lowers the cost of development. More effort is put into the object-oriented analysis and design, which lowers the overall cost of development.
* It makes software***easier to maintain.*** Since the design is modular, part of the system can be updated in case of issues without a need to make large-scale changes
* Reuse also enables ***faster development***. Object-oriented programming languages come with rich libraries of objects, and code developed during projects is also reusable in future projects.
* It provides a good framework for code libraries where the supplied software components can be ***easily adapted and modified by the programmer***. This is particularly useful for developing graphical user interfaces.

# Differentiate between function and method?

A **function** is a piece of code that is called by name. It can be passed data to operate on (i.e. the parameters) and can optionally return data (the return value). All data that is passed to a function is explicitly passed.

A **method** is a piece of code that is called by a name that is associated with an object. In most respects it is identical to a function except for two key differences:

1. A method is implicitly passed the object on which it was called.
2. A method is able to operate on data that is contained within the class (remembering that an object is an instance of a class - the class is the definition, the object is an instance of that data).

# Define the following terms:

1. **Class:**  
   Classes and objects are the fundamental components of OOP's. Often there is confusion between classes and objects.
2. **Object:**

objects are the things you think about first in designing a program and they are also the units of code that are eventually derived from the process

1. **Attribute:**

Attributes are data stored inside a class or instance and represent the state or quality of the class or instance

1. **Behavior:**

The behavior of an object is defined by its methods, which are the functions and subroutines defined within the object class. ... Methods determine what type of functionality a class has, how it modifies its data, and its overall behavior