Functional programming

Functional programming is a programming paradigm in which we try to bind everything in **pure functions** style. It is a **declarative** type of programming style. Its main focus is on "**what to solve**" in contrast to an **imperative** style where the main focus is "**how to solve**". It uses expressions instead of statements. An expression is evaluated to produce a value whereas a statement is executed to assign variables.

Procedural programming

Procedural Programming can be defined as a programming paradigm which is **derived** from **imperative** programming, based upon the concept of calling procedure. Procedures, also known as **routines**, **subroutines**, or **functions**, simply consist of a series of computational steps to be carried out. During a program's execution, any given procedure might be called at any point, including by other procedures or itself.

Object-Oriented programming

Object-oriented programming can be defined as a programming paradigm which is based upon the concept of **objects**. Objects contain data in the form of **attributes** and code in the form of **methods**. In object-oriented programming, computer programs are designed using the concept of objects that interact with the real world. Object-oriented programming languages are various, but the most popular ones are **class-based**, meaning that objects are instances of classes, which also determine their types.