**Functional programming**

Functional programming is a programming paradigm in which we try to bind everything in pure mathematical 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. Those functions have some special features discussed below.

*Example:*

1. C++
2. Python
3. Java
4. C#
5. Swift

**OOP**

Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

*EXAMPLE:*

1. Ruby
2. Python
3. Small Talk
4. JavaScript
5. TypeScript
6. Dart
7. Java
8. C#
9. C++

**Procedural programming**

In procedural programming, the program is divided into small parts called functions.Procedural programming follows a top-down approach. Procedural Programming can be defined as a programming model which is derived from structured 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.

*Example:*

1. FORTRAN
2. pascal
3. COBOL
4. Basic