MyAnatomy

Difference between Procedural Programming and Object-Oriented Programming

The most important difference to be noted is that procedural programming follows a step-by-step approach to break down a job into a collection of routines and variables by following a series of instructions, whereas object-oriented programming uses objects and classes to create models.

Parameter	Object Oriented Programming	Procedural Programming
Definition	Object-oriented Programming is a programming language that uses classes and objects to create models based on the real world environment. In OOPs, it makes it easy to maintain and modify existing code as new objects are created inheriting characteristics from existing ones.	Procedural Programming is a programming language that follows a step-by-step approach to break down a task into a collection of variables and routines (or subroutines) through a sequence of instructions. Each step is carried out in order in a systematic manner so that a computer can understand what to do.
Approach	In OOPs concept of objects and classes is introduced and hence the program is divided into small chunks called objects which are instances of classes.	In procedural programming, the main program is divided into small parts based on the functions and is treated as separate program for individual smaller program.
Access modifiers	In OOPs access modifiers are introduced namely as Private, Public, and Protected.	No such modifiers are introduced in procedural programming.
Security	Due to abstraction in OOPs data hiding is possible and hence it is more secure than POP.	Procedural programming is less secure as compare to OOPs.
Complexity	OOPs due to modularity in its programs is less complex and hence new data objects can be created easily from existing objects making object-oriented programs easy to modify	There is no simple process to add data in procedural programming, at least not without revising the whole program.
Program division	OOP divides a program into small parts and these parts are referred to as objects.	Procedural programming divides a program into small programs and each small program is referred to as a function.
Importance	OOP gives importance to data rather than functions or procedures.	Procedural programming does not give importance to data. In POP, functions along with sequence of actions are followed.
Inheritance	OOP provides inheritance in three modes i.e. protected, private, and public	Procedural programming does not provide any inheritance.
Examples	C++, C#, Java, Python, etc. are the examples of OOP languages.	C, BASIC, COBOL, Pascal, etc. are the examples POP languages.