非面向对象更关注功能，它将一个大的问题细分成很多小功能，每个功能就表现为一个函数，通过函数之间的相互连接，完成问题的求解。使用这种功能分解方式编程会出现一个问题：当需求发生改变时，必须要修改某个函数或模块，模块的变化同时会引起其他依赖该模块的程序的正常运行，因此带来了程序不易维护和扩展的缺点。

面向对象编程用对象封装解决了上面的问题。利用面向对象的基本特征，并灵活使用设计模式，能给程序带来良好的可复用性、可扩展性以及可维护性。

面向过程语言以函数作为基本组成结构，在面向过程程序设计中，问题被看作一系列需要完成的任务，函数则用于完成这些任务，解决问题的焦点集中于函数。 其概念最早由E．W．Dijikstra在1965年提出，是软件发展的一个重要里程碑。例如C语言，FORTRAN语言。用这些语言编程也称为面向过程编程。

面向过程编程即在实现一定功能的过程中将功能拆分为一个个任务，并用若干书写形式互相独立的函数去完成任务。

面向对象语言（Object-Oriented Language）缩写为POP，是一类以对象作为基本程序结构单位的程序设计语言，指用于描述的设计是以对象为核心，而对象是程序运行时刻的基本成分。

面向对象语言借鉴了20世纪50年代的人工智能语言LISP，引入了动态绑定的概念和交互式开发环境的思想；始于20世纪60 年代的离散事件模拟语言SIMULA67，引入了类的要领和继承，成形于20世纪70年代的Smalltalk。

面向对象语言的发展有两个方向：一种是纯面向对象语言，如Smalltalk、EIFFEL等；另一种是混合型面向对象语言，即在过程式语言及其它语言中加入类、继承等成分，如C++、Objective-C等。