

**Imperative programming**

**Imperative programming** is a paradigm of computer **programming** where the **program** describes steps that change the state of the computer

The canonical **examples** of **imperative programming** languages are Fortran and Algol. Others include Pascal, C, and Ada.

Structured programming

**Structured programming** is a **programming** paradigm aimed at improving the clarity, quality, and development time of a computer **program** by making extensive use of the **structured** control flow constructs of selection (if/then/else) and repetition (while and for), block structures, and subroutines.

Some of the **languages** initially used for **structured programming** include: ALGOL, Pascal, PL/I and Ada

**Procedural programming**

**Procedural programming** is a **programming** paradigm, derived from structured **programming**, based upon the concept of the procedure call. Procedures, also known as routines, subroutines, or functions, simply contain a series of computational steps to be carried out.”

**Examples** of computer **procedural** languages are BASIC, C, C++ FORTRAN, Java, and Pascal

**Modular programming**

**Modular programming** (also referred to as **modular** architecture) is a general **programming** concept. It involves separating a **program's** functions into independent pieces or building blocks, each containing all the parts needed to execute a single aspect of the functionality

Languages that support the module concept are IBM Assembler, COBOL, RPG, FORTRAN, Morpho, Zonnon and Erlang, among others.

**Declarative programming**

**Declarative programming** is a non-imperative style of **programming** in which programs describe their desired results without explicitly listing commands or steps that must be performed

Some well-known general **declarative** programming **languages** include Ruby, R and Haskell. SQL (Structured Query **Language**) is a **declarative** query **language** and is the industry standard for relational databases

**Logic programming**

**Logic programming** is a **programming** paradigm which is largely based on formal **logic**

Some **logic programming languages**, such as Datalog and ASP (Answer Set **Programming**), are purely declarative

**Functional programming**

**Functional programming** is a **programming** paradigm in which we try to bind everything in pure mathematical functions style

**Haskell , SML , Scala , Erlang**