

Allen Downey, in his book *How To Think Like A Computer Scientist*, writes: Many computer languages provide a mechanism to call functions provided by shared libraries. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. One approach popular for requirements analysis is Use Case analysis. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit. Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine. Many applications use a mix of several languages in their construction and use. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the *Book of Ingenious Devices*. The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems. Integrated development environments (IDEs) aim to integrate all such help. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. The following properties are among the most important: In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. Ideally, the programming language best suited for the task at hand will be selected.