

High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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. 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. Different programming languages support different styles of programming (called programming paradigms). Ideally, the programming language best suited for the task at hand will be selected. It is very difficult to determine what are the most popular modern programming languages. Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Programs were mostly entered using punched cards or paper tape. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code. 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. For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Code-breaking algorithms have also existed for centuries. While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Computer programmers are those who write computer software. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display.