

Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). Code-breaking algorithms have also existed for centuries. Following a consistent programming style often helps readability. Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language. Techniques like Code refactoring can enhance readability. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. 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. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Programming languages are essential for software development. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. 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. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Different programming languages support different styles of programming (called programming paradigms). 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. There are many approaches to the Software development process. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input. Computer programmers are those who write computer software. Use of a static code analysis tool can help detect some possible problems. Code-breaking algorithms have also existed for centuries. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.