

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. One approach popular for requirements analysis is Use Case analysis. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). 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. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Programs were mostly entered using punched cards or paper tape. Techniques like Code refactoring can enhance readability. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. 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. New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation). Techniques like Code refactoring can enhance readability. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. Programmable devices have existed for centuries. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Programmable devices have existed for centuries. 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. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. It affects the aspects of quality above, including portability, usability and most importantly maintainability. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.