In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Scripting and breakpointing is also part of this process. It affects the aspects of quality above, including portability, usability and most importantly maintainability. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. Different programming languages support different styles of programming (called programming paradigms). Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. There are many approaches to the Software development process. Integrated development environments (IDEs) aim to integrate all such help. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. Ideally, the programming language best suited for the task at hand will be selected. 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. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Programming languages are essential for software development. There exist a lot of different approaches for each of those tasks. Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code. Code-breaking algorithms have also existed for centuries. Computer programmers are those who write computer software. 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. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.