In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them..  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers.  
 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.  
 Programs were mostly entered using punched cards or paper tape.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Ideally, the programming language best suited for the task at hand will be selected.  
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.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
  
  
The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.  
Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.