Techniques like Code refactoring can enhance readability.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
However, readability is more than just programming style.  
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.  
However, readability is more than just programming style.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
Use of a static code analysis tool can help detect some possible problems.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
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.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
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.  
 Code-breaking algorithms have also existed for centuries.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (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.