Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability..  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years.  
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.  
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.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
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.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
There are many approaches to the Software development process.  
Integrated development environments (IDEs) aim to integrate all such help.  
Their jobs usually involve:  
 Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language.  
 Different programming languages support different styles of programming (called programming paradigms).  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Scripting and breakpointing is also part of this process.  
 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).