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.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
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.  
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.  
 Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications.  
 Programmable devices have existed for centuries.  
Normally the first step in debugging is to attempt to reproduce the problem.  
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.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
 Computer programmers are those who write computer software.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.