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..  
 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.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
 The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine.  
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.  
 Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Scripting and breakpointing is also part of this process.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software.  
Techniques like Code refactoring can enhance readability.  
 Following a consistent programming style often helps readability.