Different programming languages support different styles of programming (called programming paradigms)..  
Programming languages are essential for software development.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
 Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine 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.  
 Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages.  
 Allen Downey, in his book How To Think Like A Computer Scientist, writes:  
 Many computer languages provide a mechanism to call functions provided by shared libraries.  
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.  
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.  
Ideally, the programming language best suited for the task at hand will be selected.  
 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.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.