One approach popular for requirements analysis is Use Case analysis..  
 Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications.  
Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
 The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems.  
 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.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
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.  
FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research.  
Techniques like Code refactoring can enhance readability.  
For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash.  
In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.