It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).  
Integrated development environments (IDEs) aim to integrate all such help.  
The source code of a program is written in one or more languages that are intelligible to programmers, rather than machine code, which is directly executed by the central processing unit.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.  
When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.