Use of a static code analysis tool can help detect some possible problems..  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
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.  
  
Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic.  
 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.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
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).  
Techniques like Code refactoring can enhance readability.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.