However, readability is more than just programming style..  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Scripting and breakpointing is also part of this process.  
 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.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
 Programmable devices have existed for centuries.  
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.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
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.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
Normally the first step in debugging is to attempt to reproduce the problem.  
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).  
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.