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..  
Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
Normally the first step in debugging is to attempt to reproduce the problem.  
Many applications use a mix of several languages in their construction and use.  
  
Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
There exist a lot of different approaches for each of those tasks.  
 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.  
 It is very difficult to determine what are the most popular modern programming languages.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).