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.  
Many applications use a mix of several languages in their construction and use.  
There are many approaches to the Software development process.  
 These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics.  
 Programmable devices have existed for centuries.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se.  
Normally the first step in debugging is to attempt to reproduce the problem.  
However, readability is more than just programming style.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Scripting and breakpointing is also part of this process.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
 New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation).