Integrated development environments (IDEs) aim to integrate all such help.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
Ideally, the programming language best suited for the task at hand will be selected.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
Many applications use a mix of several languages in their construction and use.  
There are many approaches to the Software development process.  
Compiling takes the source code from a low-level programming language and converts it into machine 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.  
As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices.  
  
 Computer programming is the process of performing particular computations (or more generally, accomplishing specific computing results), usually by designing and building executable computer programs.  
  
Proficient programming thus usually requires expertise in several different subjects, including knowledge of the application domain, specialized algorithms, and formal logic.