Scripting and breakpointing is also part of this process..  
  
Integrated development environments (IDEs) aim to integrate all such help.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Use of a static code analysis tool can help detect some possible problems.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
 Programmable devices have existed for centuries.  
 Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
 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.  
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.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).