Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic..  
There are many approaches to the Software development process.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
Integrated development environments (IDEs) aim to integrate all such help.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear.  
There exist a lot of different approaches for each of those tasks.  
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.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
 Programs were mostly entered using punched cards or paper tape.  
 Allen Downey, in his book How To Think Like A Computer Scientist, writes:  
 Many computer languages provide a mechanism to call functions provided by shared libraries.  
 Following a consistent programming style often helps readability.  
Programming languages are essential for software development.  
Techniques like Code refactoring can enhance readability.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.