High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware..  
Techniques like Code refactoring can enhance readability.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages.  
One approach popular for requirements analysis is Use Case analysis.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
 Following a consistent programming style often helps readability.  
 The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
However, readability is more than just programming style.  
 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.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists.