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.  
Some of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Programs were mostly entered using punched cards or paper tape.  
For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
Techniques like Code refactoring can enhance 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.  
  
 Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks.  
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.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
 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.