There are many approaches to the Software development process..  
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.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
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.  
The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
One approach popular for requirements analysis is Use Case analysis.  
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.  
Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
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.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
 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.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
Programming languages are essential for software development.