Programming languages are essential for software development.  
FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
One approach popular for requirements analysis is Use Case analysis.  
The following properties are among the most important:  
  
 In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code.  
However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory.  
To produce machine code, the source code must either be compiled or transpiled.  
Integrated development environments (IDEs) aim to integrate all such help.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses.