A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
To produce machine code, the source code must either be compiled or transpiled.  
There exist a lot of different approaches for each of those tasks.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
However, while these might be considered part of the programming process, often the term software development is more likely used for this larger overall process – whereas the terms programming, implementation, and coding tend to be focused on the actual writing of code.  
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).  
For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
However, readability is more than just programming style.  
However, readability is more than just programming style.  
Programming languages are essential for software development.  
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.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
Compiling takes the source code from a low-level programming language and converts it into machine code.