Techniques like Code refactoring can enhance readability..  
 Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code.  
Their jobs usually involve:  
 Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language.  
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.  
There exist a lot of different approaches for each of those tasks.  
One approach popular for requirements analysis is Use Case analysis.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
 Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line.  
 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.  
 Programs were mostly entered using punched cards or paper tape.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Following a consistent programming style often helps readability.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.