Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA)..  
 Code-breaking algorithms have also existed for centuries.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
 Different programming languages support different styles of programming (called programming paradigms).  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
 Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine 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.  
Programming languages are essential for software development.  
Ideally, the programming language best suited for the task at hand will be selected.  
  
The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.  
 It is very difficult to determine what are the most popular modern programming languages.  
Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit.  
 Programs were mostly entered using punched cards or paper tape.  
There are many approaches to the Software development process.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.