Integrated development environments (IDEs) aim to integrate all such help.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
Normally the first step in debugging is to attempt to reproduce the problem.  
Programming languages are essential for software development.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
Use of a static code analysis tool can help detect some possible problems.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.