Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Programming languages are essential for software development. It affects the aspects of quality above, including portability, usability and most importantly maintainability. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. 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. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Following a consistent programming style often helps readability. 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. Programming languages are essential for software development. Integrated development environments (IDEs) aim to integrate all such help. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Programs were mostly entered using punched cards or paper tape. However, readability is more than just programming style. One approach popular for requirements analysis is Use Case analysis. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). Different programming languages support different styles of programming (called programming paradigms). Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. 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. When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear.