UNIT-01: Introduction to Software Engineering

**Software:** Software is a collection of instructions, data, or computer programs that are used to run machines and carry out activities. It is the antithesis of hardware, which refers to a computer's external components. A device's running programs, scripts, and applications are collectively referred to as "software" in this context.

**Engineering:** Engineering is the use of math and science to solve problems. While scientists and innovators create advancements, engineers are the ones who translate these discoveries into practical applications. Engineering is the science-based process of converting natural resources as efficiently as possible for human use.

# Types of Software

**System Software:** This class of software manages and controls the internal operations of a computer system. It is a group of programs, which is responsible for using computer resources efficiently and effectively. For example, an [operating system](https://ecomputernotes.com/fundamental/disk-operating-system/what-is-operating-system) is a system software, which controls the hardware, manages [memory](https://ecomputernotes.com/fundamental/input-output-and-memory/memory) and multitasking functions, and acts as an interface between application programs and the computer.

**Application software:** Developed as per user requirement. This class of software is used for both official and personal use. The [personal computer](https://ecomputernotes.com/fundamental/introduction-to-computer/personal-computer) software market has grown over in the last two decades from normal text editor to word processor and from simple paintbrush to advanced image-editing software. This software is used predominantly in almost every field, whether it is [database management system](https://ecomputernotes.com/fundamental/what-is-a-database/advantages-and-disadvantages-of-dbms), financial accounting package, or multimedia-based software. It has emerged as a versatile tool for routine applications.

**Real-time Software:** This class of software observes, analyses, and controls real world events as they occur. Generally, a real-time system guarantees a response to an external event within a specified period of time. An example of real-time software is the software used for weather forecasting that collects and processes parameters like temperature and humidity from the external environment to forecast the weather. Most of the defence organizations all over the world use real-time software to control their military hardware.

**Business Software:** This class of software is widely used in areas where management and control of financial activities is of utmost importance. The fundamental component of a business system comprises payroll, inventory, and accounting software that permit the user to access relevant data from the [database](https://ecomputernotes.com/fundamental/what-is-a-database/advantages-and-disadvantages-of-dbms). These activities are usually performed with the help of specialized business software that facilitates efficient framework in business operations and in management decisions.

**Engineering and Scientific Software:** This class of software has emerged as a powerful tool in the research and development of next generation technology. Applications such as the study of celestial bodies, under-surface activities, and programming of an orbital path for space shuttles are heavily dependent on engineering and scientific software. This software is designed to perform precise calculations on complex numerical data that are obtained during real time environment.

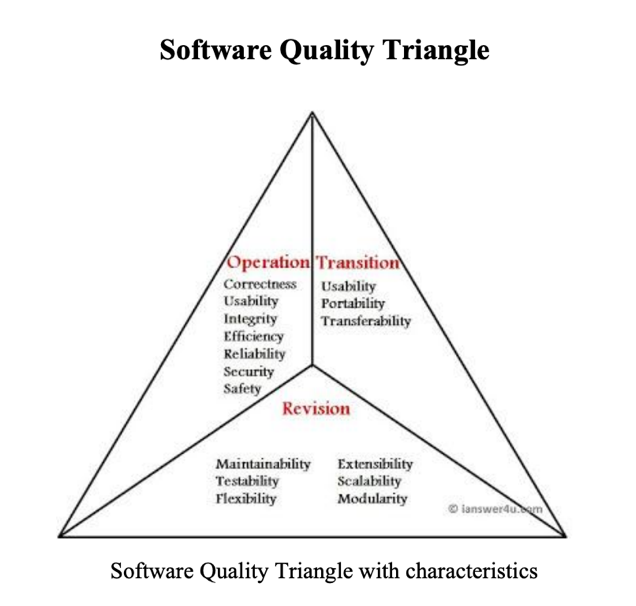
**Artificial Intelligence (AI) Software:** This class of software is used where the problem-solving technique is non-algorithmic in nature. The solutions of such problems are generally non-agreeable to computation or straightforward analysis. Instead, these problems require specific problem-solving strategies that include expert system, pattern recognition, and game-playing techniques. In addition, they involve different kinds of search techniques which include the use of heuristics. The role of artificial intelligence software is to add certain degrees of intelligence to the mechanical hardware in order to get the desired work done in an agile manner.

**Embedded Software:**  ﻿Embedded software is a piece of software that is embedded in hardware or non-PC devices. It is written specifically for the particular hardware that it runs on and usually has processing and memory constraints because of the device’s limited computing capabilities. Examples of embedded software include those found in dedicated GPS devices, factory robots, some calculators and even modern smartwatches

**Internet Software:** This class of software acts as an interface between the user and the [Internet](https://ecomputernotes.com/computernetworkingnotes/services-and-applications/what-is-internet). Data on the [Internet](https://ecomputernotes.com/computernetworkingnotes/services-and-applications/what-is-internet) is in the form of text, audio, or video format, linked with hyperlinks. Web browser is a software that retrieves web pages from the Internet. The software incorporates executable instructions written in special scripting languages such as CGI or ASP. Apart from providing navigation on the Web, this software also supports additional features that are useful while surfing the Internet.

**Software Tools and CASE environment:** Tools and programs that help the construction of application software and systems. For example, test tools, version control tools.

# Characteristics of Software



**Operational Characteristics**

These are functionality based factors and related to 'exterior quality' of software. Various Operational Characteristics of software are :

**a) Correctness:** The software which we are making should meet all the specifications stated by the customer.

**b) Usability/Learnability:** The amount of efforts or time required to learn how to use the software should be less. This makes the software user-friendly even for IT illiterate people.

**c) Integrity:** Just like medicines have side-effects, in the same way a software may have a side-effect i.e. it may affect the working of another application. But a quality software should not have side effects.

**d) Reliability :** The software product should not have any defects. Not only this, it shouldn't fail while execution.

**e) Efficiency :** This characteristic relates to the way software uses the available resources. The software should make effective use of the storage space and execute command as per desired timing requirements.

**f) Security :** With the increase in security threats nowadays, this factor is gaining importance. The software shouldn't have ill effects on data / hardware. Proper measures should be taken to keep data secure from external threats.

**g) Safety :** The software should not be hazardous to the environment/life.

**Maintenance or Revision Characteristics**  
These engineering based factors of the relate to 'interior quality' of the software like efficiency, documentation and structure. These factors should be in-build in any good software. Various Revision Characteristics of software are :-

**a) Maintainability :** Maintenance of the software should be easy for any kind of user.

**b) Flexibility :** Changes in the software should be easy to make.

**c) Extensibility : I**t should be easy to increase the functions performed by it.

**d) Scalability :** It should be very easy to upgrade it for more work(or for more number of users)

**e) Testability :** Testing the software should be easy.

**f) Modularity :** Any software is said to made of units and modules which are independent of each other. These modules are then integrated to make the final software. If the software is divided into separate independent parts that can be modified, tested separately, it has high modularity

**Transition Characteristics**

**a) Interoperability :** Interoperability is the ability of software to exchange information with other applications and make use of information transparently.

**b) Reusability :** If we are able to use the software code with some modifications for different purpose then we call software to be reusable.

**c)Portability :** The ability of software to perform same functions across all environments and platforms, demonstrate its portability.