* Component Object Model (COM) is a binary-interface standard for software component introduced by Microsoft in 1993.
* It is used to enable inter-process communication and dynamic object creation in a large range of programming languages.
* COM is the basis for several other Microsoft technologies and frameworks including OLE, OLE automation, ActiveX, COM+, DCOM, and Windows shell, DirectX and the Windows runtime.

**Overview**

* The essence of COM is a language-neutral way of implementing objects that can be used in environments different from the one in which they were created, even across machine boundaries.
* For well-authored components, COM allows reuse of objects with no knowledge of their internal implementation, as it forces component implementers to provide well-defined interfaces that are separated from the implementation.
* The different allocation semantics of languages are accommodated by making objects accountable for their own creation and destruction through reference-counting.
* Casting between different interfaces of an object is achieved through the QueryInterface method.
* The preferred method of inheritance within COM is the creation of sub-objects to which method calls are delegated.
* COM is a technology defined and implemented as standard only on Microsoft Windows and Apple’s Core Foundation 1.3 and later plug-in API, that in any case implement only a subset of the whole COM interface.
* For some applications, COM has been replaced at least to some extent by the Microsoft .NET framework, and support for Web services through the Windows Communication Foundation.
* However, COM objects can be used with all .NET languages through the .NET COM Interop.
* Networked DCOM uses binary proprietary formats, while WCF encourages the use of XML-based SOAP messaging.
* COM is very similar to other component interface technologies, such as CORBA and Java Beans, although each has its own strength and weaknesses.