A library provides functions used by many programs.

A static library(.a) is a library that can be linked directly into the final executable produced by the linker,it is contained in it and there is no need to have the library into the system where the executable will be deployed.

A shared library(.so) is a library that is linked but not embedded in the final executable, so will be loaded when the executable is launched and need to be present in the system where the executable is deployed.

A dynamic link library on windows(.dll) is like a shared library(.so) on linux but there are some differences between the two implementations that are related to the OS (Windows vs Linux) :

A DLL can define two kinds of functions: exported and internal. The exported functions are intended to be called by other modules, as well as from within the DLL where they are defined. Internal functions are typically intended to be called only from within the DLL where they are defined.

An SO library on Linux doesn't need special export statement to indicate exportable symbols, since all symbols are available to an interrogating process.

Linux systems have two types of executable programs:

* **Statically linked executables**: Contain all the library functions that they need to execute; all library functions are linked into the executable. They are complete programs that do not depend on external libraries to run. One advantage of statically linked programs is that they work without your needing to install prerequisites.
* **Dynamically linked executables**: Much smaller programs; they are incomplete in the sense that they require functions from external shared libraries to run. Besides being smaller, dynamic linking permits a package to specify prerequisite libraries without needing to include the libraries in the package. By using dynamic linking, many running programs can share one copy of a library rather than occupying memory with many copies of the same code. For these reasons, most programs today use dynamic linking.

Statically linked executables là những chương trình mà các thư viện nằm sẵn trong chương trình luôn. Không cần đến thư viện bên ngoài để chương trình execute. For example: sln

Dynamically linked executables là những chương trình cần thêm những thư viện bên ngoài thì chương trình mới execute được. For example: ln

Ln chức năng tương tự sln. Nhưng sln là đã có bao gồm tất cả các thư viện bên trong nên dung lương sln lớn hơn ln