A compiler breaks a computer program into multiple objects and libraries. Linkers basically convert these multiple objects and libraries into a single object. A linker which writes the output into the memory is called a Loader. An object file refers/is referred to other objects by means of symbols which are of three kinds, Defined Symbols(for calling by other modules), Undefined Symbols(include references that help to call other modules) and Local Symbols(used for relocation). Linkers are also held responsible for the task of memory management. There are two branches of Linkers, Static and Dynamic Linkers. Linkers get objects from a library, by including the symbols that are referenced by the objects or directly including the whole library. Static Linkers include all the defined symbols, undefined symbols, and local symbols into a single unified exe image, and it requires more disk space and memory as compared to dynamic linking but it does not require loading any libraries. On the other hand, dynamic linkers only have the name of the shareable library, and the exe contains some libraries and objects that is associated with undefined symbols. When we load the program, the program automatically load these objects and libraries and performs a final linking.