

## Assignment# 6.3

### **Part 1:**

- Provide a template Sphinx command line command which auto documents your project. Explain what each flag and each element of your command do.

### **Answer:**

short details:

**sphinx-apidoc -o Destination\_folder Source\_folder Optional(flag) to more understanding.**

**sphinx-apidoc:** The Sphinx-apidoc command is used to generate API documentation. It automatically generates reStructuredText files that outline the public API of a Python project.

**-o Destination\_folder:** -o Destination\_folder flag specifies the folder location where the newly created document files will be saved. Destination\_folder should be changed to the correct path of the folder where the output files should be saved.

**source\_folder:** The folder containing the Python source code that you want to record is located at source\_folder. Sphinx will scan this folder and its subdirectories for Python modules, classes, methods, and other objects that need to generate API documentation.

**Optional (flags):** This element, optional (flags), refers to a flag that you can pass to the command to change the way sphinx-apidoc works. Sphinx provides several parameters to control actions such as excluding specific modules or packages, specifying which file extensions to consider, and can be used to set the maximum recursion depth when scanning source folders. The specific flag and its purpose will be chosen based on the needs of your project.

When you run this command with the appropriate values for Source\_folder and Destination\_folder, Sphinx will search the search folder, create ReStructuredText files (.rst) based on anything Python finds, and send them to the specified destination. will assign to the folder. After processing these restructured text files, the Sphinx API can provide documents in a variety of output formats, including HTML, PDF, or LaTeX.

### **Part 2:**

- What flag should you add if you use implicit namespaces in your project?

### **Answer:**

**sphinx-apidoc --implicit-namespaces -o Destination\_folder Source\_folders**

Now! Let's analyze the given command:

**sphinx-apidoc:** The Sphinx-apidoc command is used to generate API documentation. It automatically generates reStructuredText files that outline the public API of a Python project.

**Flag: --implicit-namespaces**

**The generation of documentation for implicit namespaces in addition to standard modules and packages is enabled by this flag.**

**Flag: -o Destination\_folder**

**The output directory where the created documentation will be saved is specified by this flag. instead the desired path to the folder where you would like to save the documentation in place of Destination\_folder.**

**Argument: Source\_folders**

The directory path(s) to the source code that your project that you want to document are specified by this option. The path(s) to the source code directories to your project should be replaced by Source\_folders.

Hence, the sphinx-apidoc tool will be executed with the --implicit-namespaces flag and will provide documentation for the specified Source\_folders when the command sphinx-apidoc --implicit-namespaces -o Destination\_folder Source\_folders are entered. The Destination\_folder directory will hold the created documentation.