

Unit 6.3 Graded Assignment:

Syed Muhammad Raqim Ali Shah (2303.KHI.DEG.008)

Maaz Javaid Siddique (2303.KHI.DEG.004)

Qadeer Hussain (2303.KHI.DEG.006)

Daily Assignment 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:

sphinx-apidoc -o Destination_folder Source_folder Optional(flag)

This is a basic Sphinx command to auto-generate documentation using the apidoc module. This how to use this command:

- **sphinx-apidoc:** This is the command to run Sphinx with the apidoc module, which automatically generates API documentation from your project's source code.
- **-o Destination_folder:** This flag specifies the output directory where the generated documentation will be saved. Replace Destination_folder with the desired path to the folder where you want to store the documentation.
- **Source_folder:** This is the directory path to your project's source code that you want to document. Replace Source_folder with the actual path to your project directory.
- **[optional(flag)]:** This section allows you to include additional options and flags to customize the documentation generation process. Here are some common options:
 - **-f or --force:** This flag forces the re-generation of all documentation files, even if they already exist. It is useful when you want to update the documentation with the latest changes.
 - **-M or --module-first:** This flag rearranges the generated module documentation in the file hierarchy, placing it under a module-first

structure. This is particularly useful when documenting a project with multiple modules or packages.

By running this command, Sphinx will analyze the Python modules within the `Source_folder` and generate corresponding reStructuredText files in the `Destination_folder` for documentation purposes. These files will include class and function documentation extracted from your source code, which can later be built into HTML, PDF, or other formats using Sphinx.

Daily Assignment 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`

The **`--implicit-namespaces`** flag tells Sphinx to include implicit namespaces in the generated documentation. Implicit namespaces are sub packages or modules that are not explicitly imported but are referred to within your project's code base.

By using this flag, Sphinx will traverse the source code within the `Source_folder` and automatically include any implicit namespaces it encounters in the generated documentation. This ensures that the documentation accurately reflects the structure of your project, even if some namespaces are implicitly referenced.