Python Essentials 2
Section
Modules and Packages

Introduction

In this section, you will learn:

- What are modules and packages
- How to create your first module
- How to build your first package
- Summary and review questions

1.3.1 What is a Module?

- A module is a single Python file (.py) containing functions, variables, or classes.
- Modules allow code reuse and better organization.

- Example:
  - math_utils.py:
    - def add(a, b):
        return a + b

      def subtract(a, b):
        return a - b

Using a Module

- You can import and use functions from your module in another script.

- Example:
  - main.py:
    - import math_utils

      print(math_utils.add(3, 4))   # Output: 7

1.3.2 What is a Package?

• A package is a directory containing multiple modules.

• It must contain a special file __init__.py.

• Packages allow building structured projects.

Example: A Simple Package

- mypackage/
- __init__.py
- math_utils.py
- string_utils.py
  - main.py:
    - from mypackage import math_utils

      print(math_utils.add(10, 5))   # Output: 15

Importing Techniques

- You can import in different ways:

  - import module_name
  - from module_name import function_name
  - import module_name as alias
  - from package.module import function_name

1.3.4 Section Summary

- Module = Python file containing reusable code.

- Package = Directory of modules + __init__.py.

- import/from import = keywords for usage.

- Organizing code with modules/packages improves readability and reusability.

1.3.5 Section Quiz – Review Questions

- 1) What is the difference between a module and a package?

- 2) How do you import a single function from a module?

- 3) What is the purpose of __init__.py?

- 4) Write an example of importing a function from a package.
    - Suggested Answers:
        - 1) Module = .py file, Package = directory with modules.
          2) from module import function
          3) Defines directory as a package.
          4) from mypackage.math_utils import add