Module 1 Summary: Python Coding Practices and Packaging Concepts

Congratulations! You have completed this module. At this point, you know that:

- The application development lifecycle has seven phases, including:
  - Requirement Gathering: You collect user, business, and technical requirements for the app
  - Analysis: You analyze the requirements
  - Design: You design the complete solution
  - Code and test: You build and test the different components of the app
  - User and system test: Users test the app for usability, and you perform system integration testing and performance testing
  - Production: The application is available to all end users
  - Maintenance: You upgrade or fix any user or system issues
- All web apps are APIs, but not all APIs are web apps. Both share data between apps, but not all APIs require networks like web apps do.
- The PEP8 guidelines for code readability include the following:
  - Four spaces for indentation
  - Blank lines to separate functions and classes
  - Spaces around operators and after commas
- The PEP8 coding conventions for consistency and manageability include:
  - Add larger blocks of code inside functions
  - Name functions and files using lowercase with underscores
  - Name classes using CamelCase
  - Name constants in capital letters with underscores separating words
- To ensure that your code adheres to the predefined style and standard without executing the code, you can use the Static code analysis method.
- Unit testing is a method to validate if code units are operating as designed. You must test every unit before integration with the final codebase.

- To create a package:
  - Create a folder with the package name
  - Create an empty __init__.py file
  - Create the required modules
  - In the __init__.py file, add code to reference the modules needed in the package
- You can verify the package via the bash terminal in a Python shell.