

Intermediate Python Nanodegree

Glossary – Large Code Bases Using Libraries



1. Foundations

Exception

A special Python object that is raised when the program fails

Instantiation

Creating a new object in memory.

Linters

A tool that checks the quality of your source code. Linters can check for a variety of errors, ranging from programmatic mistakes to stylistic issues.

Pure function

A method where the output is based solely on its inputs and no variables outside the function's scope are affected or changed.

PEPs (Python Enhancement Proposals)

A set of community driven guidelines that help Python developers understand key language decisions and implement uniform code.

PEP-8

The Python style guide.

Raise an Exception

Disrupting the program by “throwing a wrench in” the execution flow.

self

In Python, the *self* keyword is a reference to the specific object we're working with in memory

Type Hints

A simple way to add types to variables and methods.

Unified Modeling Language (UML)

A visual language for expressing software structure, including objects, class relationships, and sequences of tasks.

2. Building Modules

`__init__.py`

Special file that declares that a directory is a Python module. The code in `__init__.py` is executed when the module is imported.

Absolute Import

Pythonic concept to import a module by searching for the module in the project's root folder (e.g. `import src.module.utility`).

Abstract Class

A partially defined class that defines a common interface for other, concrete, classes.

Class Method

A special instance method that is used to generate new object instances of the class. Decorated with `@classmethod`.

Decorators

Extends a function's behavior without needing to rewrite the entire function. Written as `@decorator`

Object-Oriented Programming (OOP)

A programming design pattern that groups data attributes (instance variables) and related methods or functions (instance methods) into *objects*.

Overrides

During object inheritance, if a method body is rewritten by a child object, it is considered an override. Common overrides include `__eq__`, `__str__`, and `__len__`.

Partial Import

Pythonic concept to import only a specific method, class, or variable from a module, instead of importing the entire module. (e.g., `from src.module import utility`).

Relative Import

Pythonic concept to import a module by searching for the module based on the location of the executed code—for example, by searching in a file located in the current directory, as in:
`from .module import utility`

Static Method

A special instance method that does not depend on any other instance methods or variables. It does not have a reference to *self*. Decorated with `@static`.

3. Using Libraries

Encapsulation

Bundling of methods or features into a new interface.

Pip Installs Python (pip)

Python utility to install python packages from PyPI and other repositories. (Note: The acronym "pip" is recursive, since it is itself one of the words in the acronym!)

Python Package Index (PyPI)

A repository of software for the Python programming language.

requirements.txt

Commonly used filename for a list of all python library dependencies. Generated in the CLI with:

```
pip freeze > requirements.txt
```

Strategy Object

Design pattern that provides a common interface for methods that have different execution algorithms (For example, methods for converting multiple types of units.)

Virtual Environment

A Python module that creates a segmented directory where Python modules can be installed without interfering with system modules.

Created in CLI with:

```
python3 -m venv venv
```

4. Python in Systems

Argparser

Python module that simplifies the creation of command-line tools with input arguments.

Command-Line Tool

A utility that can be executed from the terminal window (shell or BASH).

Legacy Code

Software that exists within a system that is not necessarily written in a desirable language, but cannot be replaced.

Subprocess

Python module used to spawn system processes (e.g., legacy code).

5. Python for Web

Application Program Interface (API)

Communication protocol that allows two computer systems to interact. On the web, specific URLs commonly provide this protocol.

Flask

Python framework for creating a web server.

HyperText Transfer Protocol (HTTP)

The protocol—or set of rules—governing how requests are sent over the Internet.

Requests

Popular Python module for consuming information from web services.

Request/Response

Common terminology for sending and receiving data over HTTP network connections.

Server

Specialized software that is designed to handle networked requests and responses.

Uniform Resource Locator (URL)

A web address for a specific server action, which includes a protocol, domain name, and path.