Discrete Structures: Setting Up Projects CMPSC 102

Oliver BONHAM-CARTER

Fall 2022
Week 5
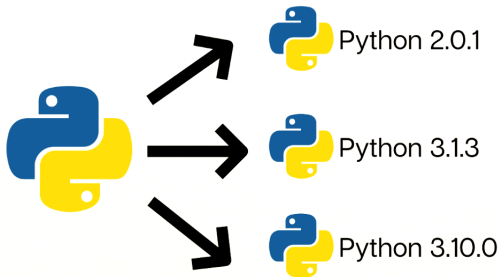Slides 01

Let's Discuss

Key Questions

How do I use virtual environments like `Venv`, `Poetry`, `Typer` and other resources to create a professional project?

Learning Objectives

To learn how to use libraries and dependencies for development with Python code and programming techniques to create the foundations for a professional project.

Virtual Environments

- Projects' dependencies and specific versions of libraries.
- Not all projects require the same dependencies; How do you mix projects on your computer?

Python 2.0.1

Python 3.1.3

Python 3.10.0

Go Virtual

- Virtual environments maintain specific libraries and dependencies projects
- Shipping software: build an exact copy of development environment on client's machine to use software.

- Decide on the purpose and composition of the project
  - Our project: a regression analysis demonstration from SciKit-Learn
    - `https://scikit-learn.org/stable/supervised_learning.html#supervised-learning`
  - No command line parameters
  - No output, other than screen printing
  - Execution: Program complete regression analysis of random values
  - One function in project: `main()`
  - **Dependencies**: scikit-learn, numpy, seaborn

Setting Up Virtual Environment

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss
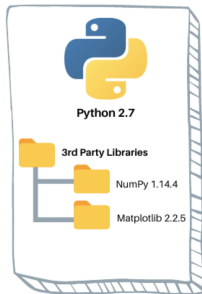
Virtual
Environments

Venv Project

Setup
Executing
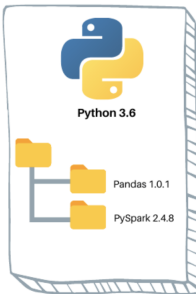Black

Another
Virtual
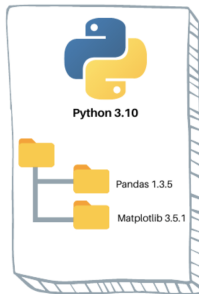Environment

New Poetry
Project

Test

Execute Project

Create a project directory

```
mkdir projects
cd projects
```

Create virtual environment using Python

```
python3 -m venv myenv
# see the file tree
find . -not -path  '*/\.*'
```

Activate *myenv* the virtual environment

```
source myenv/bin/activate
```

Install Dependencies

```
pip install numpy
pip install seaborn
pip install scikit-learn
```

Output From Executing the Script

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss

Virtual
Environments

Venv Project

Setup

Executing

Black

Another
Virtual
Environment

New Poetry
Project

Test

Execute Project

Execute given Script in *myenv*

```
python3 sciKitDemo.py
```


File: regressionPlot.png

ALLEGHENY COLLEGE

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Black: a Python Script *formatter*

Got Clutter?

- How to maintain *readable* code?

- How to reduce white-space in code to improve readability?

- A code formatting resource

- https://black.readthedocs.io/en/stable/
  getting_started.html

Black: a Python Script *formatter*

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss

Virtual
Environments

Venv Project
Setup
Executing
Black

Another
Virtual
Environment

New Poetry
Project
Test
Execute Project

Install Dependencies

```
pip install black
```

Linting example: Standard Input

```
echo "print ( 'hello, world' )" | black -
```

Linting example: As a String

```
black --code "print ( 'hello, world' )"
```

https://black.readthedocs.io/en/stable/usage_and_
configuration/the_basics.html

We will use this with Poetry (up next)

A *Bigger* Virtual Environment

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

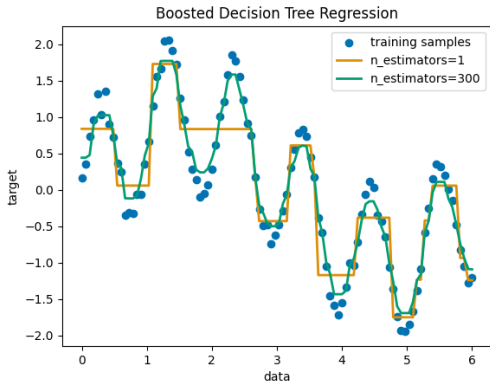Let's Discuss

Virtual
Environments

Venv Project
Setup
Executing
Black

Another
Virtual
Environment

New Poetry
Project
Test
Execute Project

Use a New Terminal

Poetry

PYTHON PACKAGING AND DEPENDENCY MANAGEMENT MADE EASY

➡️ **Typer**

https://typer.tiangolo.com/

Typer

- Command line interface support for program inputs and parameters
- Annotations: assigns types to functions that accept arguments (parameters)
- Productivity: types aid in the creation of the interface
- Checking: Confirm that inputs match expected types.

ALLEGHENY
COLLEGE

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss

Virtual
Environments

Venv Project
Setup
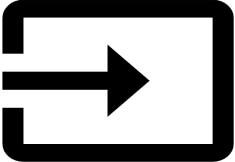Executing
Black

Another
Virtual
Environment

New Poetry
Project
Test
Execute Project

PYTHON PACKAGING AND DEPENDENCY MANAGEMENT MADE EASY

Poetry

https://python-poetry.org/

Poetry

- Management support for Python and its resources
- Environments: manage dependencies in isolation
- Package: create a stand-alone executable application
- Publish: expedite and simplify the release of program to PyPI

A Hello User Program

- Decide on the purpose and composition of the project
  - Our project: Say hello to the user
  - Parameters firstName, middleName and lastName as parameters
  - No output, other than screen printing
  - Execution: Program greets user by full name
  - One function in project: main()

Setup Steps

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss

Virtual
Environments

Venv Project
Setup
Executing
Black

Another
Virtual
Environment

New Poetry
Project

Test
Execute Project

Make a working directory

```
mkdir projects
cd projects
```

Use Poetry to create new project

```
poetry new hello_user
cd hello_user
```

Add Project Dependencies

```
poetry add typer
poetry add rich
```

Add Project Development Dependencies

```
poetry add -D black mypy
```

Mypy: http://mypy-lang.org/

Setup Steps

File: Add File: projects/hello_user/hello_user/__init__.py

```
"""Required docstring for an __init__ file."""

__version__ = "0.1.0"
```

File: Add File:projects/hello_user/pyproject.toml

```
[tool.poetry] ...

[tool.poetry.scripts]
hello_user = "hello_user.main:cli"

[tool.poetry.dependencies] ...
```

Update Poetry

```
poetry install
```

ALLEGHENY COLLEGE

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss

Virtual
Environments

Venv Project
Setup
Executing
Black

Another
Virtual
Environment

New Poetry
Project
Test
Execute Project

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
from rich.console import Console
import typer

# create a Typer object to support the command-line interface
cli = typer.Typer()
@cli.command()
def main(first: str = "", middle: str = "", last: str = ""):
    """Say hello to the person having a name
    of first, middle and last name"""
    console = Console()
    console.print("Hello to;")
    console.print(f"    first = {first}")
    console.print(f"    middle = {middle}")
    console.print(f"    last = {last}")
# end of main()
```

```
poetry run black hello_user tests
```

```
obonhamcarter@MacBookPro-2017 hello_user % poetry run black hello_user tests
reformatted hello_user/main.py

All done! ✨ 🍰 ✨
1 file reformatted, 2 files left unchanged.
```

Execute Project

Discrete
Structures:
Setting Up
Projects
CMPSC 102

Oliver
BONHAM-
CARTER

Let's Discuss

Virtual
Environments

Venv Project
Setup
Executing
Black

Another
Virtual
Environment

New Poetry
Project
Test
Execute Project

What do you see?

```
# run from projects/hello_user
poetry run hello_user --help
```

```
obonhamcarter@MacBookPro-2017 hello_user % poetry run hello_user --help

 Usage: hello_user [OPTIONS]

 Say hello to the person having a name of first, middle and last name

 ┌─ Options ────────────────────────────────────────────────────────────────┐
 --first                         TEXT
 --middle                        TEXT
 --last                          TEXT
 --install-completion            [bash|zsh|fish|powershell    Install completion for
                                 |pwsh]                       the specified shell.
                                                              [default: None]
 --show-completion               [bash|zsh|fish|powershell    Show completion for the
                                 |pwsh]                       specified shell, to copy
                                                              it or customize the
                                                              installation.
                                                              [default: None]
 --help                                                       Show this message and
                                                              exit.
```

What do you see?

```
# run from projects/hello_user
poetry run hello_user
```

Without parameters

```
poetry run hello_user
Hello to;
    first =
    middle =
    last =
```

What do you see?

```
# run from projects/hello_user
poetry run hello_user
```

Without parameters

```
Hello to;
    first =
    middle =
    last =
```

Execute Project

ALLEGHENY COLLEGE

Discrete Structures: Setting Up Projects CMPSC 102

Oliver BONHAM-CARTER

Let's Discuss

Virtual Environments

Venv Project
Setup
Executing
Black

Another Virtual Environment

New Poetry Project
Test
Execute Project

What do you see?

```
# run from projects/hello_user
poetry run hello_user
    --first Darth
    --middle Cornelius
    --last Vader
```

With parameters

```
Hello to;
    first = Darth
    middle = Cornelius
    last = Vader
```