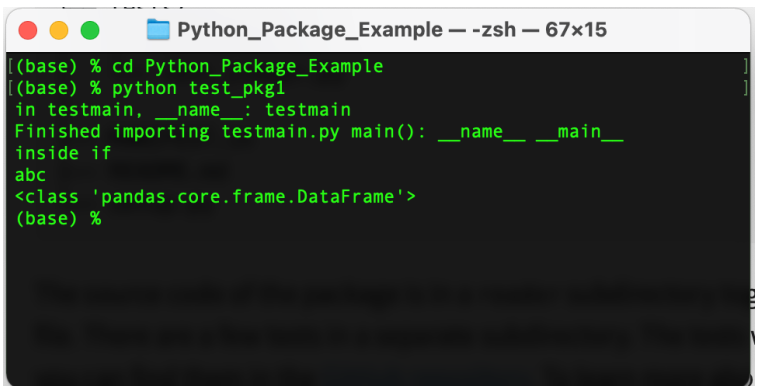
Python Project Setup

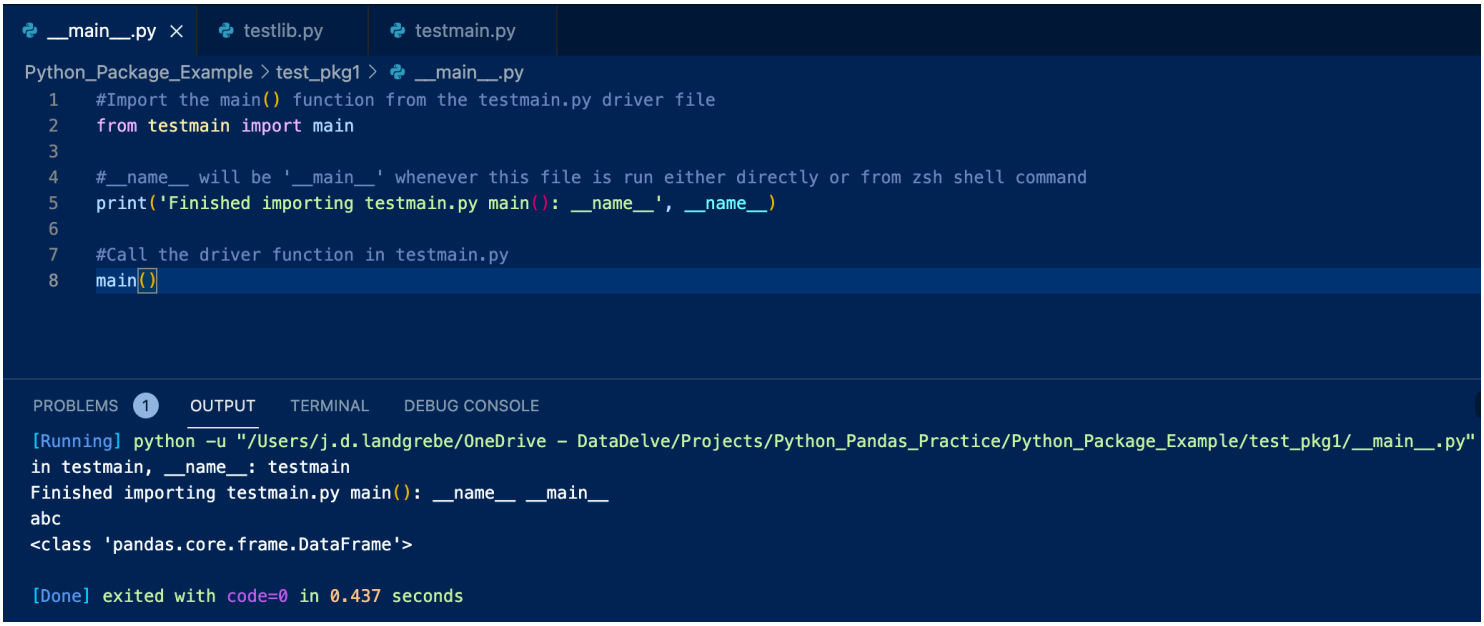
From the command line, running the package (>>python pkg_name) calls the __main__.py code at the package subfolder's root level. In the example, **test_pkg1** is self-contained with all needed code in its sub-folder.

Example folder structure:

```
Python_Pandas_Practice\
    |__ Python_Package_Example\
        |
        |__ test_pkg1\
        |    |
        |    |__ __main__.py
        |    |__ testmain.py
        |    |__ testlib.py
        |
        |__ test_pkg2\
        |    |__etc.
        |
        |__ tests\
        |__ Readme.md
        |__ etc.
```

Running from the Python_Package_Example folder is equivalent to running **__main__.py** from within VS Code:





The driver file **testmain.py** and an imported Class in **testlib.py** showing how code can be subdivided into

The driver file, **testmain.py** and an imported class in **testlib.py** showing how code can be subdivided into multiple files within the package.

- Note that the package top-level driver code is in the **main()** function to control not running it when simply importing **testmain** as a libary by other code
- The lines at bottom are standard and usually placed at the end of a *.py file. These lines run the **main()** function if testmain.py is run

__main__.py | testlib.py | testmain.py ×

Python_Package_Example > test_pkg1 > testmain.py > ...

```python
import testlib

def main():
    #Instantiate the class
    cl = testlib.TestClass('abc')
    print(cl.param)
    print(type(cl.df))

print('in testmain, __name__:', __name__)
if __name__ == '__main__':
    print('Running standalone from testmain.py')
    main()
```

PROBLEMS 1 | OUTPUT | TERMINAL | DEBUG CONSOLE

```
[Running] python -u "/Users/j.d.landgrebe/OneDrive - DataDelve/Projects/Python_Pandas_Practice/Python_Package_Example/test_pkg1/testmain.py"
in testmain, __name__: __main__
Running standalone
abc
<class 'pandas.core.frame.DataFrame'>

[Done] exited with code=0 in 0.564 seconds
```

__main__.py | testlib.py × | testmain.py

Python_Package_Example > test_pkg1 > testlib.py > ...

```python
import pandas as pd
import numpy as np

class TestClass:
    def __init__(self, param):
        self.param = param
        self.df = pd.DataFrame()
        return
```