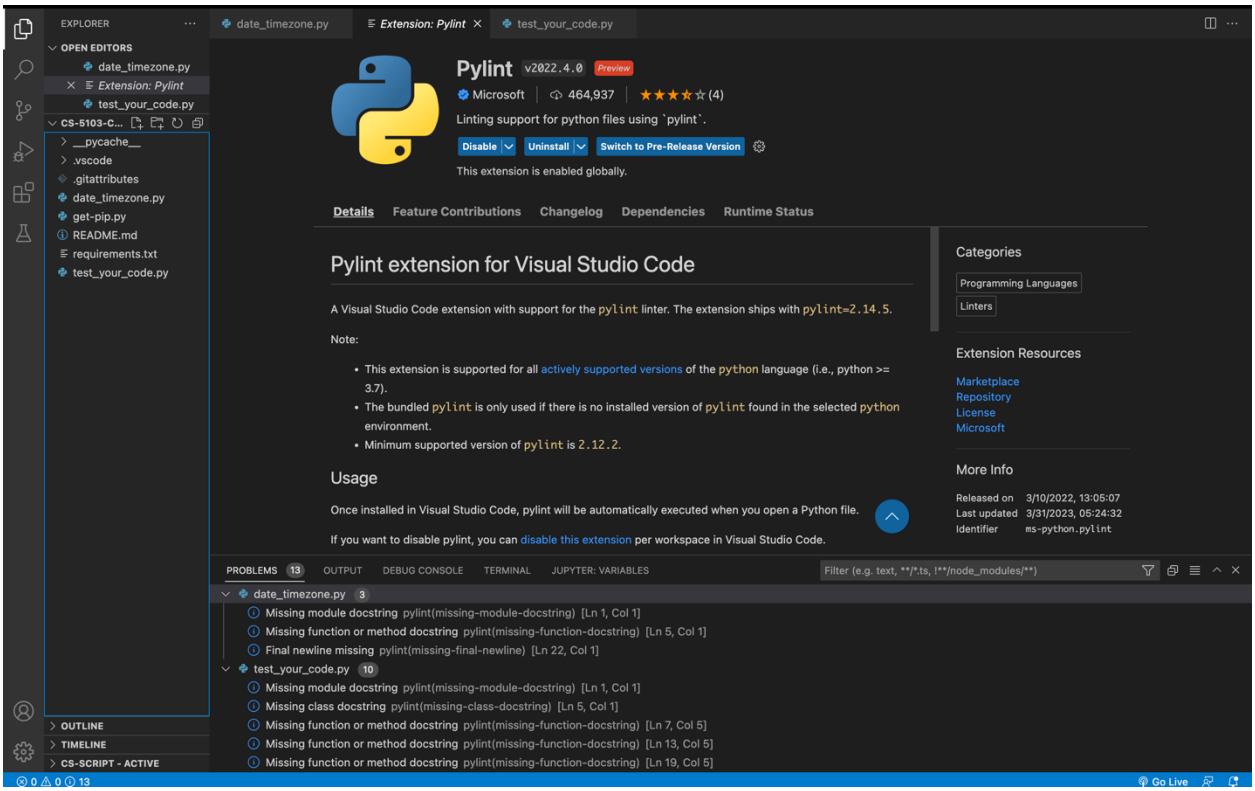
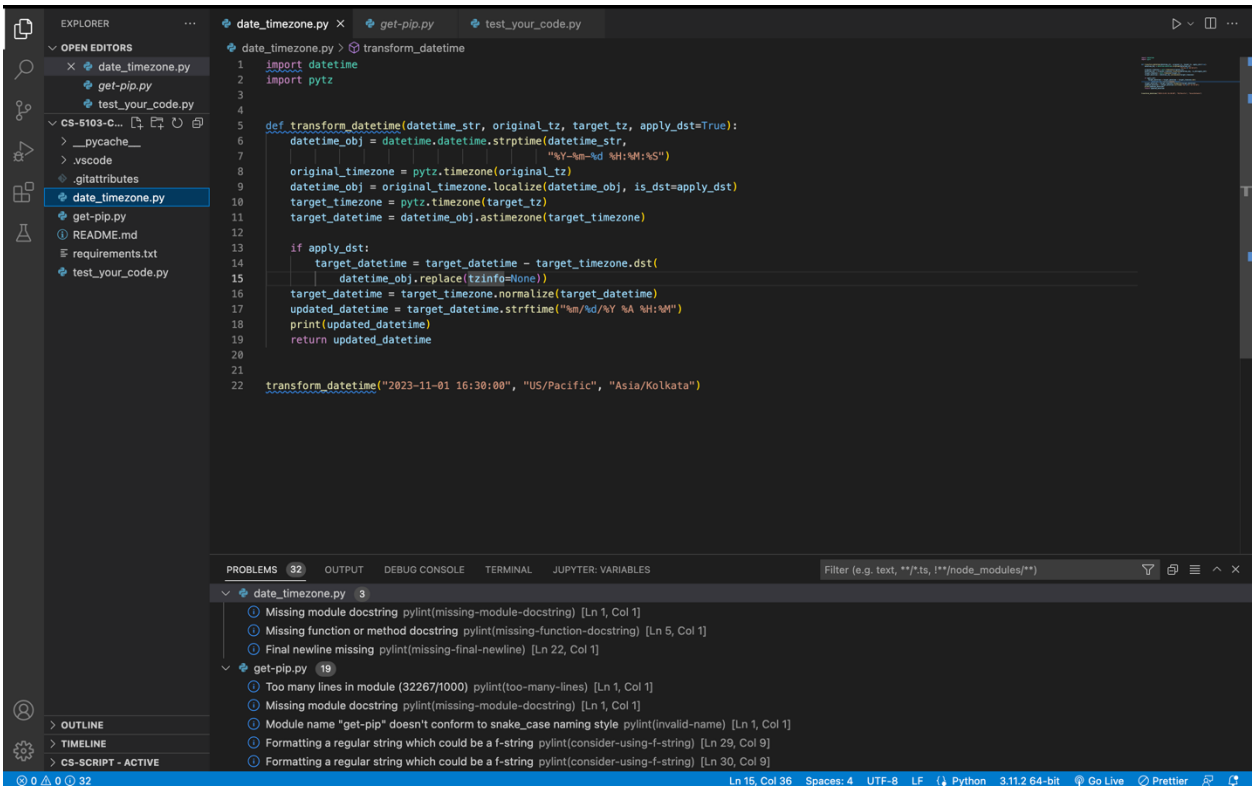
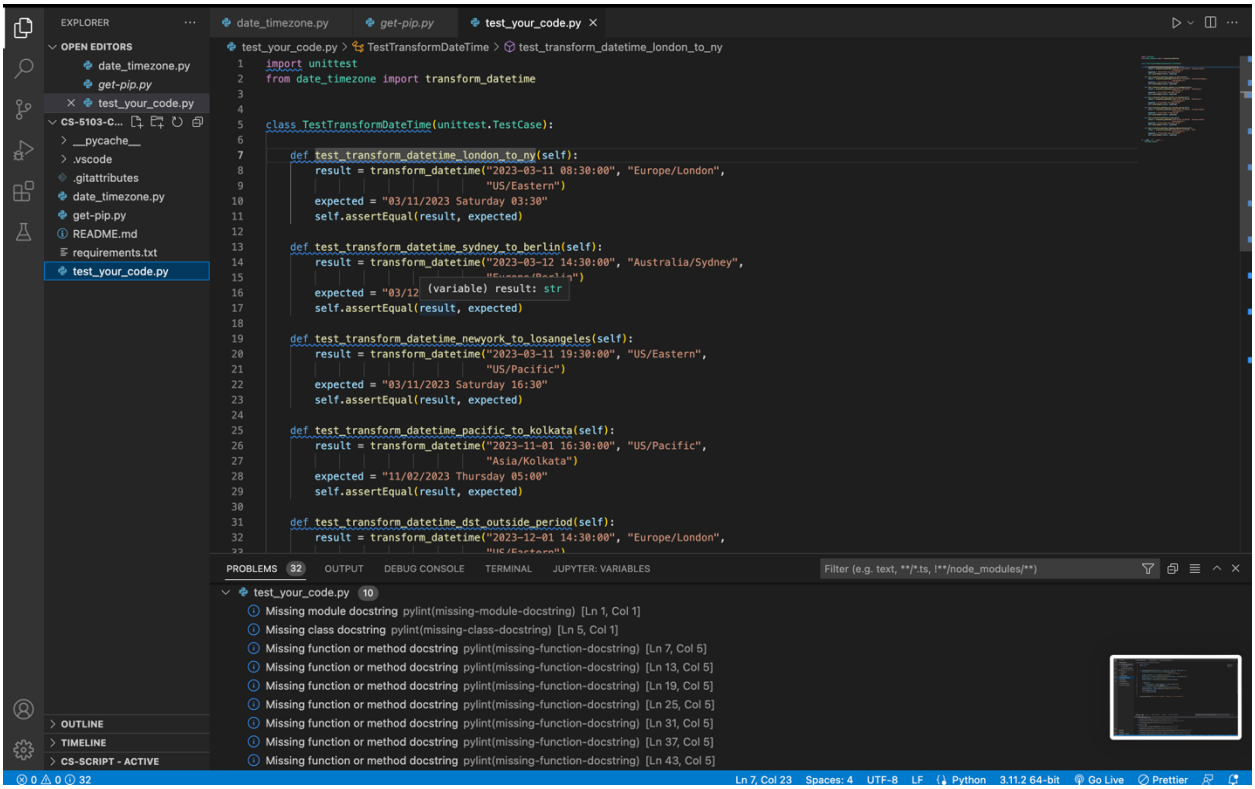
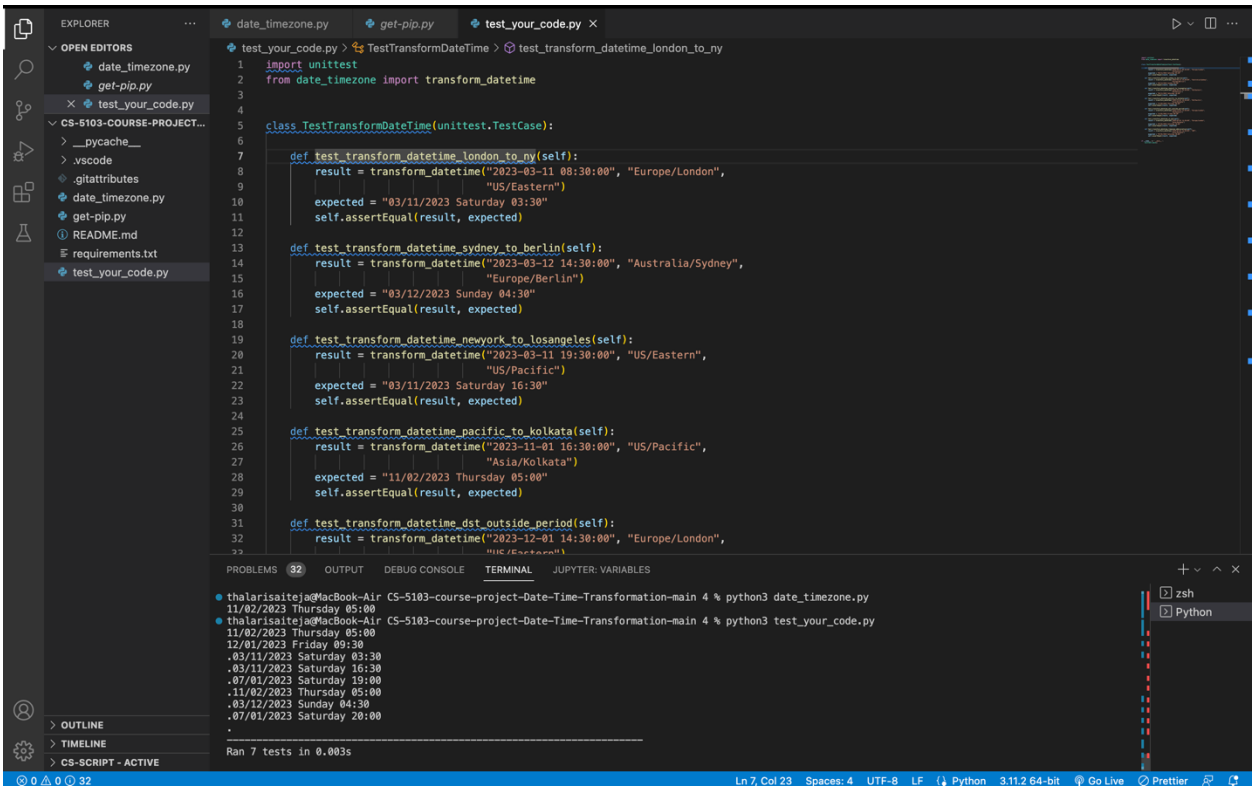

EXPLORER
OPEN EDITORS
date_timezone.py
Extension: Pylint
test_your_code.py
CS-5103-C...
__pycache__
.vscode
.gitattributes
date_timezone.py
get-pip.py
README.md
requirements.txt
test_your_code.py
OUTLINE
TIMELINE
CS-SCRIPT - ACTIVE
date_timezone.py
Extension: Pylint
test_your_code.py
Pylint v2022.4.0 Preview
Microsoft 464,937 ★★★★☆(4)
Linting support for python files using `pylint`.
Disable Uninstall Switch to Pre-Release Version
This extension is enabled globally.
Details Feature Contributions Changelog Dependencies Runtime Status
Pylint extension for Visual Studio Code
A Visual Studio Code extension with support for the pylint linter. The extension ships with pylint=2.14.5.
Note:
This extension is supported for all actively supported versions of the python language (i.e., python >= 3.7).
The bundled pylint is only used if there is no installed version of pylint found in the selected python environment.
Minimum supported version of pylint is 2.12.2.
Usage
Once installed in Visual Studio Code, pylint will be automatically executed when you open a Python file.
If you want to disable pylint, you can disable this extension per workspace in Visual Studio Code.
Categories
Programming Languages
Linters
Extension Resources
Marketplace
Repository
License
Microsoft
More Info
Released on 3/10/2022, 13:05:07
Last updated 3/31/2023, 05:24:32
Identifier ms-python.pylint
PROBLEMS 13 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER: VARIABLES
Filter (e.g. text, **/*.ts, !**/node_modules/**)
date_timezone.py 3
Missing module docstring pylint(missing-module-docstring) [Ln 1, Col 1]
Missing function or method docstring pylint(missing-function-docstring) [Ln 5, Col 1]
Final newline missing pylint(missing-final-newline) [Ln 22, Col 1]
test_your_code.py 10
Missing module docstring pylint(missing-module-docstring) [Ln 1, Col 1]
Missing class docstring pylint(missing-class-docstring) [Ln 5, Col 1]
Missing function or method docstring pylint(missing-function-docstring) [Ln 7, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 13, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 19, Col 5]
0 0 13
Go Live


EXPLORER
OPEN EDITORS
date_timezone.py
get-pip.py
test_your_code.py
CS-5103-C...
__pycache__
.vscode
.gitattributes
date_timezone.py
get-pip.py
README.md
requirements.txt
test_your_code.py
OUTLINE
TIMELINE
CS-SCRIPT - ACTIVE
date_timezone.py
get-pip.py
test_your_code.py
date_timezone.py > transform_datetime
import datetime
import pytz
def transform_datetime(datetime_str, original_tz, target_tz, apply_dst=True):
datetime_obj = datetime.datetime.strptime(datetime_str,
"%Y-%m-%d %H:%M:%S")
original_timezone = pytz.timezone(original_tz)
datetime_obj = original_timezone.localize(datetime_obj, is_dst=apply_dst)
target_timezone = pytz.timezone(target_tz)
target_datetime = datetime_obj.astimezone(target_timezone)
if apply_dst:
target_datetime = target_datetime - target_timezone.dst(
datetime_obj.replace(tzinfo=None))
target_datetime = target_timezone.normalize(target_datetime)
updated_datetime = target_datetime.strftime("%m/%d/%Y %A %H:%M")
print(updated_datetime)
return updated_datetime
transform_datetime("2023-11-01 16:30:00", "US/Pacific", "Asia/Kolkata")
PROBLEMS 32 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER: VARIABLES
Filter (e.g. text, **/*.ts, !**/node_modules/**)
date_timezone.py 3
Missing module docstring pylint(missing-module-docstring) [Ln 1, Col 1]
Missing function or method docstring pylint(missing-function-docstring) [Ln 5, Col 1]
Final newline missing pylint(missing-final-newline) [Ln 22, Col 1]
get-pip.py 19
Too many lines in module (32267/1000) pylint(too-many-lines) [Ln 1, Col 1]
Missing module docstring pylint(missing-module-docstring) [Ln 1, Col 1]
Module name "get-pip" doesn't conform to snake_case naming style pylint(invalid-name) [Ln 1, Col 1]
Formatting a regular string which could be a f-string pylint(consider-using-f-string) [Ln 29, Col 9]
Formatting a regular string which could be a f-string pylint(consider-using-f-string) [Ln 30, Col 9]
0 0 32
Ln 15, Col 36 Spaces: 4 UTF-8 LF Python 3.11.2 64-bit Go Live Prettier


EXPLORER
OPEN EDITORS
date_timezone.py
get-pip.py
test_your_code.py
CS-5103-C...
__pycache__
.vscode
.gitattributes
date_timezone.py
get-pip.py
README.md
requirements.txt
test_your_code.py
OUTLINE
TIMELINE
CS-SCRIPT - ACTIVE
test_your_code.py > TestTransformDateTime > test_transform_datetime_london_to_ny
import unittest
from date_timezone import transform_datetime
class TestTransformDateTime(unittest.TestCase):
def test_transform_datetime_london_to_ny(self):
result = transform_datetime("2023-03-11 08:30:00", "Europe/London",
"US/Eastern")
expected = "03/11/2023 Saturday 03:30"
self.assertEqual(result, expected)
def test_transform_datetime_sydney_to_berlin(self):
result = transform_datetime("2023-03-12 14:30:00", "Australia/Sydney",
(variable) result: str
expected = "03/12
self.assertEqual(result, expected)
def test_transform_datetime_newyork_to_losangeles(self):
result = transform_datetime("2023-03-11 19:30:00", "US/Eastern",
"US/Pacific")
expected = "03/11/2023 Saturday 16:30"
self.assertEqual(result, expected)
def test_transform_datetime_pacific_to_kolkata(self):
result = transform_datetime("2023-11-01 16:30:00", "US/Pacific",
"Asia/Kolkata")
expected = "11/02/2023 Thursday 05:00"
self.assertEqual(result, expected)
def test_transform_datetime_dst_outside_period(self):
result = transform_datetime("2023-12-01 14:30:00", "Europe/London",
PROBLEMS 32
OUTPUT
DEBUG CONSOLE
TERMINAL
JUPYTER: VARIABLES
Filter (e.g. text, **/*.ts, !**/node_modules/**)
test_your_code.py 10
Missing module docstring pylint(missing-module-docstring) [Ln 1, Col 1]
Missing class docstring pylint(missing-class-docstring) [Ln 5, Col 1]
Missing function or method docstring pylint(missing-function-docstring) [Ln 7, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 13, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 19, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 25, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 31, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 37, Col 5]
Missing function or method docstring pylint(missing-function-docstring) [Ln 43, Col 5]
0 0 32
Ln 7, Col 23
Spaces: 4
UTF-8
LF
Python
3.11.2 64-bit
Go Live
Prettier


EXPLORER
OPEN EDITORS
date_timezone.py
get-pip.py
test_your_code.py
CS-5103-COURSE-PROJECT...
__pycache__
.vscode
.gitattributes
date_timezone.py
get-pip.py
README.md
requirements.txt
test_your_code.py
OUTLINE
TIMELINE
CS-SCRIPT - ACTIVE
test_your_code.py > TestTransformDateTime > test_transform_datetime_london_to_ny
import unittest
from date_timezone import transform_datetime
class TestTransformDateTime(unittest.TestCase):
def test_transform_datetime_london_to_ny(self):
result = transform_datetime("2023-03-11 08:30:00", "Europe/London",
"US/Eastern")
expected = "03/11/2023 Saturday 03:30"
self.assertEqual(result, expected)
def test_transform_datetime_sydney_to_berlin(self):
result = transform_datetime("2023-03-12 14:30:00", "Australia/Sydney",
"Europe/Berlin")
expected = "03/12/2023 Sunday 04:30"
self.assertEqual(result, expected)
def test_transform_datetime_newyork_to_losangeles(self):
result = transform_datetime("2023-03-11 19:30:00", "US/Eastern",
"US/Pacific")
expected = "03/11/2023 Saturday 16:30"
self.assertEqual(result, expected)
def test_transform_datetime_pacific_to_kolkata(self):
result = transform_datetime("2023-11-01 16:30:00", "US/Pacific",
"Asia/Kolkata")
expected = "11/02/2023 Thursday 05:00"
self.assertEqual(result, expected)
def test_transform_datetime_dst_outside_period(self):
result = transform_datetime("2023-12-01 14:30:00", "Europe/London",
PROBLEMS 32
OUTPUT
DEBUG CONSOLE
TERMINAL
JUPYTER: VARIABLES
thalarisaiteja@MacBook-Air CS-5103-course-project-Date-Time-Transformation-main 4 % python3 date_timezone.py
11/02/2023 Thursday 05:00
thalarisaiteja@MacBook-Air CS-5103-course-project-Date-Time-Transformation-main 4 % python3 test_your_code.py
11/02/2023 Thursday 05:00
12/01/2023 Friday 09:30
.03/11/2023 Saturday 03:30
.03/11/2023 Saturday 16:30
.07/01/2023 Saturday 19:00
.11/02/2023 Thursday 05:00
.03/12/2023 Sunday 04:30
.07/01/2023 Saturday 20:00
.
Ran 7 tests in 0.003s
zsh
Python
0 0 32
Ln 7, Col 23
Spaces: 4
UTF-8
LF
Python
3.11.2 64-bit
Go Live
Prettier