Python TUTORIAL

Python assertIsNone()

Summary: in this tutorial, you'll learn how to use the Python `assertIsNone()` method to test if an expression is `None`.

Introduction to the Python assertIsNone() method

The `assertIsNone()` is a method of the `TestCase` class of the unittest (https://www.pythontutorial.net/python-unit-testing/python-unittest/) module. The `assertIsNone()` test if an expression is None (https://www.pythontutorial.net/advanced-python/python-none/) :

```
assertIsNone(expr, msg=None)
```

If the `expr` is `None`, the test passes. Otherwise, the test will fail.

The `msg` is optional. It'll be displayed in the test result if the test fails.

Python assertIsNone() method examples

Let's take some examples of using the `assertIsNone()` method.

1) Using assertIsNone() with a success case

The following example uses the `assertIsNone()` to test if the message variable is None:

```
import unittest


class TestNone(unittest.TestCase):
    def test_variable_none(self):
        message = None
        self.assertIsNone(message)
```

Run the test:

```
python -m unittest -v
```

Output:

```
test_variable_none (test_none.TestNone) ... ok

----------------------------------------------------------------------
Ran 1 test in 0.000s
```

2) Using assertIsNone() with a failed case

The following example uses the `assertIsNone()` method to test if the `message` variable is None:

```
import unittest


class TestNone(unittest.TestCase):
    def test_variable_not_none(self):
        message = 'Hello'
        self.assertIsNone(message)
```

Run the test:

```
python -m unittest -v
```

Output:

```
test_variable_not_none (test_none.TestNone) ... FAIL

======================================================================
FAIL: test_variable_not_none (test_none.TestNone)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "D:\python-unit-testing\test_none.py", line 7, in test_variable_not_non
    self.assertIsNone(message)
AssertionError: 'Hello' is not None

----------------------------------------------------------------------
Ran 1 test in 0.001s

FAILED (failures=1)
```

Since the message is `'Hello'`, it is not None. Therefore, the test failed.

3) Using assertIsNone() with a failed case with a message

The following example uses the `assertIsNone()` to test if the message variable is None. Also, we show a message when the test fails:

```
import unittest


class TestNone(unittest.TestCase):
    def test_variable_not_none(self):
        message = 'Hello'
        self.assertIsNone(
            message,
            f'The message is "{message}" so it is not None.'
        )
```

Run the test:

```
python -m unittest -v
```

Output:

```
test_variable_not_none (test_none.TestNone) ... FAIL

======================================================================
FAIL: test_variable_not_none (test_none.TestNone)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "D:\python-unit-testing\test_none.py", line 7, in test_variable_not_non
    self.assertIsNone(
AssertionError: 'Hello' is not None : The message is "Hello" so it is not None

----------------------------------------------------------------------
Ran 1 test in 0.001s

FAILED (failures=1)
```

Python assertIsNotNone() method

The `assertIsNotNone()` is opposite of the `assertIsNone()` method. The `assertIsNotNone()` method tests if a variable is not None.

```
assertIsNotNone(expr, msg=None)
```

The test passes if the `expr` is not `None` or fails otherwise. For example:

```
import unittest


class TestNone(unittest.TestCase):
    def test_variable_is_not_none(self):
        message = 'Bye'
        self.assertIsNotNone(message)
```

Run the test:

```
python -m unittest -v
```

Output:

```
test_variable_is_not_none (test_not_none.TestNone) ... ok

----------------------------------------------------------------------
Ran 1 test in 0.001s

OK
```

Summary

- Use the `assertIsNone()` method to test if a variable is `None`.
- use the `assertIsNotNone()` method to test if a variable is not `None`.