Perform unit testing in Python with the Unittest module

• unittest, functions, methods, and assert methods (assertIsInstance, assertEqual, assertTrue, assertIs, assertIn)

What is Unit Testing

Unit testing is a process of verifying the functionality of individual units of code, such as functions or methods, in isolation from other components. Unit testing can help to detect and prevent errors, improve code quality, and facilitate refactoring. In Python, one of the most popular frameworks for unit testing is **Unittest**, which is part of the standard library. **Unittest** provides a set of tools to create test cases, run tests, and report results.

How to Implement Unit Testing

To use Unittest, we need to import the **unittest module** and create a **subclass of unittest.TestCase** that contains the test methods. A test method is a function that starts with the prefix test_ and performs some assertions using the assert methods provided by Unittest. For example, assertIsInstance checks if an object is an instance of a class, assertEqual checks if two objects are equal, assertTrue checks if an expression is True, assertIs checks if two objects are the same object, and assertIn checks if an object is in a container.

Example unittest01.py is an example of a test case that tests some functions that perform arithmetic operations and demonstrates methods assertEqual and assertRaise.

The assertIsInstance() method is a built-in assertion method in Python's unittest module, which helps in validating object classes during unit testing. The assertIsInstance() method checks if an object belongs to a specific class or its subclass and raises an AssertionError if it does not. The syntax of the assertIsInstance() method is:

assertIsInstance(obj, cls, msg=None)

where obj is the object to test, cls is a class or a tuple of classes, and msg is an optional message to display in case of failure. The assertIsInstance() method uses the isinstance() function internally to perform the check.

Example untittesting02.py /shapes.py

Example untittesting03.py /shapes.py

The python unittest assertTrue method is used to check if an expression evaluates to True in a test case. It is one of the assert methods provided by the unittest module to compare an actual result with an expected result.

The syntax of the assertTrue method is:

assertTrue(expr, msg=None)

where expr is the expression to be tested, and msg is an optional message to display in case the test fails.

The assertTrue method passes the test if expr is True, and fails the test if expr is False. The msg parameter can be used to provide more information about the failure.

For example, suppose we have a function that returns True if a number is even, and False otherwise: