

### Szablon klasy testowej

```
plik hpp:
#include <cppunit/extensions/HelperMacros.h>
class FooTest: public CppUnit::TestFixture
{
    CPPUNIT_TEST_SUITE(FooTest);
    CPPUNIT_TEST(testMethod);
    CPPUNIT_TEST_SUITE_END();
public:
    FooTest();
    ~FooTest();
    void setUp();
    void tearDown();
private:
    void testMethod();
};
plik cpp:
CPPUNIT_TEST_SUITE_NAMED_REGISTRATION(FooTest,
"nazwa_zestawu");
CPPUNIT_REGISTRY_ADD_TO_DEFAULT("nazwa_zestawu");
```

### Makra testowe

<b>CPPUNIT_ASSERT</b> (warunek)	Weryfikuje czy <i>warunek</i> ma wartość <code>true</code>
<b>CPPUNIT_ASSERT_MESSAGE</b> (komunikat, warunek)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia
<b>CPPUNIT_FAIL</b> (komunikat)	Wymusza niepowodzenie testu i wyświetla <i>komunikat</i>
<b>CPPUNIT_ASSERT_EQUAL</b> (oczekiwana, faktyczna)	Weryfikuje czy wartości <i>oczekiwana</i> i <i>faktyczna</i> są sobie równe
<b>CPPUNIT_ASSERT_EQUAL_MESSAGE</b> (komunikat, oczekiwana, faktyczna)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia
<b>CPPUNIT_ASSERT_DOUBLES_EQUAL</b> (oczekiwana, faktyczna, delta)	Weryfikuje czy różnica wartości <i>oczekiwana</i> i <i>faktyczna</i> typu <code>double</code> jest nie większa niż <i>delta</i>
<b>CPPUNIT_ASSERT_DOUBLES_EQUAL_MESSAGE</b> (komunikat, oczekiwana, faktyczna, delta)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia

### Makra testowe - wyjątki

<b>CPPUNIT_ASSERT_THROW</b> (wyrażenie, typWyjątku)	Weryfikuje czy wyrażenie rzuca wyjątek typu <i>typWyjątku</i>
<b>CPPUNIT_ASSERT_THROW_MESSAGE</b> (komunikat, wyrażenie, typWyjątku)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia testu
<b>CPPUNIT_ASSERT_NO_THROW</b> (wyrażenie)	Weryfikuje czy wyrażenie <b>nie</b> rzuca wyjątku typu <i>typWyjątku</i>
<b>CPPUNIT_ASSERT_NO_THROW_MESSAGE</b> (komunikat, wyrażenie)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia testu

### Makra testowe - asercje

<b>CPPUNIT_ASSERT_ASSERTION_FAIL</b> (asercja)	Weryfikuje niepowodzenie <i>asercji</i>
<b>CPPUNIT_ASSERT_ASSERTION_FAIL_MESSAGE</b> (komunikat, asercja)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia
<b>CPPUNIT_ASSERT_ASSERTION_PASSED</b> (asercja)	Weryfikuje powodzenie <i>asercji</i>
<b>CPPUNIT_ASSERT_ASSERTION_PASSED_MESSAGE</b> (message, assertion)	jw. oraz wyświetla <i>komunikat</i> w przypadku niepowodzenia

### Zestaw testów

<b>CPPUNIT_TEST_SUITE</b> (klasa)	Rozpoczyna zestaw testów dla <i>klasy</i>
<b>CPPUNIT_TEST_SUITE_END</b> ()	Kończy zestaw testów
<b>CPPUNIT_TEST_SUITE_SETUP</b> (metoda)	Rejestruje metodę wykonywaną tylko raz przed wszystkimi testami
<b>CPPUNIT_TEST_SUITE_TEAR_DOWN</b> (metoda)	Rejestruje metodę wykonywaną tylko raz po wszystkich testach
<b>CPPUNIT_TEST</b> (metoda)	Dodaje <i>metodę</i> do zestawu
<b>CPPUNIT_TEST_EXCEPT</b> (metoda, wyjątek)	Dodaje do zestawu <i>metodę</i> , która powinna rzucić <i>wyjątek</i>
<b>CPPUNIT_TEST_FAIL</b> (metoda)	Dodaje do zestawu <i>metodę</i> , której test powinien zakończyć się niepowodzeniem