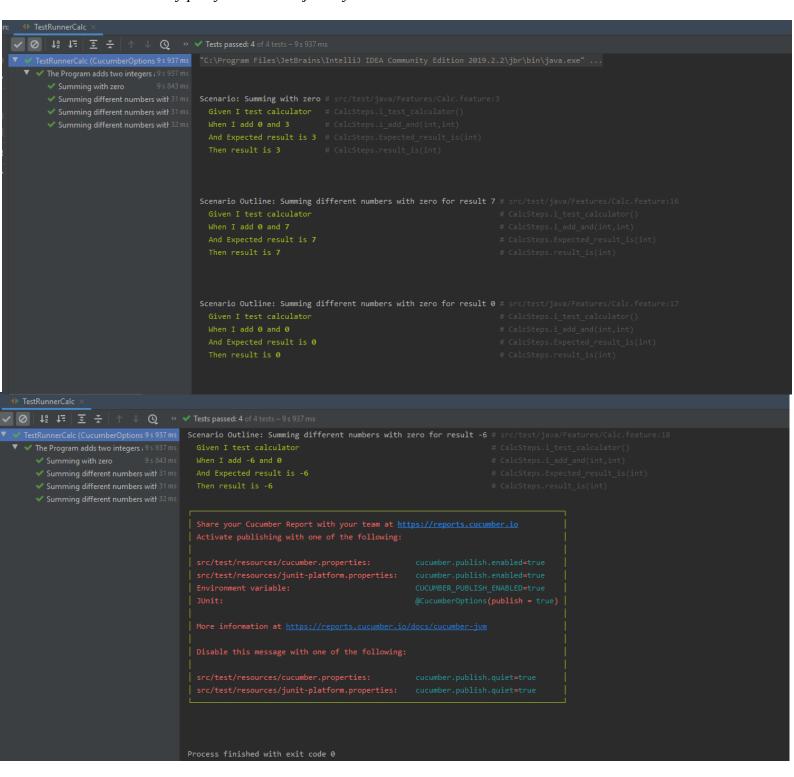
Kalkulator – funkcja dodawania Na przykładzie wykonanego testu jednostkowego

Wynik Testu:

```
Scenario: Summing with zero # src/test/java/Features/Calc.feature:3
  Given I test calculator # CalcSteps.i_test_calculator()
  When I add 0 and 3 # CalcSteps.i_add_and(int,int)
  And Expected result is 3 # CalcSteps.Expected_result_is(int)
  Then result is 3 # CalcSteps.result_is(int)
Scenario Outline: Summing different numbers with zero for result 7 # src/test/java/Features/Calc.feature:16
  Given I test calculator
                                                                  # CalcSteps.i_test_calculator()
  When I add 0 and 7
                                                                  # CalcSteps.i_add_and(int,int)
  And Expected result is 7
                                                                  # CalcSteps.Expected_result_is(int)
  Then result is 7
                                                                  # CalcSteps.result_is(int)
Scenario Outline: Summing different numbers with zero for result 0 # src/test/java/Features/Calc.feature:17
                                                                  # CalcSteps.i_test_calculator()
  Given I test calculator
  When I add 0 and 0
                                                                  # CalcSteps.i_add_and(int,int)
  And Expected result is 0
                                                                  # CalcSteps.Expected_result_is(int)
  Then result is 0
                                                                  # CalcSteps.result_is(int)
Scenario Outline: Summing different numbers with zero for result -6 # src/test/java/Features/Calc.feature:18
  Given I test calculator
                                                                   # CalcSteps.i_test_calculator()
  When I add -6 and 0
                                                                   # CalcSteps.i_add_and(int,int)
  And Expected result is -6
                                                                   # CalcSteps.Expected_result_is(int)
  Then result is -6
                                                                   # CalcSteps.result_is(int)
  Share your Cucumber Report with your team at https://reports.cucumber.io
  Activate publishing with one of the following:
  src/test/resources/cucumber.properties:
                                                  cucumber.publish.enabled=true
  src/test/resources/junit-platform.properties: cucumber.publish.enabled=true
  Environment variable:
                                                   CUCUMBER_PUBLISH_ENABLED=true
  JUnit:
                                                   @CucumberOptions(publish = true)
  More information at https://reports.cucumber.io/docs/cucumber-jvm
  Disable this message with one of the following:
  src/test/resources/cucumber.properties:
                                                  cucumber.publish.quiet=true
  src/test/resources/junit-platform.properties:
                                                  cucumber.publish.quiet=true
```

Test został zdany pomyślnie. Poniżej zrzuty z ekranu.



Komentarz do testu:

Na podstawie wykonanych testów sprawdzających działanie programu Calc, dodawanie dwóch liczb całkowitych można stwierdzić, że działanie programu jest poprawne.

- 1. Test pierwszy dodawanie liczb 0 i 3 przeszedł pomyślnie z rezultatem 3.
- 2. Test drugi dodawanie liczb 0 i 7 przeszedł pomyślnie z rezultatem 7.
- 3. Test trzeci dodawanie liczb 0 i 0 przeszedł pomyślnie z rezultatem 0.
- 4. Test czwarty dodawanie liczb -6 i 0 przeszedł pomyślnie z rezultatem -6.