

Zadanie – Testy jednostkowe

Utwórz w języku Java metody, które generują liczby oraz pobierają tekst od użytkownika z walidacją długości. Następnie napisz testy jednostkowe JUnit dla obu metod.

1. Utwórz klasę `GeneratorWniosku`

W klasie mają znaleźć się **dwie metody**:

Metoda 1 — `public int generujLiczbe()`

Wymagania:

Metoda ma generować **losową liczbę całkowitą z zakresu 1–100**.

Metoda 2 — `public String wprowadzTekst()`

Wymagania:

1. Metoda ma **przyjąć od użytkownika dowolny tekst**.
2. Tekst ten ma zostać zwrócony tylko wtedy, jeśli:
 - o jego długość jest **od 1 do 20 znaków**.

2. Przygotuj testy jednostkowe

Stwórz klasę testową `GeneratorWnioskuTest`.

2.2 Testy dla `generujLiczbe()`

Napisz testy:

Test 1 — Czy liczba ≥ 1

Test 2 — Czy liczba ≤ 100

2.3 Testy dla `wprowadzTekst()`

Napisz testy:

Sprawdź, czy metoda zwraca właściwą długość tekstu.

Test 1 — Podanie poprawnego tekstu

Sprawdź, czy metoda zwraca właściwą długość tekstu.

3. Struktura projektu

```
src/  
└─ main/java/  
    └─ GeneratorWniosku.java  
  
└─ test/java/  
    └─ GeneratorWnioskuTest.java
```

Dla chętnych:

Jeśli tekst jest:

- *pusty,*
- *dłuższy niż 20 znaków,*
metoda powinna wyrzucić wyjątek typu `IllegalArgumentException`.

Sprawdź, czy metoda zwraca właściwą długość tekstu.

Test 2 — Tekst pusty

Test 3 — Tekst > 20 znaków

Sprawdź, czy rzucony jest wyjątek.

Podpowiedź: użyj `assertThrows()`.

Teoria:

Metody oznaczone adnotacją `@Test` to testy. Metody te nie mogą zwracać żadnej wartości, nie mogą być prywatne ani statyczne.

Wewnątrz testów używa się asercji. Asercje dostarczone przez JUnit zgrupowane są wewnątrz klasy `Assertions`. W przykładach powyżej użyłem asercji `assertEquals`.

Zobacz:

<https://www.samouczekprogramisty.pl/testy-jednostkowe-z-junit5/>