

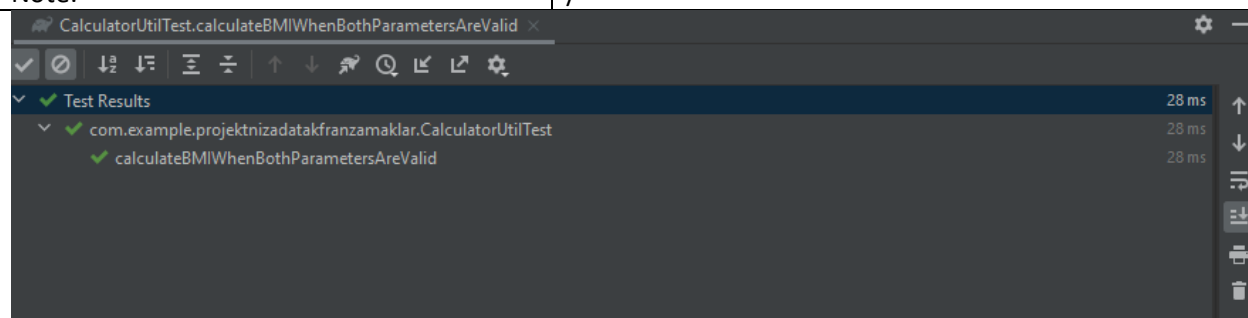
# TESTNA IZVJEŠĆA ZA

## “ProjektniZadatakFranZamaklar”

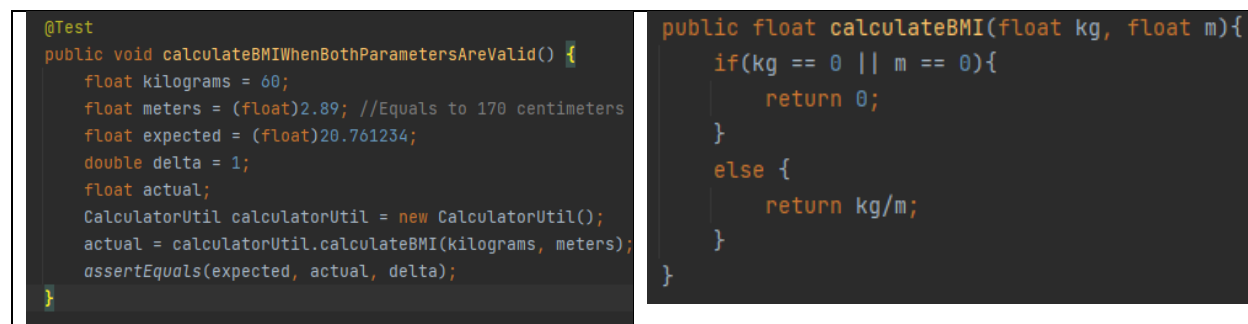
### [MTTPP – Projektni zadatak]

**Fran Zamaklar, 1.diplomski, izborni blok DRC, GRUPA LV2**

Test ID:	CalculatorUtilTest: calculateBMIWhenBothParametersAreValid
Area of functionality:	Testiranje primarne funkcionalnosti aplikacije – izračun BMI na temelju predanih parametara vrijednosti
Objective:	Provjera validnosti rada kalkulatora prilikom davanja validnih argumenata
Test Case Results:	Ispravno izvršen testni slučaj
Note:	/



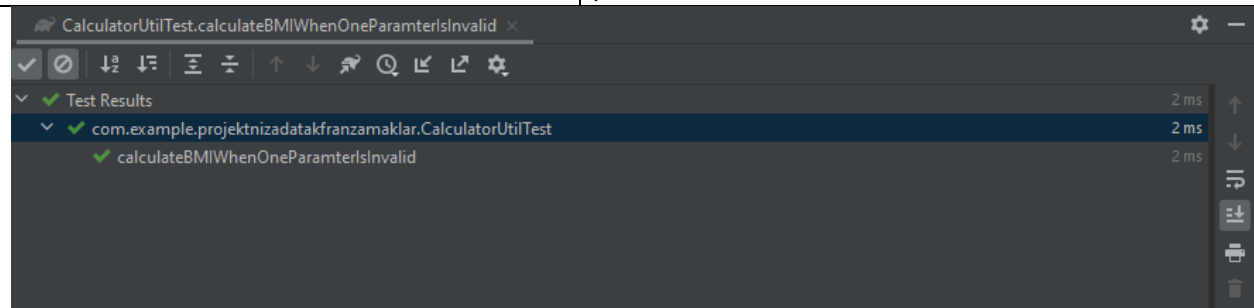
*Slika 1. Uspješno izvršeni Unit testni slučaj*



*Slika 2. Testna metoda navedenog testnog slučaja*

*Slika 3. Metoda u klasi CalculatorUtil koja ostvaruje implementaciju Unit testiranja*

<b>Test ID:</b>	CalculatorUtilTest: calculateBMIWhenOneParameterIsInvalid
<b>Area of functionality:</b>	Testiranje primarne funkcionalnosti aplikacije – izračun BMI na temelju predanih parametara vrijednosti
<b>Objective:</b>	Provjera validnosti rada kalkulatora prilikom davanja invalidnih argumenata
<b>Test Case Results:</b>	Uspješno izvršeni testni slučaj
<b>Note:</b>	/

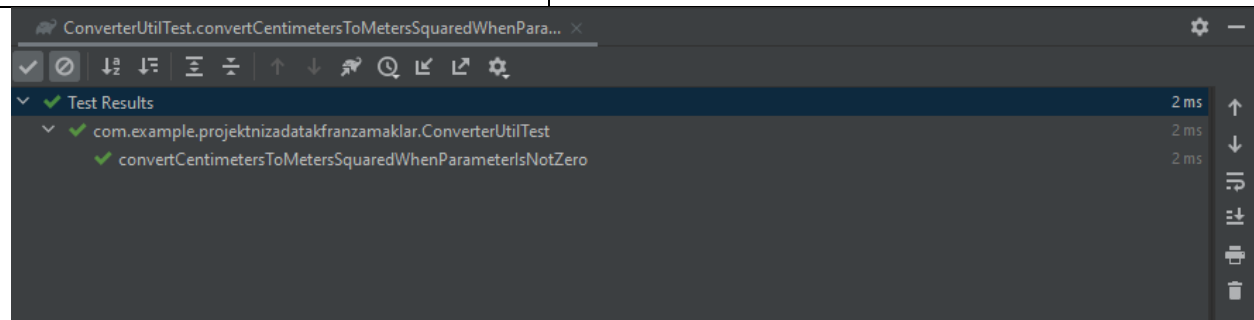


*Slika 4. Uspješno izvršeni Unit testni slučaj*

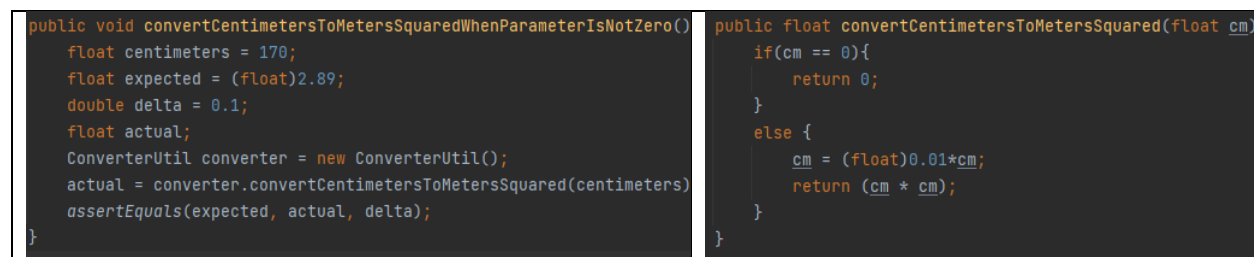
```
@Test
public void calculateBMIWhenOneParameterIsInvalid() {
    float kilograms = 0;
    float meters = (float)2.89; //Equals to 170 centimeters
    float expected = 0;
    double delta = 1;
    float actual;
    CalculatorUtil calculatorUtil = new CalculatorUtil();
    actual = calculatorUtil.calculateBMI(kilograms, meters);
    assertEquals(expected, actual, delta);
}
```

*Slika 5. Testna metoda navedenog testnog slučaja*

<b>Test ID:</b>	ConverterUtilTest: convertCentimetersToMetersSquaredWhenParameterIsNotZero
<b>Area of functionality:</b>	Pretvorba unesenih centimetara u metre kvadratne
<b>Objective:</b>	Provjera funkcionalnosti rada funkcije pretvorbe centimetara u metre kvadratne kada su uneseni validni parametri (brojevi veći od nule)
<b>Test Case Results:</b>	Uspješno izvršen testni slučaj
<b>Note:</b>	Korisnički unosi se prvobitno konvertiraju iz tipa <i>EditText</i> na način da se dohvati u obliku <i>stringa</i> koji se parsira u <i>int</i> koji bi se naposljetku castao u <i>float</i> . Zatim se pristupa obradi centimetara u metre.



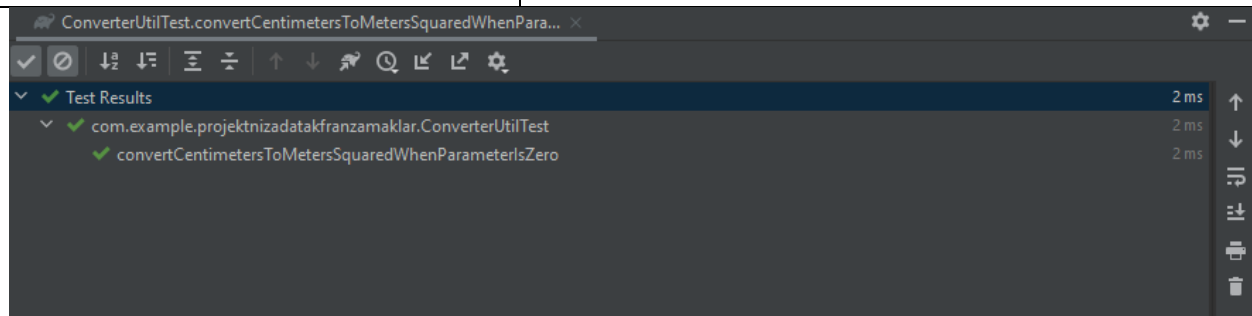
**Slika 6. Uspješno izvršeni Unit testni slučaj**



**Slika 7. Testna metoda navedenog testnog slučaja**

**Slika 8. Metoda u klasi ConverterUtil koja ostvaruje implementaciju Unit testiranja**

Test ID:	ConverterUtilTest: convertCentimetersToMetersSquaredWhenParameterIsZero
Area of functionality:	Pretvorba unesenih centimetara u metre kvadratne
Objective:	Provjera funkcionalnosti rada funkcije pretvorbe centimetara u metre kvadratne kada je kao parametar unesena nula
Test Case Results:	Uspješno izvršen testni slučaj
Note:	Korisnički unosi se prvobitno konvertiraju iz tipa <i>EditText</i> na način da se dohvati u obliku <i>stringa</i> koji se parsira u <i>int</i> koji bi se naposljetku castao u <i>float</i> . Zatim se pristupa obradi centimetara u metre.

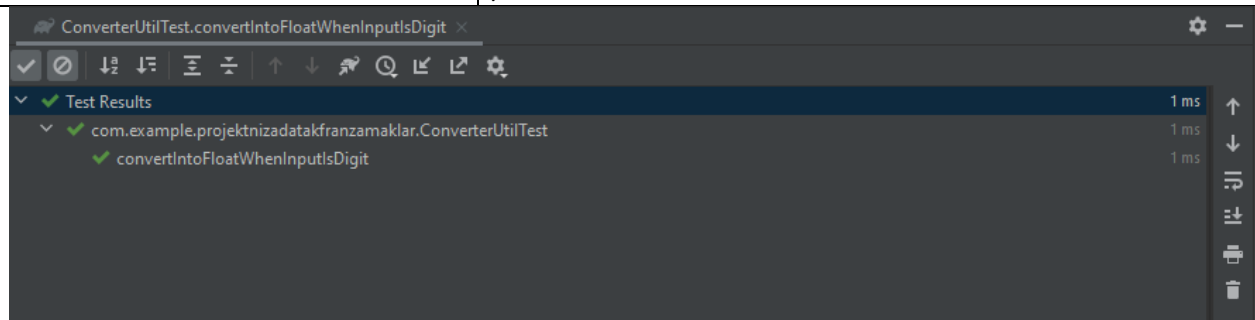


Slika 9. Uspješno izvršen Unit testni slučaj

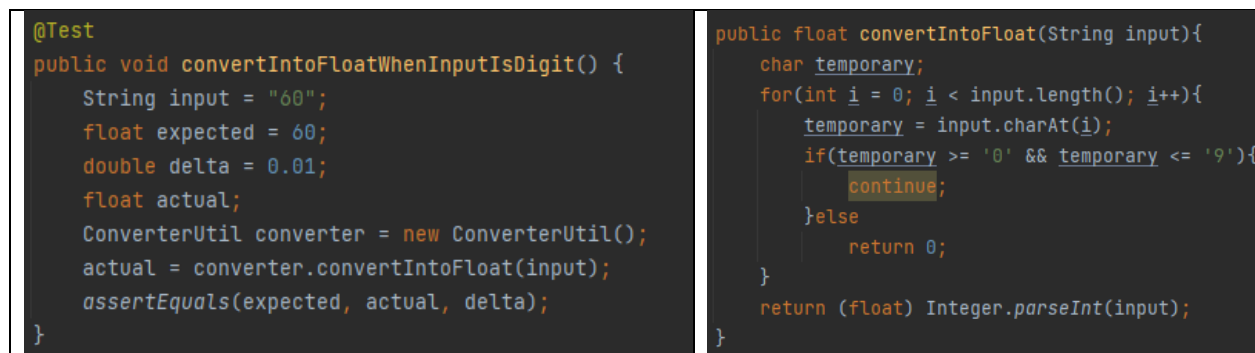
```
@Test
public void convertCentimetersToMetersSquaredWhenParameterIsZero() {
    float centimeters = 0;
    float expected = 0;
    double delta = 0.1;
    float actual;
    ConverterUtil converter = new ConverterUtil();
    actual = converter.convertCentimetersToMetersSquared(centimeters);
    assertEquals(expected, actual, delta);
}
```

Slika 10. Testna metoda navedenog testnog slučaja

<b>Test ID:</b>	ConverterUtilTest: convertIntoFloatWhenInputIsDigit
<b>Area of functionality:</b>	Pretvorba korisnikova unosa u <i>float</i> tip
<b>Objective:</b>	Provjera funkcionalnosti rada funkcije pretvorbe korisnikova unosa u <i>float</i> kada je za parametar unesen validni podatak (broj veći od nule i input je brojka)
<b>Test Case Results:</b>	Uspješno izvršen testni slučaj
<b>Note:</b>	/



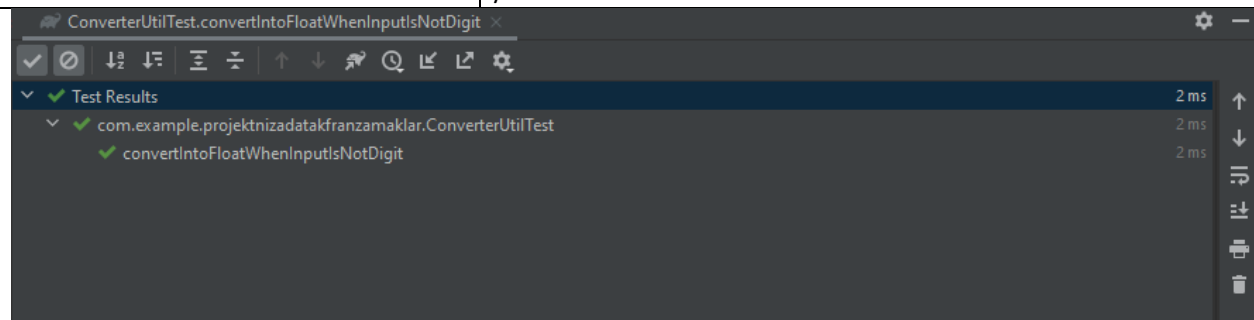
**Slika 11. Uspješno izvršen Unit testni slučaj**



**Slika 12. Testna metoda navedenog testnog slučaja**

**Slika 13. Metoda u klasi ConverterUtil koja ostvaruje implementaciju Unit testiranja**

<b>Test ID:</b>	ConverterUtilTest: convertIntoFloatWhenInputsNotDigit
<b>Area of functionality:</b>	Pretvorba korisnikova unosa u <i>float</i> tip
<b>Objective:</b>	Provjera funkcionalnosti rada funkcije pretvorbe korisnikova unosa u <i>float</i> kada je za parametar unesen invalidni podatak (podatak je znak drugačiji od brojke)
<b>Test Case Results:</b>	Uspješno izvršen testni slučaj
<b>Note:</b>	/

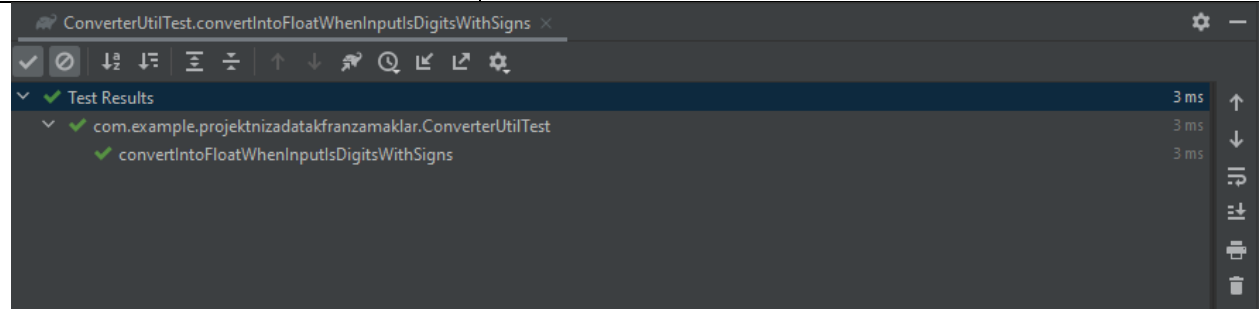


**Slika 14. Uspješno izvršen Unit testni slučaj**

```
@Test
public void convertIntoFloatWhenInputIsNotDigit() {
    String input = " ";
    float expected = 0;
    double delta = 0.01;
    float actual;
    ConverterUtil converter = new ConverterUtil();
    actual = converter.convertIntoFloat(input);
    assertEquals(expected, actual, delta);
}
```

**Slika 15. Testna metoda navedenog testnog slučaja**

<b>Test ID:</b>	ConverterUtilTest: onvertIntoFloatWhenInputIsNotDigit
<b>Area of functionality:</b>	Pretvorba korisnikova unosa u <i>float</i> tip
<b>Objective:</b>	Provjera funkcionalnosti rada funkcije pretvorbe korisnikova unosa u <i>float</i> kada je za parametar unesen invalidni podatak (podatak se sastoji od brojke i od znakova različitih od brojki)
<b>Test Case Results:</b>	Uspješno izvršen testni slučaj
<b>Note:</b>	/



*Slika 16. Uspješno izvršen Unit testni slučaj*

```
@Test
public void convertIntoFloatWhenInputIsDigitsWithSigns() {
    String input = "40!";
    float expected = 0;
    double delta = 0.01;
    float actual;
    ConverterUtil converter = new ConverterUtil();
    actual = converter.convertIntoFloat(input);
    assertEquals(expected, actual, delta);
}
```

*Slika 17. Testna metoda navedenog testnog slučaja*