SD2 - Worksheet 2 - 6%

GRIFFITH COLLEGE DUBLIN

Student name:	**Roman Polishchuk**				
Student number:	**3135838**				
Faculty:	**Computing Science**				
Course:	**BSCH/BSCO/EXCH**		**Stage/year:**	**2**	
Subject:	**Software Development 2**				
Study Mode:	Full time	☑		Part-time	
Lecturer Name:	**Gemma Deery**				
Assignment Title:	**Worksheet 2**				
Date due:	**5/3/25**				
Date submitted:	**4/3/25**				

Plagiarism disclaimer:

I understand that plagiarism is a serious offence and have read and understood the college policy on plagiarism. I also understand that I may receive a mark of zero if I have not identified and properly attributed sources which have been used, referred to, or have in any way influenced the preparation of this assignment, or if I have knowingly allowed others to plagiarise my work in this way.

I hereby certify that this assignment is my own work, based on my personal study and/or research, and that I have acknowledged all material and sources used in its preparation. I also certify that the assignment has not previously been submitted for assessment and that I have not copied in part or whole or otherwise plagiarised the work of anyone else, including other students.

Signed: _____ **Date:** _____

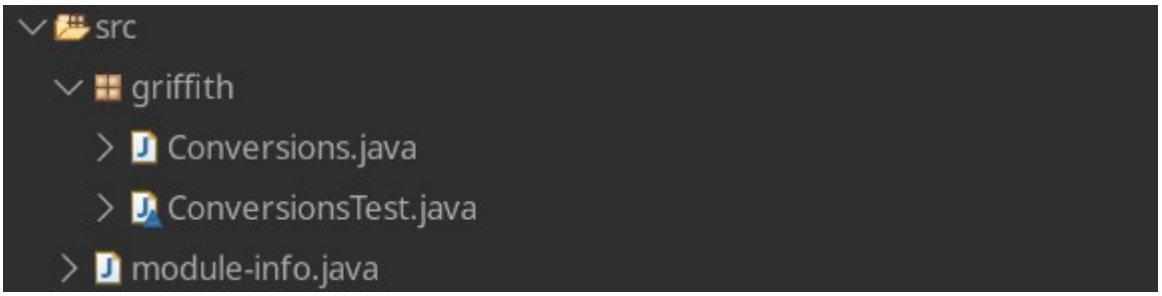
Please note: **Students** MUST **retain a hard / soft copy of** ALL assignments as well as a receipt issued and signed by a member of Faculty as proof of submission.

Repo Link: *https://github.com/romanpolishchuk-griffith-college/WorksheetTwo_3135838*

Task One:

Part one

Create Package and Project files



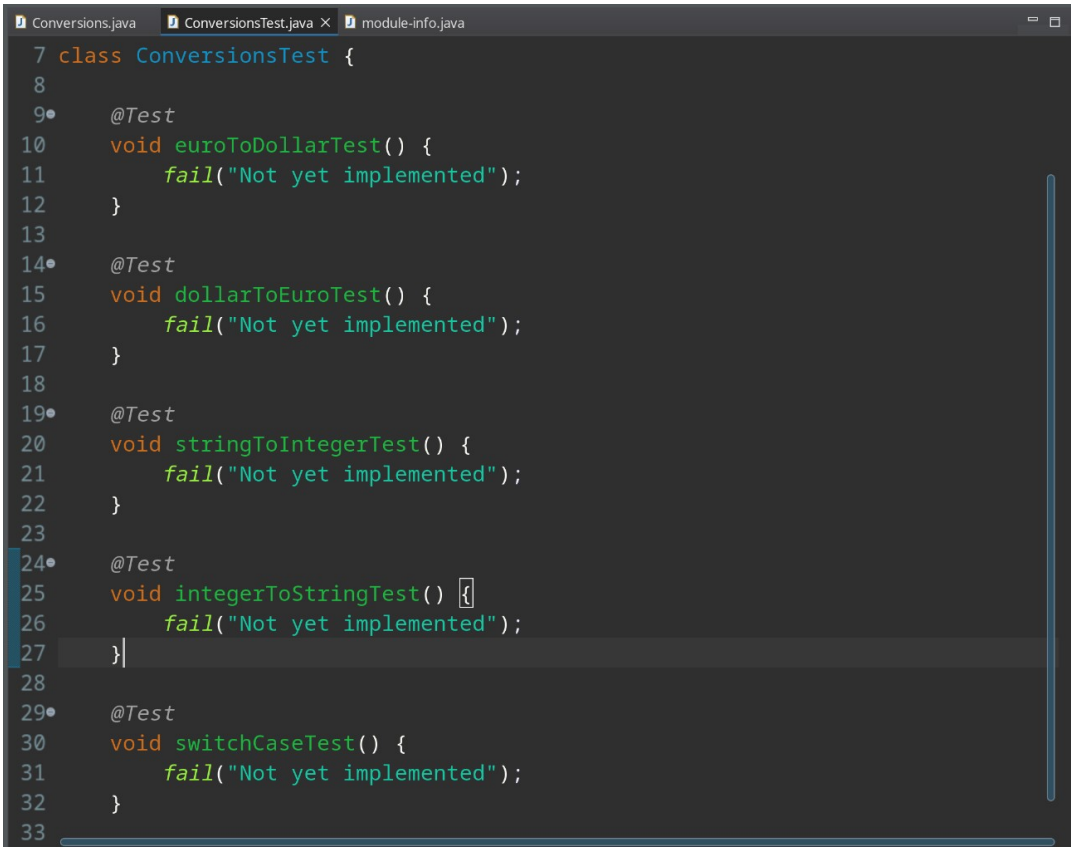
Part two

Add Conversions class stab methods

```java
package griffith;

public class Conversions {
    public double euroToDollar(double euro) {
        return 0.0;
    }
    public double dollarToEuro(double dollar){
        return 0.0;
    }
    public int stringToInteger (String val){
        return 0;
    }
    public String integerToString (int val){
        return "";
    }
    public String switchCase() // change uppercase to lowercase and vice vers
    {
        return "";
    }
}

```

Part three

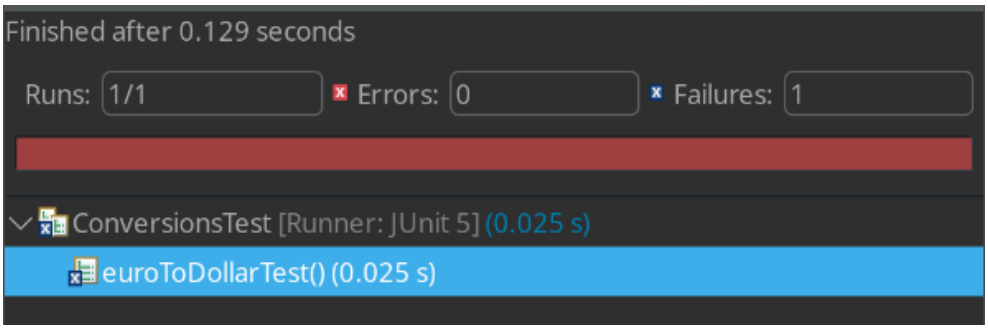
Add ConversionsTest class stab methods



Implement euroToDollarTest method

```
@Test
void euroToDollarTest() {
    //Create Conversions class instance
    Conversions conversions = new Conversions();

    //Test euro is 10
    assertEquals(10.48, conversions.euroToDollar(10));
    //Test euro is 5
    assertEquals(5.24, conversions.euroToDollar(5));
    //Test euro is 0
    assertEquals(0, conversions.euroToDollar(0));
    //Test euro is -999
    assertEquals(0, conversions.euroToDollar(-999));
}
```

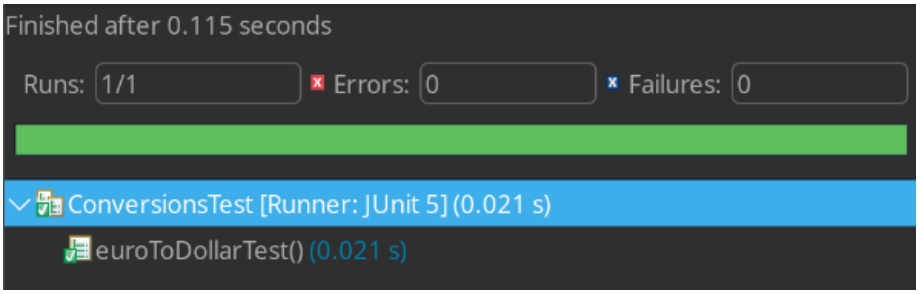
Test fails



Implement euroToDollar method

```
public double euroToDollar(double euro) {
    return Math.max(0, euro * 1.048);
}
```

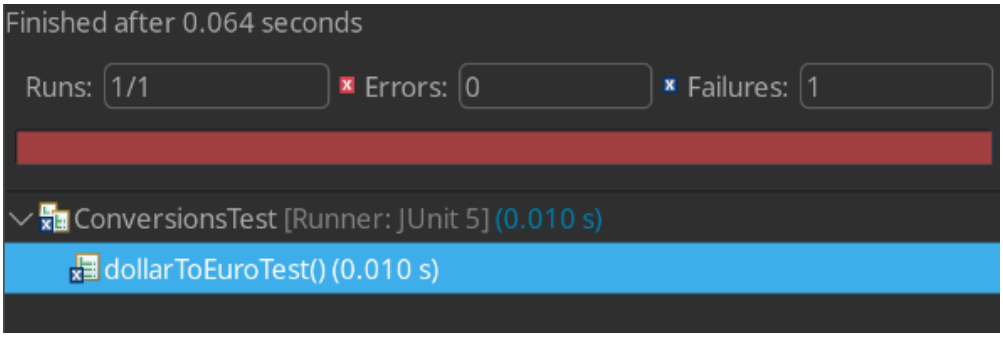
Test passes



Implement dollarToEuroTest method

```java
@Test
void dollarToEuroTest() {
    //Create Conversions class instance
    Conversions conversions = new Conversions();

    //Test dollar is 9.432
    assertEquals(9, conversions.dollarToEuro(9.432));
    //Test dollar is 102.704
    assertEquals(98, conversions.dollarToEuro(102.704));
    //Test dollar is 0
    assertEquals(0, conversions.dollarToEuro(0));
    //Test dollar is -999
    assertEquals(0, conversions.dollarToEuro(-999));
}
```

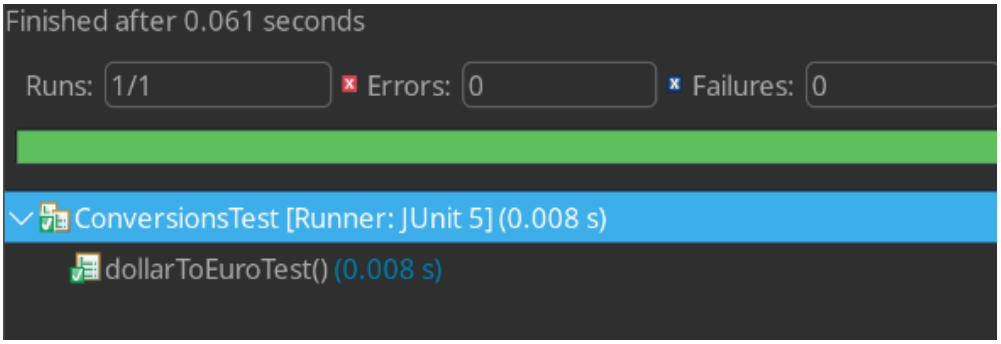
Test fails



Implement dollarToEuro method

```java
public double dollarToEuro(double dollar){
    return Math.round(Math.max(0, dollar * 0.954198473282));
}
```

Test passes



Implement stringToIntegerTest method

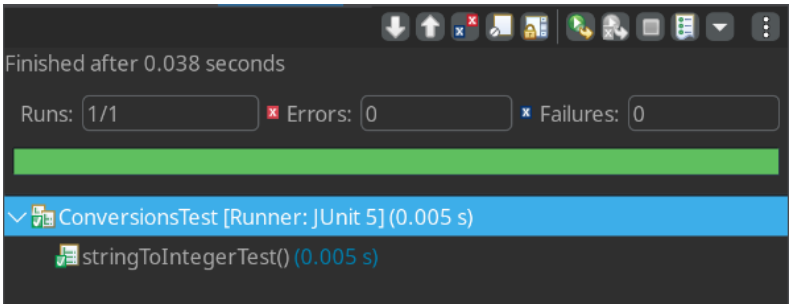
```java
@Test
void stringToIntegerTest() {
    //Create Conversions class instance
    Conversions conversions = new Conversions();

    //Test string is 9
    assertEquals(9, conversions.stringToInteger("9"));
    //Test string is 14.55
    assertEquals(14, conversions.stringToInteger("14.55"));
    //Test string is 0
    assertEquals(0, conversions.stringToInteger("0"));
    //Test string is -999
    assertEquals(-999, conversions.stringToInteger("-999"));
    //Test string is null
    assertEquals(0, conversions.stringToInteger(null));
}
```

Test fails

Implement stringToInteger method

```java
public int stringToInteger (String val){
    if(val == null) {
        return 0;
    }
    return (int)Double.parseDouble(val);
}
```

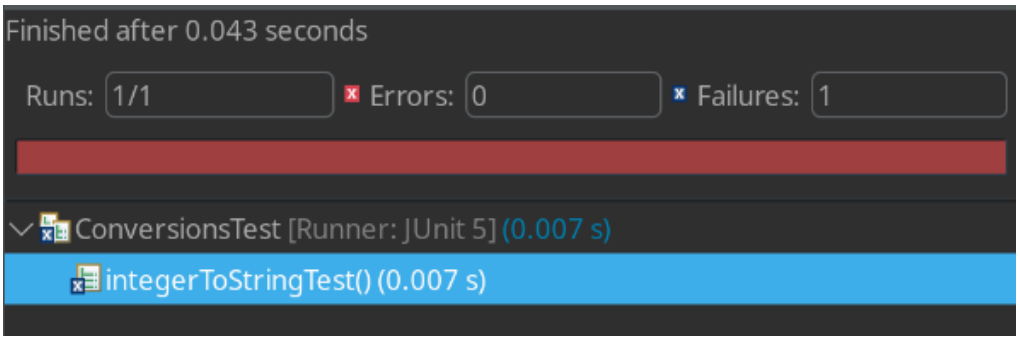
Test passes



Implement integerToStringTest method

```java
@Test
void integerToStringTest() {
    //Create Conversions class instance
    Conversions conversions = new Conversions();

    //Test int is 10
    assertEquals("10", conversions.integerToString(10));
    //Test int is 99
    assertEquals("99", conversions.integerToString(99));
    //Test int is 0
    assertEquals("0", conversions.integerToString(0));
    //Test int is -1000
    assertEquals("-1000", conversions.integerToString(-1000));
}
```

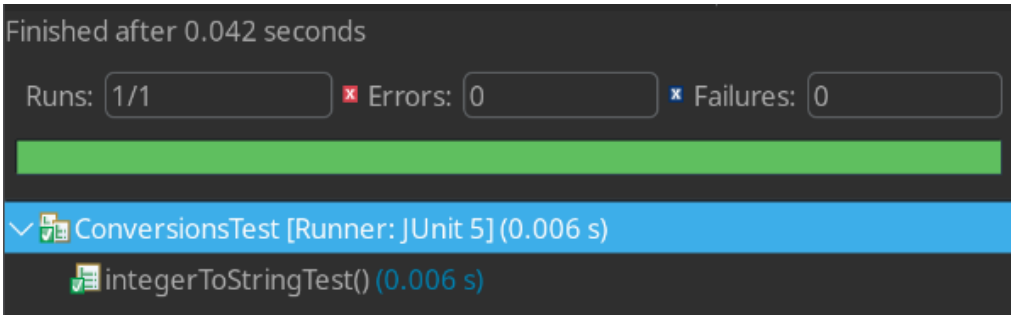
Test fails



Implement integerToString method

```java
public String integerToString (int val){
    return String.valueOf(val);
}
```

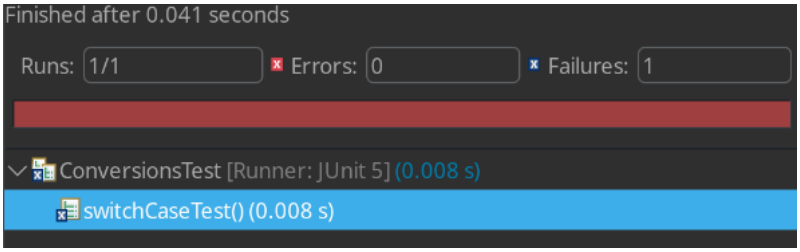
Test passes



Implement switchCaseTest method

```java
@Test
void switchCaseTest() {
    //Create Conversions class instance
    Conversions conversions = new Conversions();

    //Test string is HELLO
    assertEquals("hello", conversions.switchCase("HELLO"));
    //Test string is planet
    assertEquals("PLANET", conversions.switchCase("planet"));
    //Test string is LaPtOp
    assertEquals("lApToP", conversions.switchCase("LaPtOp"));
    //Test string is lower_case_AnD_UPPER_CASE
    assertEquals("LOWER_CASE_aNd_upper_case",
            conversions.switchCase("lower_case_AnD_UPPER_CASE"));
}
```

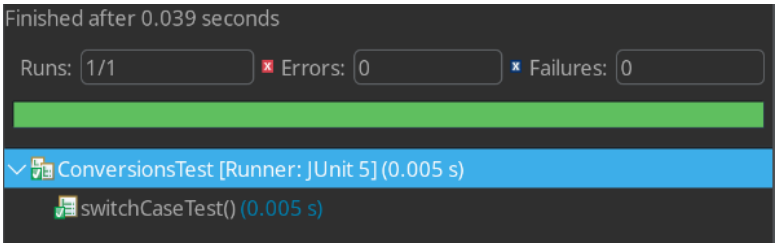
Test fails



Implement switchCase method

```java
//Switch case of a character
public String switchCase(String string) // change uppercase to lowercase
{
    StringBuilder newStringBuilder = new StringBuilder();
    //Check all characters
    for(int i = 0; i < string.length(); i++) {
        //If character is lower case, set it to upper case
        if(Character.isLowerCase(string.charAt(i))) {
            newStringBuilder.
                append(Character.toUpperCase(string.charAt(i)));
        }
        //If character is upper case, set it to lower case
        else {
            newStringBuilder.
                append(Character.toLowerCase(string.charAt(i)));
        }
    }

    //Return final string
    return newStringBuilder.toString();
}
```

Test passes



All tests pass