# **Activity - Unit Test - Banking Account**

For this exercise, you will create two files from scratch: ***Account.java*** and ***AccountTest.java***. You are to develop the source-code and test-code using Java programming language, use Eclipse and JUnit 5 (a.k.a jupiter).

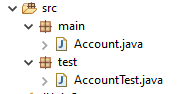
In Eclipse, JUnit comes already installed - ask your instructor if you have trouble setting it up.

The objective of this assignment is to first write a class to demonstrate bank account functionality and then create corresponding JUnit test cases to ensure their correctness.

The ***Account.java***class has to have the following methods and member variables.

* A **double** private member variable to hold the current account **balance**.
* public Account() {...} The default constructor should initialize the balance to 0.0.
* public void deposit(double amount) {...} A deposit method to add money to the account.
* public double withdraw(double amount) {...} A withdraw method that withdraws the given amount from the account.
  + If the amount given can be withdrawn, it should return that amount.
  + If not, it should return 0.0.
* public double getBalance() {...} A method to get the current balance in the account.
* public void transferMoney(Account toAccount, double amount){…} Transfers money from the current account to toAccount (withdraw rules apply, call withdraw member function in order not to have code duplication)

Add your full name at the beginning of each source file you are submitting

Exercise these annotations:  

@Test   
@DisplayName("Custom test name containing spaces")  
@Disabled("Disabled until bug #99 has been fixed")  
@BeforeEach

Submission:

* (6 Points) Implementation of the given requirements above in Java language.
* (9 Points) JUnit test file of the implementation
  + At least 7 test methods, testing 7 different test cases,
  + Additional, one method for @Disabled, and one method for @BeforeEach
  + Have good meaningful test function names,
  + Don’t forget to put your name in each source file,
* Upload compressed (zip, rar) files to Canvas

References:<https://junit.org/junit5/docs/current/user-guide>