Agile Software Development

Assignment Two 2020
Due: 21/12/2020

The following document contains the instructions for Assignment Two 2020 for Agile Software Development. The project will be worth 30% of your mark for this module. You are required to develop a **JUnit automated test suite** that is capable of testing the Java application available for download on learnonline (InsuranceProgram.java).

- You are required to **refactor** this code so that can be tested using JUnit5.

- You test should contain a test suite.

- Your test code should contain the following annotations @BeforeAll, @BeforeEach, @Test, @ParameterizedTest, @Timeout, @AfterAll and @AfterEach.

- Your code should also be able to test for two different kinds of exceptions.

This program calculates car insurance premium based on age. The test suite should test every method of your refactored source code.

NB: Students should consider refactoring the provided program in order to write meaningful test cases.

Submission

Your will be required to submit your JUnit test suite and the downloaded program through Learnonline in a zip file.

Student Conduct

You should familiarise yourself with GMIT's code of student conduct and the policy on plagiarism. In particular, note two things. First, students are expected to treat other students and staff politely and with courtesy. Second, it is assumed that all work you submit is being presented as your own work, unless referenced otherwise.

Expected standard

Please note that this is a level 8 module. Significant effort is made to ensure that the standard is fair and consistent across third level institutes, both nationally and internationally. The standard we set for modules in computing is informed by Quality and Qualifications Ireland's Award

Standard for Computing. Below is a particularly relevant selection of the learning outcomes contained in that document.

Level 8

The learner will be able to:

- describe the limitations of some current computing theories.

- evaluate information through online research.

- model and design complex computer-based systems in a way that demonstrates comprehension of the trade-off involved in design choices.

- demonstrate mastery of a complex and specialised area of skills and tools;

- manage one's own learning and development, including time management and organisational skills.

- manage a computer-based project throughout all stages of the life-cycle.

- apply quality concepts to products and processes of own work.