

Software Development 1, Coursework 4

This is individual assessed coursework. You are allowed to discuss this assessment with other students, but you should not copy their code, and you should not share your own code with other students. Note that we will carry out plagiarism checks on all submissions.

This coursework contributes 12.5% of your overall mark for the course and requires you to apply your knowledge of variables, types, operators, expressions and conditional execution. These topics are covered in the learning materials in Weeks 8 and 9 on Vision, and it is strongly recommended that you work through the tutorial exercises from these weeks before attempting the coursework.

The **deadline for completing this lab is the end of Week 12**. Before this deadline, you should submit your work for marking (whether you have completed everything or not) during your timetabled lab session. To do this, check your final solution into GitLab (as instructed in the familiarisation lab in Week 1) and use the lab form (via the link provided by your lecturer) to ask for marking. You will need to specify the URL of your GitLab project, which you can copy from the address field of your browser when viewing your project in a web browser.

Instructions

Go to https://gitlab-student.macs.hw.ac.uk/F27SA_2021-22/f27sa-coursework-4 in GitLab, fork the project, and import the forked project into Eclipse. Don't forget to fork the project; if you don't do this, you won't be able to save your changes back to Gitlab.

Save your changes to GitLab every time you get something working. If you later break something, this will make it easy for you to restore the earlier working version. Once you've finished, **make sure you save your code to GitLab**, and then show it to a lab helper or your lecturer during a timetabled lab session. If you have not been able to finish everything, please show us what you have been able to do, and save what you have completed to GitLab.

Problem Description:

Design a class named MyInteger. The class contains:

- An int data field named value that stores the int value represented by this object.
- A constructor that creates a MyInteger object for the specified int value.
- A get method that returns the int value.
- Methods isEven(), isOdd(), and isPrime() that return true if the value is even, odd, or prime, respectively.
- Static methods isEven(int), isOdd(int), and isPrime(int) that return true if the specified value is even, odd, or prime, respectively.
- Static methods isEven(MyInteger), isOdd(MyInteger), and isPrime(MyInteger) that return true if the specified value is even, odd, or prime, respectively.

- Methods equals(int) and equals(MyInteger) that return true if the value in the object is equal to the specified value.
- A static method parseInt(char[]) that converts an array of numeric characters to an int value.
- A static method parseInt(String) that converts a string into an int value.

Write a client program that tests all methods in the class.

You will get a mark out of 10 for your work, according to this marking scheme:

- 1 mark for correctly implementing **each bullet point (9 in total)**.
- 1 mark for correctly testing your code.

In gitlab you are provided with a skeleton of the MyInteger class and with an example of how you may want to approach testing of your class.

Late submissions will be marked according to the university's late submissions policy, i.e. a 30% deduction if submitted within 5 working days of the deadline (e.g. in the following week's labs), and no mark after that. If you have mitigating circumstances (e.g. illness), please submit the form available at: <https://www.hw.ac.uk/students/studies/examinations/mitigating-circumstances.htm>
