

Assignment 2

INSE 6615

Due: March 28 by 6pm (18h00)

*Assignments are to be completed individually. Any reference to external material should be cited. Each student has available one slip day for use on one, and only one, assignment of his/her choosing. Using the slip day allows the assignment to be submitted at **noon (12h00)** on the **Friday** that follows the due date without penalty. Late assignments will not otherwise be accepted (exceptions made for medical certificates).*

Assignments should be typeset (not handwritten) in PDF format. Please name the file `sid.pdf` where `sid` is your student ID.

To use EAS, you must be on the campus internet or use a VPN from off campus, otherwise the website will not load. Follow these instructions for Concordia's VPN (pay attention to the "for students" parts):

<https://www.concordia.ca/it/security/mfa/use-with-vpn.html>

Then upload to EAS (link below) under "Assignment" and "2." If you want to modify your assignment before the deadline, upload it again to "Assignment" and "2" and EAS will replace the file.

<https://fis.encs.concordia.ca/eas/>

EAS requires your GCS username/password (note: the wifi and VPN uses your Concordia username/password). Please ensure you have both of these well in advance of the deadline (if you are from outside GCS, you can obtain a username/password from the AITS service desk in EV or Hall buildings).

Assignment

This assignment requires the student to know how Ethereum blockchain works and be able to deploy a smart contract and interact with the smart contract. All the required steps, as well as the basics of Solidity programming using Remix IDE, are explained in this video, however some finer details may be out of date: https://youtu.be/lv8ai_V-E0

Requirements:

- Firefox or Chrome browser
- Metamask extension (<https://metamask.io/>) or a wallet of your choosing

- Switch Metamask to Sepolia test network (or a test network of your choosing—the video uses Ropsten)
- Create a new address in Metamask
- Research how to get free test coins from a faucet: <https://ethereum.org/en/developers/docs/networks/>
- Remix IDE: <http://remix.ethereum.org>

Copy and paste the following code in Remix (Also you can find the code here: <https://gist.github.com/shayanb/8509abe91f7c3dd43ff32fd1be047739>)

```
pragma solidity ^0.4.24;

contract assignmentTwo {
    uint public studentNumber;
    address public student;

    constructor() public {
        student = msg.sender;
    }

    function setStudentNumber(uint _studentNumber) public {
        studentNumber = _studentNumber;
    }
}
```

For this assignment, you are required to:

1. Add a new public Unsigned Integer variable named `GasUsed`
2. Add a function as a setter for the `GasUsed` variable
3. Set your student number in the contract using `setStudentNumber(uint _studentNumber)` function
4. Set the gas used by step 3 to `GasUsed` using the function in step 2

As explained in the video, when you call a function on the smart contract using a transaction, you spent some amount of Ether, named Gas, to pay for the computation. When you call `setStudentNumber()` to set your student number on the contract it uses some amount of gas which is visible on blockchain explorers such as etherscan.io. It is required that you read that number and store it in the variable `GasUsed`.

Verify your deployed smart contract is live on etherscan.io and take a screenshot. (For example, <https://ropsten.etherscan.io/address/0x0f3d2fae1370a5700a896c46a22cb975c89a687e#code>)

For the assignment, upload a simple PDF with your name, student ID, testnet you used, address of your contract, and screenshot of it on Etherscan.

If you want to learn more about Smart contract development, check out Solidity documentation: <https://solidity.readthedocs.io/en/latest/introduction-to-smart-contracts.html>