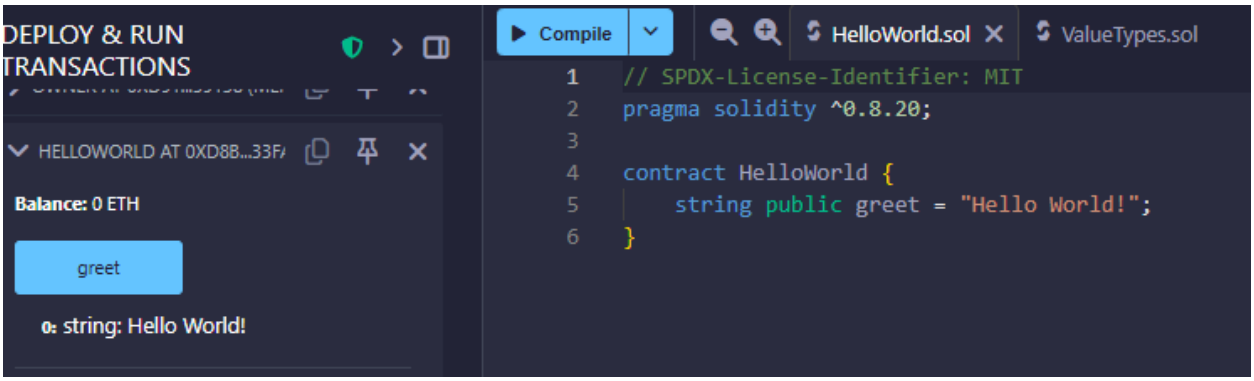
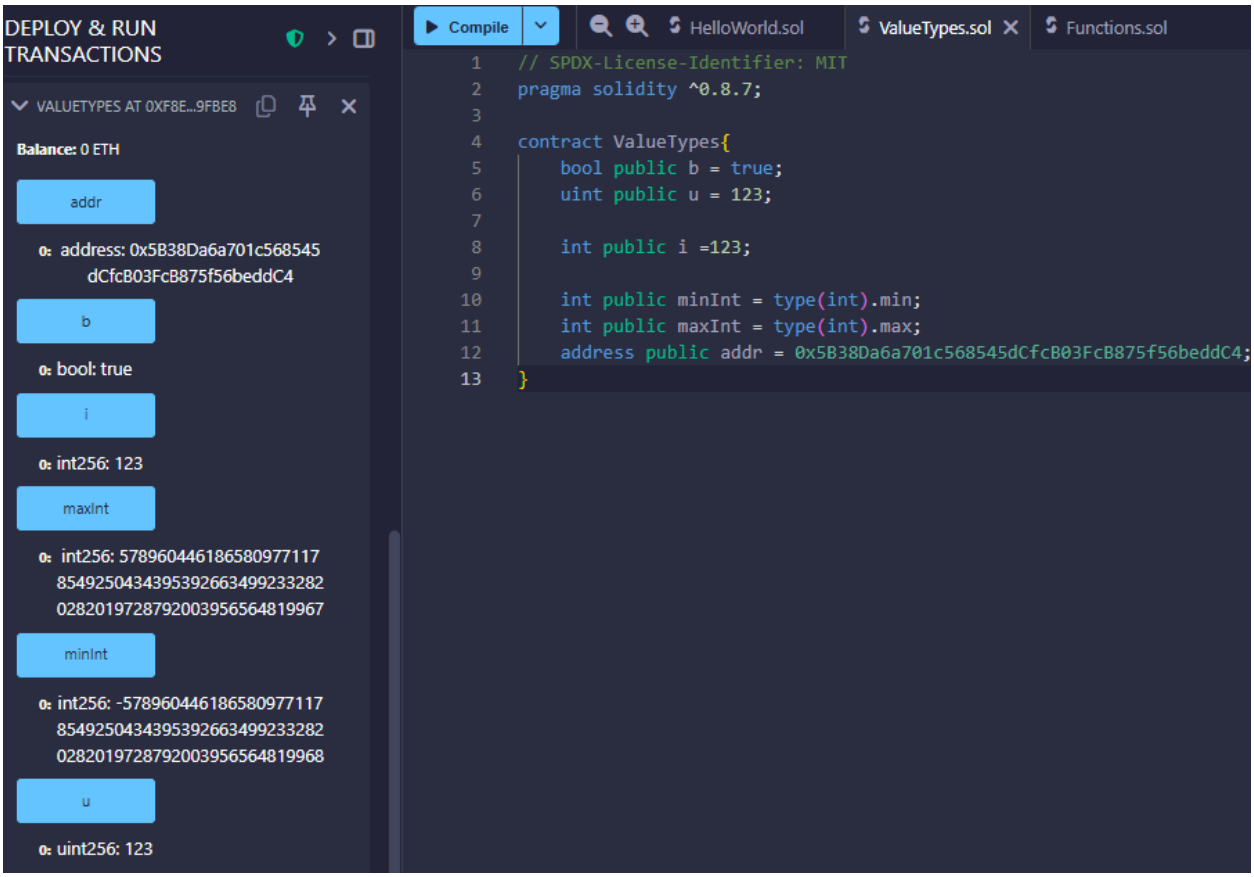
Dungo, Adrian Jeferson T. NW – 301 11/18/25

HelloWorld.sol



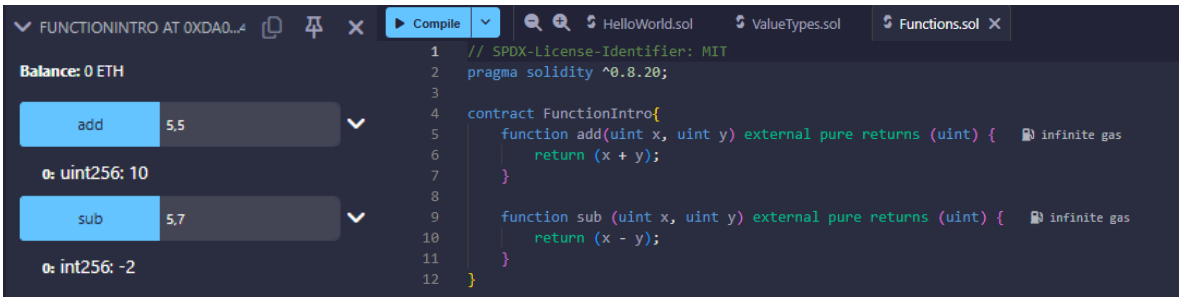
It defines a public string variable greet that is set to "Hello World!". The contract is deployed and the greet function can be called to return the string "Hello World!".

ValueTypes.sol



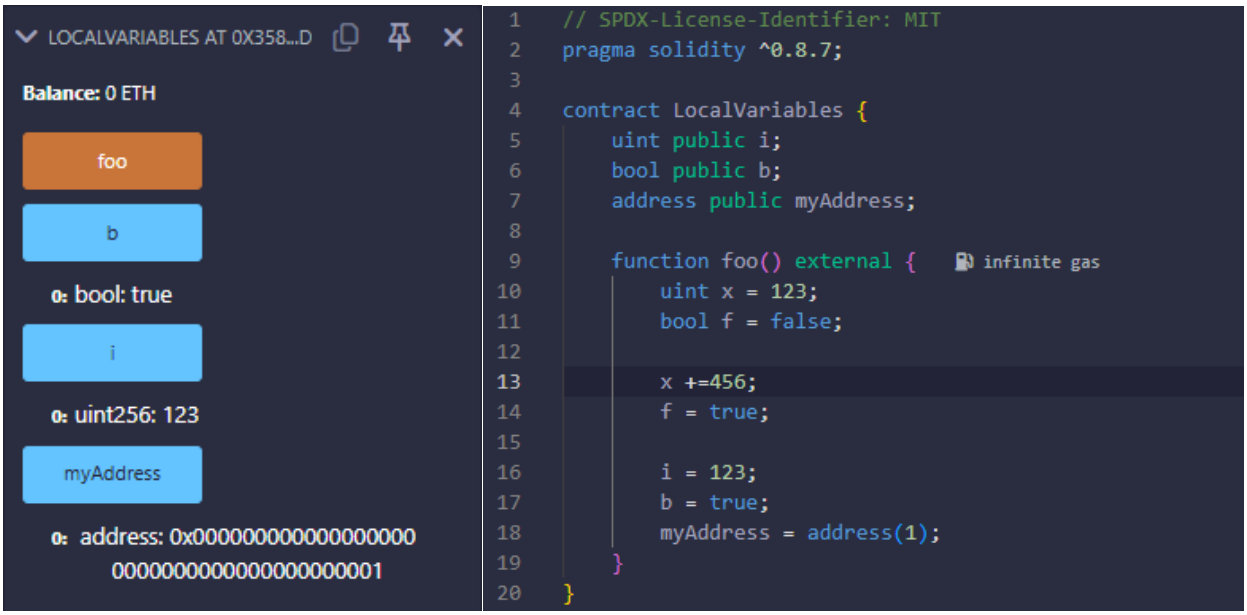
This solidity contract defines different data types. It has a boolean b, an unsigned integer u, and an integer i, as well as the minimum and maximum values of an integer type (minInt and maxInt). Additionally, an Ethereum address addr is also defined in the contract.

Functions.sol



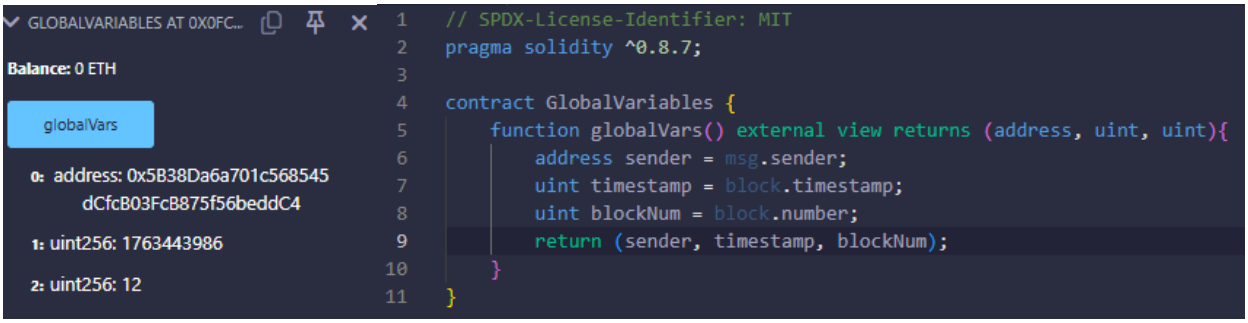
This contract uses functions to perform addition and subtraction. The values of x and y are based on the input of the sender. The result of the addition (add) is 10, and the result of the subtraction (sub) is -2.

LocalVariables.sol



This solidity contract defines variables of different data types (uint, Boolean, and an address). It has a function foo that assigns value to the variables when called.

GlobalVariables.sol



This contract reads three global values from the blockchain. It gathers the caller address, current block timestamp, and block number. It returns these values because the function is marked view and does not change state.

Review Questions

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.7;

contract ViewAndPure {
    function add(uint a, uint b) public ? returns (uint) {
        return a + b;
    }
}
```

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.7;

contract ViewAndPure {
    function getBalance(address account) public ? returns (uint) {
        return account.balance;
    }
}
```

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.7;

contract ViewAndPure {
    function multiply(uint a,uint b) internal ? returns (uint){
        return a*b;
    }
}
```

1. The function add is pure because it only uses the inputted values. It does not read contract storage.
2. The second function getBalance is view because it reads the balance field from the blockchain and it does not change any state variables.
3. The third function multiple works only using the provided parameters and does not interfere with contract storage.