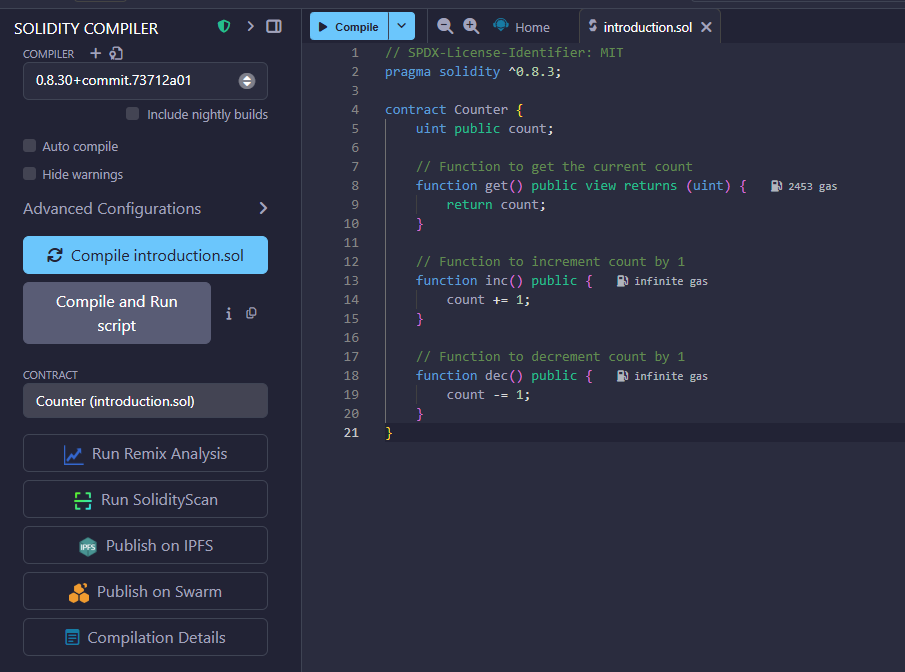
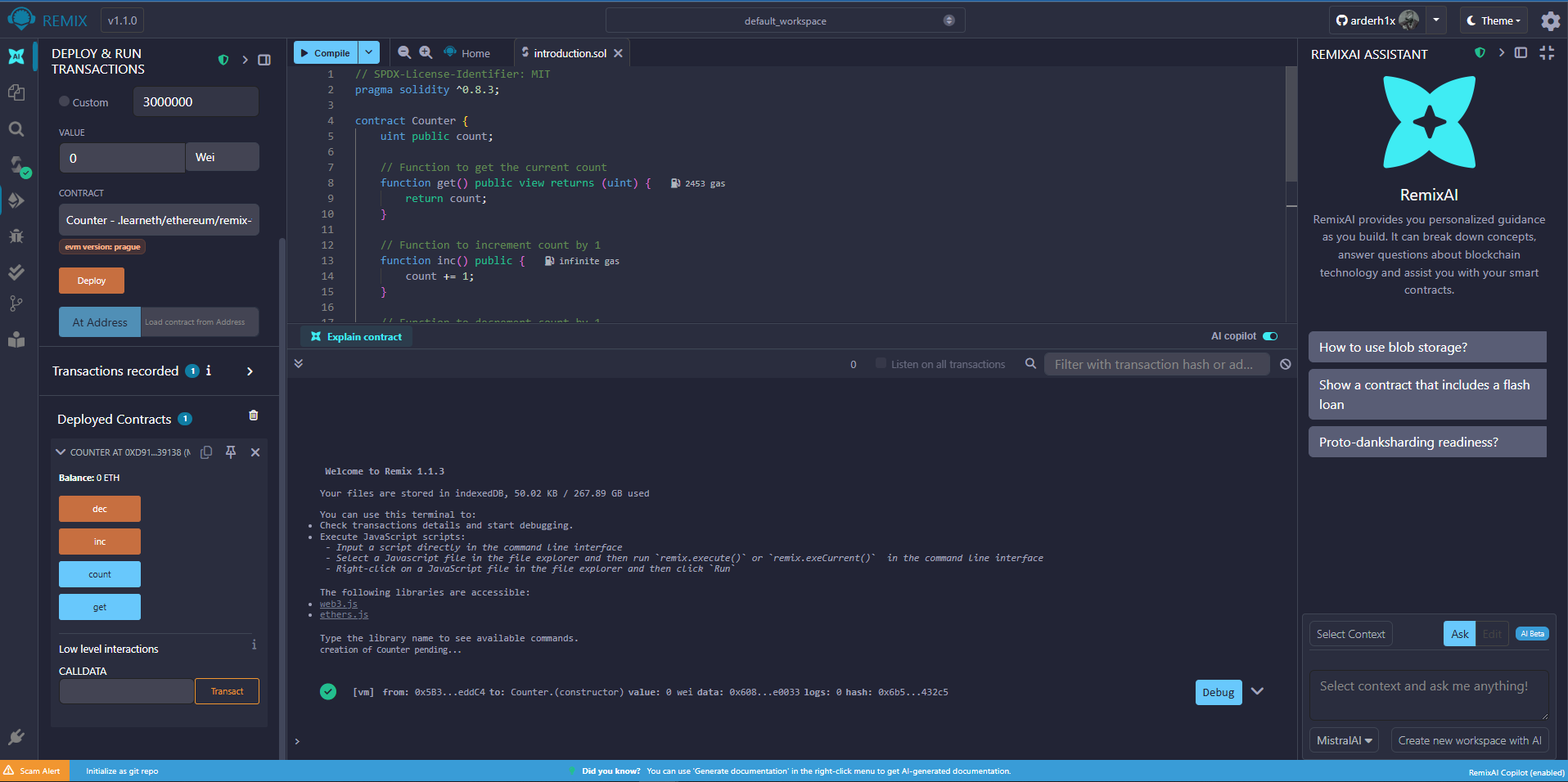
***Report for completing Solidity Beginner Course [1-7.2 chapters].***

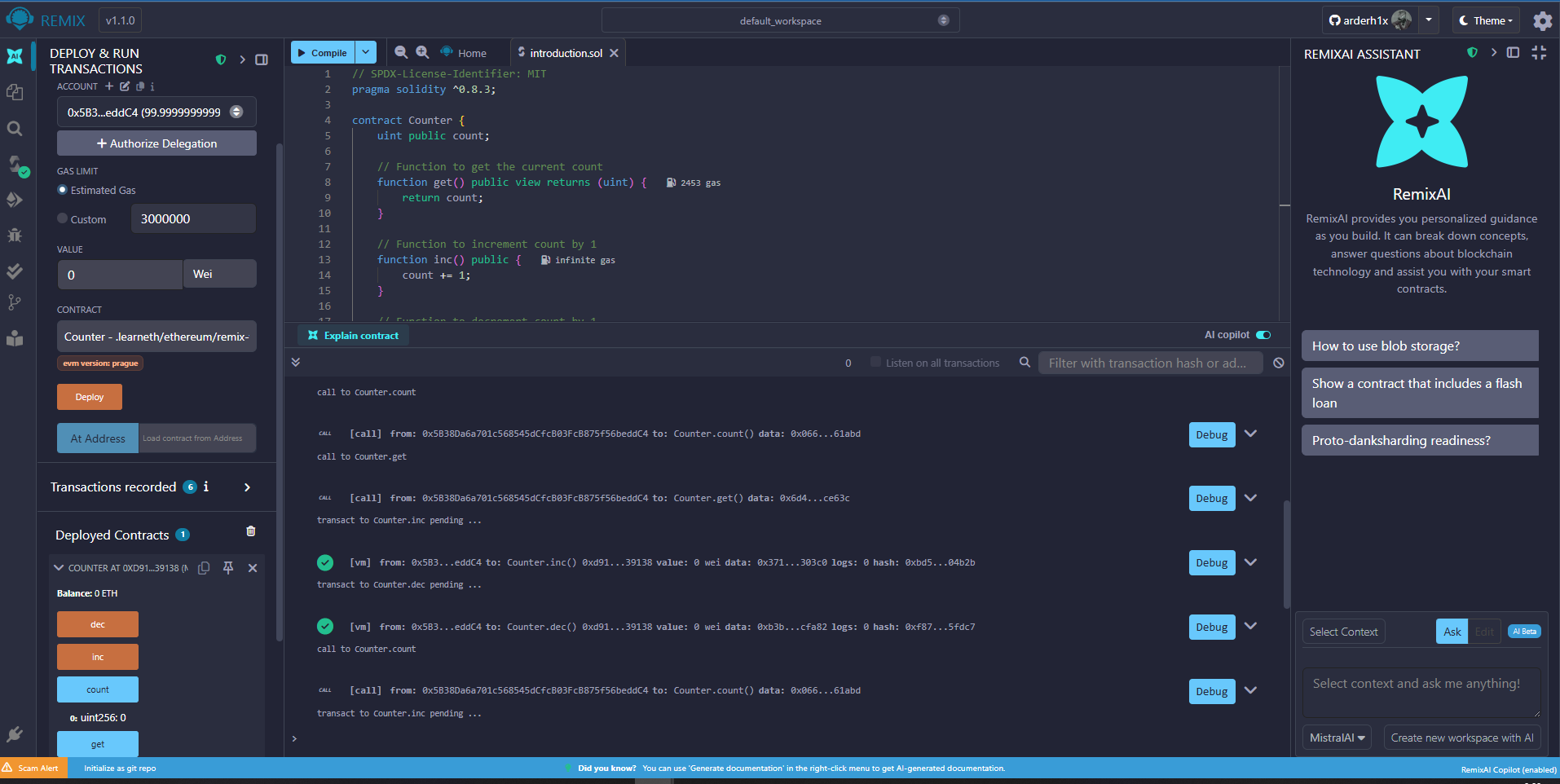
**1. Introduction.**

1) Compile this contract.



2) Deploy it to the Remix VM.



3) Interact with your contract.

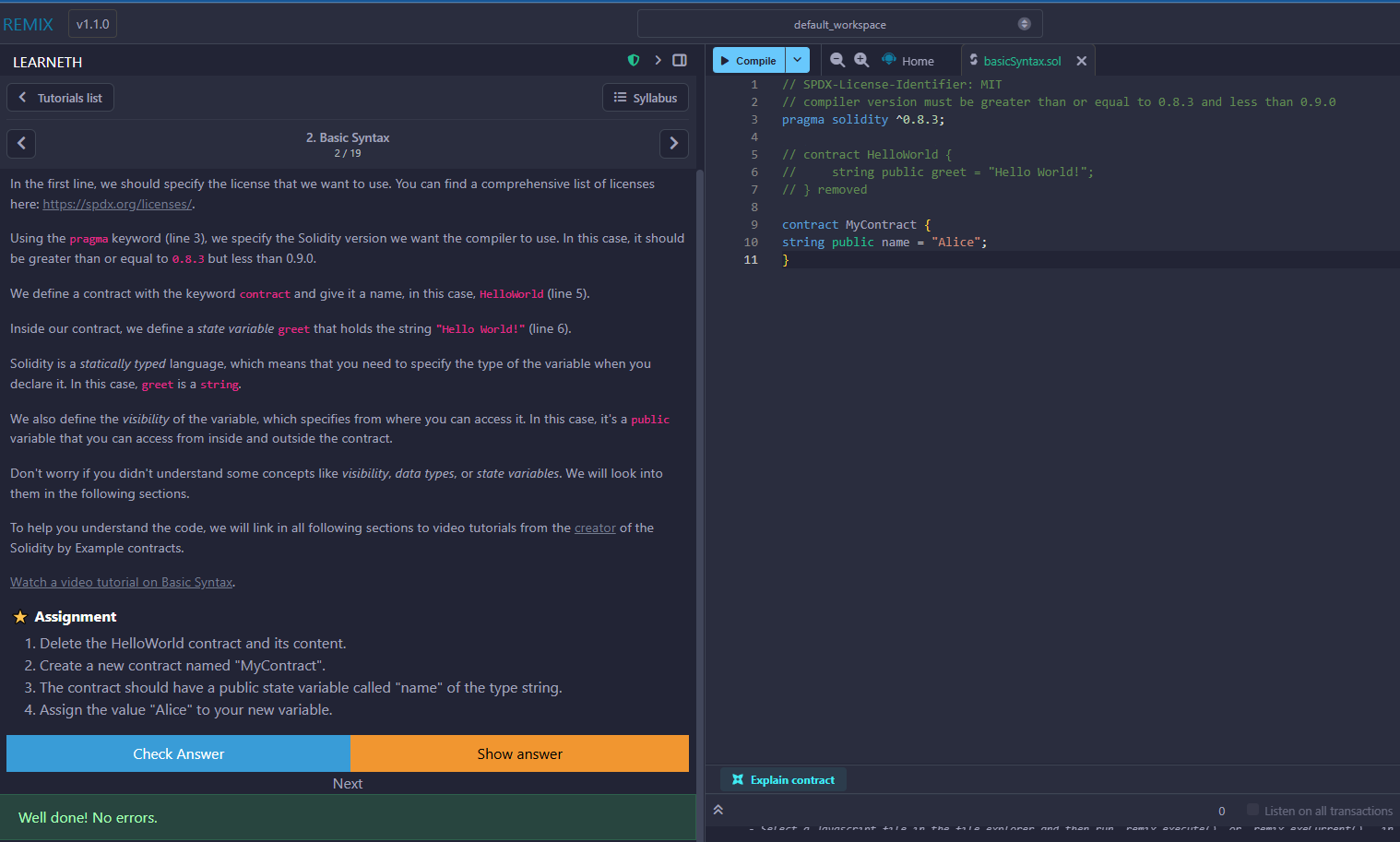
**2. Basic Syntax.**

1) Delete the HelloWorld contract and its content.

2) Create a new contract named "MyContract".

3) The contract should have a public state variable called "name" of the type string.

4) Assign the value "Alice" to your new variable.

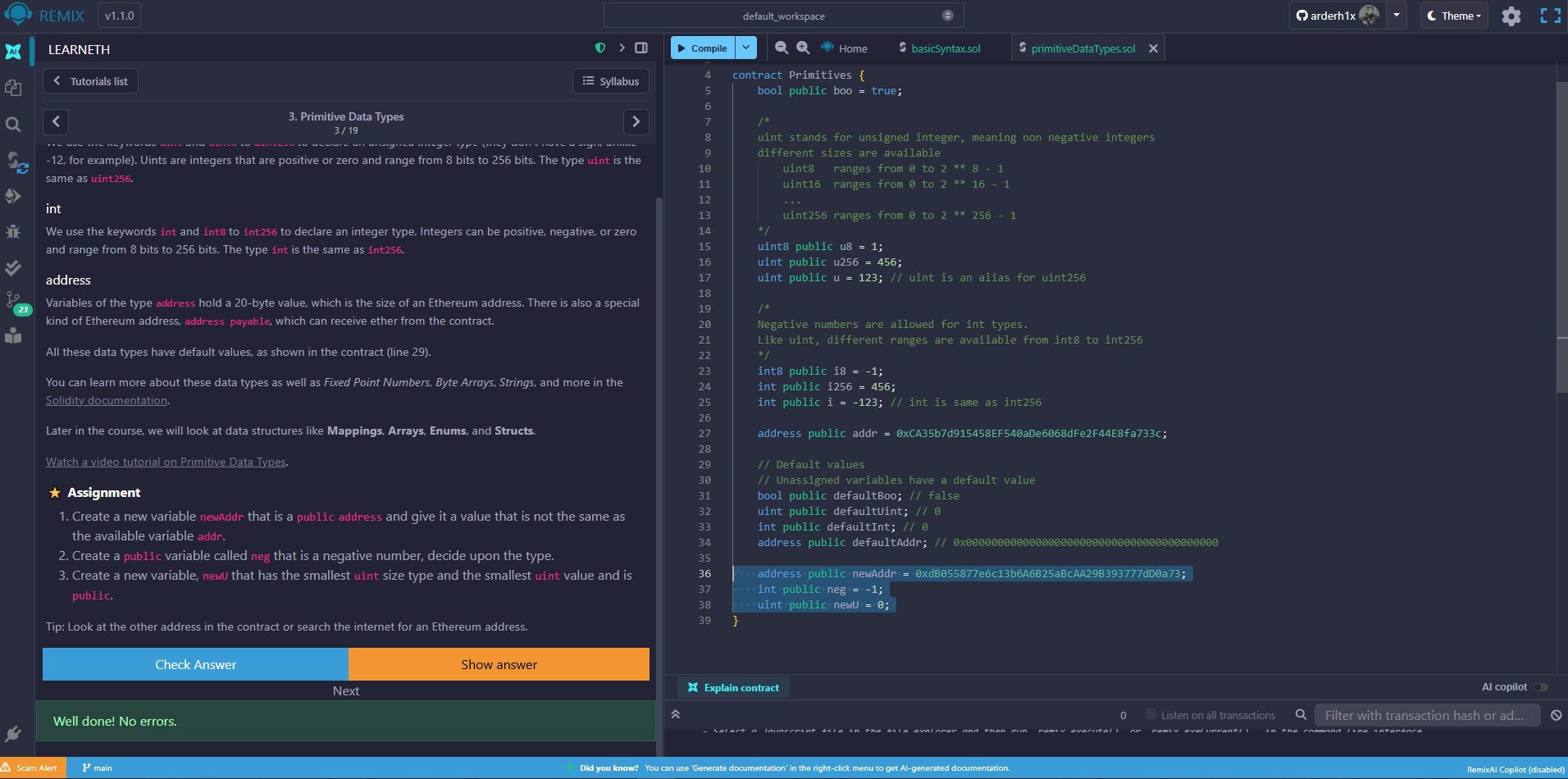


**3. Primitive Data Types.**

1) Create a new variable newAddr that is a public address and give it a value that is not the same as the available variable addr.

2) Create a public variable called neg that is a negative number, decide upon the type.

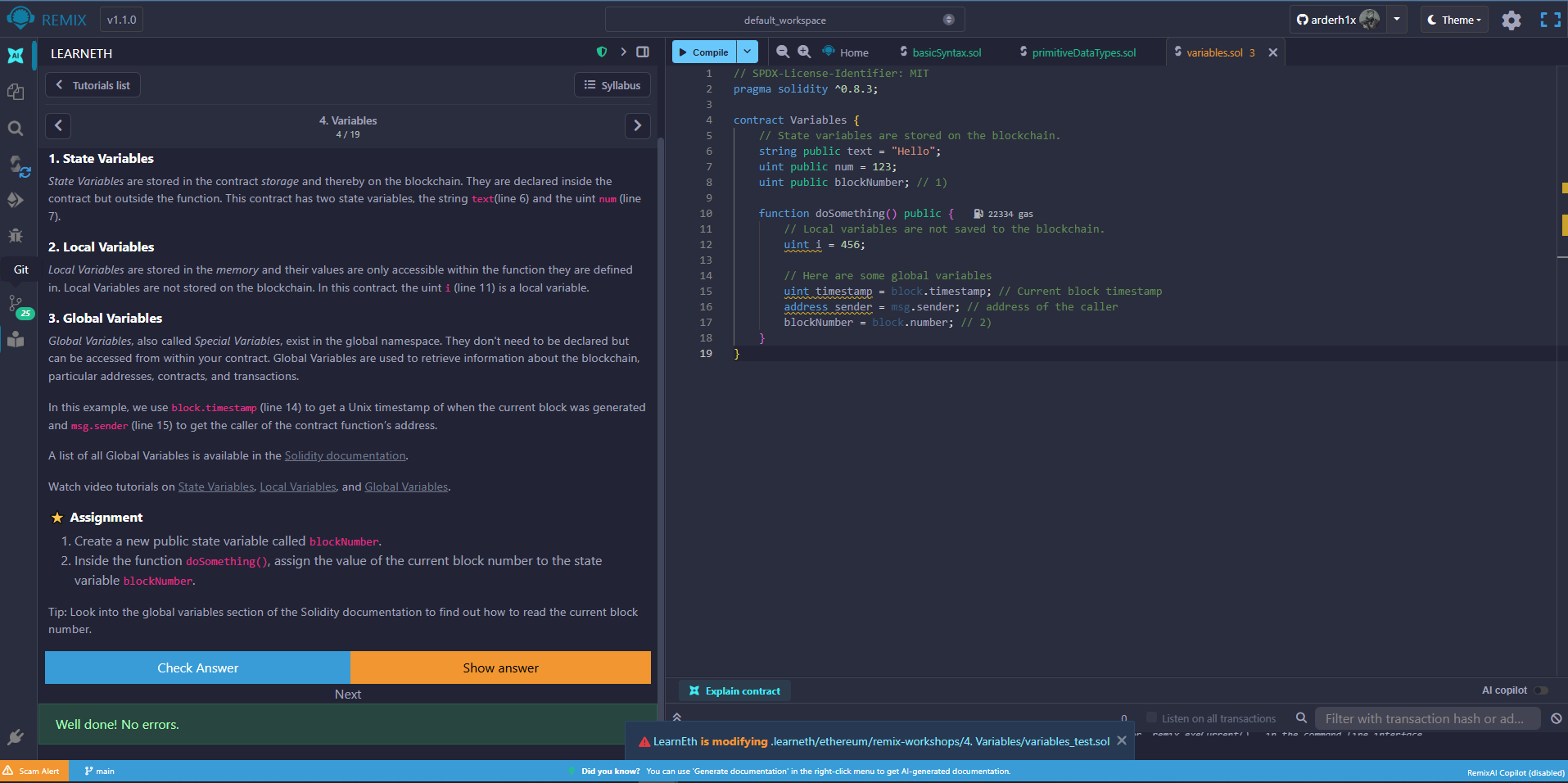
3) Create a new variable, newU that has the smallest uint size type and the smallest uint value and is public.



**4. Variables.**

1) Create a new public state variable called blockNumber.

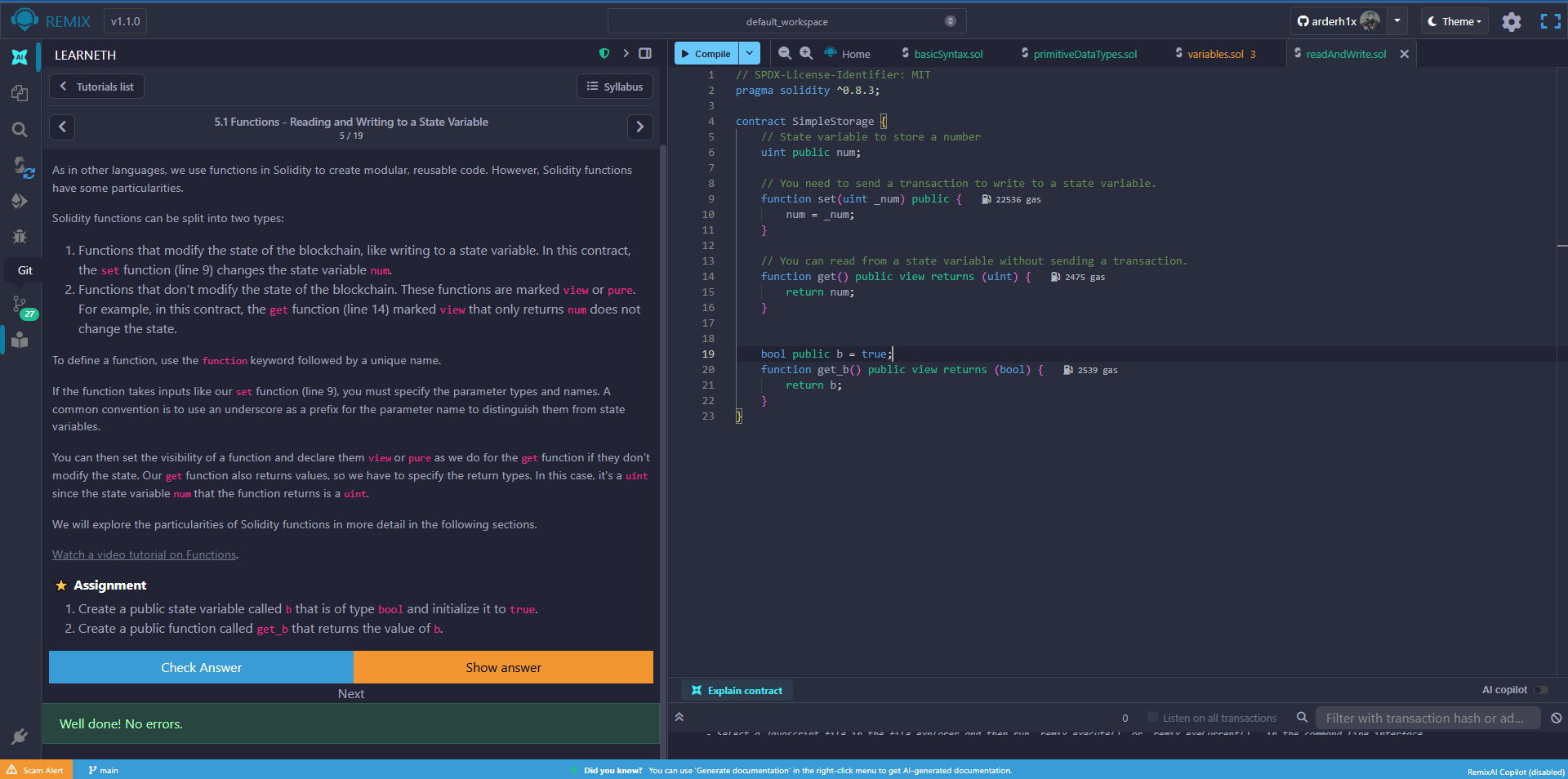
2) Inside the function doSomething(), assign the value of the current block number to the state variable blockNumber.



**5.1 Functions – Reading and Writing to a State Variable.**

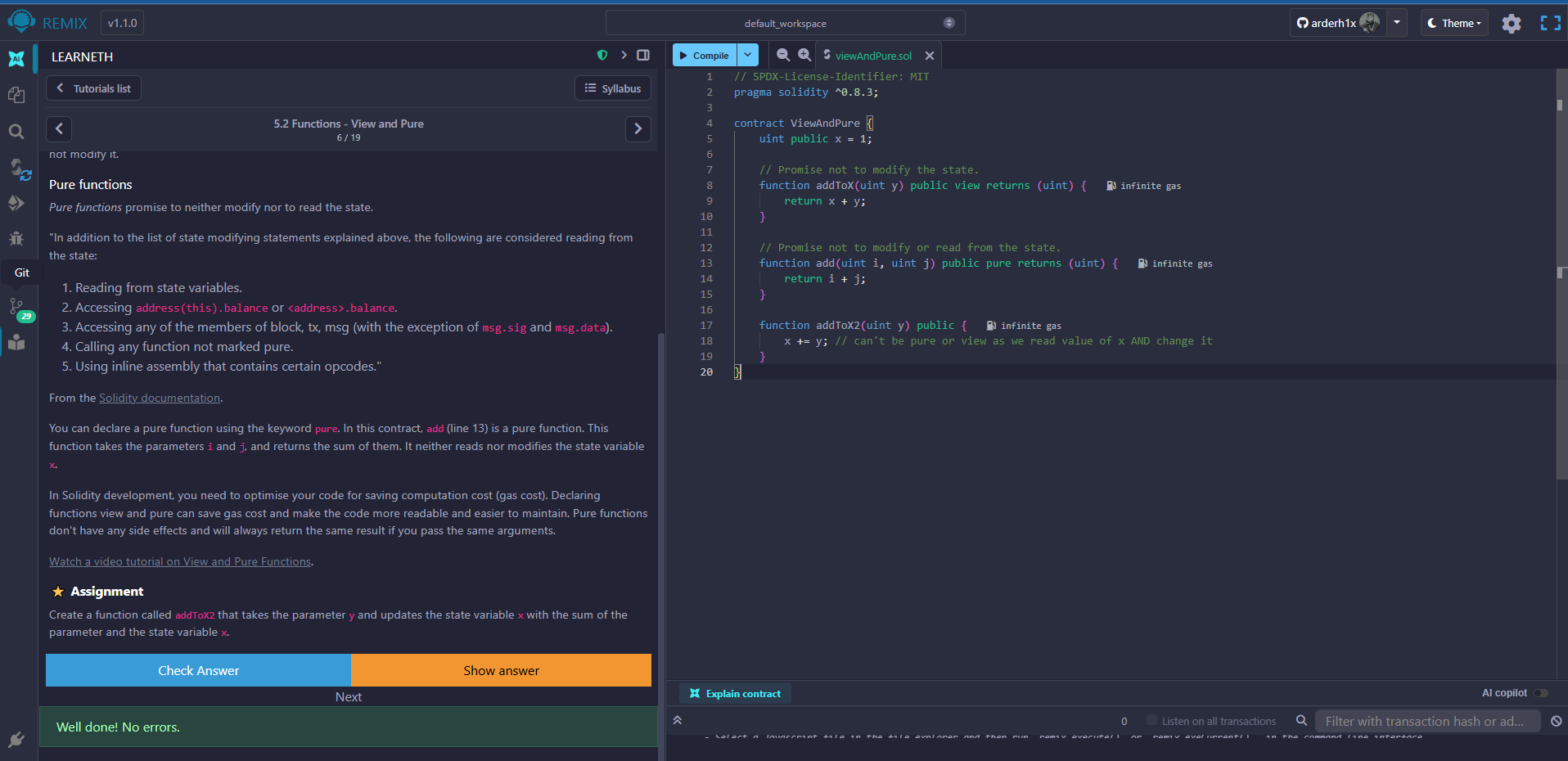
1) Create a public state variable called b that is of type bool and initialize it to true.

2) Create a public function called get\_b that returns the value of b.



**5.2 Functions - View and Pure.**

1) Create a function called addToX2 that takes the parameter y and updates the state variable x with the sum of the parameter and the state variable x.

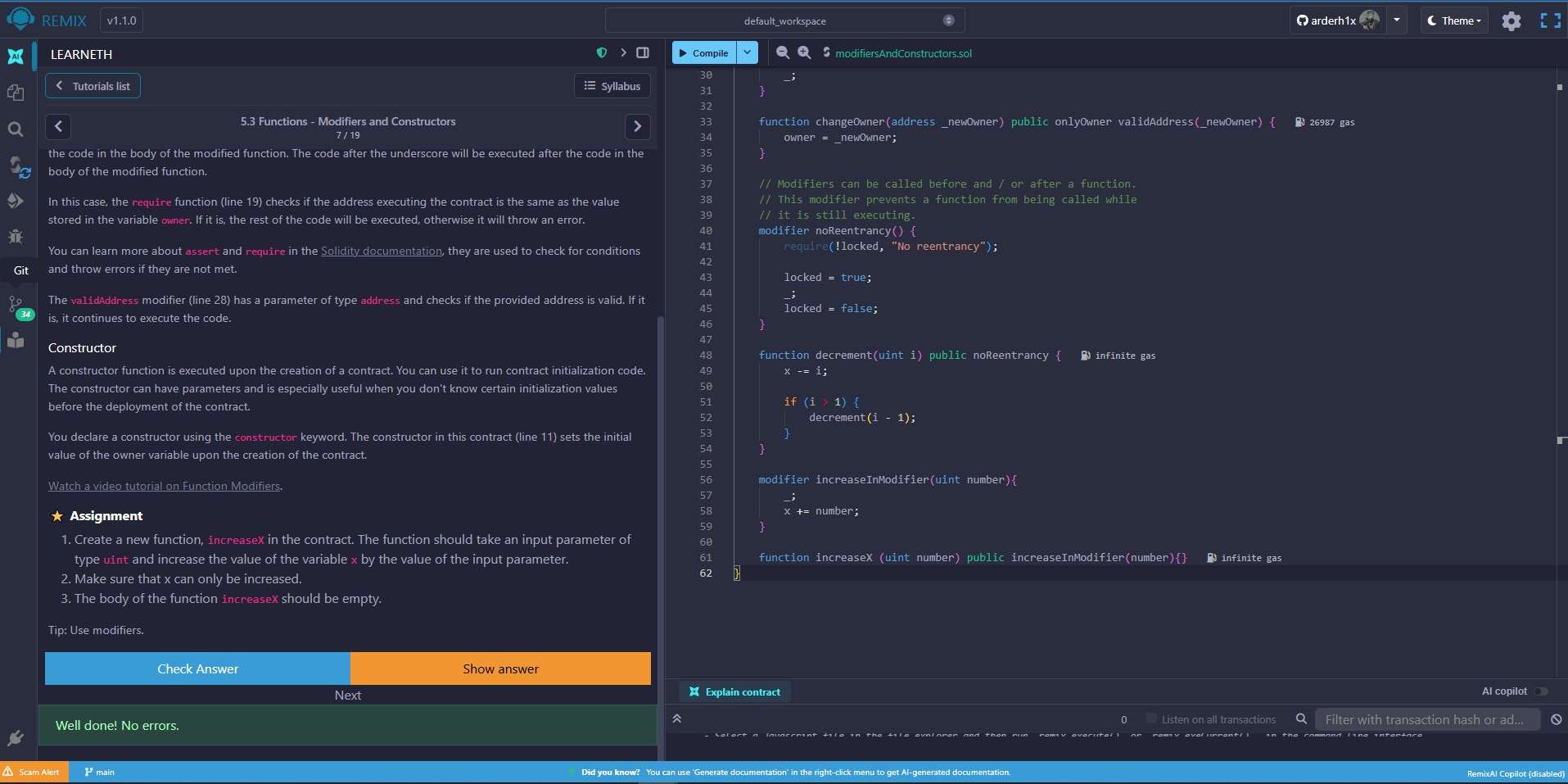


**5.3 Functions - Modifiers and Constructors.**

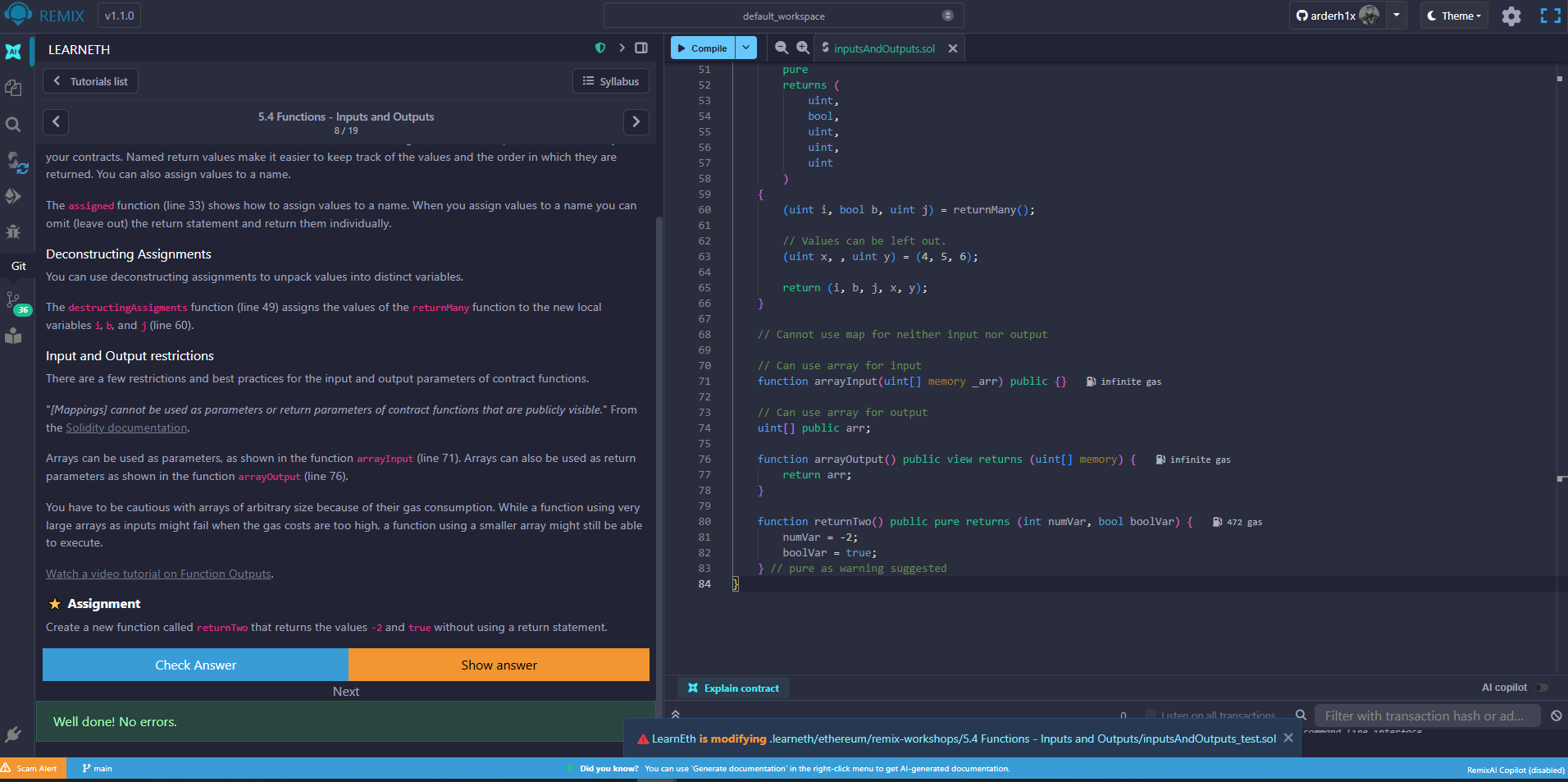
1) Create a new function, increaseX in the contract. The function should take an input parameter of type uint and increase the value of the variable x by the value of the input parameter.

2) Make sure that x can only be increased. // as uint type, it’s impossible

3) The body of the function increaseX should be empty.

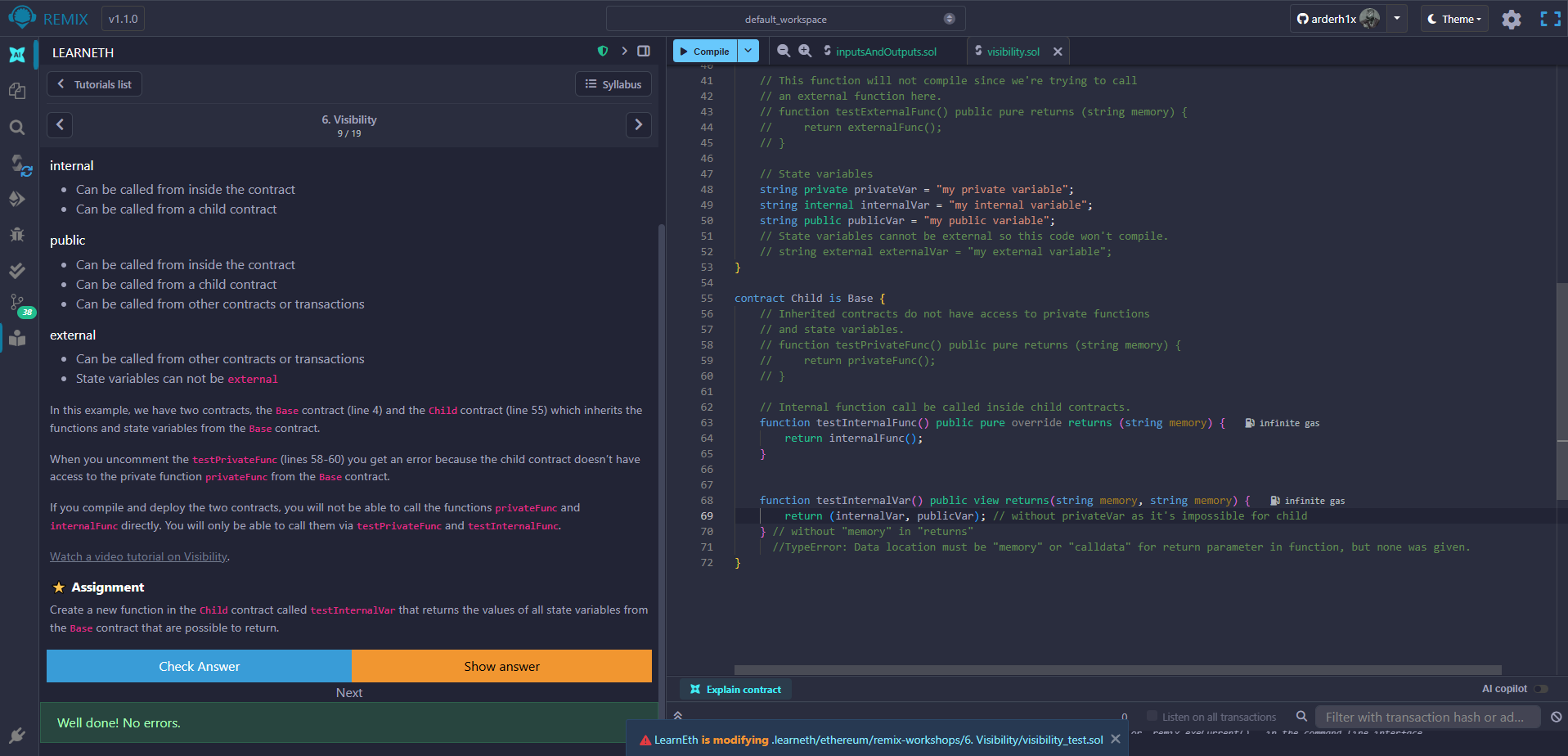


**5.4 Functions - Inputs and Outputs.**  
1) Create a new function called returnTwo that returns the values -2 and true without using a return statement.

****

**6. Visibility.**

1) Create a new function in the Child contract called testInternalVar that returns the values of all state variables from the Base contract that are possible to return.

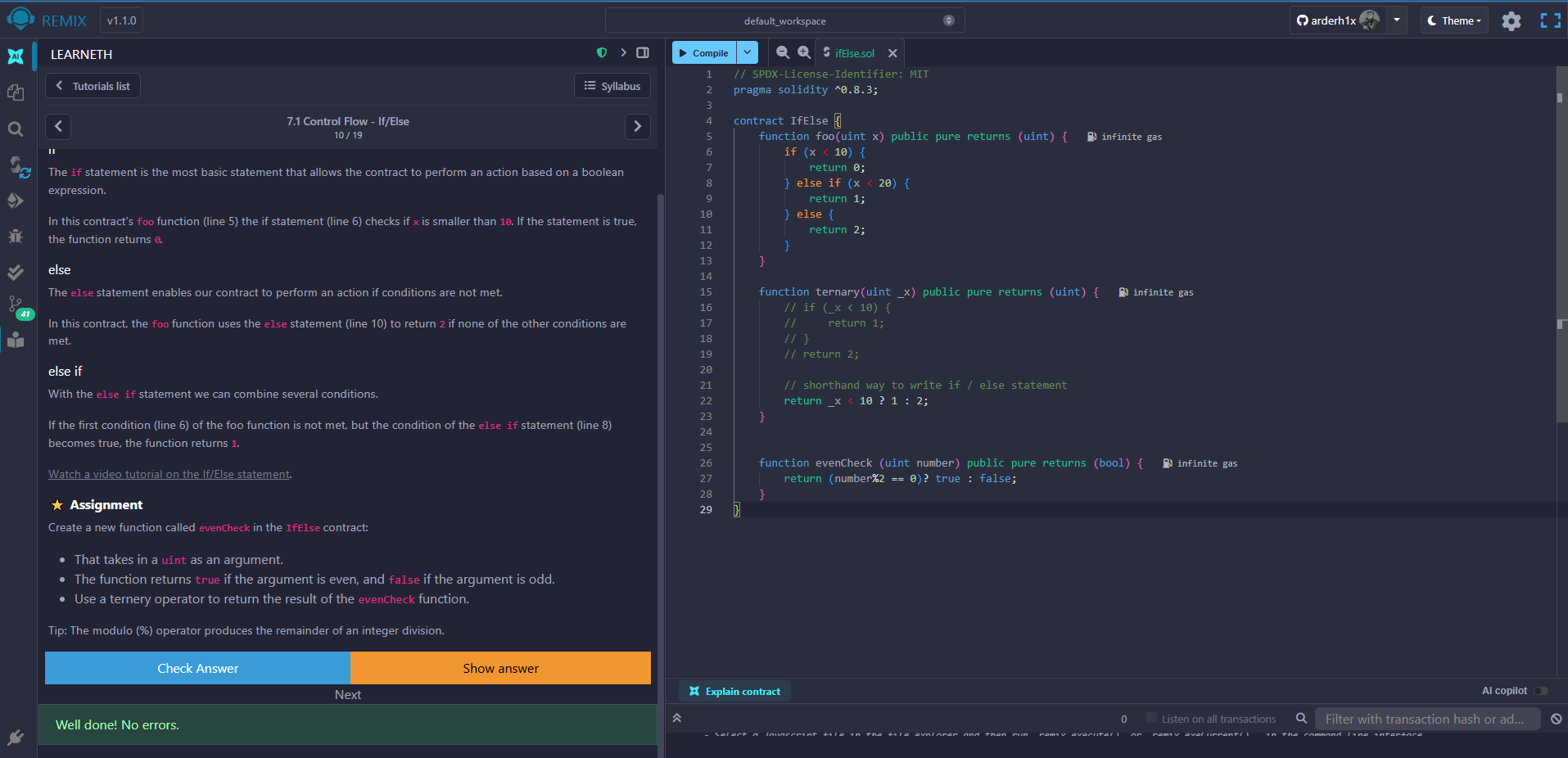
****

**7.1 Control Flow - If/Else.**

1) That takes in a uint as an argument.

2) The function returns true if the argument is even, and false if the argument is odd.

3) Use a ternery operator to return the result of the evenCheck function.



**7.2 Control Flow – Loops.**

1) Create a public uint state variable called count in the Loop contract.

2) At the end of the for loop, increment the count variable by 1.

3) Try to get the count variable to be equal to 9, but make sure you don’t edit the break statement.

