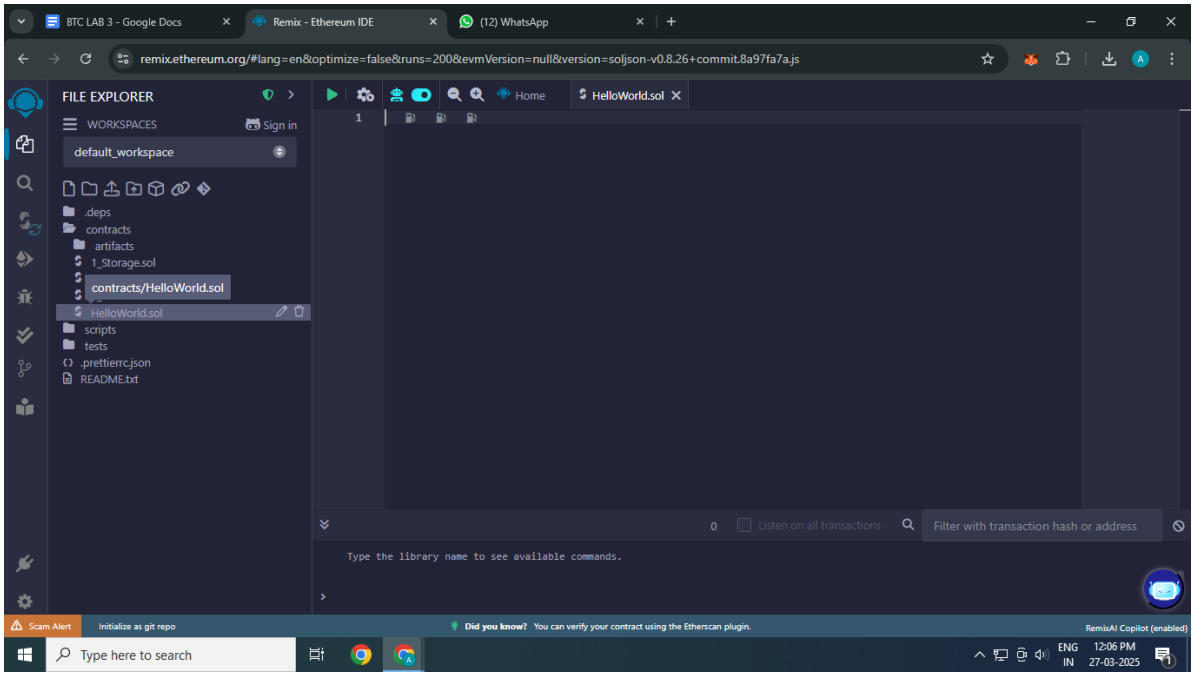
Assignment: 03

Step 1: Open Remix IDE

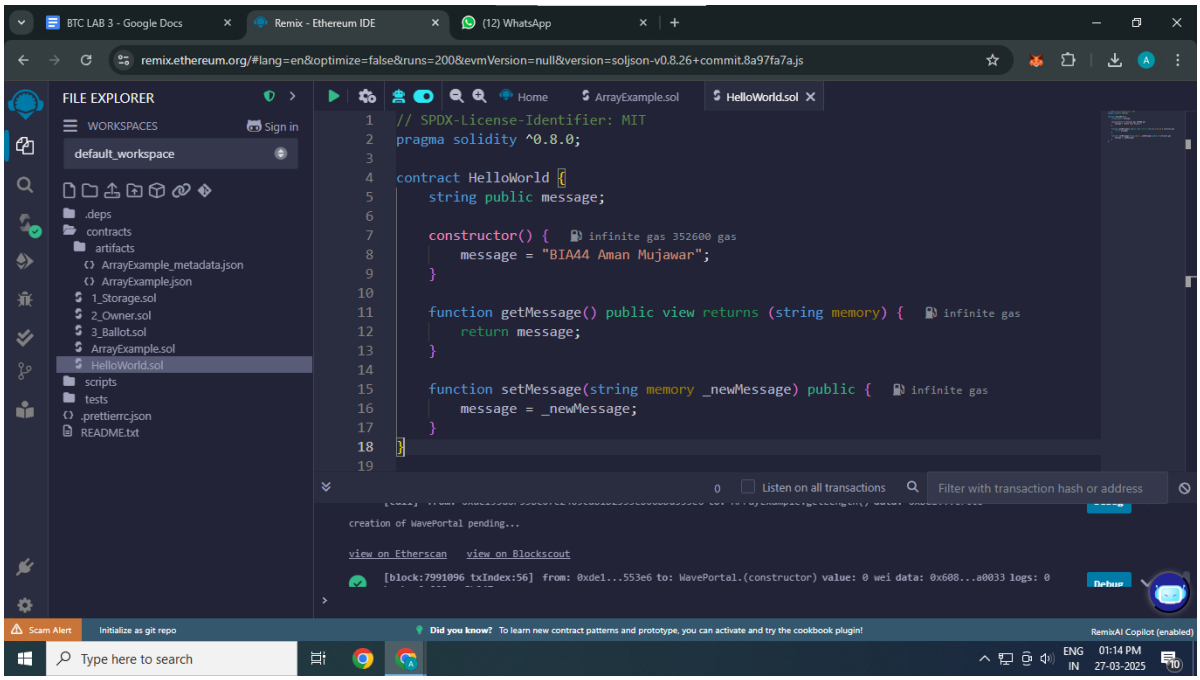
1. Go to **Remix IDE** in your web browser.
2. Click **"Create New File"**.
3. Name your file as **HelloWorld.sol**.



Step 2: Write a Hello World Smart Contract

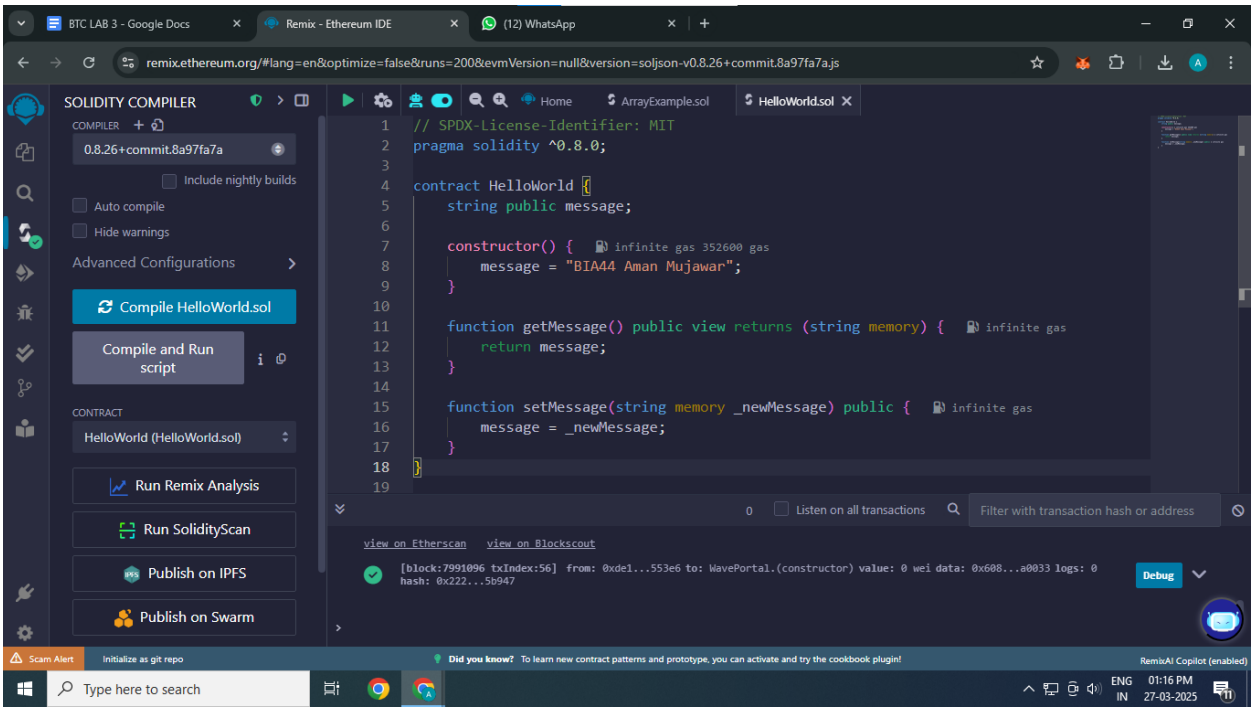
Solidity code:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract HelloWorld {
    string public message;

    constructor() {
        message = "BIA44 Aman Mujawar";
    }

    function getMessage() public view returns (string memory) {
        return message;
    }

    function setMessage(string memory _newMessage) public {
        message = _newMessage;
    }
}
```



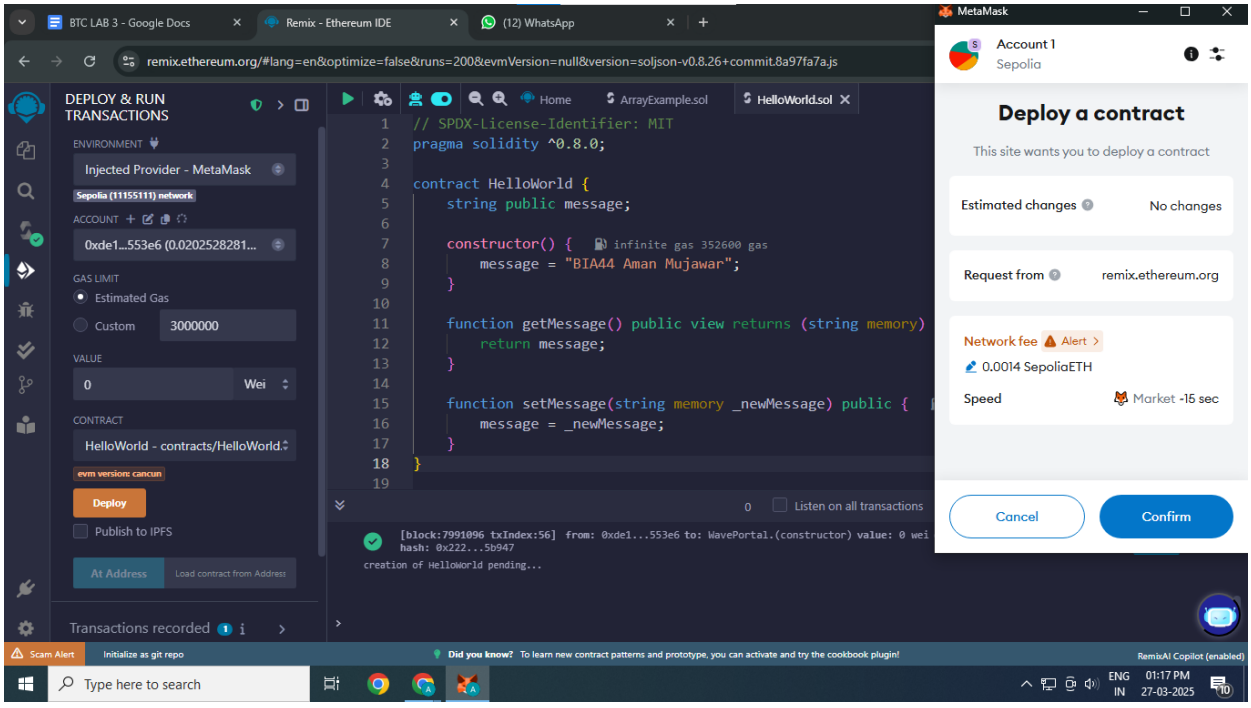
Step 3: Compile the Smart Contract

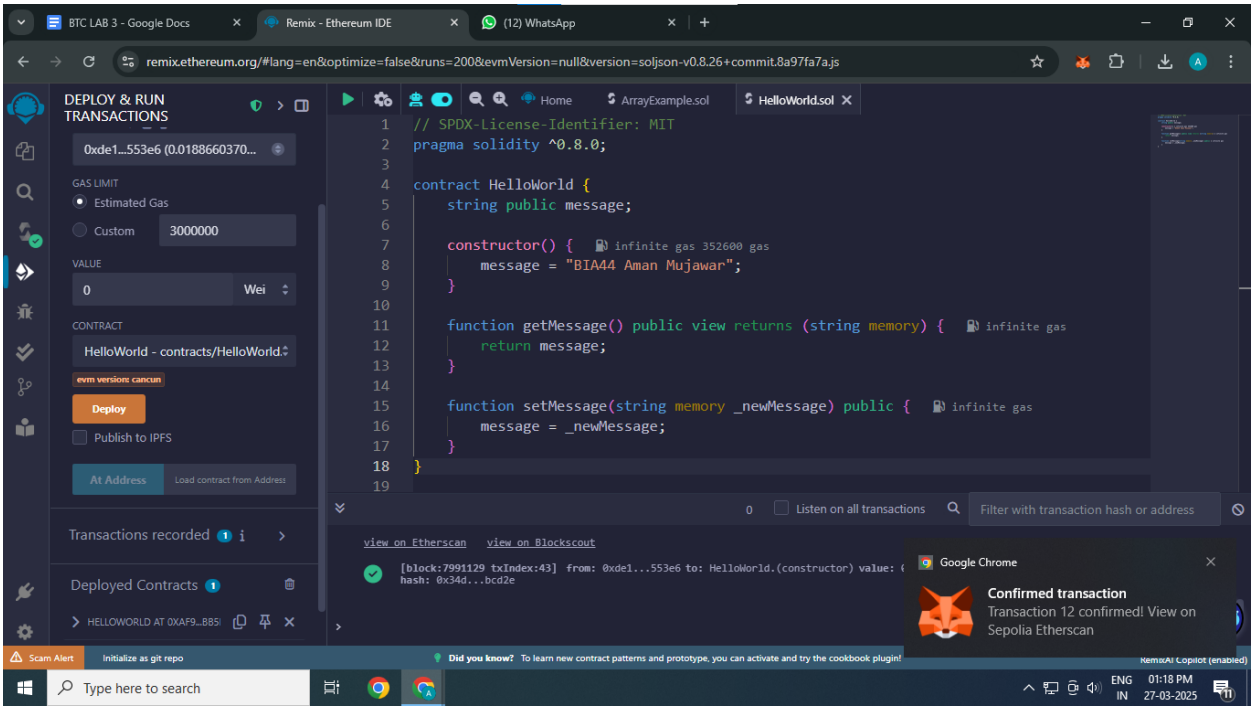
1. Click on the **Solidity Compiler**.
2. Click the **"Compile HelloWorld.sol"** button.



Step 4: Deploy the Smart Contract

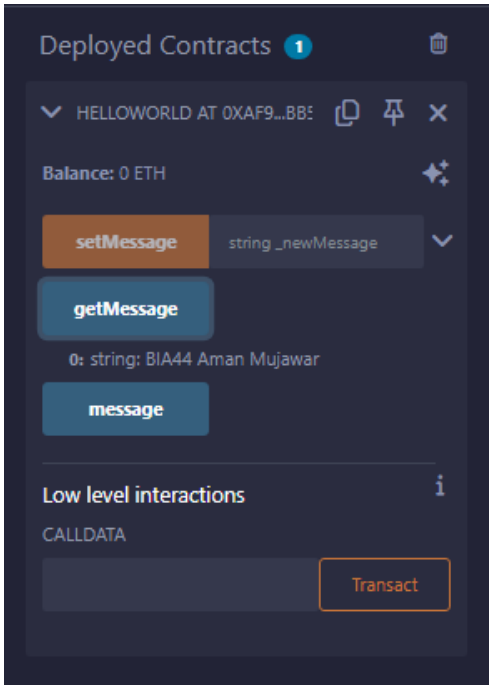
1. Click on the **Deploy & Run Transactions**
2. Under **"Environment"**, select **"Injected Provider(Meta Mask)"**.
3. Click **"Deploy"**.

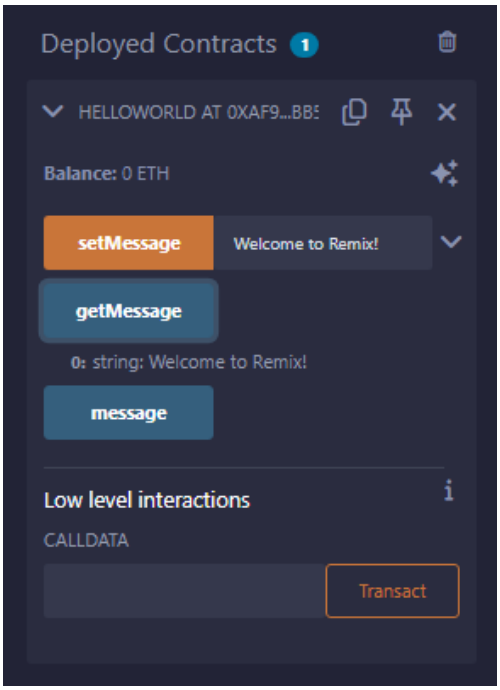




Step 5: Interact with the Smart Contract

1. Click the **"getMessage"** function → It will return **"Hello, World!"**.
2. Click the **"setMessage"** function → Enter a new message, like `"Welcome to Remix!"` → Click **"Transact"**.



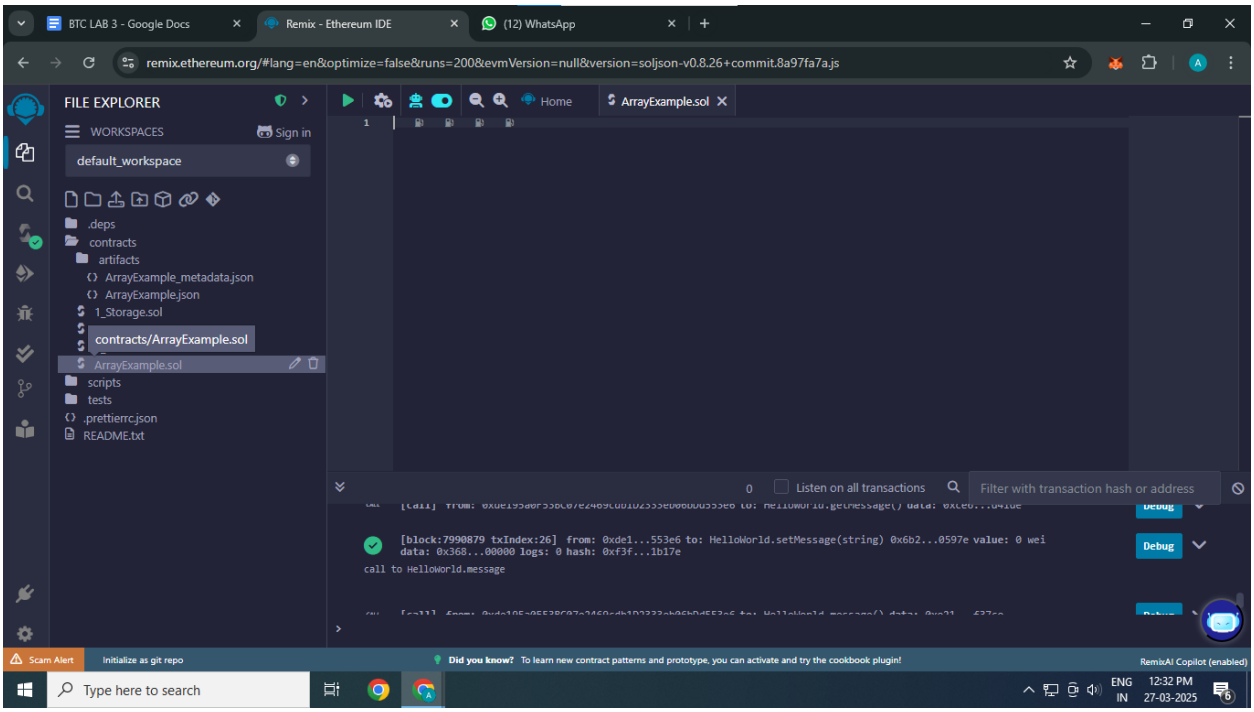


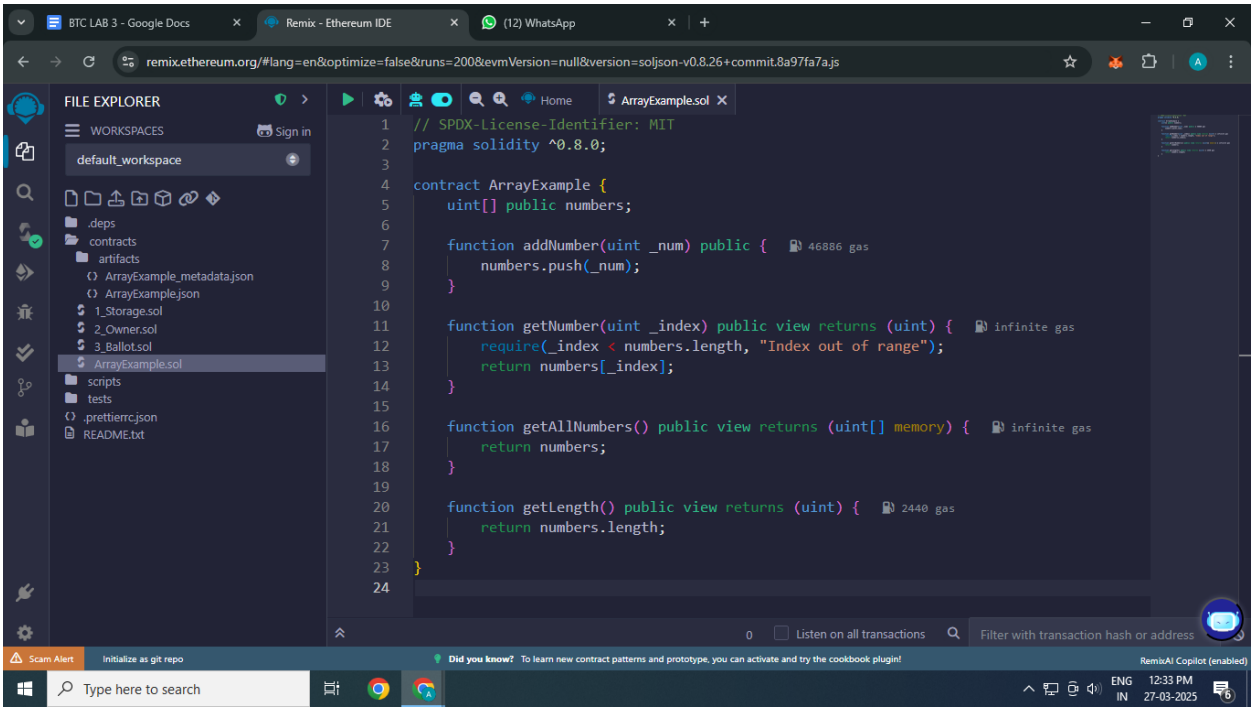
Step 6: Solidity example using Array

1. Create another file named **ArrayExample.sol**.

```
Solidity code:
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract ArrayExample {
    uint[] public numbers;

    function addNumber(uint _num) public {
        numbers.push(_num);
    }

    function getNumber(uint _index) public view returns (uint) {
        require(_index < numbers.length, "Index out of range");
        return numbers[_index];
    }

    function getAllNumbers() public view returns (uint[] memory) {
        return numbers;
    }
```

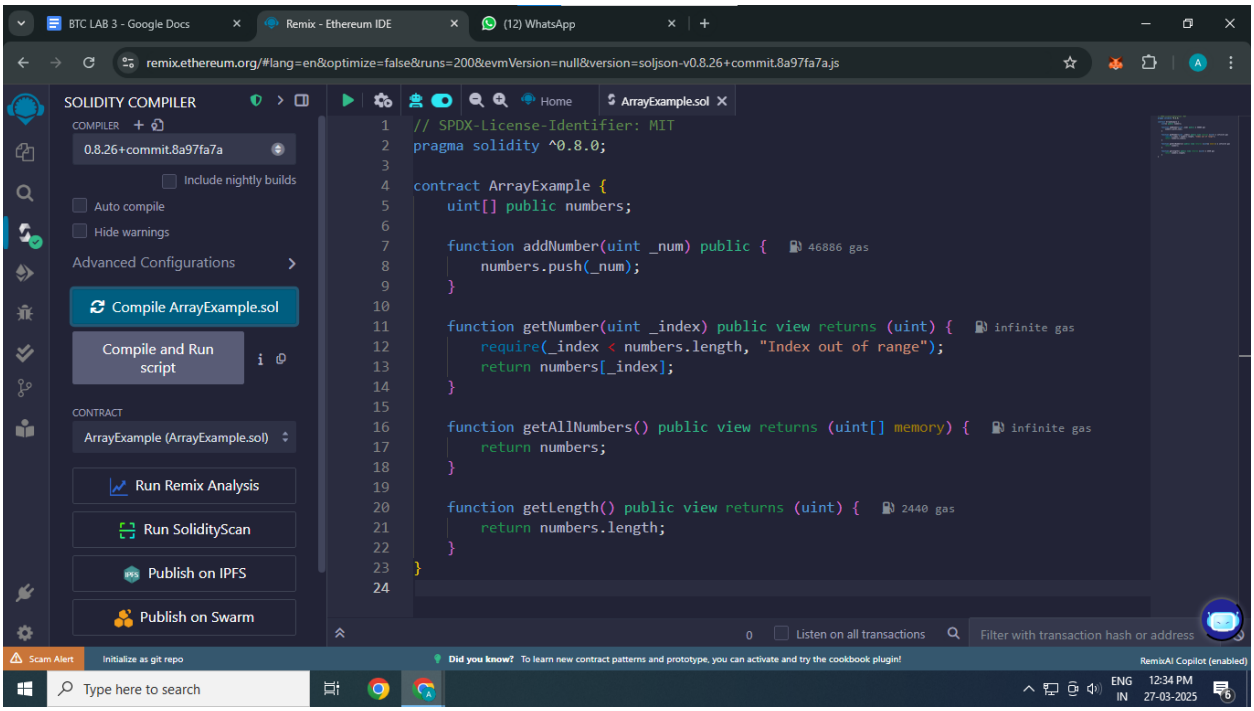
```
    function getLength() public view returns (uint) {
        return numbers.length;
    }
}
```



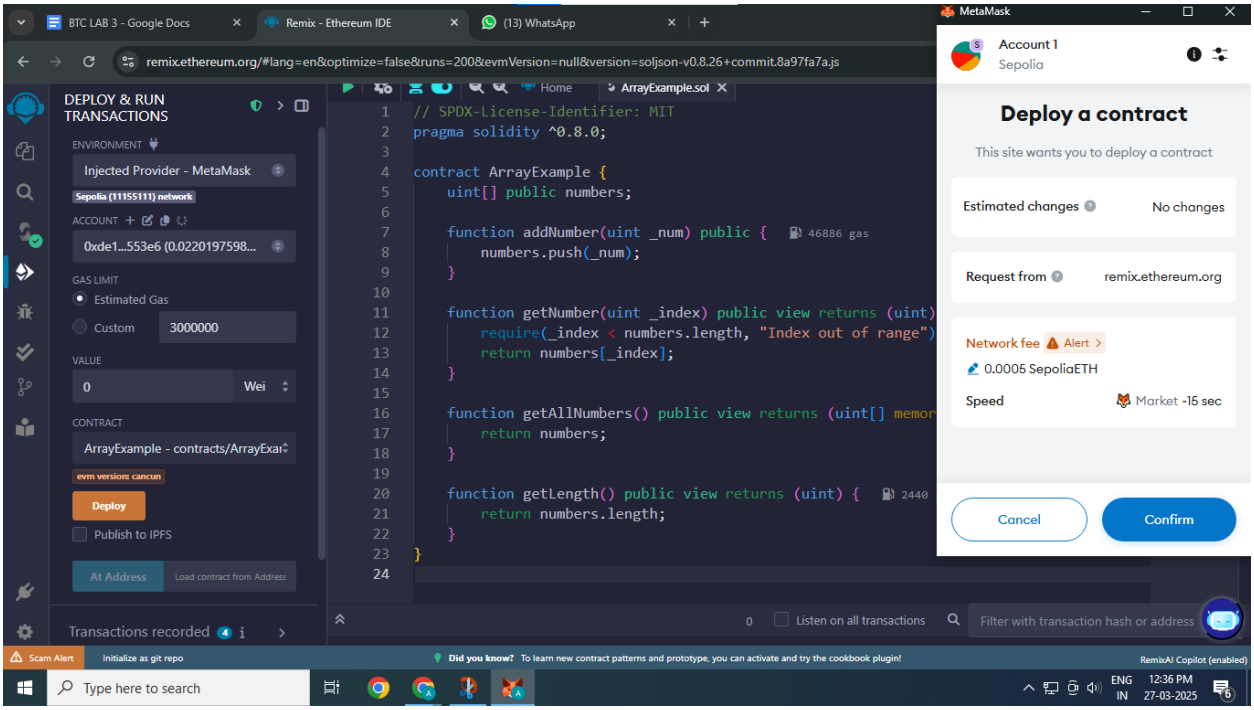


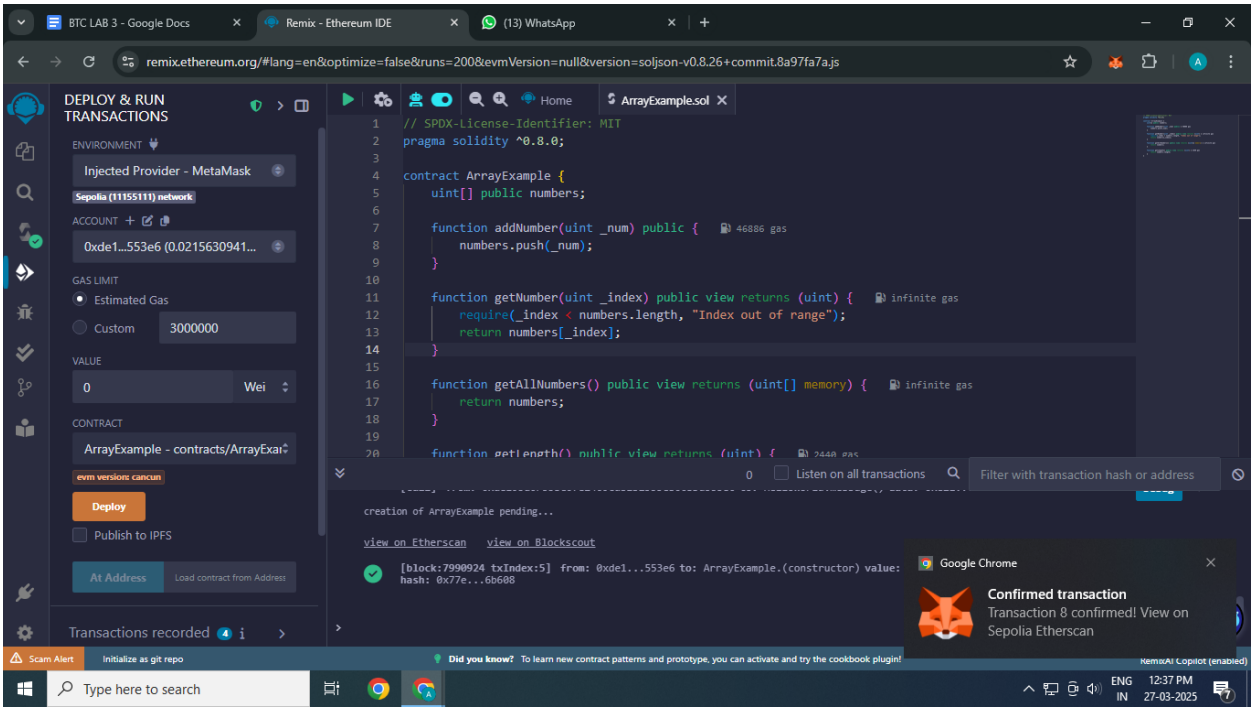
Step 7: Compile and Deploy the ArrayExample Contract

1. Click on the **Solidity Compiler**.
2. Compile **ArrayExample.sol**.



3. Click on the **Deploy & Run Transactions**.
4. Deploy the contract.





5. Interact with the contract: