

Assignment 1

NAME	SOLAI KUMAR.R https://d.docs.live.net/1517ae46bef44f96/Documents/Vishnu%20venkatesh.R%20assignment.docx
ZONE	2
COLLEGE	APOLLO ENGINEERING COLLEGE

1. GO TO THE CHROME ATFORM OPEN REMIXPLARTFORM

The screenshot displays the Remix IDE interface. The left sidebar contains the following sections:

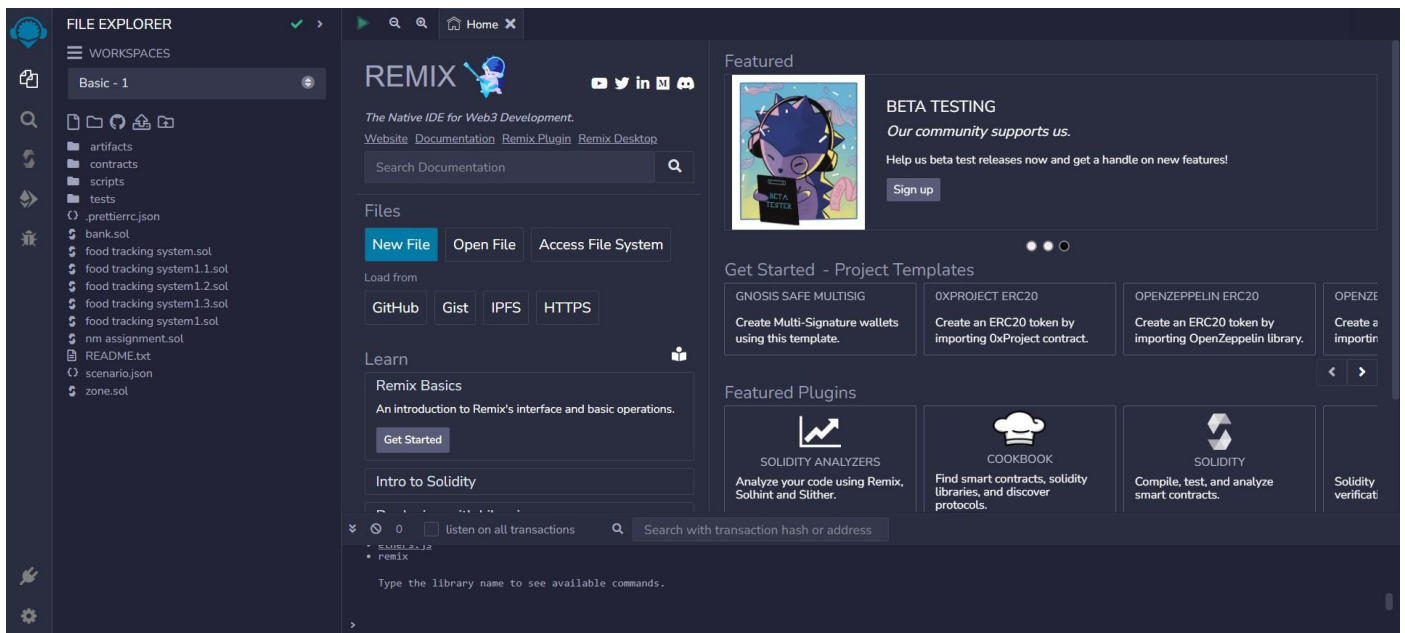
- REMIX** logo and social media links.
- The Native IDE for Web3 Development.** with links to Website, Documentation, Remix Plugin, and Remix Desktop.
- Search Documentation** bar.
- Files** section with buttons for New File, Open File, and Access File System.
- Load from** section with buttons for GitHub, Gist, IPFS, and HTTPS.
- Learn** section with links to Remix Basics, Intro to Solidity, and Deploying with Libraries.

The main workspace area includes:

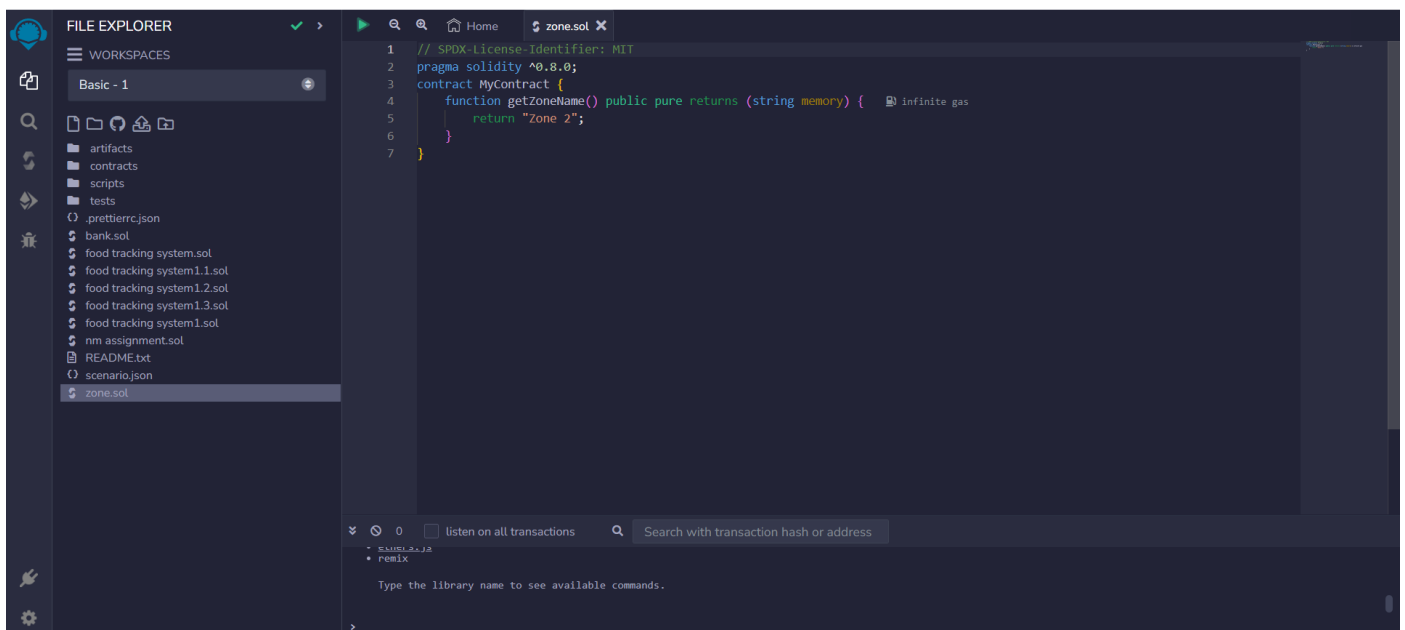
- Featured** section with a BETA TESTING announcement and a Sign up button.
- Get Started - Project Templates** section with five templates: GNOSIS SAFE MULTISIG, OXPROJECT ERC20, OPENZEPPELIN ERC20, OPENZEPPELIN ERC721, and OPENZ.
- Featured Plugins** section with four plugins: SOLIDITY ANALYZERS, COOKBOOK, SOLIDITY, and SOURCIFY.
- Scam Alert** section with a warning icon and text: "The only URL Remix uses is remix.ethereum.org. Beware of online videos promoting 'liquidity front runner bots'. Learn more. Additional safety tips: here".

The bottom status bar shows a search bar with the text "Search with transaction hash or address" and a "listen on all transactions" checkbox.

2. OPEN THE REMIX PAGE AND CREATE A NEW FILE




3. IN THE NEWLY CREATED FILE, CREATE A PROGRAM TO RETURN YOUR STRING , "ZONE.SOL"



The screenshot shows the Visual Studio Code editor interface. On the left, the 'FILE EXPLORER' sidebar displays a workspace named 'Basic - 1' with a file tree containing various files and folders. The file 'zone.sol' is selected. The main editor area shows the content of 'zone.sol', which is a Solidity program. The code is as follows:

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3 contract MyContract {
4     function getZoneName() public pure returns (string memory) {
5         return "Zone 2";
6     }
7 }
```

At the bottom of the editor, there is a terminal window with the text 'Type the library name to see available commands.'



SOLIDITY COMPILER

COMPILER +

0.8.18+commit.87f61d96

Include nightly builds

Auto compile

Hide warnings

Advanced Configurations

Compile zone.sol

Compile and Run script

CONTRACT

MyContract (zone.sol)

Publish on Ipfs

Publish on Swarm

Compilation Details

ABI

Bytecode

zone.sol

1 // SPDX-License-Identifier: MIT

2 pragma solidity ^0.8.0;

3 contract MyContract {

4 function getZoneName() public pure returns (string memory) { infinite gas

5 return "Zone 2";

6 }

7 }


0

listen on all transactions

Search with transaction hash or address

remix

Type the library name to see available commands.



SOLIDITY COMPILER

COMPILER +

0.8.18+commit.87f61d96

Include nightly builds

Auto compile

Hide warnings

Advanced Configurations

Compile zone.sol

Compile and Run script

CONTRACT

MyContract (zone.sol)

Publish on Ipfs

Publish on Swarm

Compilation Details

ABI

Bytecode

zone.sol

1 // SPDX-License-Identifier: MIT

2 pragma solidity ^0.8.0;

3 contract MyContract {

4 function getZoneName() public pure returns (string memory) { infinite gas

5 return "Zone 2";

6 }

7 }

0

listen on all transactions

Search with transaction hash or address

remix

Type the library name to see available commands.

4.SAVE THE PROGRAM AND COMPILE IT TO GET THE ABI AND BYTECODE

ABI:

```
[
  {
    "inputs": [],
    "name": "getZoneName",
    "outputs": [
      {
        "internalType": "string",
        "name": "",
        "type": "string"
      }
    ],
    "stateMutability": "pure",
    "type": "function"
  }
]
```

BYTECODE:

```
608060405234801561001057600080fd5b50610173806100206000396000f3fe6080604052348015610010576000
80fd5b506004361061002b5760003560e01c8063235ed70f14610030575b600080fd5b61003861004e565b604051
610045919061011b565b60405180910390f35b60606040518060400160405280600681526020017f5a6f6e652032
0000000000000000000000000000000000000000000000000000000000000000815250905090565b600081519050919050565b6
0082825260208201905092915050565b60005b838110156100c55780820151818401526020810190506100aa56
5b60008484015250505050565b6000601f19601f8301169050919050565b60006100ed8261008b565b6100f78185
610096565b93506101078185602086016100a7565b610110816100d1565b840191505092915050565b600060208
2019050818103600083015261013581846100e2565b90509291505056fea2646970667358221220b8236eeb307b
b1e0cc25993c06f915523963b48881be742dcf4e6e2fceb17ede64736f6c634300081200
```

FINALLY DEPLOY IT TO DISPLAY THE OUTPUT

The screenshot displays the Remix IDE interface, which is used for developing and deploying smart contracts. The interface is divided into several panels:

- Left Panel (Deploy & Run Transactions):** This panel contains settings for deploying a contract. It includes fields for the account (0x5B3...eddC4), gas limit (3000000), and value (0 Wei). The contract name is set to "MyContract - zone.sol". There are buttons for "Deploy" and "At Address". Below these, it shows "Transactions recorded" and a list of "Deployed Contracts" with addresses like 0xD91...39138 and 0xD6B...33FA8.
- Top Panel (Code Editor):** This panel shows the Solidity code for the "zone.sol" file. The code is as follows:

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3 contract MyContract {
4     function getZoneName() public pure returns (string memory) {
5         return "Zone 2";
6     }
7 }
```
- Bottom Panel (Remix Console):** This panel shows the output of the deployment process. It displays the message "Welcome to Remix 0.36.3" and "Your files are stored in indexedDB, 10.06 KB / 168.46 GB used".