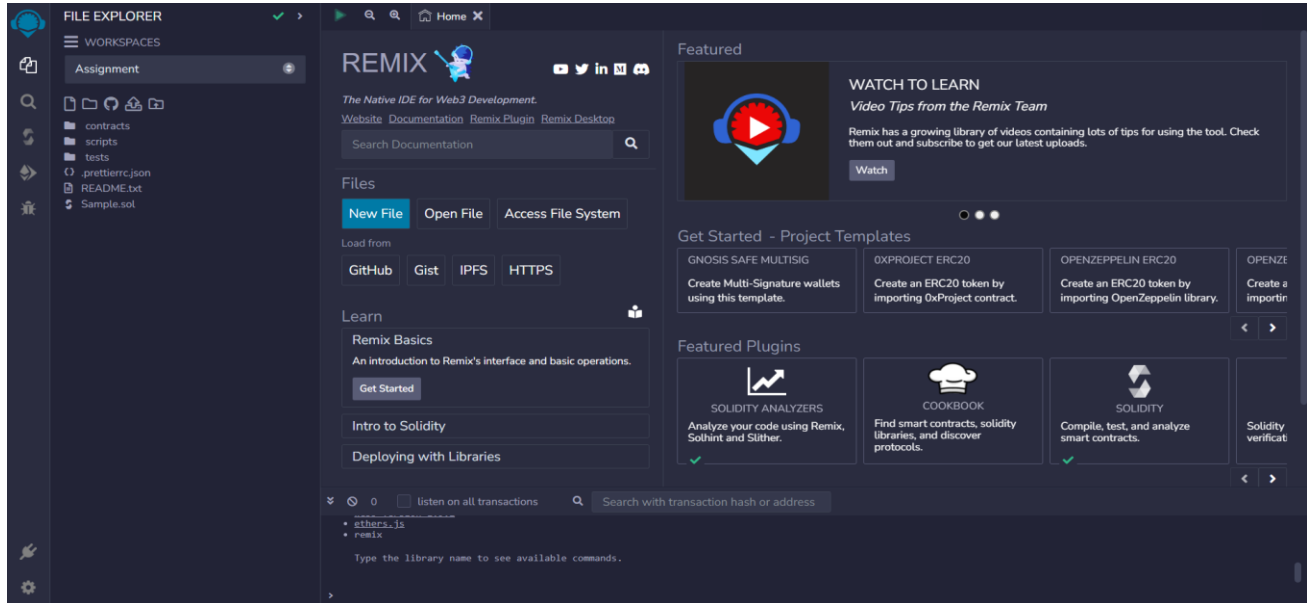


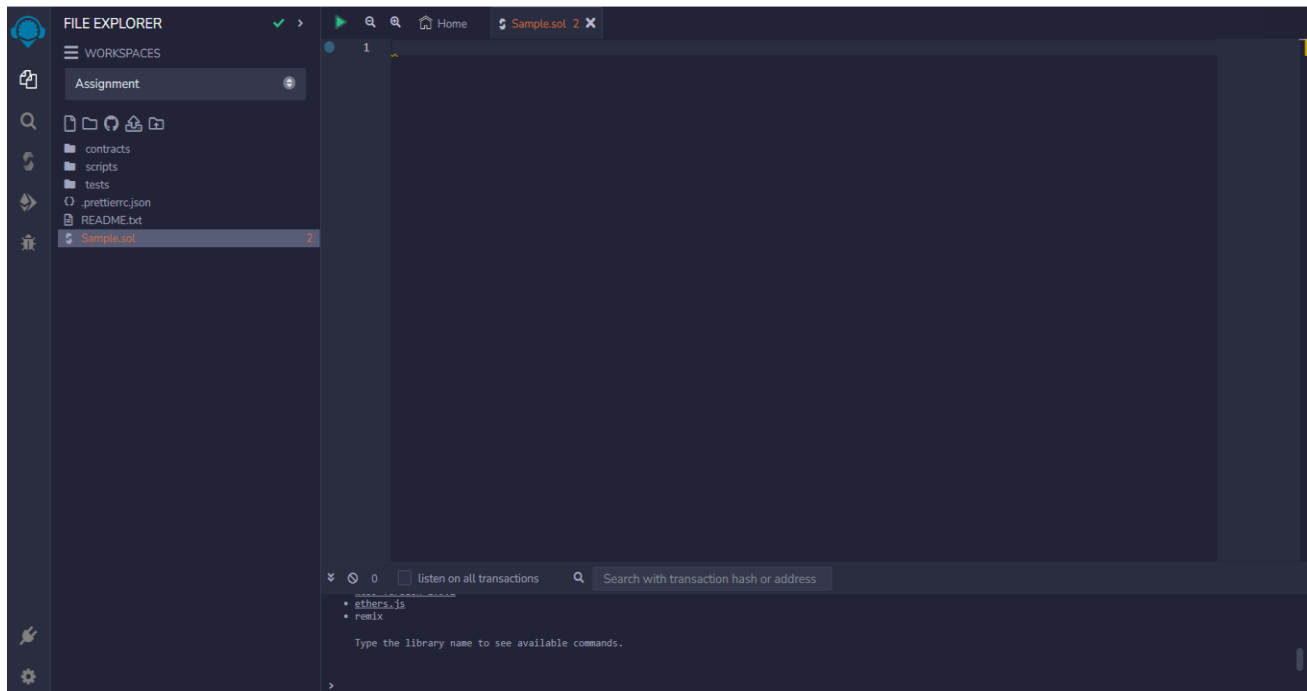
BLOCKCHAIN TECHNOLOGY

ASSIGNMENT-1

1.OPEN REMIX PLATFORM:



2.CREATE A NEW FILE IN REMIX PLATFORM:



3.PROGRAM TO RETURN STRING:

```
// SPDX-License-Identifier: MIT
```

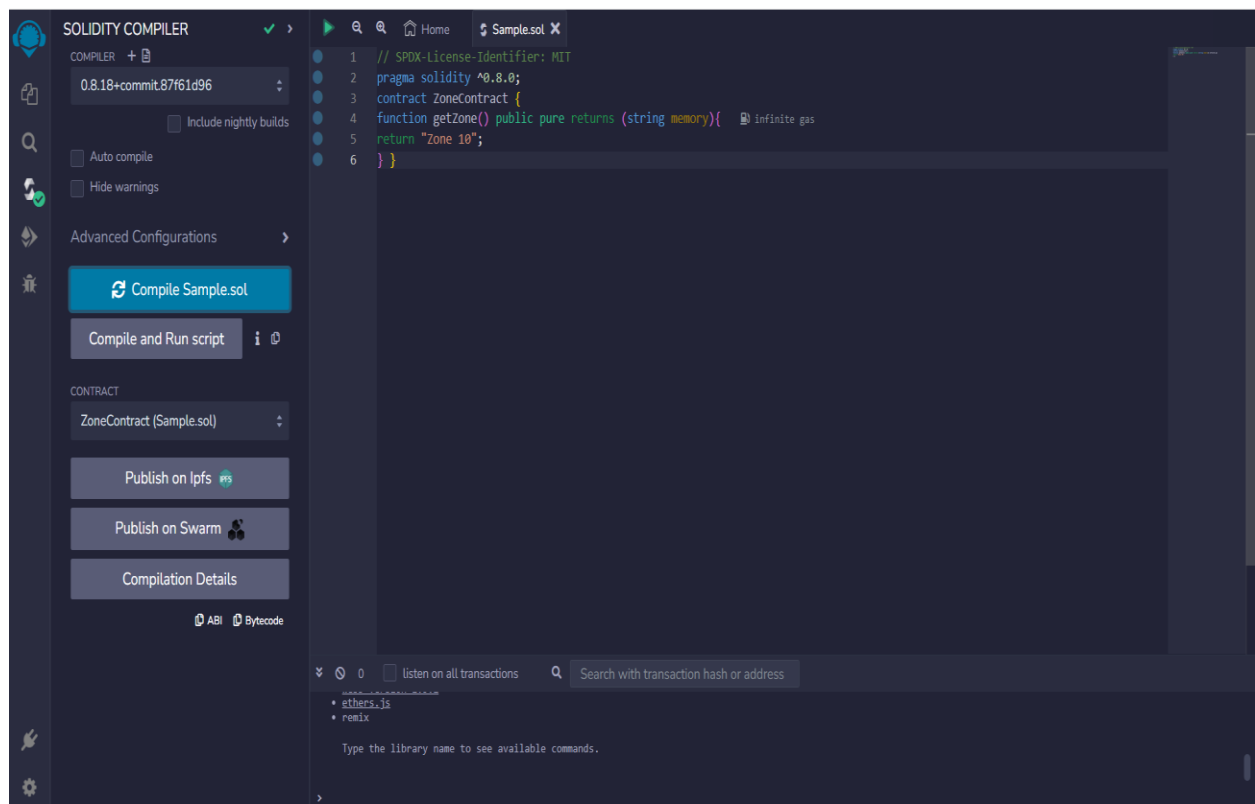
```
pragma solidity ^0.8.0;
```

```
contract ZoneContract {
```

```
function getZone() public pure returns (string memory){
```

```
return "Zone 10";
```

```
} }
```



4.ABI AND BYTECODE:

ABI:

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

BYTECODE:

```
608060405234801561001057600080fd5b50610173806100206000396000f3fe608060405234801
561001057600080fd5b506004361061002b5760003560e01c8063f9095eb714610030575b600080
fd5b61003861004e565b604051610045919061011b565b60405180910390f35b60606040518060
400160405280600781526020017f5a6f6e6520313000000000000000000000000000000000
00000000000000815250905090565b600081519050919050565b60008282526020820190509291
5050565b60005b838110156100c55780820151818401526020810190506100aa565b6000848401
5250505050565b6000601f19601f8301169050919050565b60006100ed8261008b565b6100f7818
5610096565b93506101078185602086016100a7565b610110816100d1565b84019150509291505
0565b6000602082019050818103600083015261013581846100e2565b90509291505056fea2646
970667358221220dee2b9033b703b5528e16bd91c4efa2dff98609b74f819ce575faed79ca53da364
736f6c63430008120033
```

5.DELPOY AND OUTPUT:

The screenshot displays the Remix IDE interface, which is used for developing, deploying, and interacting with Ethereum 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), **VALUE** (0 Wei), and the **CONTRACT** (ZoneContract - Sample.sol). There is a **Deploy** button and a checkbox for **Publish to IPFS**. Below these, there is a section for **Transactions recorded** and **Deployed Contracts**. The deployed contract is listed as **ZONECONTRACT AT 0xD91...391** with a balance of 0 ETH. A **getZone** button is visible, and the output of the function call is shown as **0: string: Zone 10**.
- Top Panel (Code Editor):** This panel shows the Solidity code for the **Sample.sol** contract. The code is as follows:

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3 contract ZoneContract {
4     function getZone() public pure returns (string memory){ infinite gas
5     return "Zone 10";
6 }
```
- Bottom Panel (Console/Debug):** This panel shows the execution results. It includes a search bar and a list of transactions. The first transaction is a **creation of ZoneContract pending...** with a status of **web3_version 1.5.2**, **ethers.js**, and **relix**. The second transaction is a **call to ZoneContract.getZone** with a status of **check** and a message: **[vm] from: 0x5B3...eddC4 to: ZoneContract.(constructor) value: 0 wei data: 0x688...20833 logs: 0 hash: 0xe9b...5770f**. The third transaction is a **call to ZoneContract.getZone** with a status of **check** and a message: **[call] from: 0x5B380a6a701c568545dCfcB03FcB875F56beddC4 to: ZoneContract.getZone() data: 0xf90...95eb7**. There are **Debug** buttons next to each transaction.