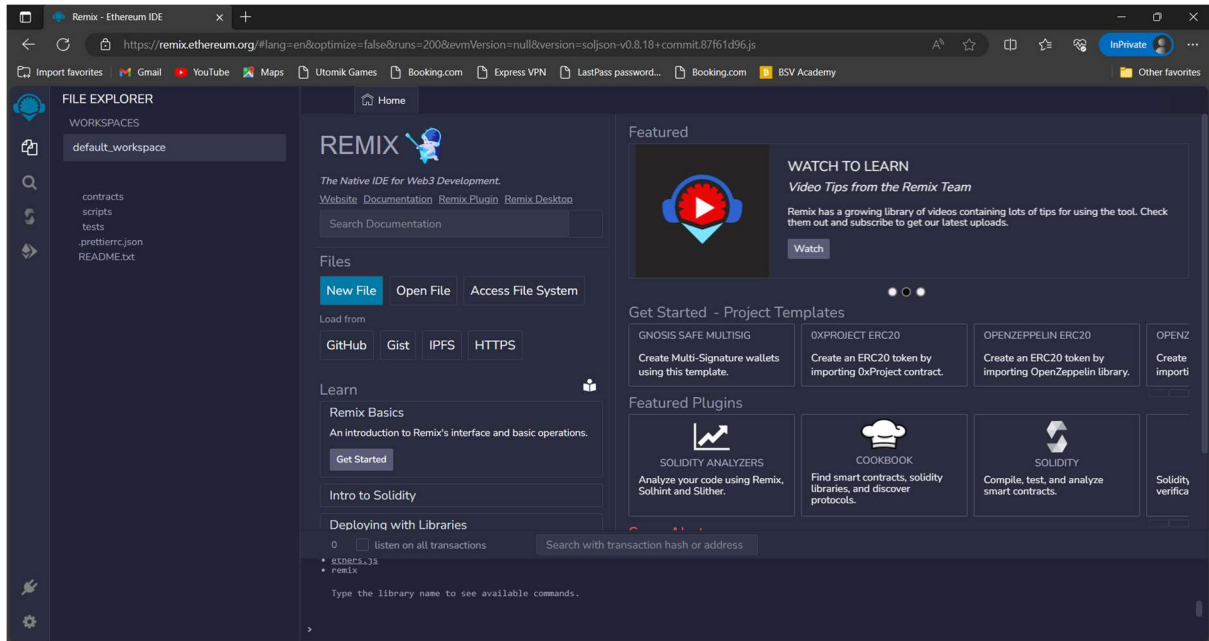


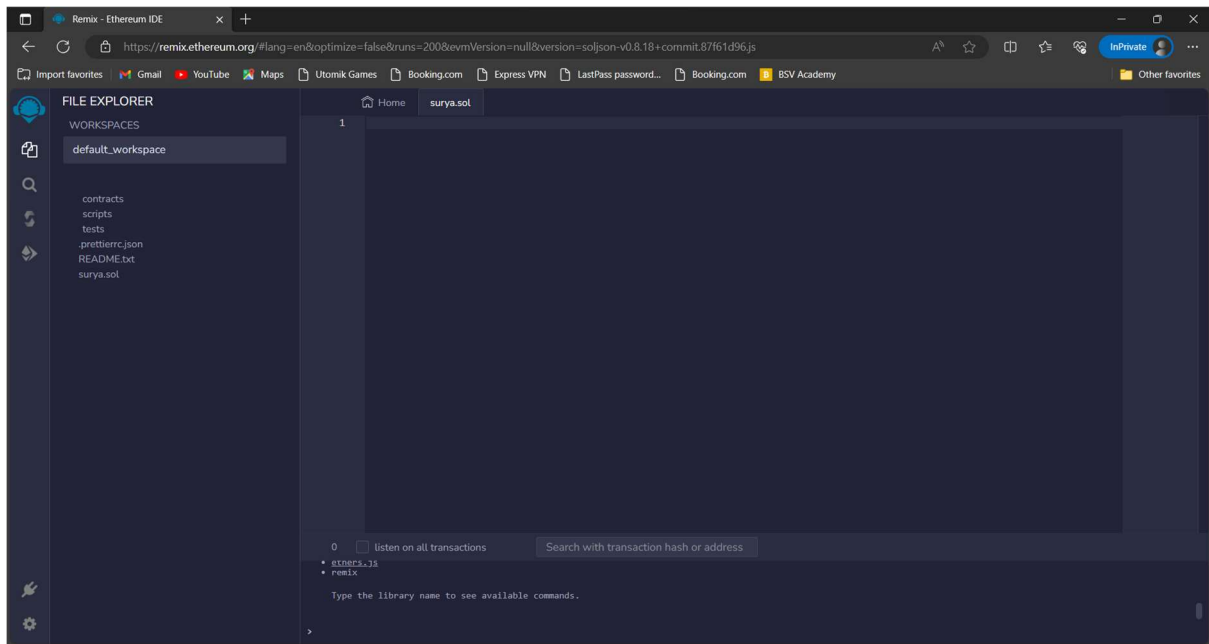
Blockchain Technology

Assignment-01

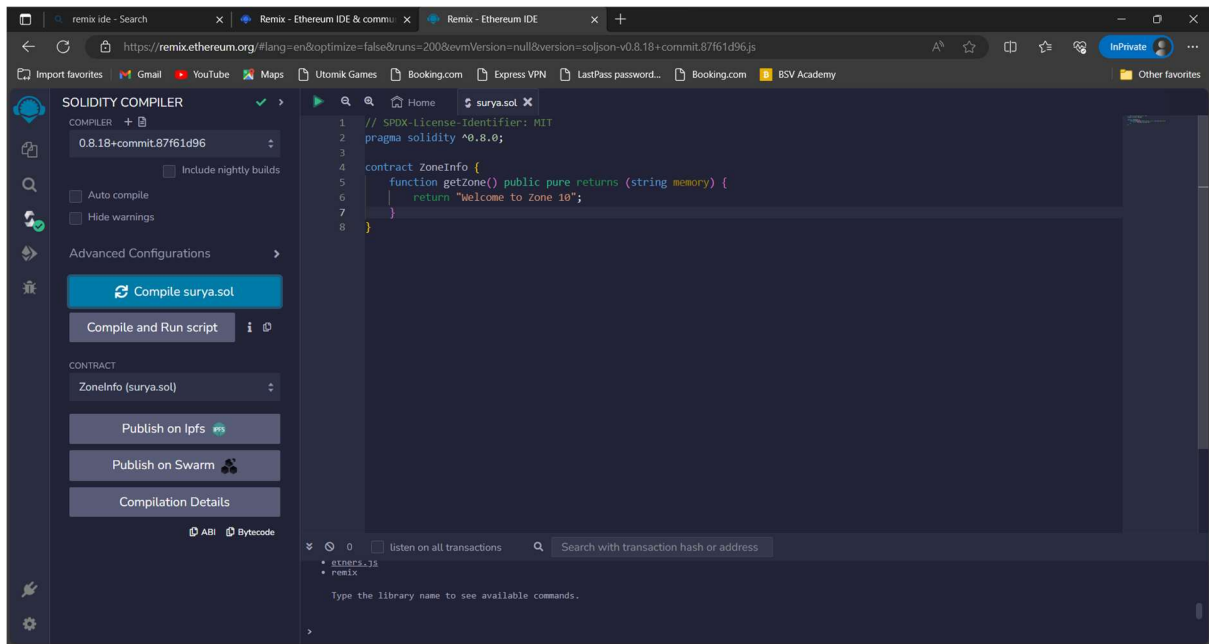
1. Open Remix Platform



2. Creating a new file



3. Program to return a string



The screenshot displays the Remix IDE interface in a web browser. The main editor window shows a Solidity file named `surya.sol` with the following code:

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract ZoneInfo {
5     function getzone() public pure returns (string memory) {
6         return "Welcome to Zone 10";
7     }
8 }
```

The left sidebar contains the **SOLIDITY COMPILER** panel, which shows the compiler version as `0.8.18+commit.87f61d96`. It includes options for `Auto compile` and `Hide warnings`, and a section for **Advanced Configurations**. Below these are buttons for `Compile surya.sol`, `Compile and Run script`, and a **CONTRACT** dropdown menu set to `ZoneInfo (surya.sol)`. Further down are buttons for `Publish on Ipfs`, `Publish on Swarm`, and `Compilation Details`. At the bottom of the sidebar are links for `ABI` and `Bytecode`.

The bottom panel shows the `ABI` tab, which lists the `getzone()` function and provides a search bar for transaction hashes or addresses.

4. ABI

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

Byte Code

[illegible]

5. Deployment

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

- Left Panel (Deploy & Run Transactions):** This panel contains controls for deploying and interacting with contracts. It shows the contract name "ZoneInfo - surya.sol", the deployment button, and options to publish to IPFS or load a contract from an address. Below this, it lists "Deployed Contracts" with a table showing the contract name, address, and balance. The "Low level interactions" section is also visible.
- Top Panel (Code Editor):** This panel displays the Solidity code for the "ZoneInfo" contract. The code is as follows:

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract ZoneInfo {
5     function getzone() public pure returns (string memory) { Infinite gas
6         return "Welcome to Zone 10";
7     }
8 }
```
- Bottom Panel (Debug Console):** This panel shows the execution results of the contract. It displays a call to the "getzone" function, which returned the string "Welcome to Zone 10". The console also shows the transaction hash and the gas used.