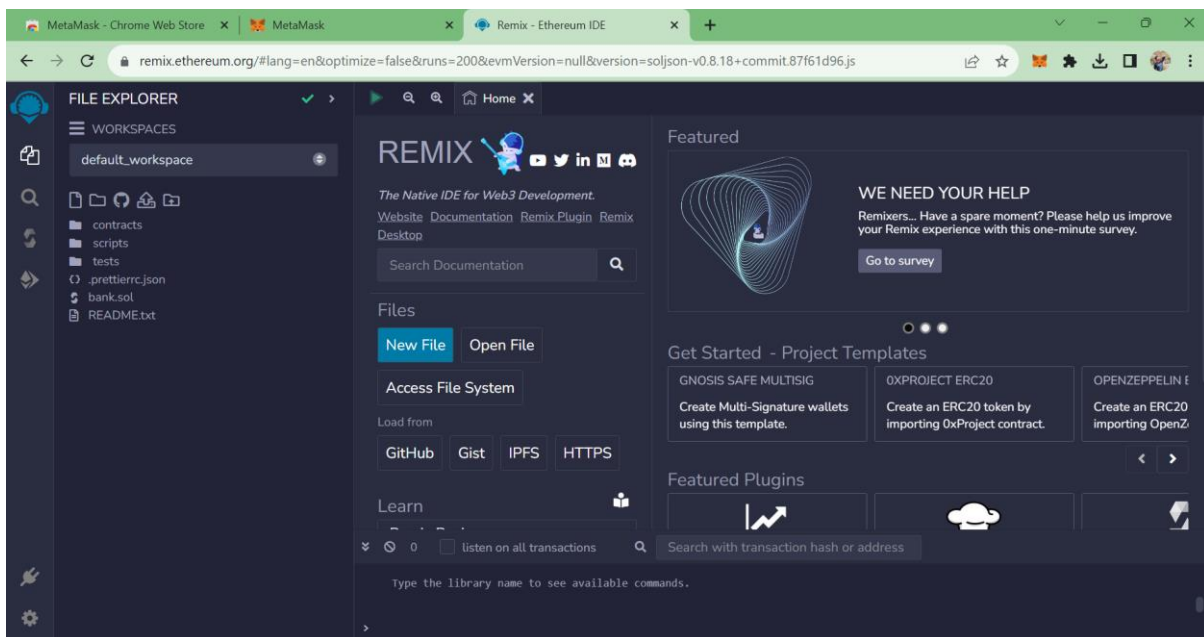


ASSIGNMENT-1

NAME	
ABINASH SJ	
ZONE	4
COLLEGE	GKMCET

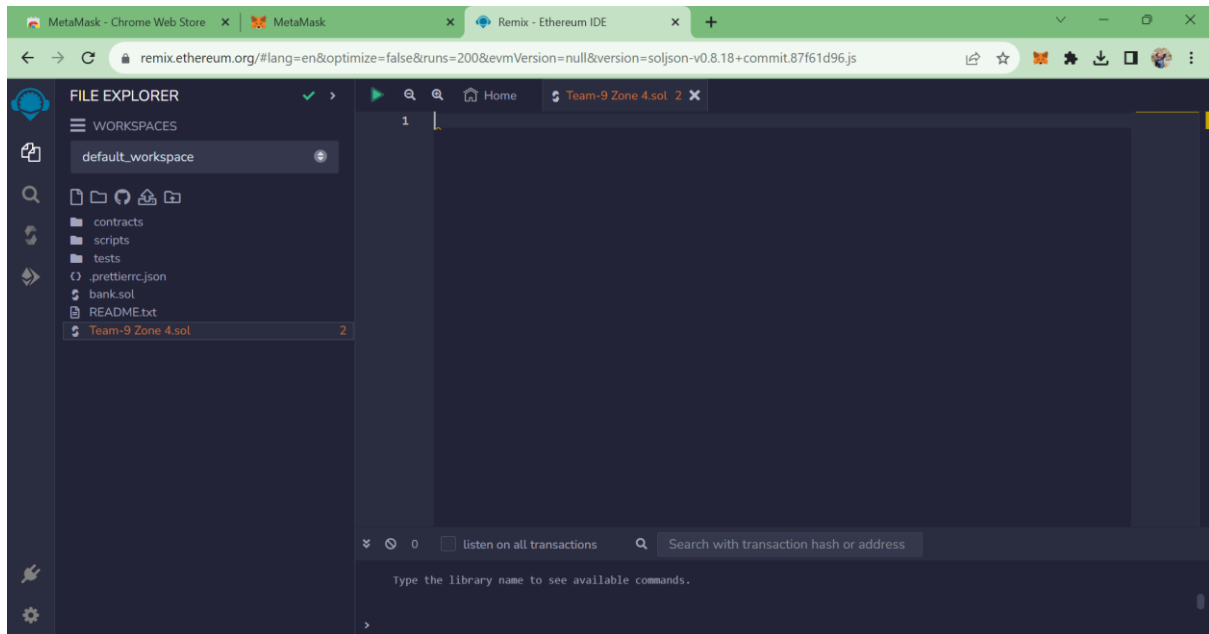
STEP 1:

Open the Remix in the chrome:



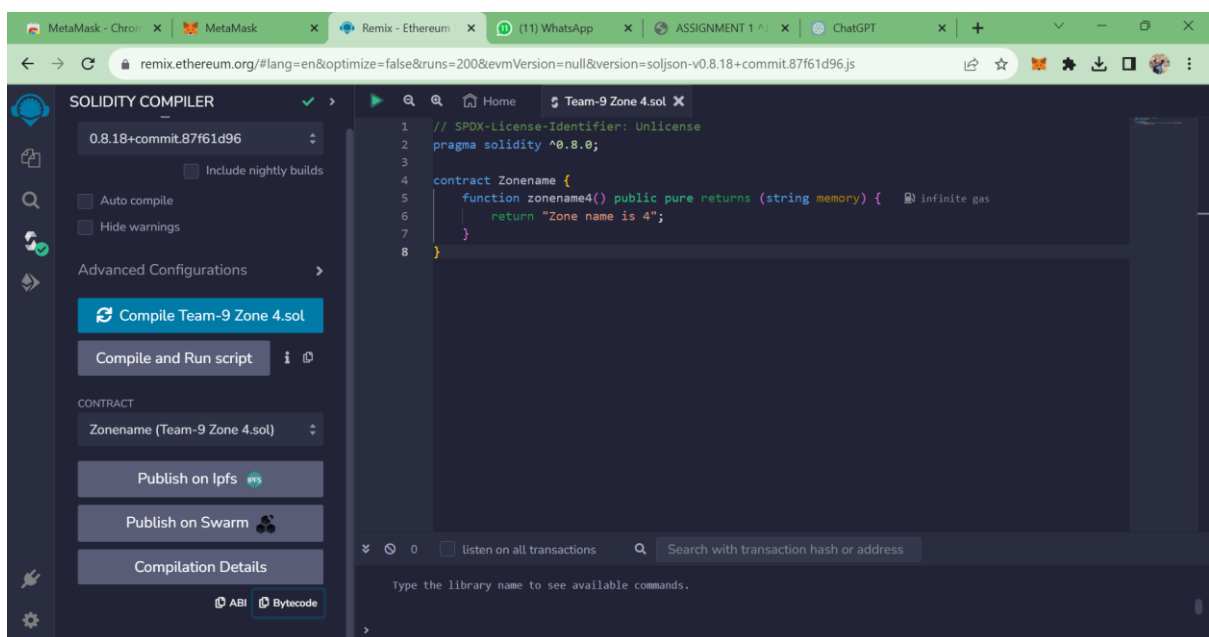
STEP 2:

Open the New file:



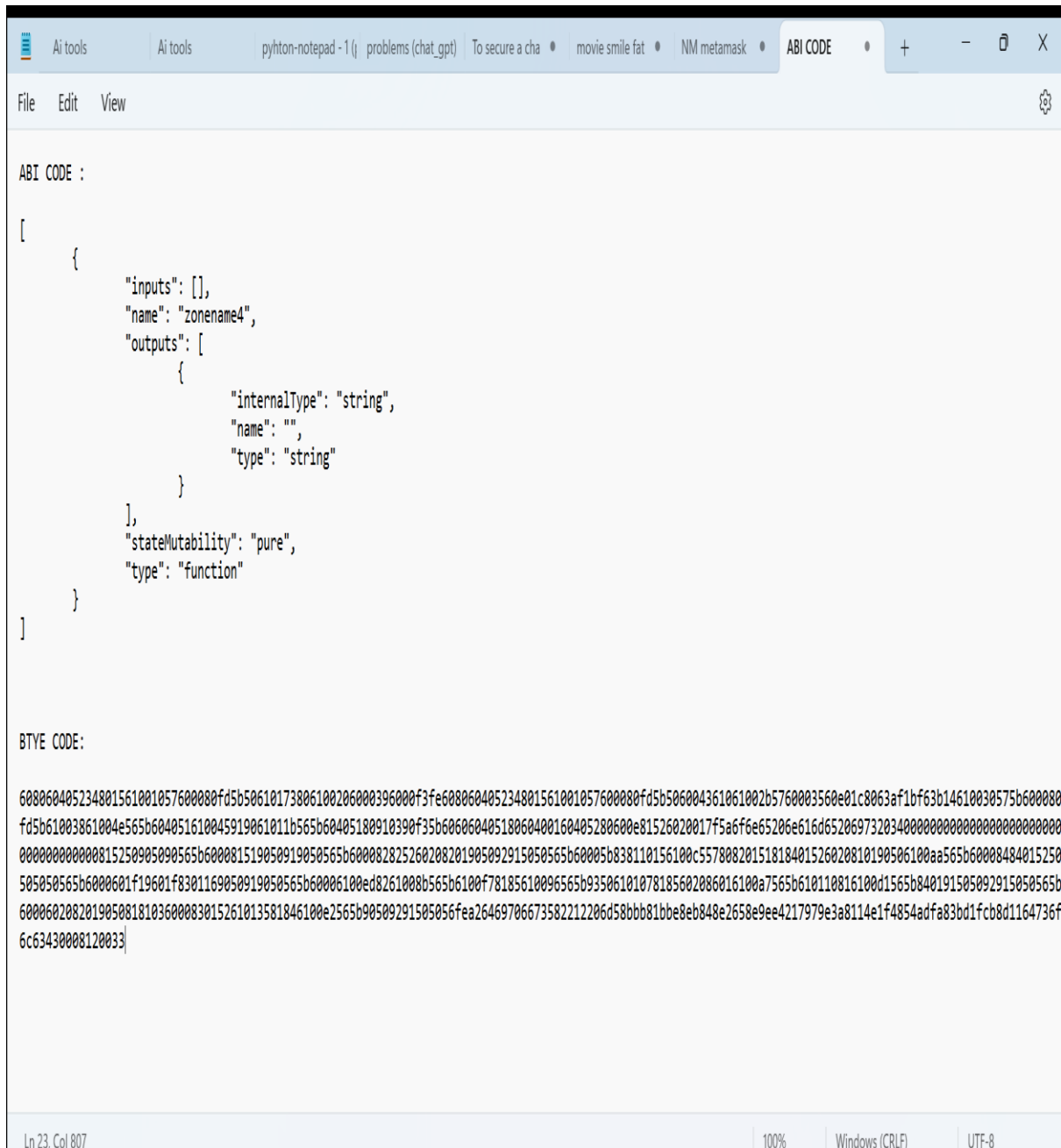
STEP 3:

Write and Compile the file:



STEP 4:

Save the program and compile it to get the ABI and BYTE CODE:



The screenshot shows a code editor window with a tab labeled 'ABI CODE'. The editor contains the following text:

```
ABI CODE :  
  
[  
  {  
    "inputs": [],  
    "name": "zonename4",  
    "outputs": [  
      {  
        "internalType": "string",  
        "name": "",  
        "type": "string"  
      }  
    ],  
    "stateMutability": "pure",  
    "type": "function"  
  }  
]  
  
BYTE CODE:  
  
608060405234801561001057600080fd5b50610173806100206000396000f3fe608060405234801561001057600080fd5b506004361061002b5760003560e01c8063af1bf63b14610030575b600080  
fd5b61003861004e565b604051610045919061011b565b60405180910390f35b6060606040518060400160405280600e81526020017f5a6f6e65206e616d652069732034000000000000000000000000  
000000000000815250905090565b600081519050919050565b600082825260208201905092915050565b60005b838110156100c55780820151818401526020810190506100aa565b60008484015250  
505050565b6000601f19601f8301169050919050565b60006100ed8261008b565b6100f78185610096565b93506101078185602086016100a7565b610110816100d1565b840191505092915050565b  
6000602082019050818103600083015261013581846100e2565b90509291505056fea26469706673582212206d58bbb81bbe8eb848e2658e9ee4217979e3a8114e1f4854adfa83bd1fcb8d1164736f  
6c63430008120033]
```

The status bar at the bottom indicates 'Ln 23, Col 807', '100%', 'Windows (CRLF)', and 'UTF-8'.

STEP 5:

Deploy the program and get the output:

The screenshot displays the Remix Ethereum IDE interface. The top bar shows the URL `remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js`. The left sidebar contains the 'DEPLOY & RUN TRANSACTIONS' panel, which includes fields for 'GAS LIMIT' (3000000), 'VALUE' (0 Wei), and a dropdown for 'CONTRACT' (Zonename - Team-9 Zone 4.sol). The 'Deploy' button is highlighted. Below this, there are options for 'Publish to IPFS' and 'At Address'. The 'Transactions recorded' section shows 2 transactions, with 'Run transactions using the latest compilation result' checked. The 'Deployed Contracts' section is at the bottom.

The main editor displays the Solidity code for the 'Zonename' contract:

```
1 // SPDX-License-Identifier: Unlicense
2 pragma solidity ^0.8.0;
3
4 contract Zonename {
5     function zonename4() public pure returns (string memory) {
6         return "Zone name is 4";
7     }
8 }
```

The right sidebar shows the 'Debug' console with a successful transaction log:

- status: true Transaction mined and execution succeed
- transaction hash: 0x777eb63db47746af193bd2ff9a42f1c19052fa1f61814ef1516c4edf36ff7
- block hash: 0xd95efa9fd6ae8e0b35c3dc8d78a91dce5a711538575b6d43b42ca256dca24a67
- block number: 3
- contract address: 0xf8e81d47203a594245e36c48e151709f0c19f8e8
- from: 0x58380a6a701c568545dcfc803fc8875f56beddC4
- to: Zonename.(constructor)
- gas: 153133 gas
- transaction cost: 133185 gas
- execution cost: 74323 gas