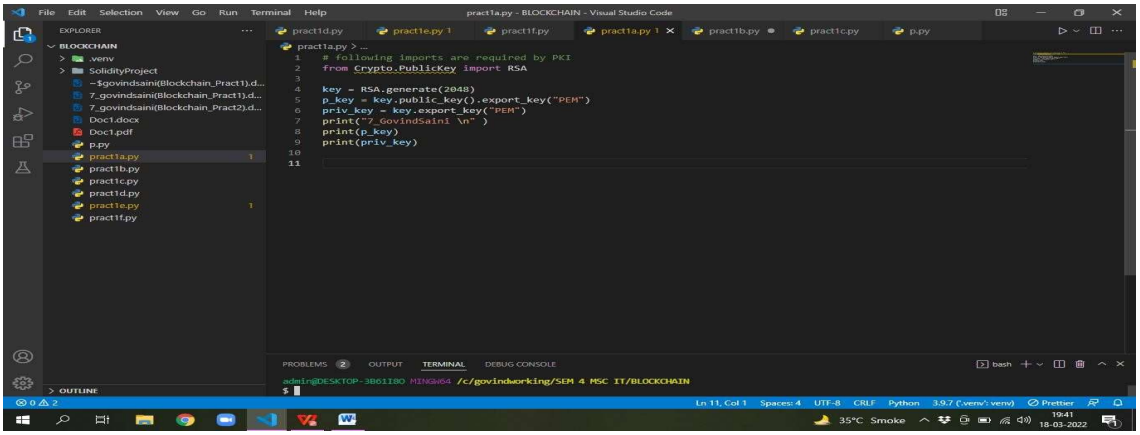
Practical No : 1

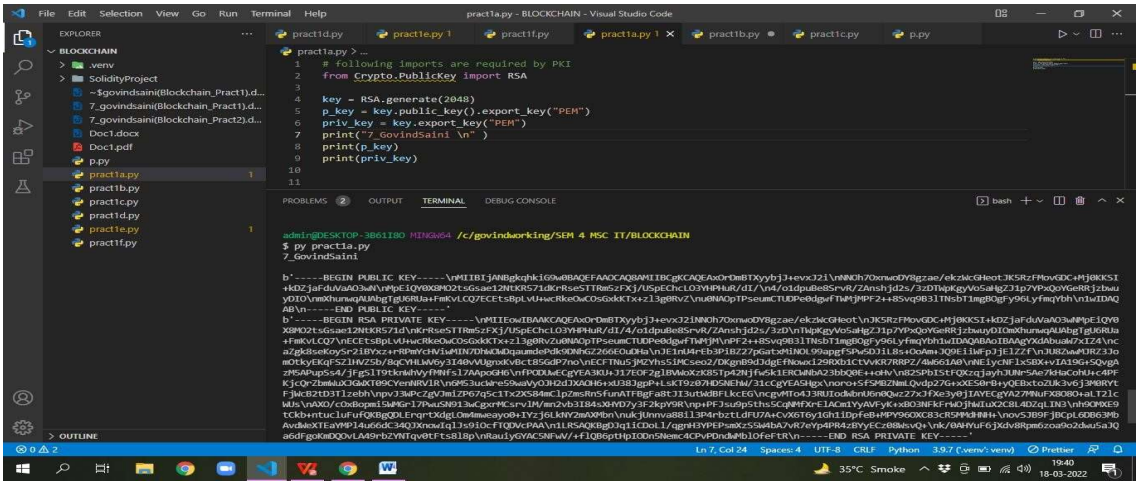
Aim : Write the following programs for Blockchain in Python.

1a) A simple client class that generates the private and public keys by using the built in Python RSA algorithm and test it.

Code:

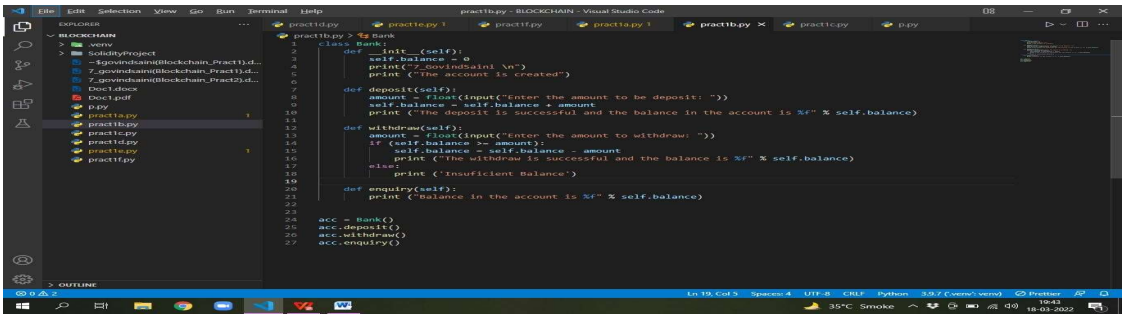


Output :

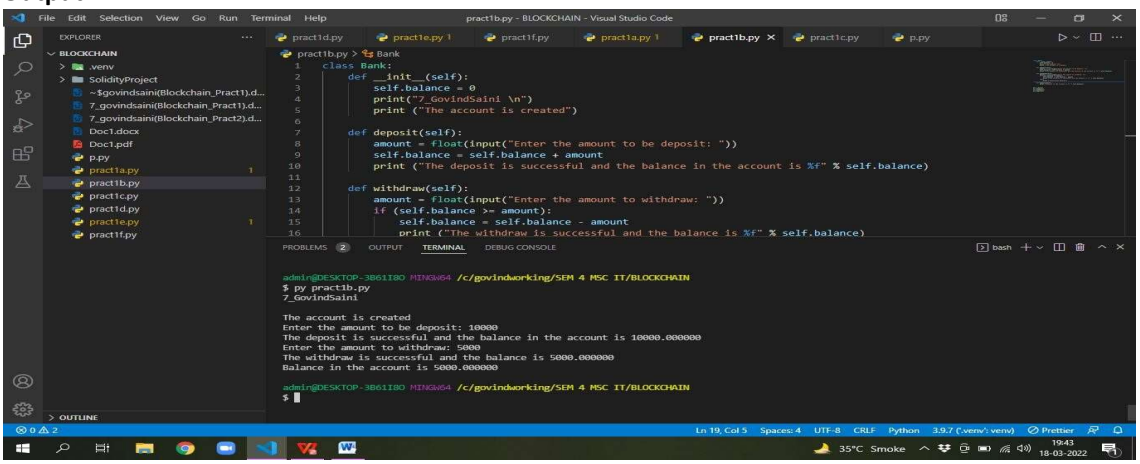


1b) A transaction class to send and receive money and test it.

Code:

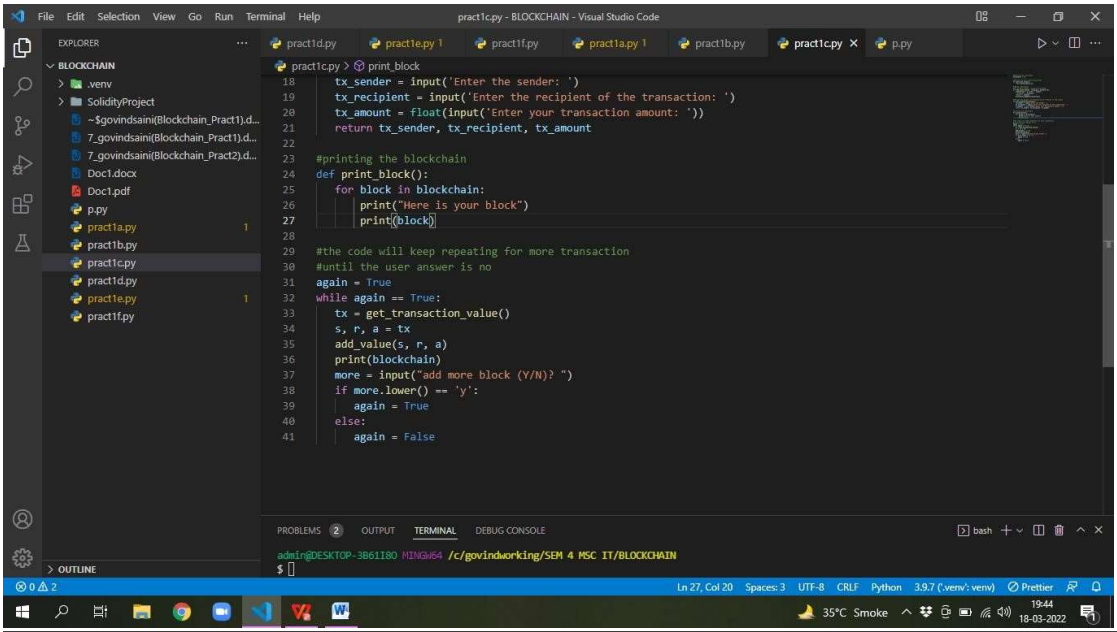


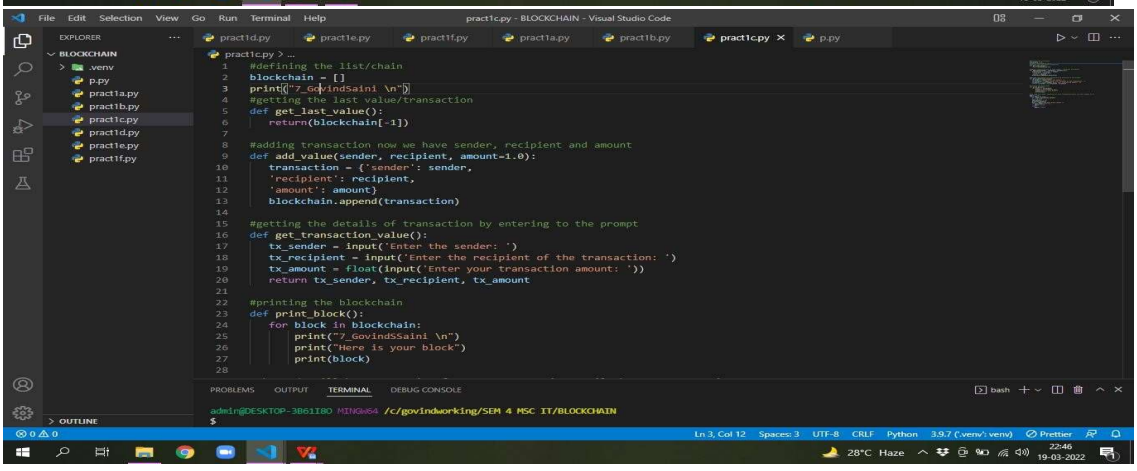
Output:



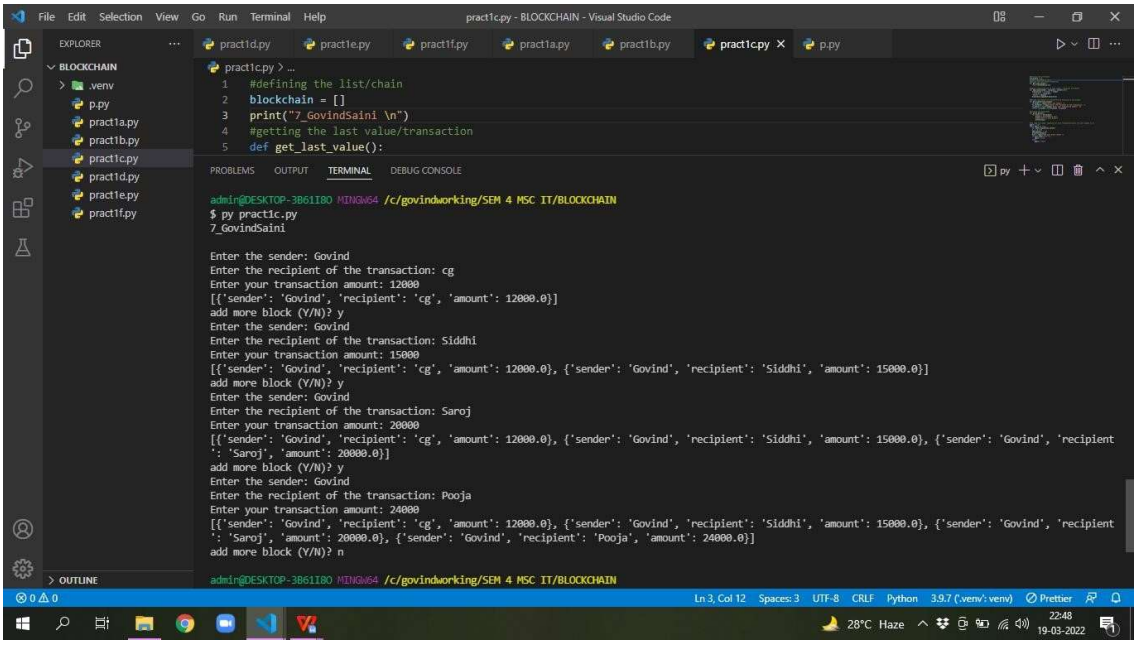
1c) Create Multiple Transaction and display them.

Code :



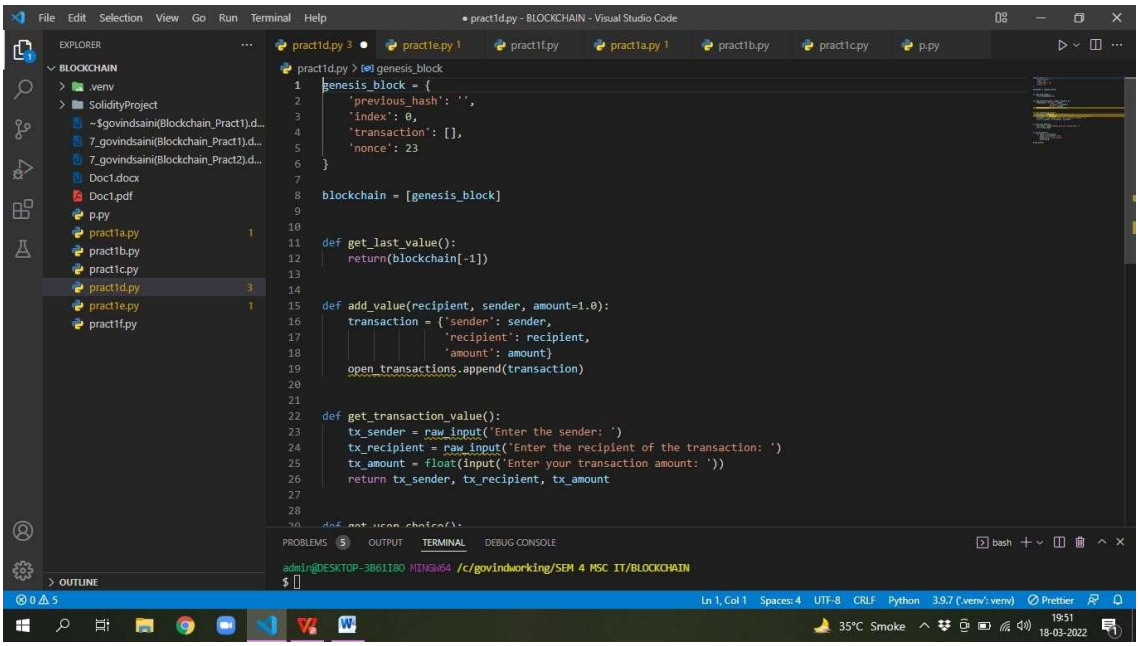


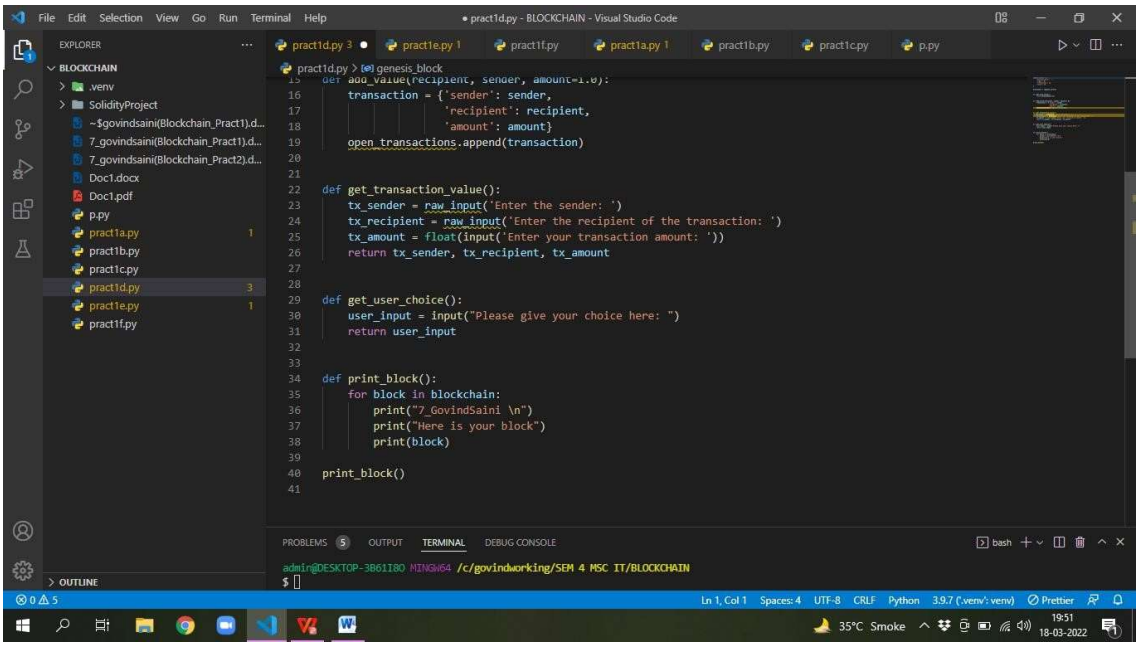
Output :



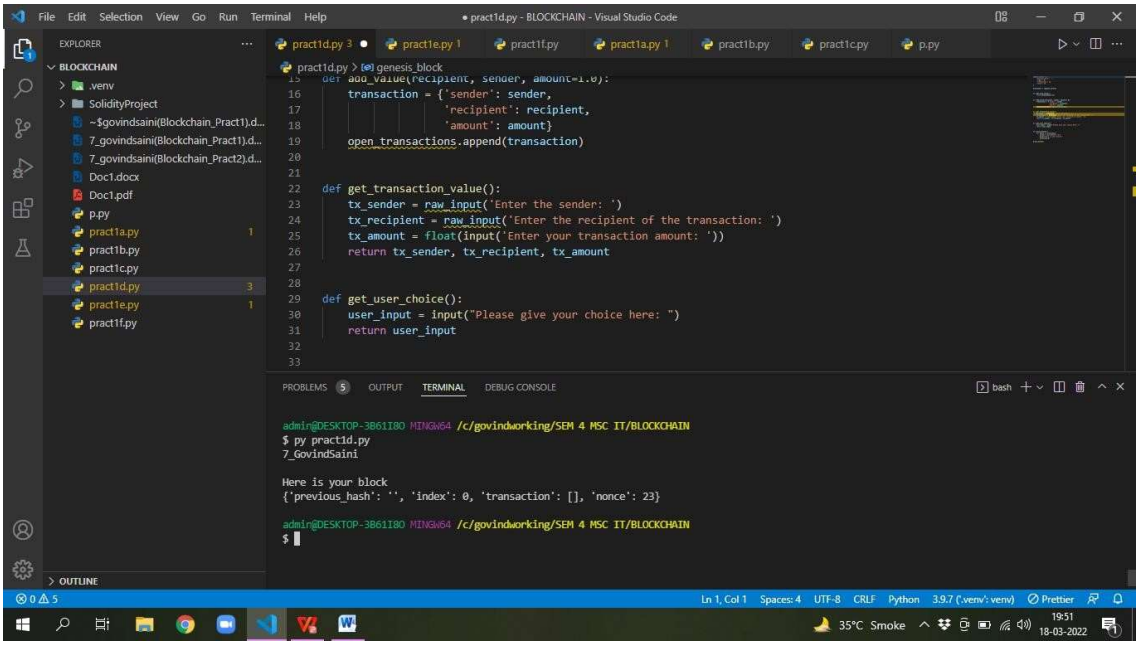
1d) Create a blockchain, a genesis block and execute it.

Code :



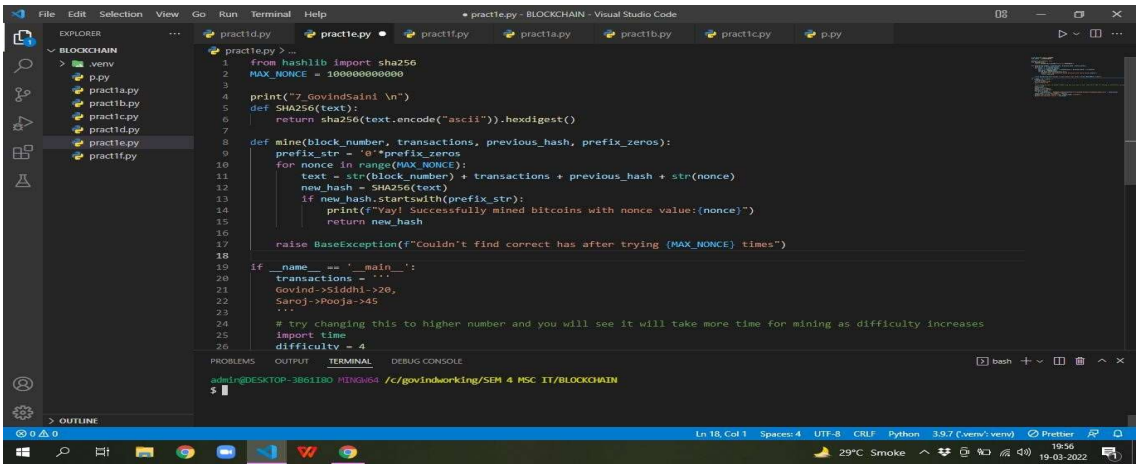


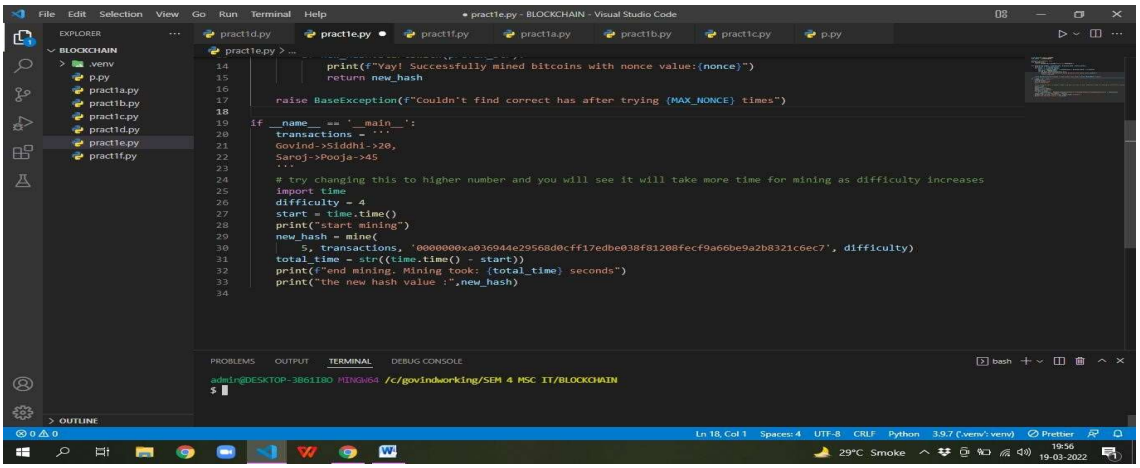
Output :



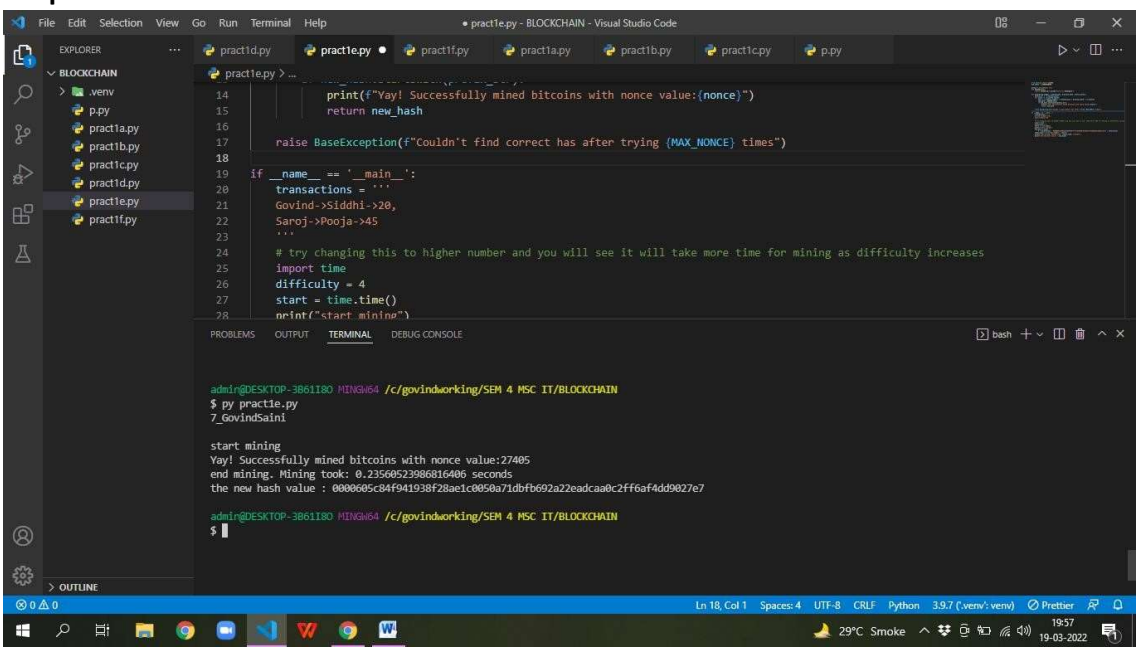
1e) Create a mining function and test it.

Code :

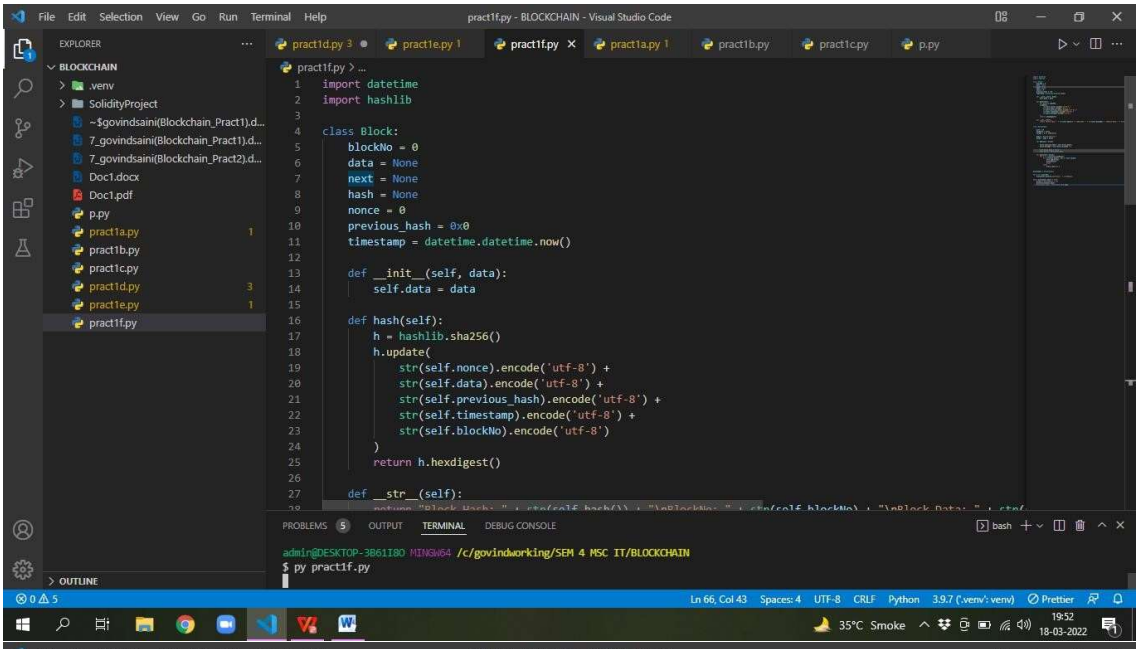

File Edit Selection View Go Run Terminal Help
• pract1e.py - BLOCKCHAIN - Visual Studio Code
EXPLORER
BLOCKCHAIN
.venv
p.py
pract1a.py
pract1b.py
pract1c.py
pract1d.py
pract1e.py
pract1f.py
pract1d.py
pract1e.py
pract1f.py
pract1a.py
pract1b.py
pract1c.py
p.py
pract1e.py > ...
from hashlib import sha256
MAX_NONCE = 100000000000
print("7_GovindSaini \n")
def SHA256(text):
return sha256(text.encode("ascii")).hexdigest()
def mine(block_number, transactions, previous_hash, prefix_zeros):
prefix_str = '0'*prefix_zeros
for nonce in range(MAX_NONCE):
text = str(block_number) + transactions + previous_hash + str(nonce)
new_hash = SHA256(text)
if new_hash.startswith(prefix_str):
print(f"Yay! Successfully mined bitcoins with nonce value:{nonce}")
return new_hash
raise BaseException(f"Couldn't find correct has after trying {MAX_NONCE} times")
if __name__ == '__main__':
transactions = '''
Govind->Siddhi->20,
Saroj->Pooja->45
'''
# try changing this to higher number and you will see it will take more time for mining as difficulty increases
import time
difficulty = 4
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
bash
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$
OUTLINE
Ln 18, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.7 ('.venv': venv) Prettier
29°C Smoke
19:56
19-03-2022


File Edit Selection View Go Run Terminal Help
• pract1e.py - BLOCKCHAIN - Visual Studio Code
print(f"Yay! Successfully mined bitcoins with nonce value:{nonce}")
return new_hash
raise BaseException(f"Couldn't find correct has after trying {MAX_NONCE} times")
if __name__ == '__main__':
transactions = '''
Govind->Siddhi->20,
Saroj->Pooja->45
'''
# try changing this to higher number and you will see it will take more time for mining as difficulty increases
import time
difficulty = 4
start = time.time()
print("start mining")
new_hash = mine(
5, transactions, '0000000xa036944e29568d0cff17edbe038f81208fecf9a66be9a2b8321c6ec7', difficulty)
total_time = str((time.time() - start))
print(f"end mining. Mining took: {total_time} seconds")
print("the new hash value :",new_hash)
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
bash
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$
OUTLINE
Ln 18, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.7 ('.venv': venv) Prettier
29°C Smoke
19:56
19-03-2022

Output :

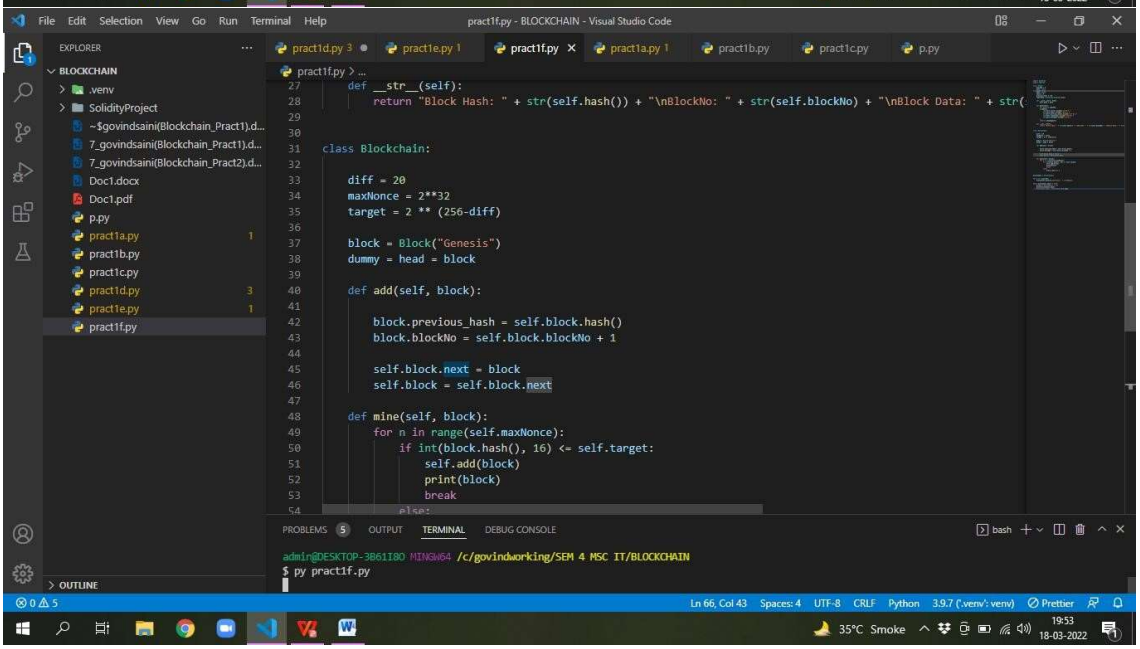

File Edit Selection View Go Run Terminal Help
• pract1e.py - BLOCKCHAIN - Visual Studio Code
print(f"Yay! Successfully mined bitcoins with nonce value:{nonce}")
return new_hash
raise BaseException(f"Couldn't find correct has after trying {MAX_NONCE} times")
if __name__ == '__main__':
transactions = '''
Govind->Siddhi->20,
Saroj->Pooja->45
'''
# try changing this to higher number and you will see it will take more time for mining as difficulty increases
import time
difficulty = 4
start = time.time()
print("start mining")
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
bash
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$ py pract1e.py
7_GovindSaini
start mining
Yay! Successfully mined bitcoins with nonce value:27405
end mining. Mining took: 0.23560523986816406 seconds
the new hash value : 0000605c84f941938f28ae1c0050a71dbfb692a22eadcaa0c2ff6af4dd9027e7
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$
OUTLINE
Ln 18, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.7 ('.venv': venv) Prettier
29°C Smoke
19:57
19-03-2022

1f) Add blocks to the miner and dump the blockchain Code :



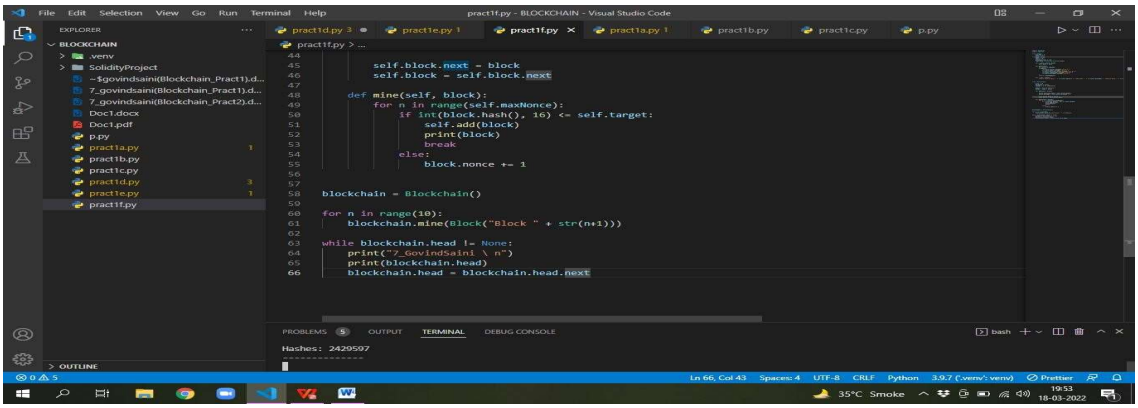
```
import datetime
import hashlib

class Block:
    blockNo = 0
    data = None
    next = None
    hash = None
    nonce = 0
    previous_hash = 0x0
    timestamp = datetime.datetime.now()

    def __init__(self, data):
        self.data = data

    def hash(self):
        h = hashlib.sha256()
        h.update(
            str(self.nonce).encode('utf-8') +
            str(self.data).encode('utf-8') +
            str(self.previous_hash).encode('utf-8') +
            str(self.timestamp).encode('utf-8') +
            str(self.blockNo).encode('utf-8')
        )
        return h.hexdigest()

    def __str__(self):
```

```
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$ py pract1f.py
```

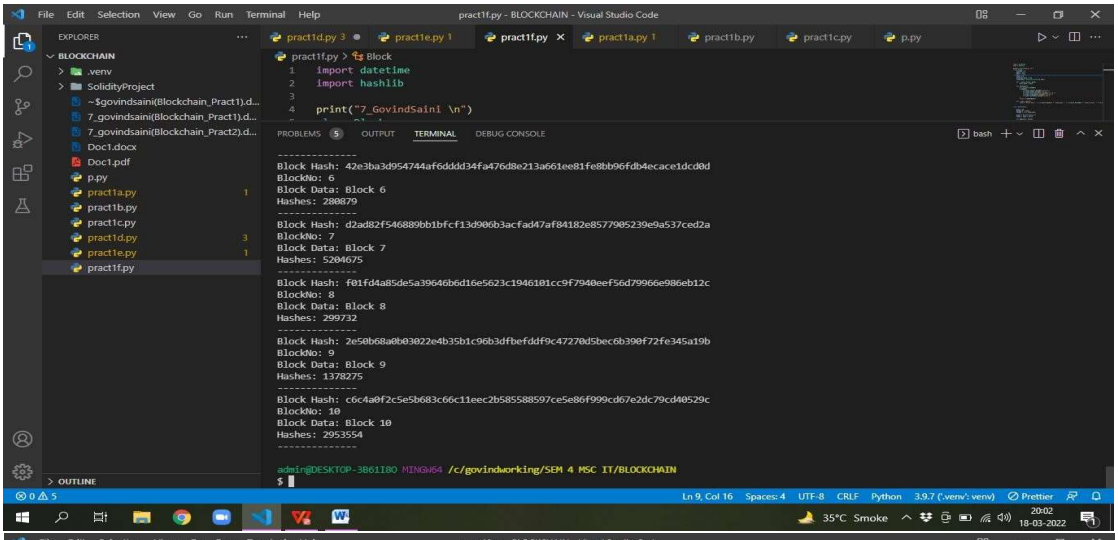


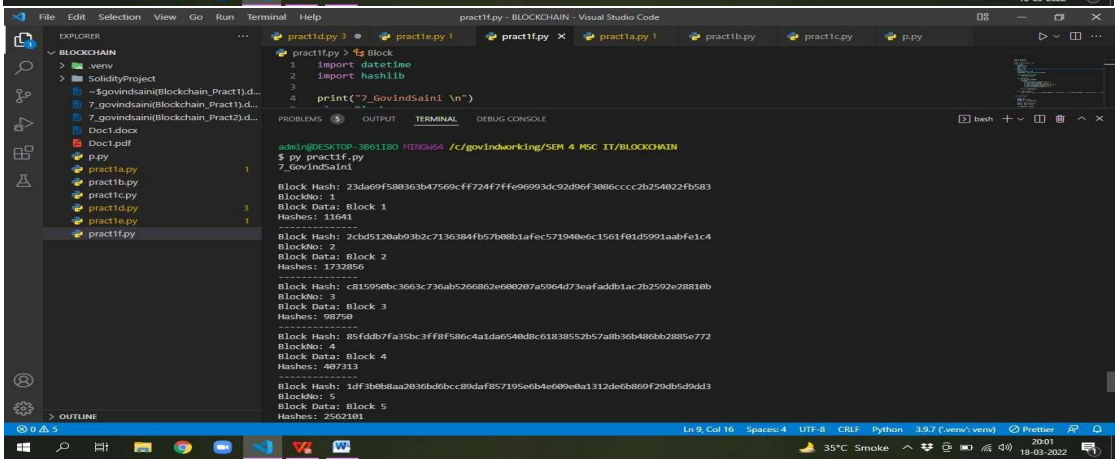
```
    def __str__(self):
        return "Block Hash: " + str(self.hash()) + "\nBlockNo: " + str(self.blockNo) + "\nBlock Data: " + str(

class Blockchain:

    diff = 20
    maxNonce = 2**32
    target = 2 ** (256-diff)

    block = Block("Genesis")
    dummy = head = block

    def add(self, block):

        block.previous_hash = self.block.hash()
        block.blockNo = self.block.blockNo + 1

        self.block.next = block
        self.block = self.block.next

    def mine(self, block):
        for n in range(self.maxNonce):
            if int(block.hash(), 16) <= self.target:
                self.add(block)
                print(block)
                break
            else:
```

```
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$ py pract1f.py
```


pract1f.py - BLOCKCHAIN - Visual Studio Code
self.block.next = block
self.block = self.block.next
def mine(self, block):
for n in range(self.maxNonce):
if int(block.hash(), 16) <= self.target:
self.add(block)
print(block)
break
else:
block.nonce += 1
blockchain = Blockchain()
for n in range(10):
blockchain.mine(Block("Block " + str(n+1)))
while blockchain.head != None:
print("7_GovindSaini \ n")
print(blockchain.head)
blockchain.head = blockchain.head.next
Hashes: 2429597

Output :

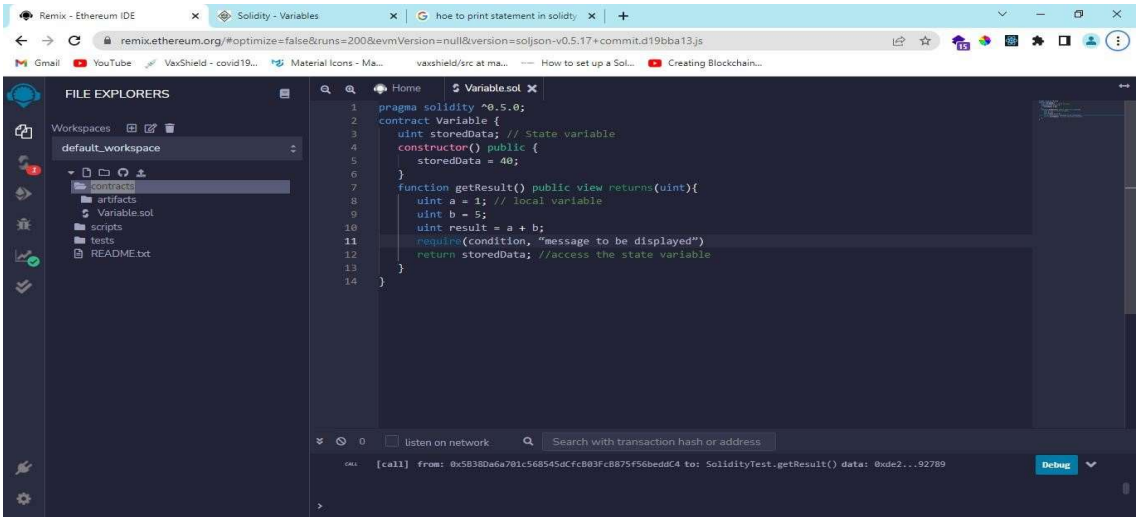

pract1f.py - BLOCKCHAIN - Visual Studio Code
import datetime
import hashlib
print("7_GovindSaini \n")
Block Hash: 42e3ba3d954744af6dddd34fa476d8e213a661ee81fe8bb96fdb4ecace1dcd0d
BlockNo: 6
Block Data: Block 6
Hashes: 280879
Block Hash: d2ad82f546889bb1bfcf13d906b3acfad47af84182e8577905239e9a537ced2a
BlockNo: 7
Block Data: Block 7
Hashes: 5204675
Block Hash: f01fd4a85de5a39646b6d16e5623c1946101cc9f7940eef56d79966e986eb12c
BlockNo: 8
Block Data: Block 8
Hashes: 299732
Block Hash: 2e50b68a0b03022e4b35b1c96b3dfbefddf9c47270d5bec6b390f72fe345a19b
BlockNo: 9
Block Data: Block 9
Hashes: 1378275
Block Hash: c6c4a0f2c5e5b683c66c11eec2b585588597ce5e86f999cd67e2dc79cd40529c
BlockNo: 10
Block Data: Block 10
Hashes: 2953554
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$


pract1f.py - BLOCKCHAIN - Visual Studio Code
import datetime
import hashlib
print("7_GovindSaini \n")
admin@DESKTOP-3B61I8O MINGW64 /c/govindworking/SEM 4 MSC IT/BLOCKCHAIN
$ py pract1f.py
7_GovindSaini
Block Hash: 23da69f580363b47569cff724f7ffe96993dc92d96f3086cccc2b254022fb583
BlockNo: 1
Block Data: Block 1
Hashes: 11641
Block Hash: 2cbd5120ab93b2c7136384fb57b08b1afec571940e6c1561f01d5991aabfe1c4
BlockNo: 2
Block Data: Block 2
Hashes: 1732856
Block Hash: c815950bc3663c736ab5266862e600207a5964d73eafaddb1ac2b2592e28810b
BlockNo: 3
Block Data: Block 3
Hashes: 98750
Block Hash: 85fddb7fa35bc3ff8f586c4a1da6540d8c61838552b57a8b36b486bb2885e772
BlockNo: 4
Block Data: Block 4
Hashes: 407313
Block Hash: 1df3b0b8aa2036bd6bcc89daf857195e6b4e609e0a1312de6b869f29db5d9dd3
BlockNo: 5
Block Data: Block 5
Hashes: 2562101

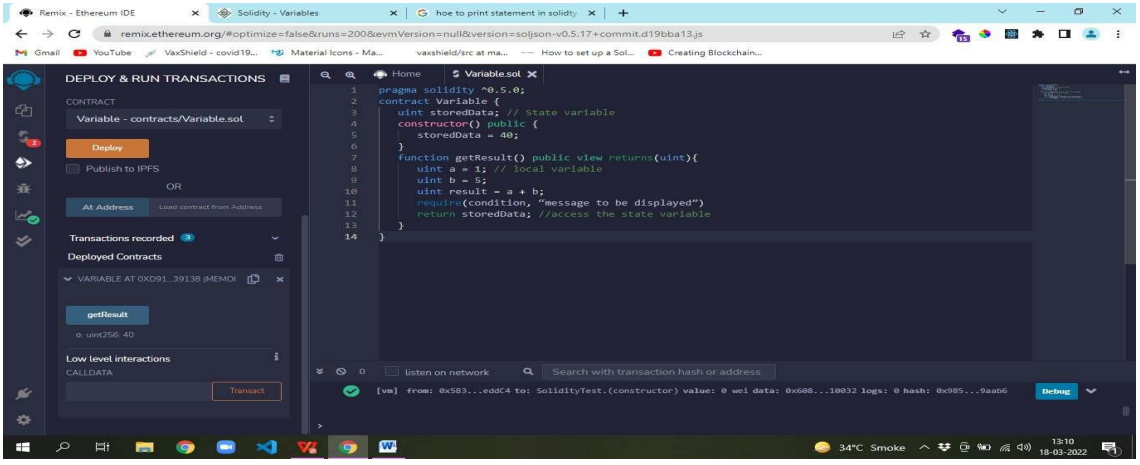
Practical No : 3

Aim : Implement and demonstrate the use of the following in Solidity

3a) Variable, Operators, Loops, Decision Making, Strings, Arrays, Enums, Structs, Mappings, Conversions, Ether Units, Special Variables. Variable Code :

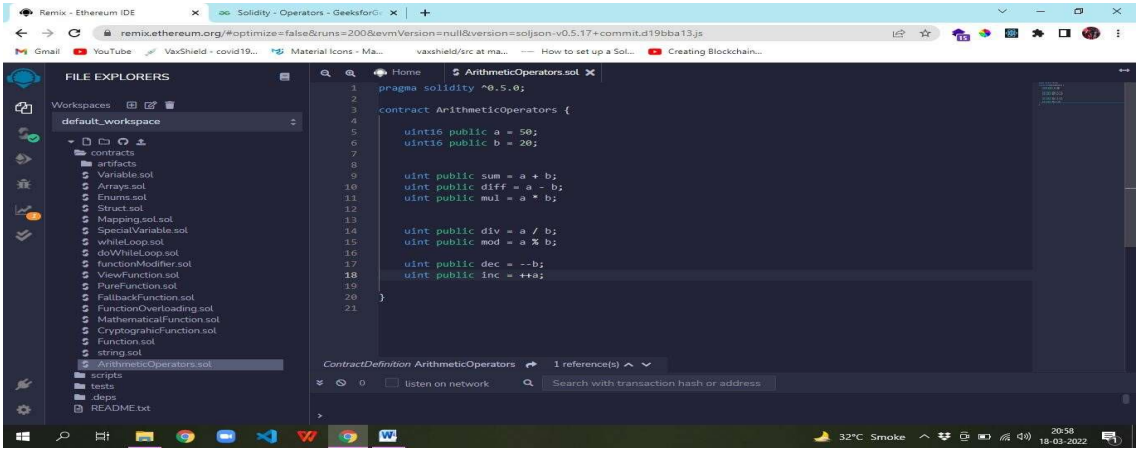


Output :

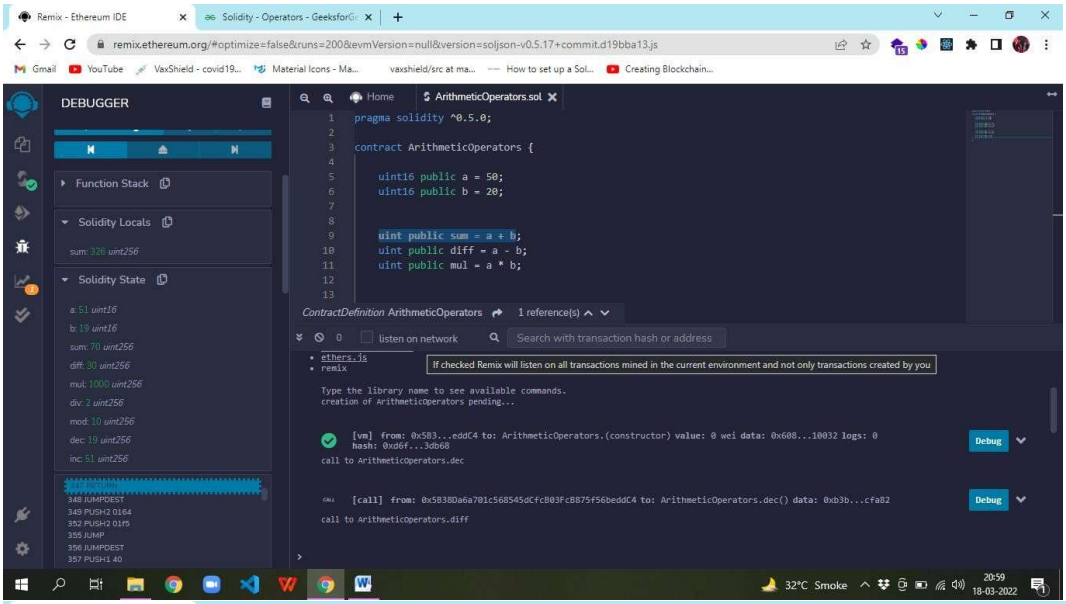


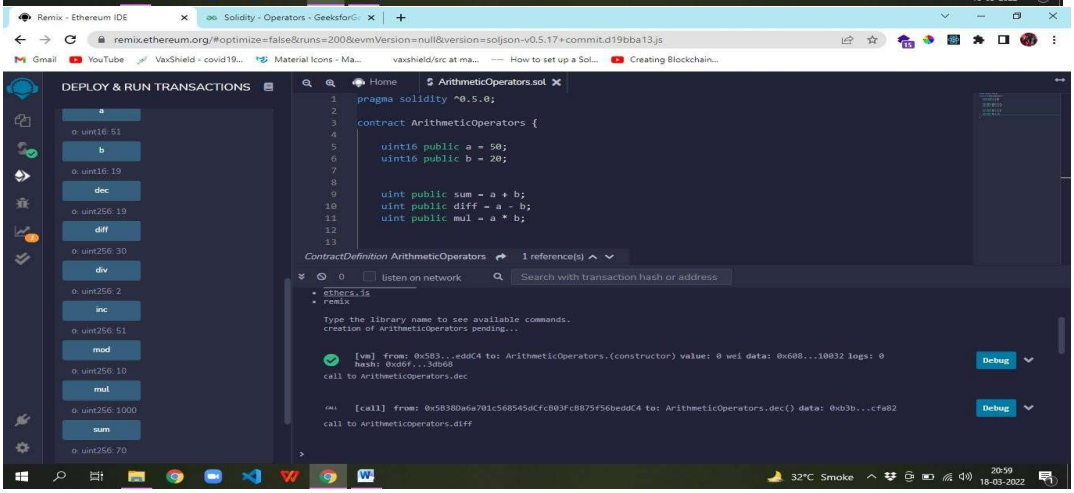
Operators : Arthmetic

Operator Code:

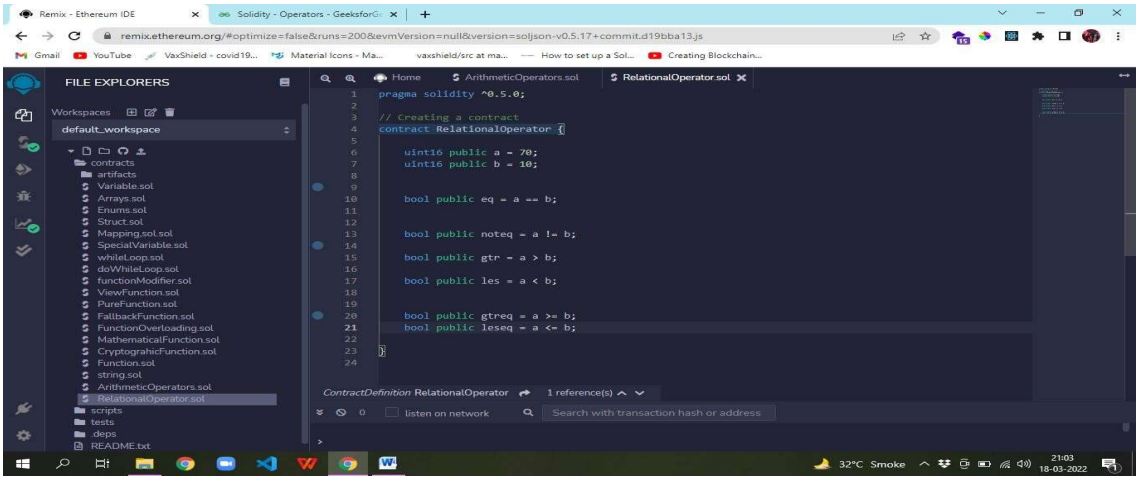

Remix - Ethereum IDE
Solidity - Operators - GeeksforG
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.17+commit.d19bba13.js
FILE EXPLORERS
Workspaces
default_workspace
contracts
artifacts
Variable.sol
Arrays.sol
Enums.sol
Struct.sol
Mapping.sol.sol
SpecialVariable.sol
whileLoop.sol
doWhileLoop.sol
functionModifier.sol
ViewFunction.sol
PureFunction.sol
FallbackFunction.sol
FunctionOverloading.sol
MathematicalFunction.sol
CryptographicFunction.sol
Function.sol
string.sol
ArithmeticOperators.sol
scripts
tests
.deps
README.txt
Home
ArithmeticOperators.sol
pragma solidity ^0.5.0;
contract ArithmeticOperators {
uint16 public a = 50;
uint16 public b = 20;
uint public sum = a + b;
uint public diff = a - b;
uint public mul = a * b;
uint public div = a / b;
uint public mod = a % b;
uint public dec = --b;
uint public inc = ++a;
}
ContractDefinition ArithmeticOperators
1 reference(s)
listen on network
Search with transaction hash or address
32°C Smoke
20:58
18-03-2022

Output:

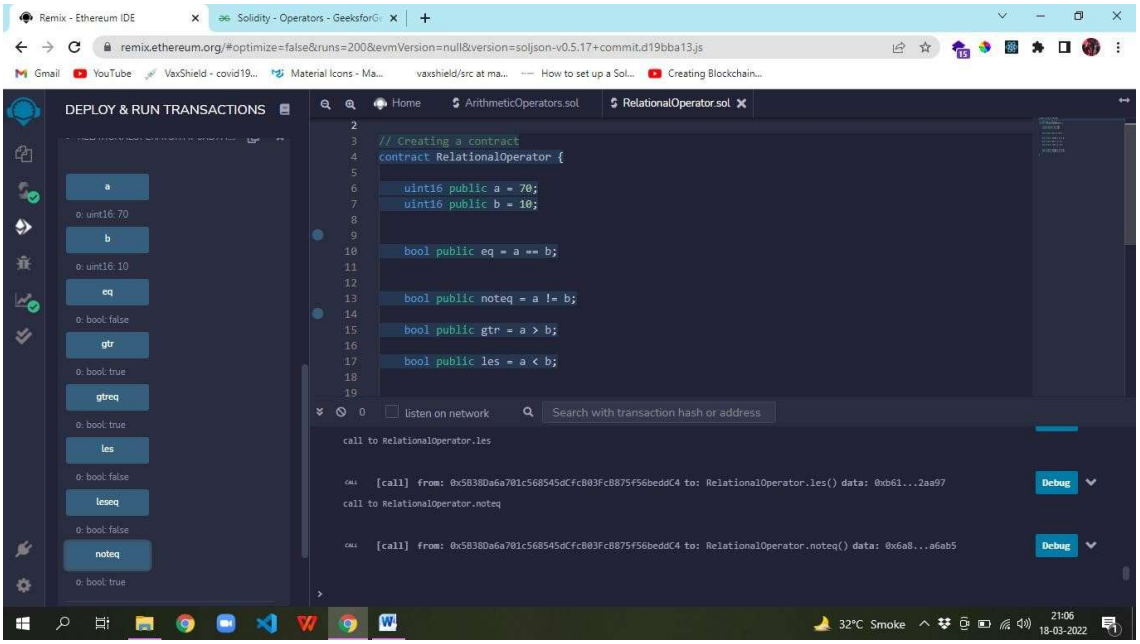

Remix - Ethereum IDE
Solidity - Operators - GeeksforG
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.17+commit.d19bba13.js
DEBUGGER
Function Stack
Solidity Locals
sum: 326 uint256
Solidity State
a: 51 uint16
b: 19 uint16
sum: 70 uint256
diff: 30 uint256
mul: 1000 uint256
div: 2 uint256
mod: 10 uint256
dec: 19 uint256
inc: 51 uint256
348 JUMPDEST
349 PUSH2 0164
352 PUSH2 01f5
355 JUMP
356 JUMPDEST
357 PUSH1 40
Home
ArithmeticOperators.sol
pragma solidity ^0.5.0;
contract ArithmeticOperators {
uint16 public a = 50;
uint16 public b = 20;
uint public sum = a + b;
uint public diff = a - b;
uint public mul = a * b;
ContractDefinition ArithmeticOperators
1 reference(s)
listen on network
Search with transaction hash or address
ethers.js
remix
If checked Remix will listen on all transactions mined in the current environment and not only transactions created by you
Type the library name to see available commands.
creation of ArithmeticOperators pending...
[vm] from: 0x583...eddC4 to: ArithmeticOperators.(constructor) value: 0 wei data: 0x608...10032 logs: 0
hash: 0xd6f...3db68
Debug
call to ArithmeticOperators.dec
[call] from: 0x5838Da6a701c568545dCfcB03FcB875f56beddC4 to: ArithmeticOperators.dec() data: 0xb3b...cfa82
Debug
call to ArithmeticOperators.diff
32°C Smoke
20:59
18-03-2022

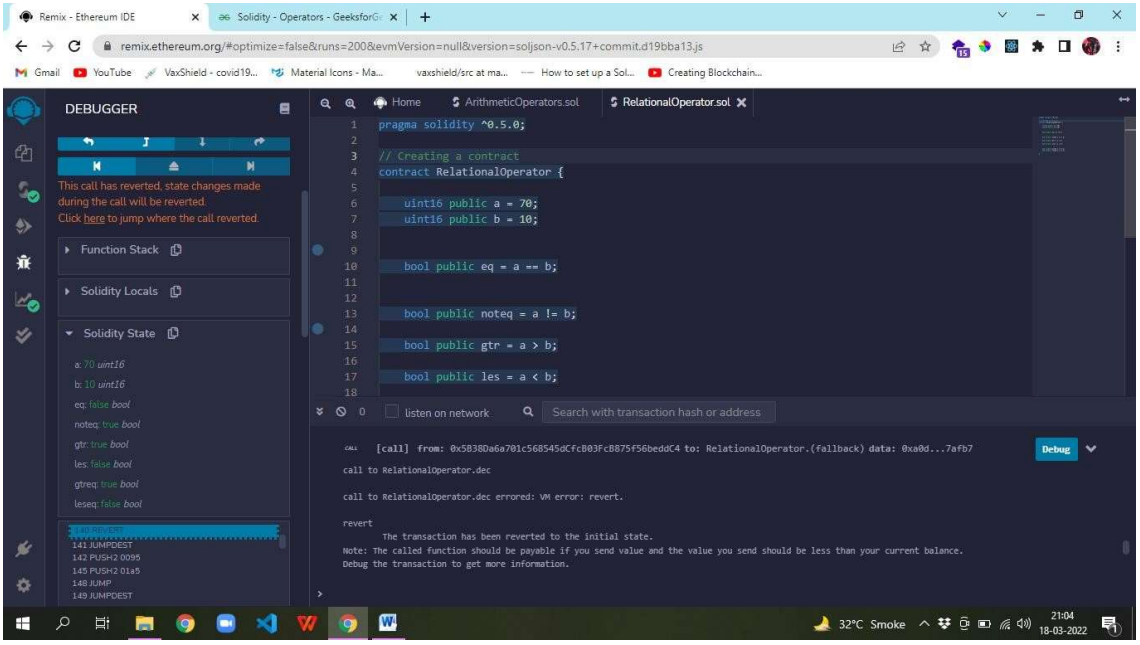

Remix - Ethereum IDE
Solidity - Operators - GeeksforG
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.17+commit.d19bba13.js
DEPLOY & RUN TRANSACTIONS
a
o: uint16: 51
b
o: uint16: 19
dec
o: uint256: 19
diff
o: uint256: 30
div
o: uint256: 2
inc
o: uint256: 51
mod
o: uint256: 10
mul
o: uint256: 1000
sum
o: uint256: 70
Home
ArithmeticOperators.sol
pragma solidity ^0.5.0;
contract ArithmeticOperators {
uint16 public a = 50;
uint16 public b = 20;
uint public sum = a + b;
uint public diff = a - b;
uint public mul = a * b;
ContractDefinition ArithmeticOperators
1 reference(s)
listen on network
Search with transaction hash or address
ethers.js
remix
Type the library name to see available commands.
creation of ArithmeticOperators pending...
[vm] from: 0x583...eddC4 to: ArithmeticOperators.(constructor) value: 0 wei data: 0x608...10032 logs: 0
hash: 0xd6f...3db68
Debug
call to ArithmeticOperators.dec
[call] from: 0x5838Da6a701c568545dCfcB03FcB875f56beddC4 to: ArithmeticOperators.dec() data: 0xb3b...cfa82
Debug
call to ArithmeticOperators.diff
32°C Smoke
20:59
18-03-2022

Relational Operator Code:

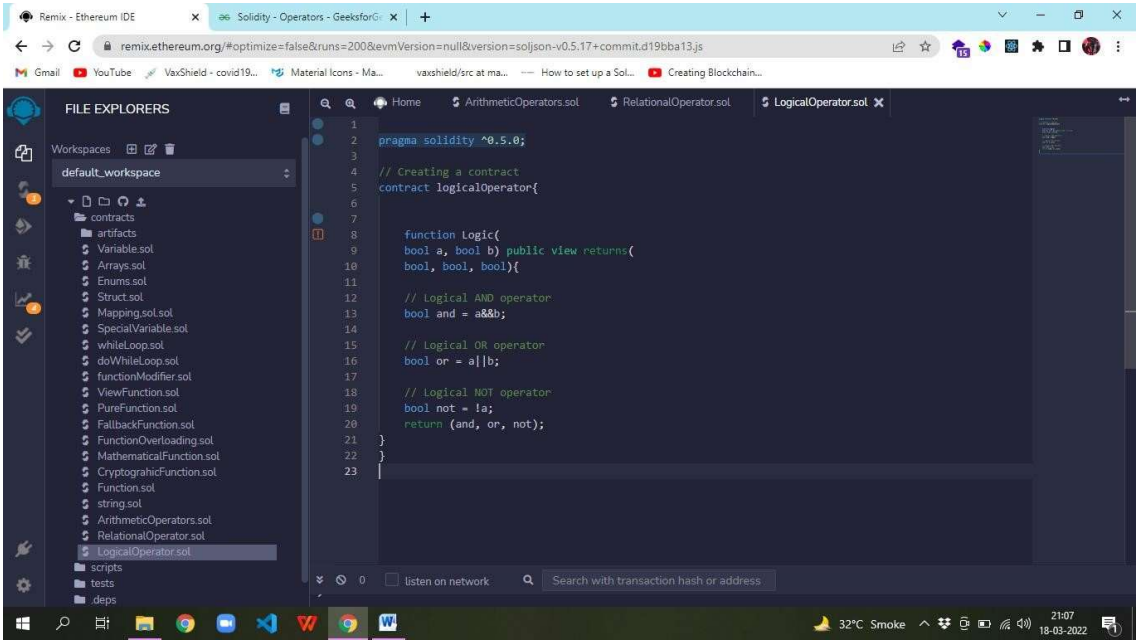


Output :

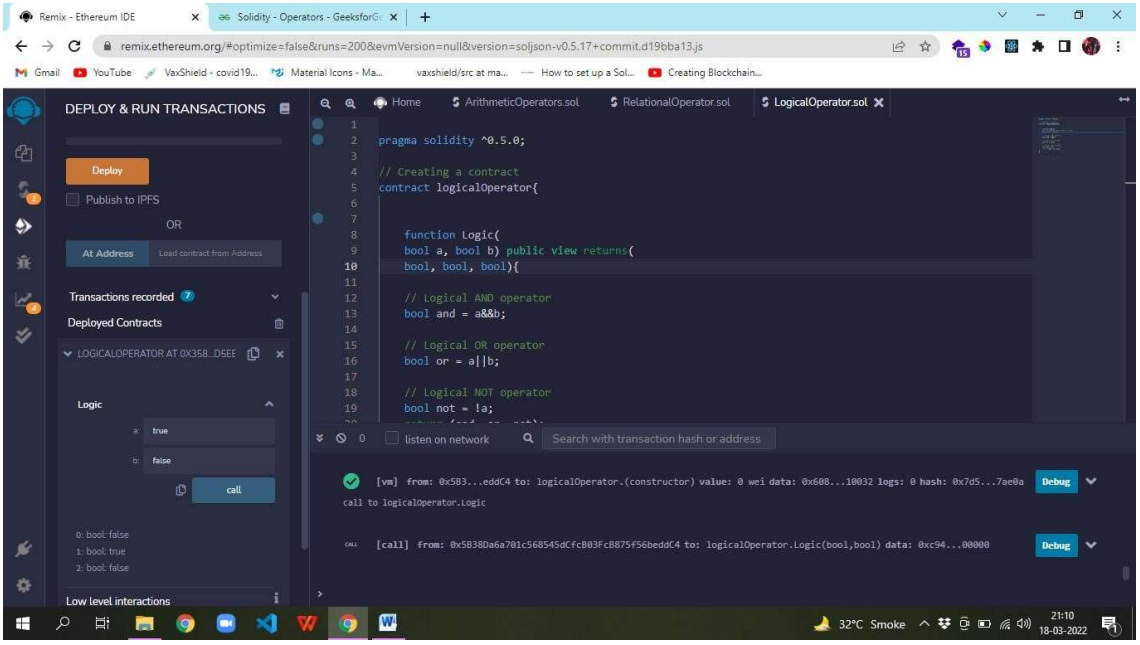




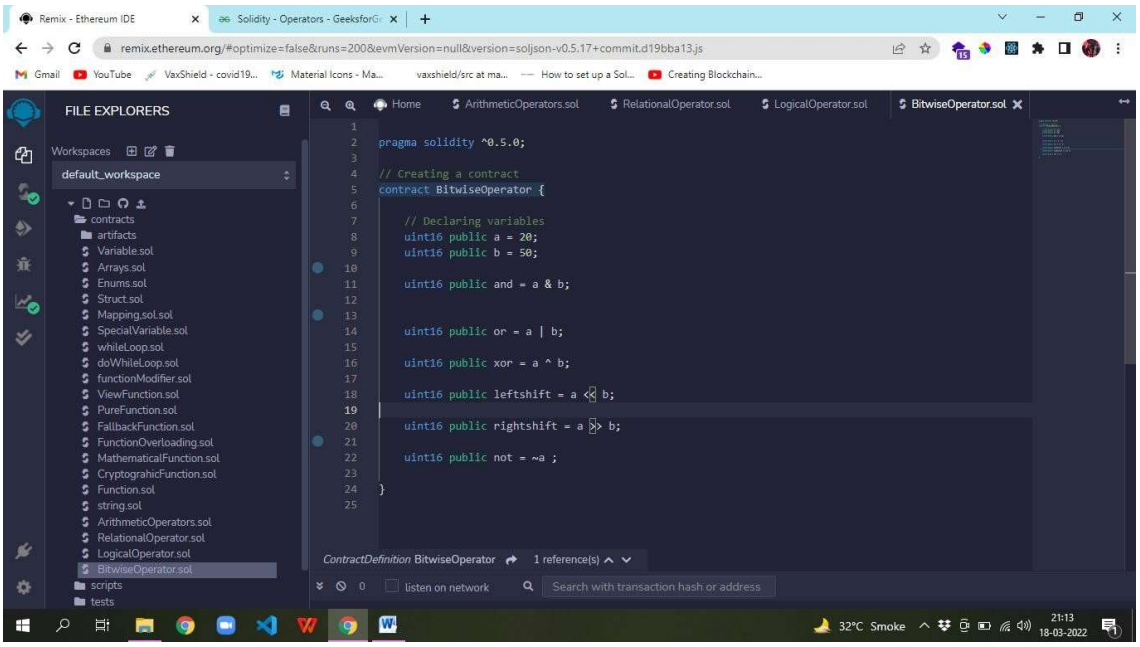
Logical Operator Code :



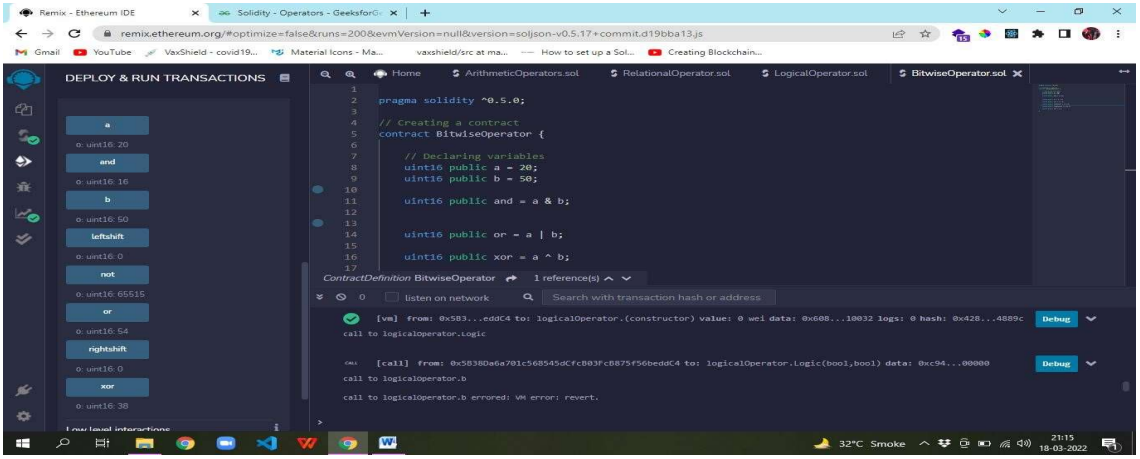
Output:

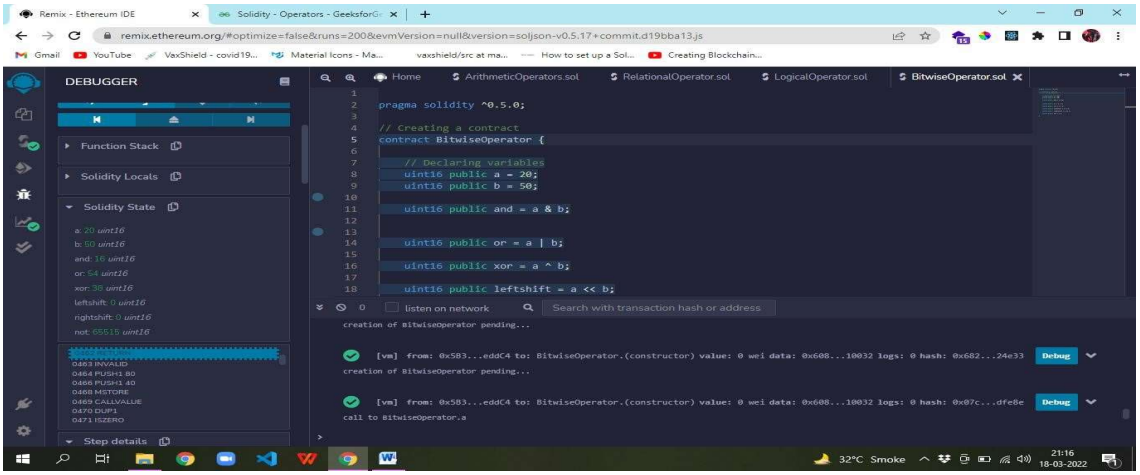


Bitwise Operator Code:

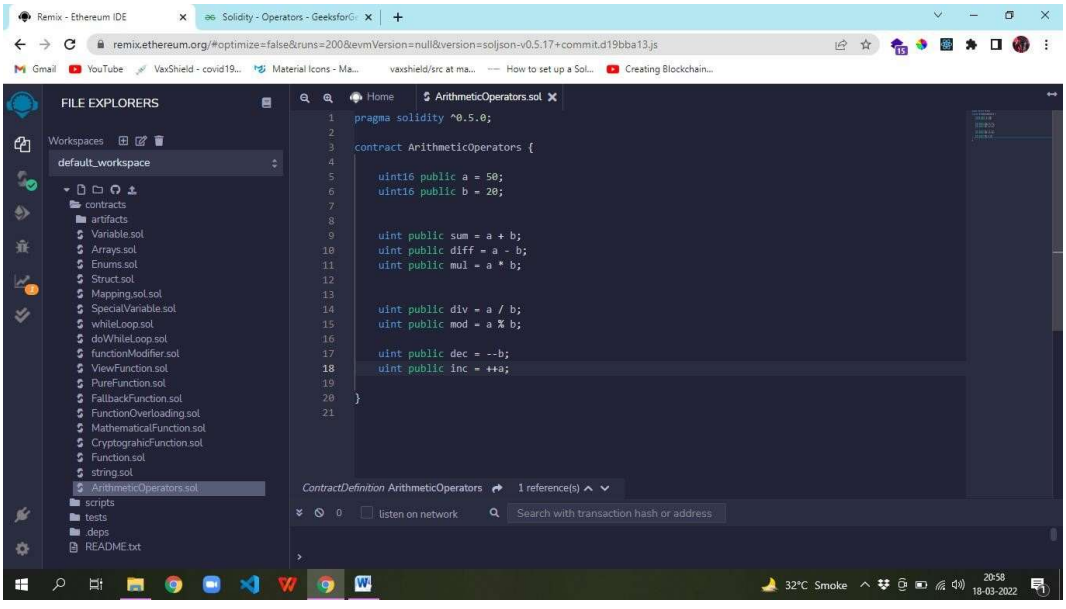


Output :

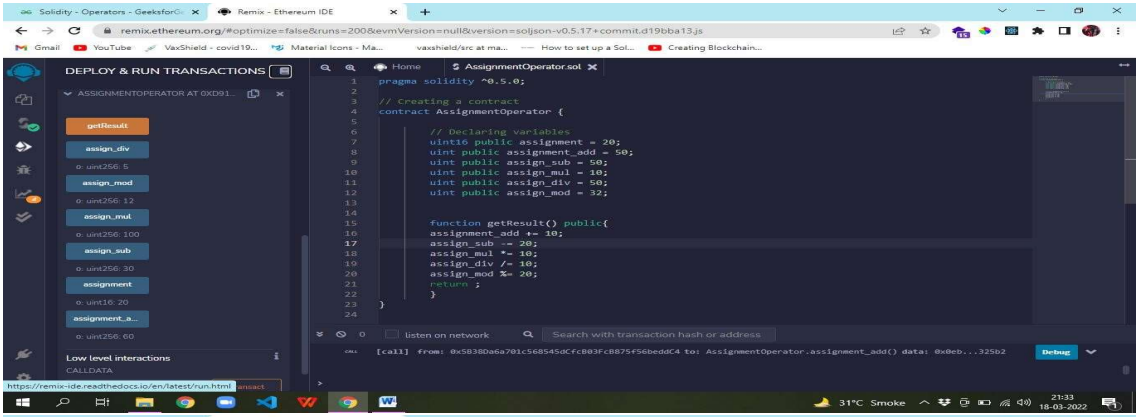


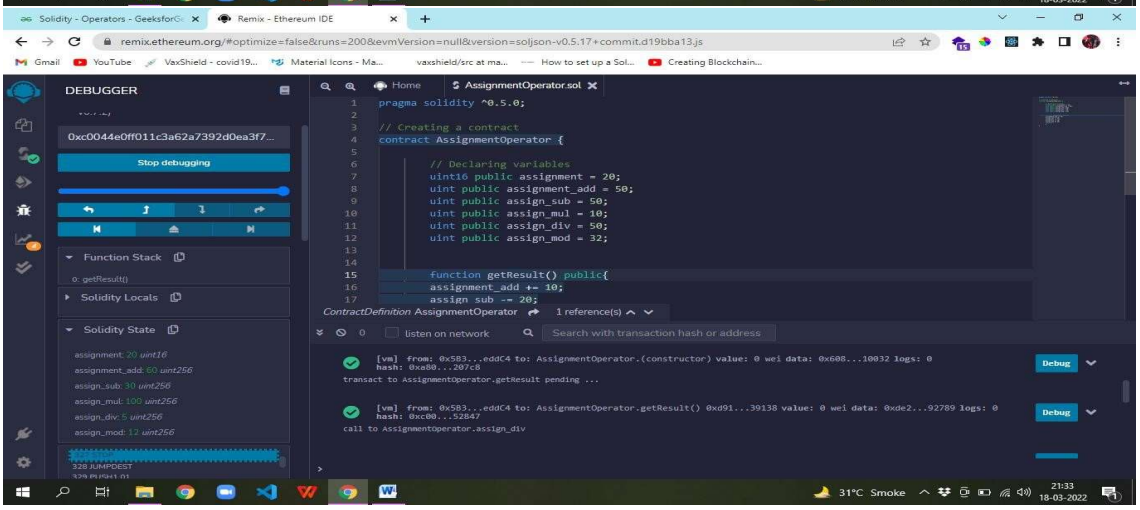


Assignment Operator Code :

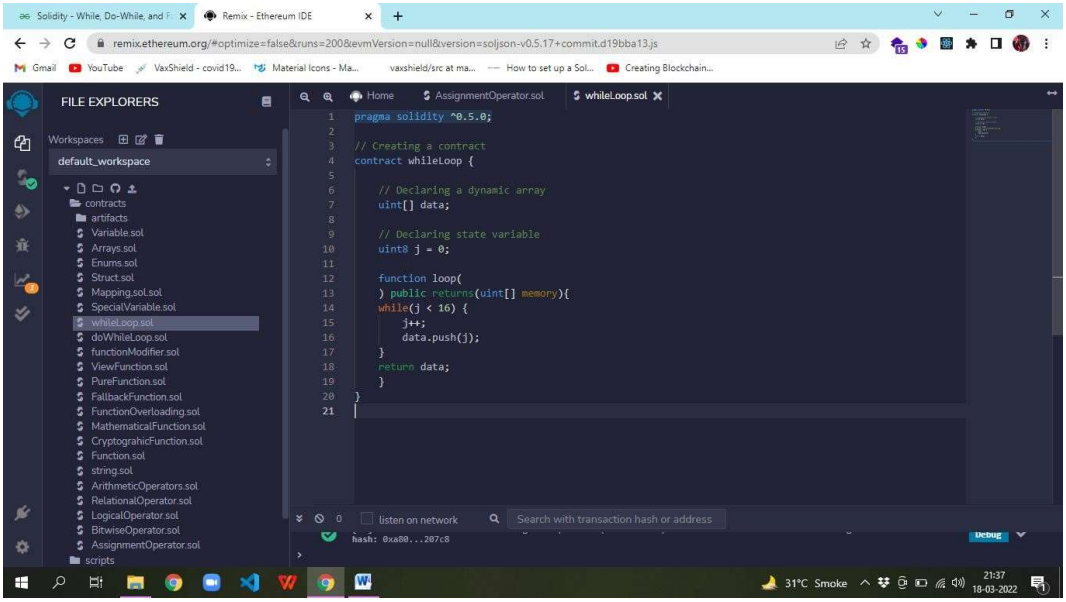


Ouput :

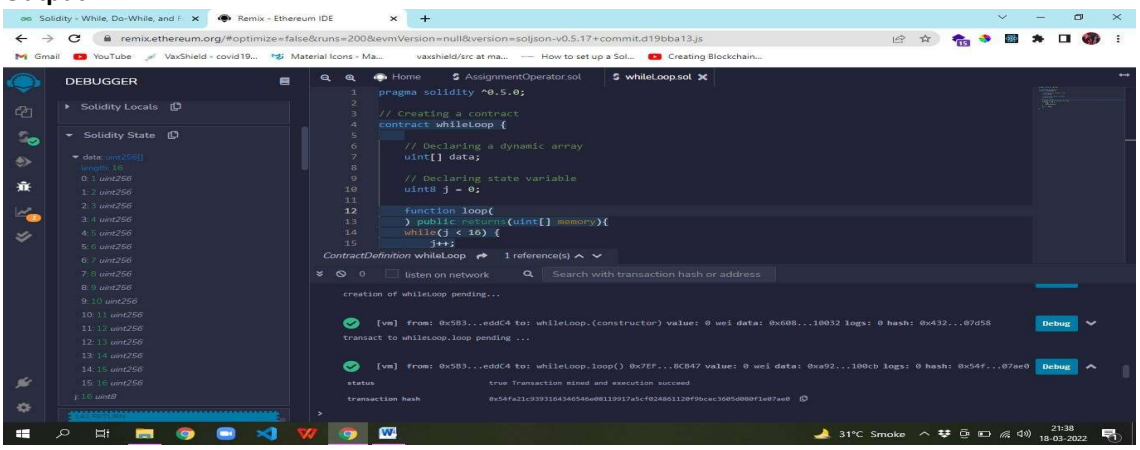


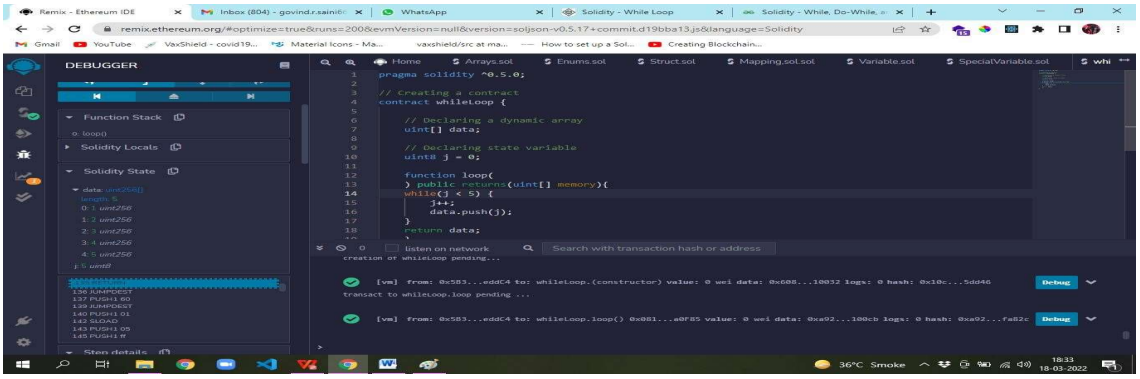


Loops whileLoop code:

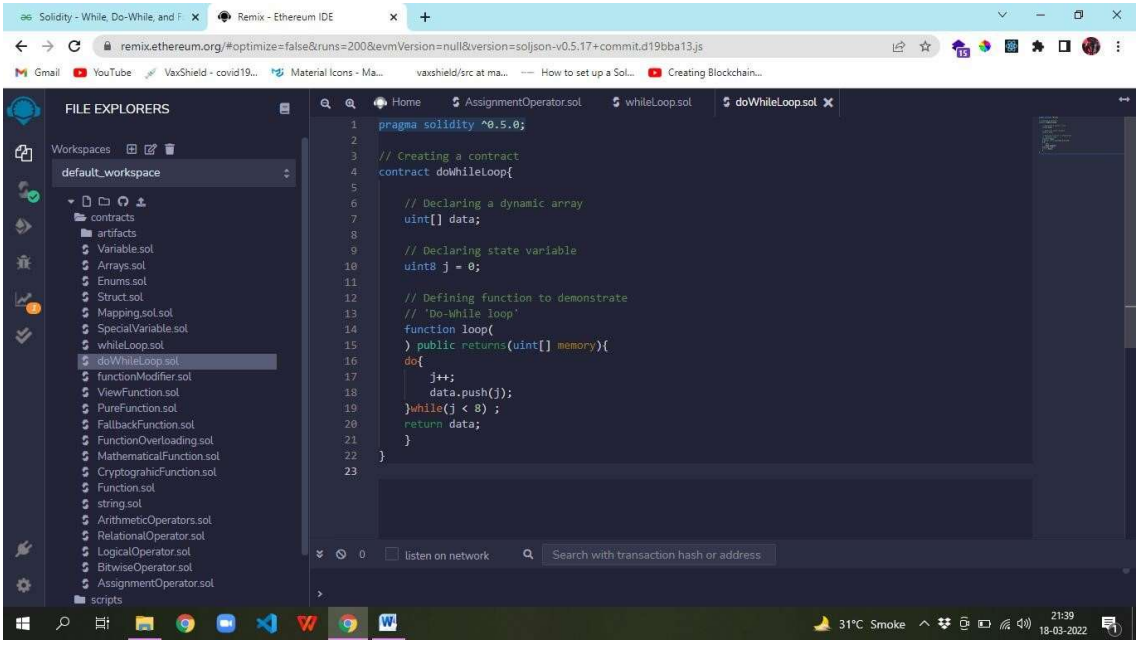


Output

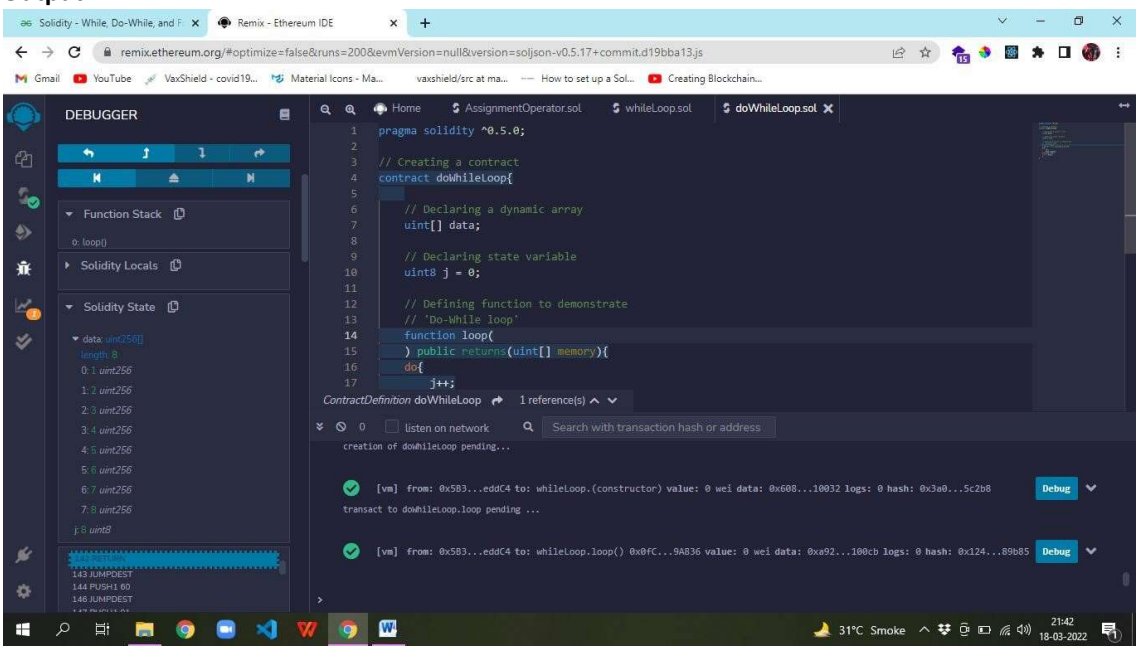




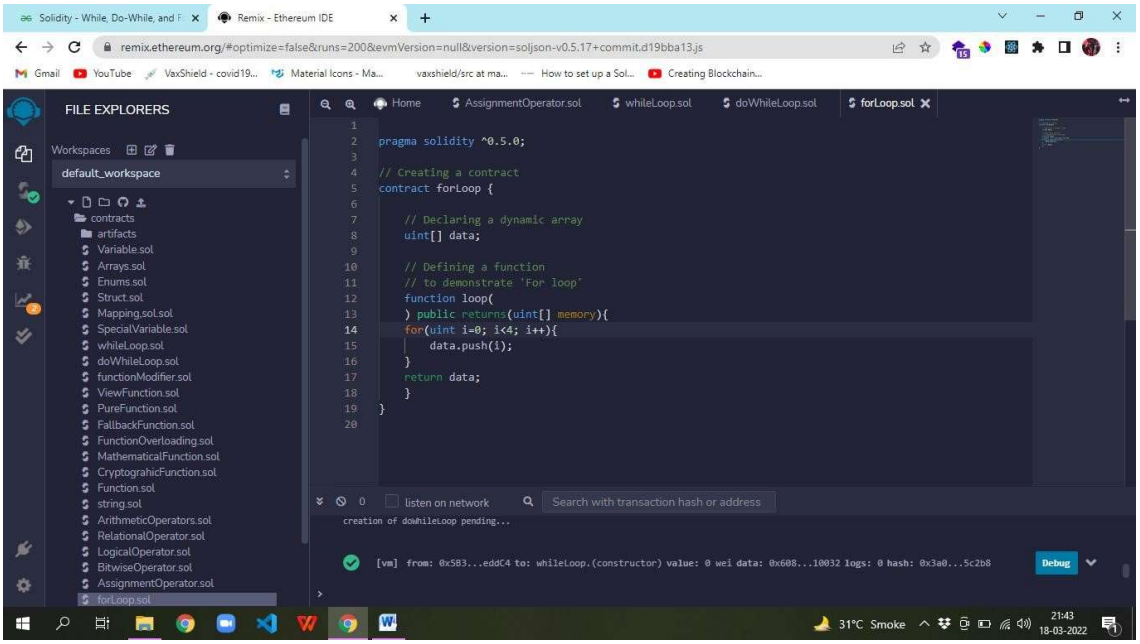
Dowhile loop Code:



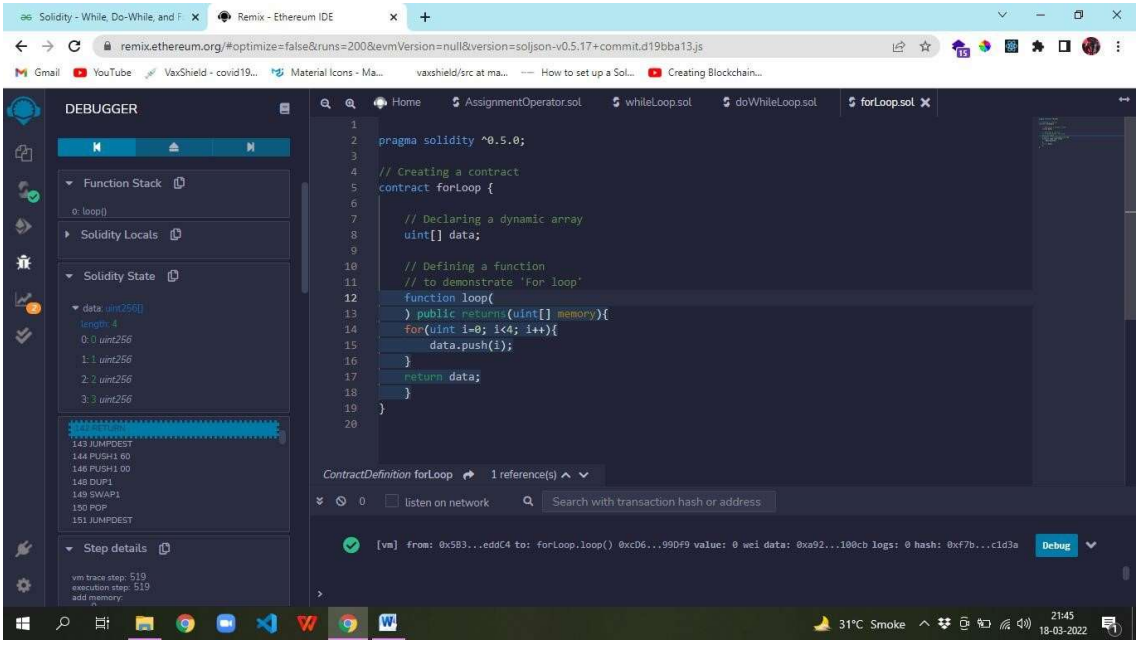
Output :



Forloop Code :

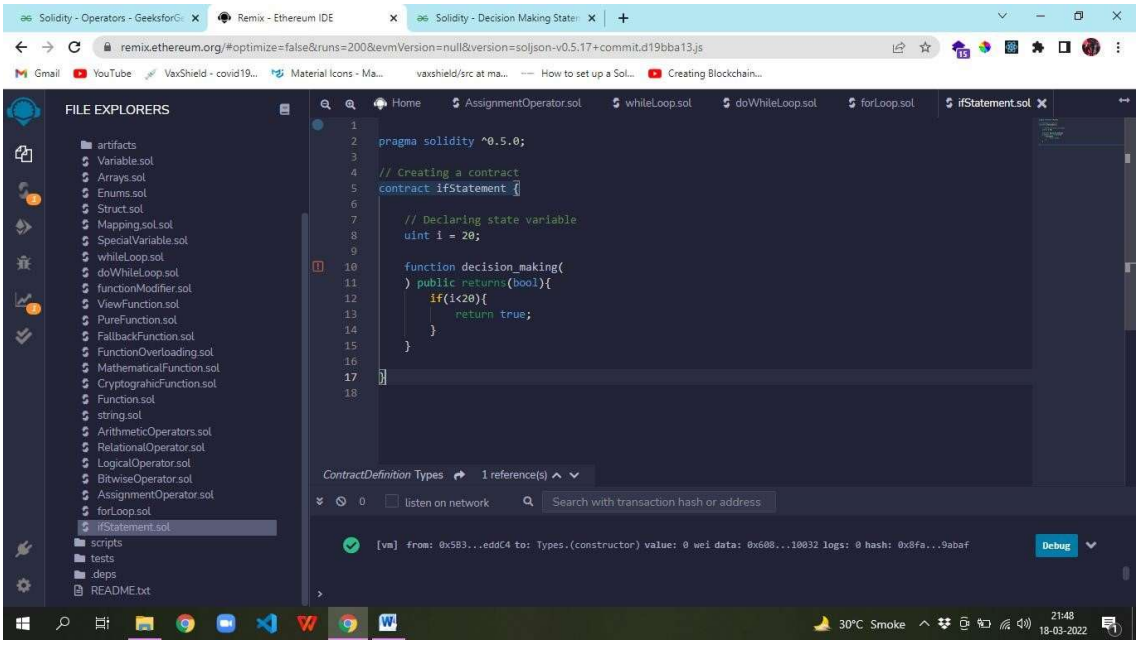


Output :

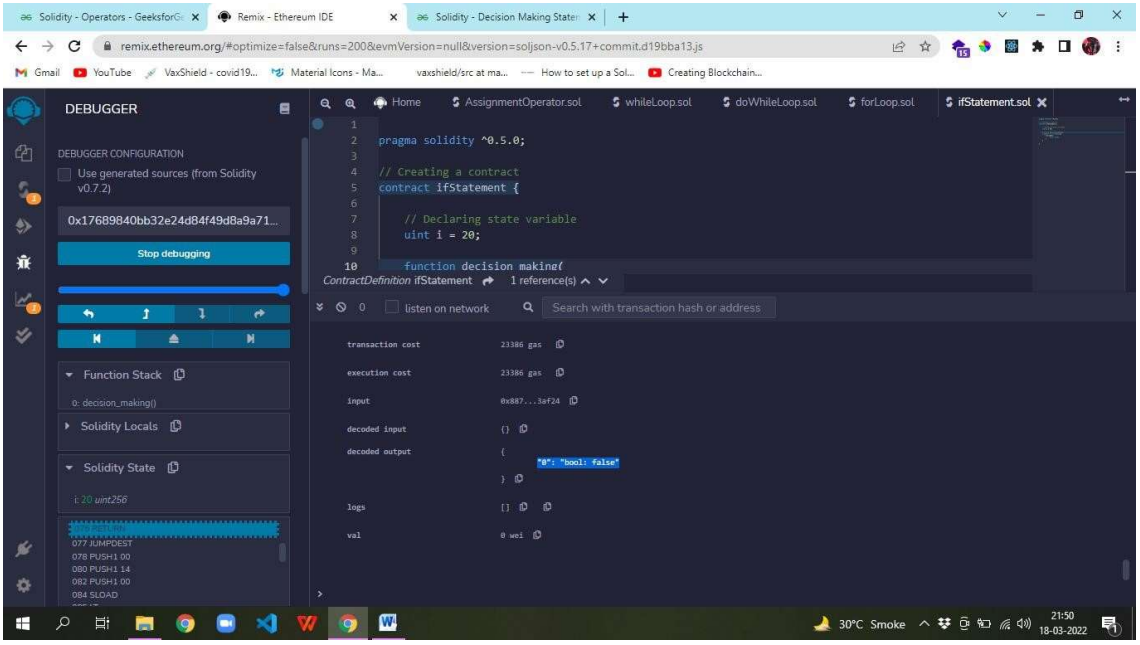


Decision making

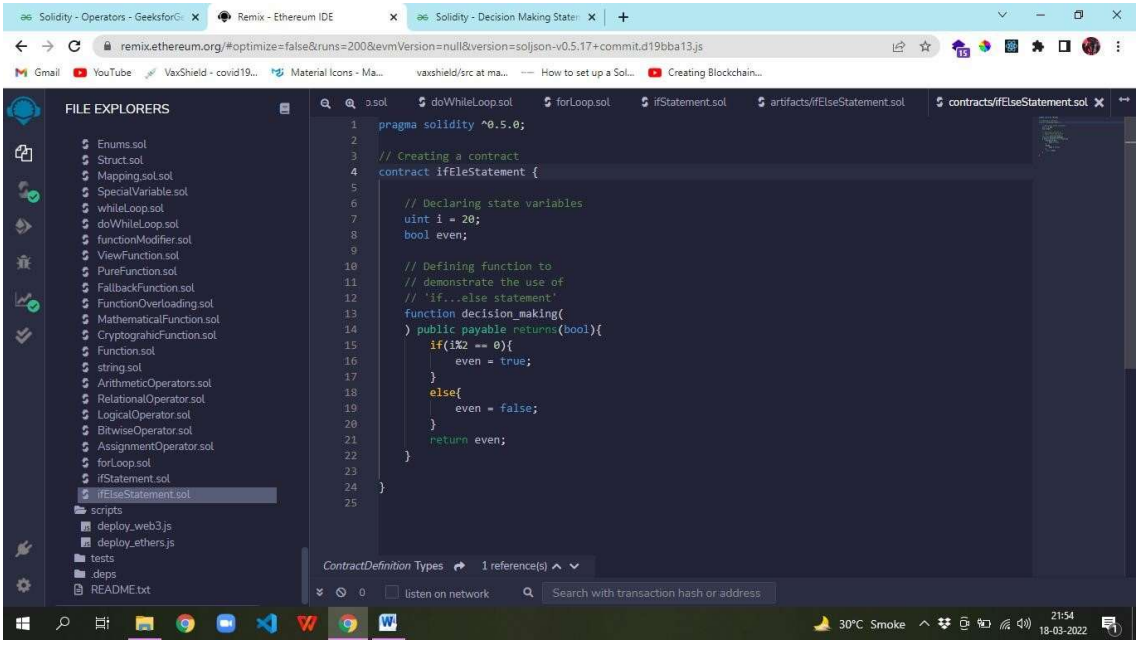
If statement Code:



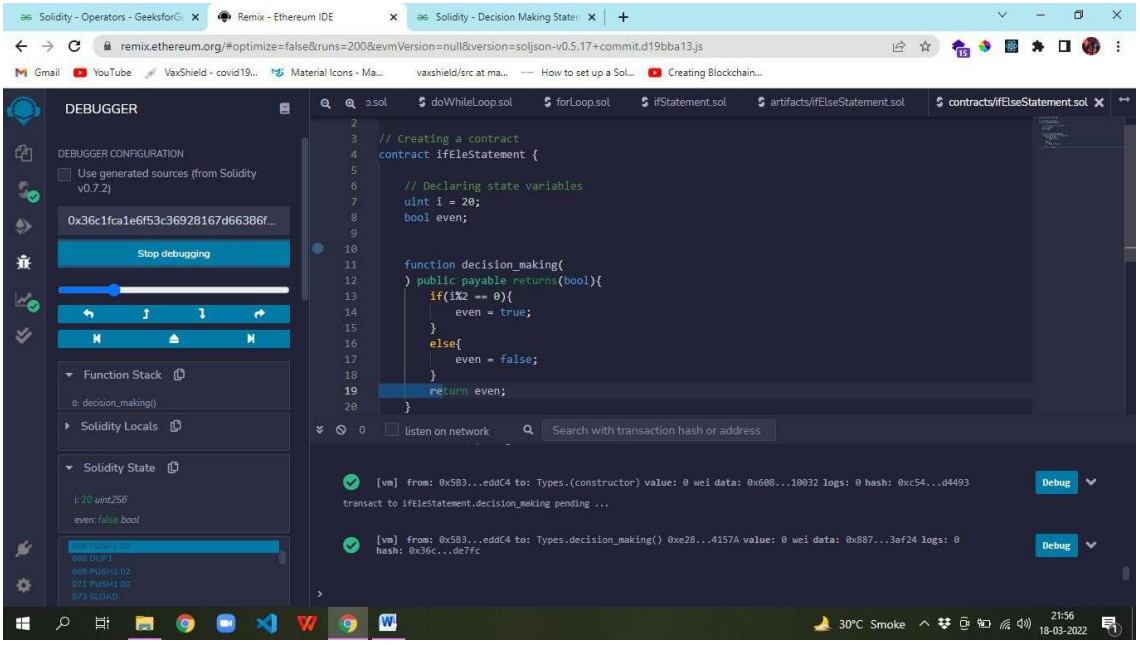
Code :



If...else statement Code :

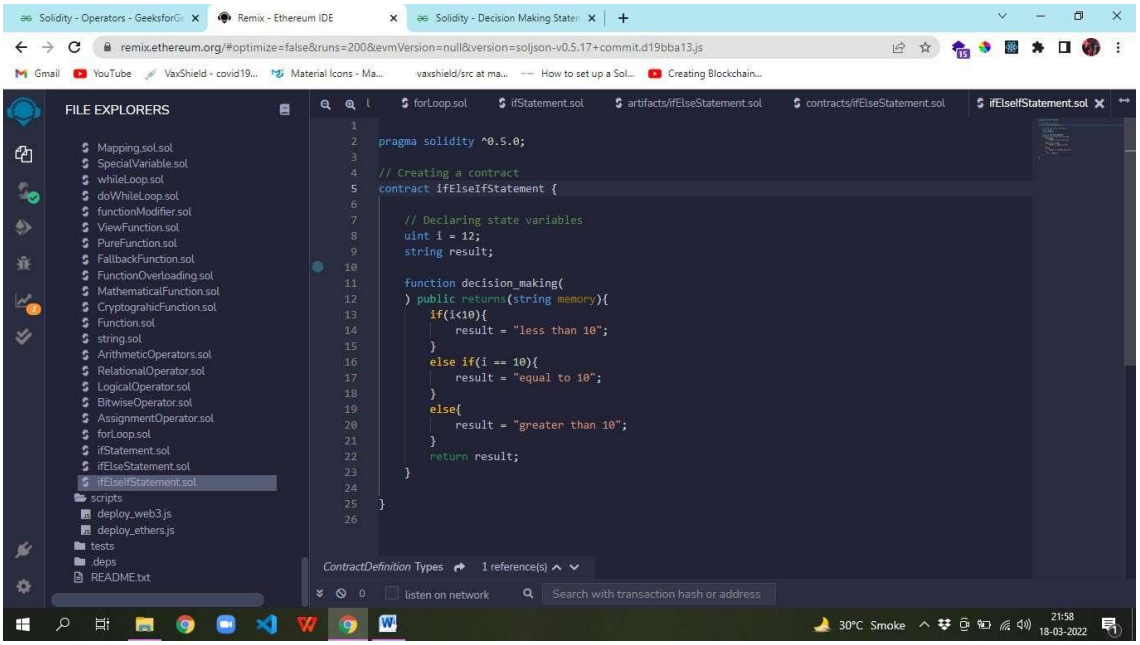


Output :

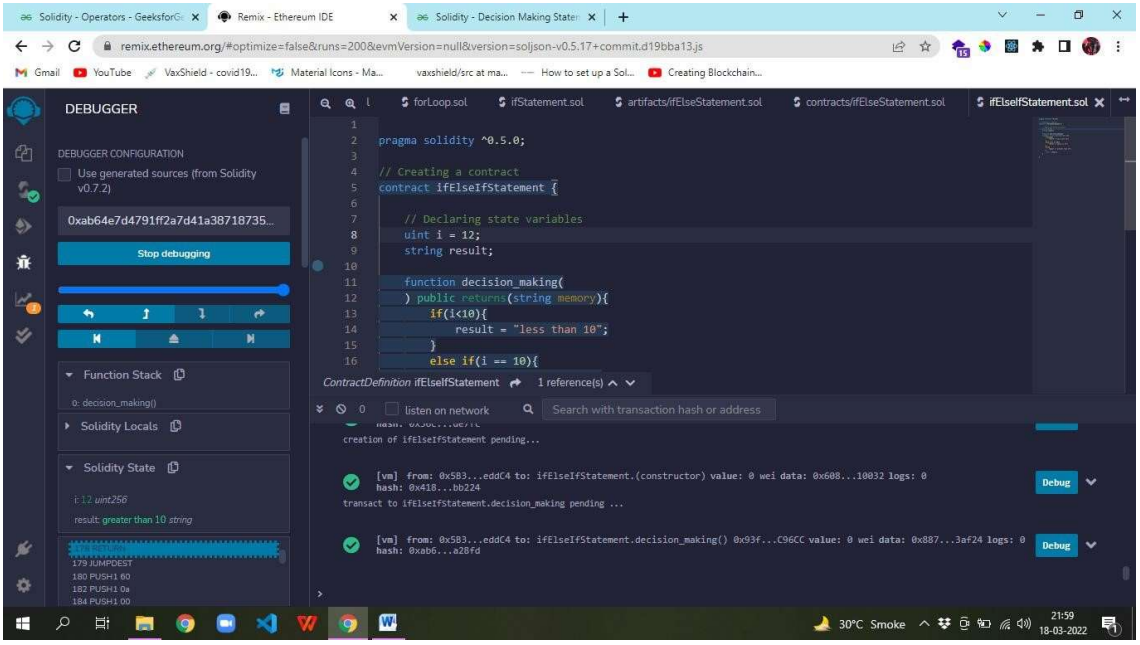


If...else if...else statement

Code :

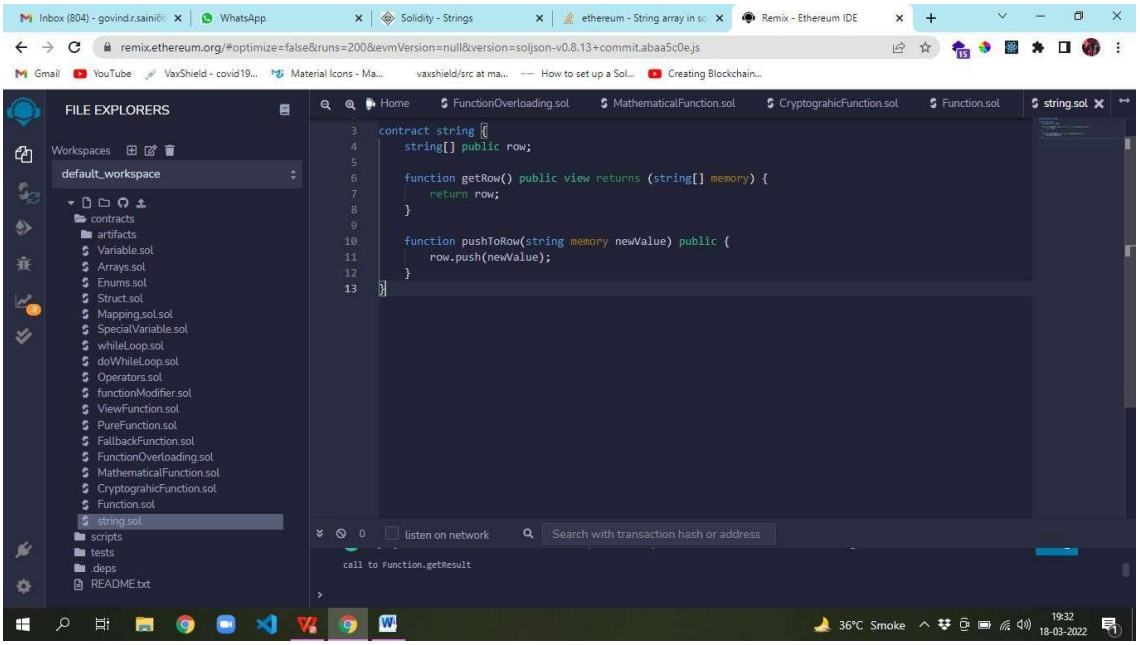


Output:

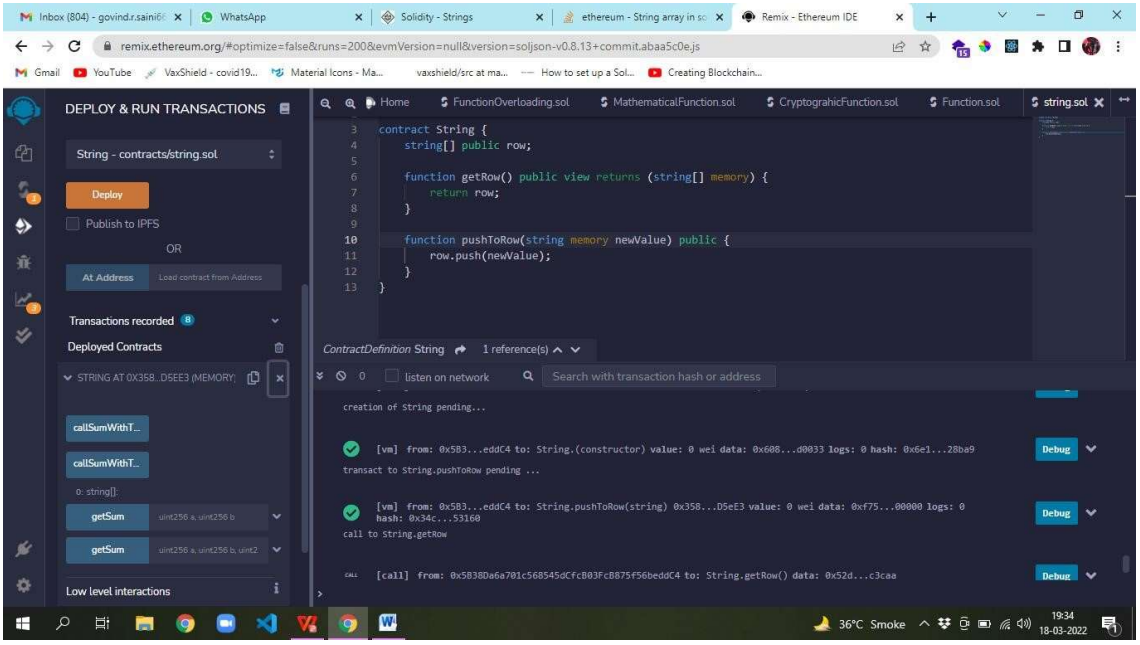


String

Code:

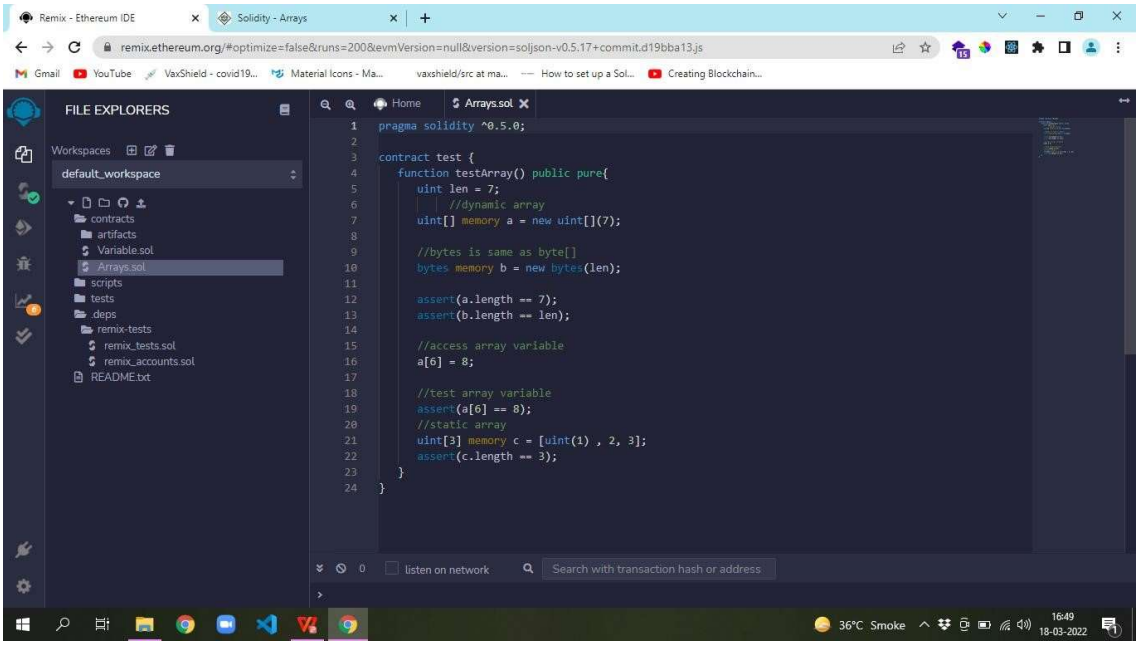


Output :

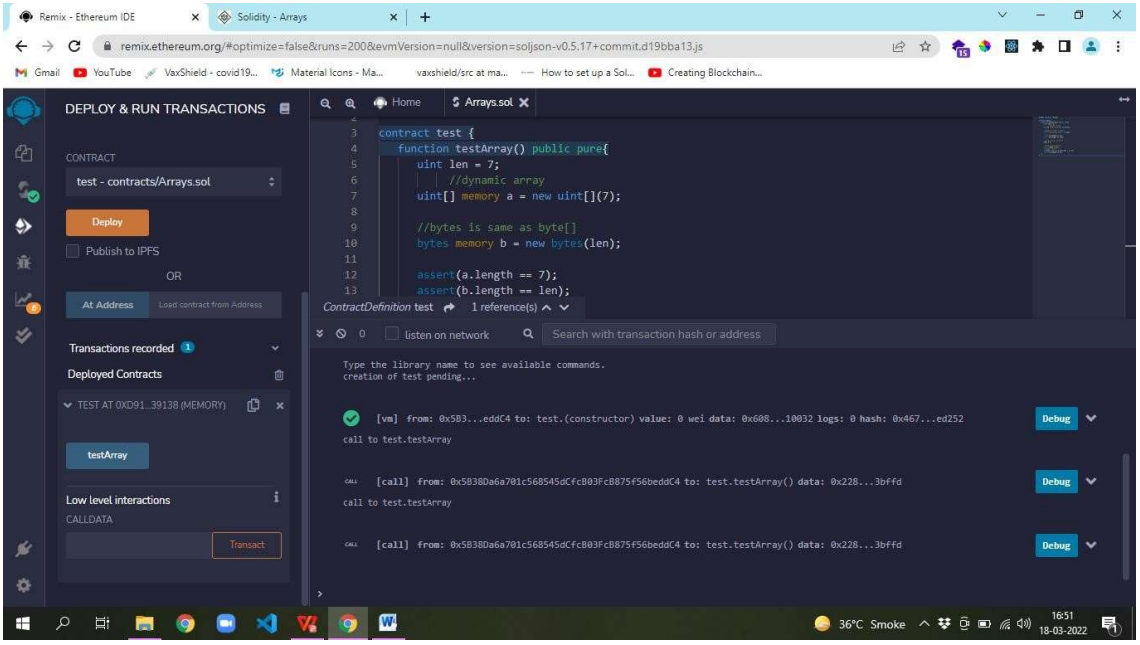


Array

Code :

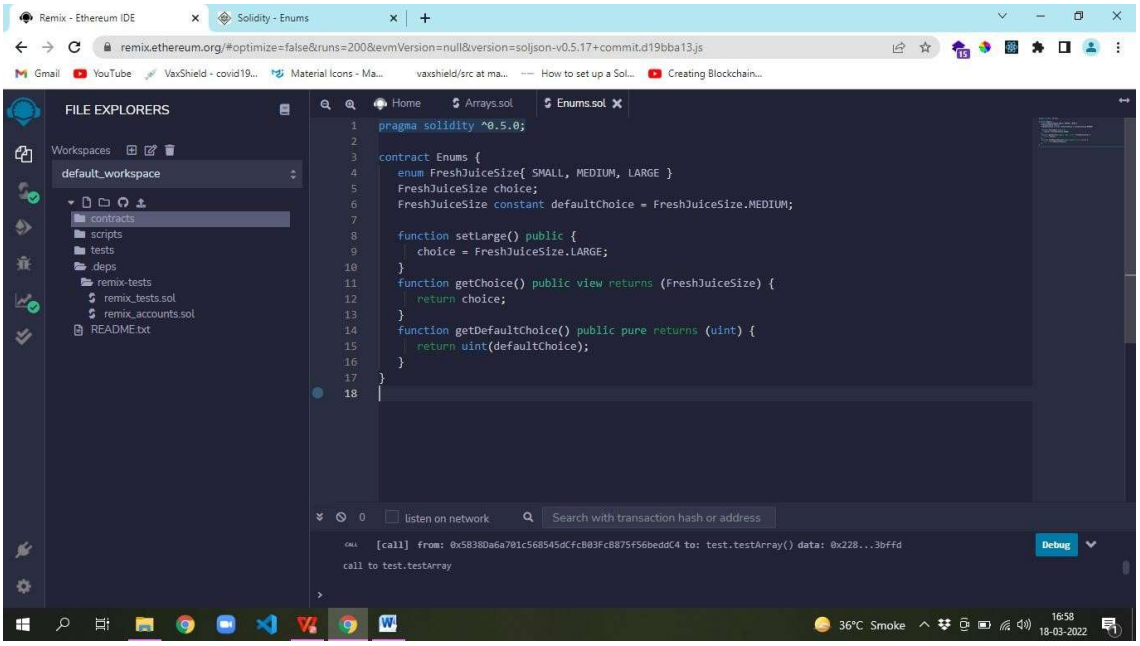


Output:

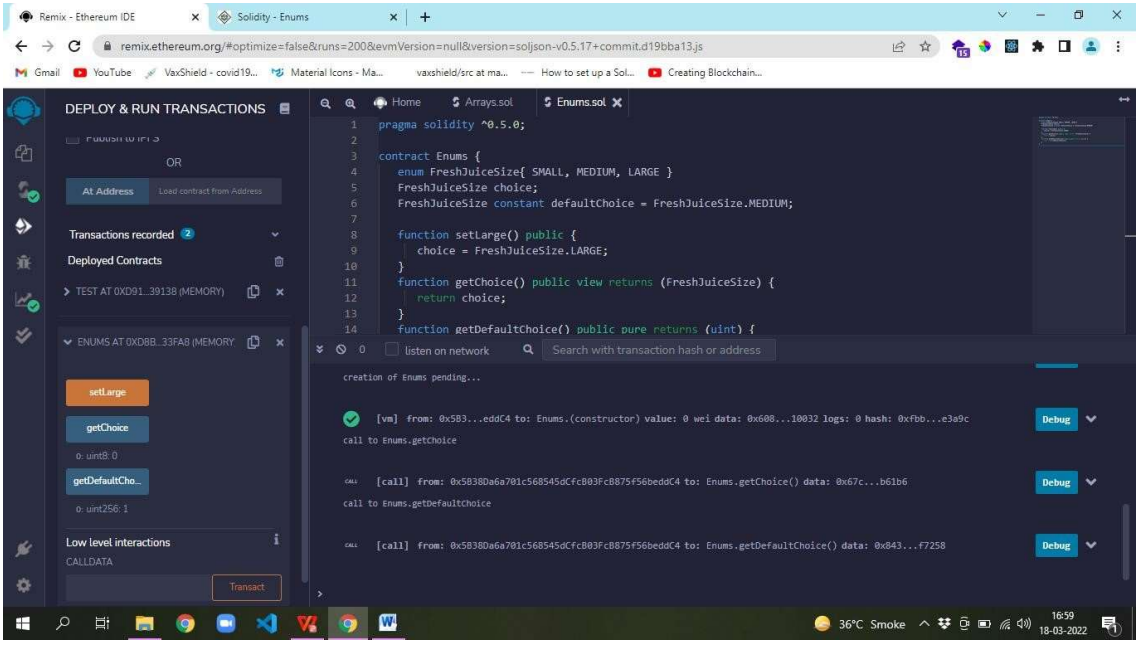


Enums

Code :

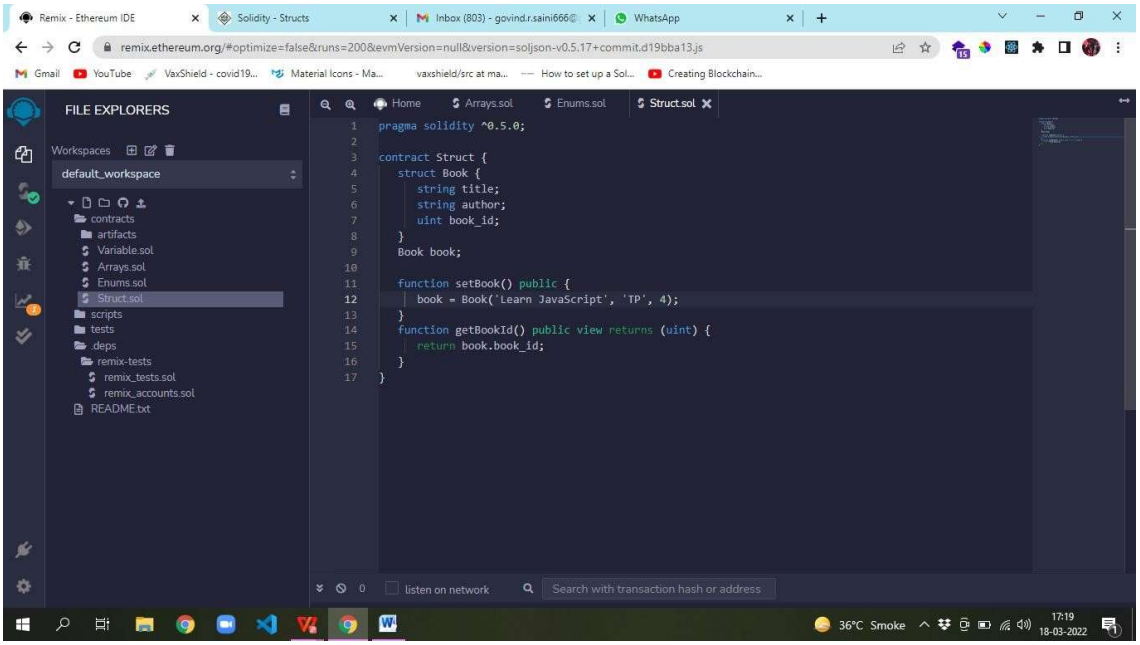


Output :

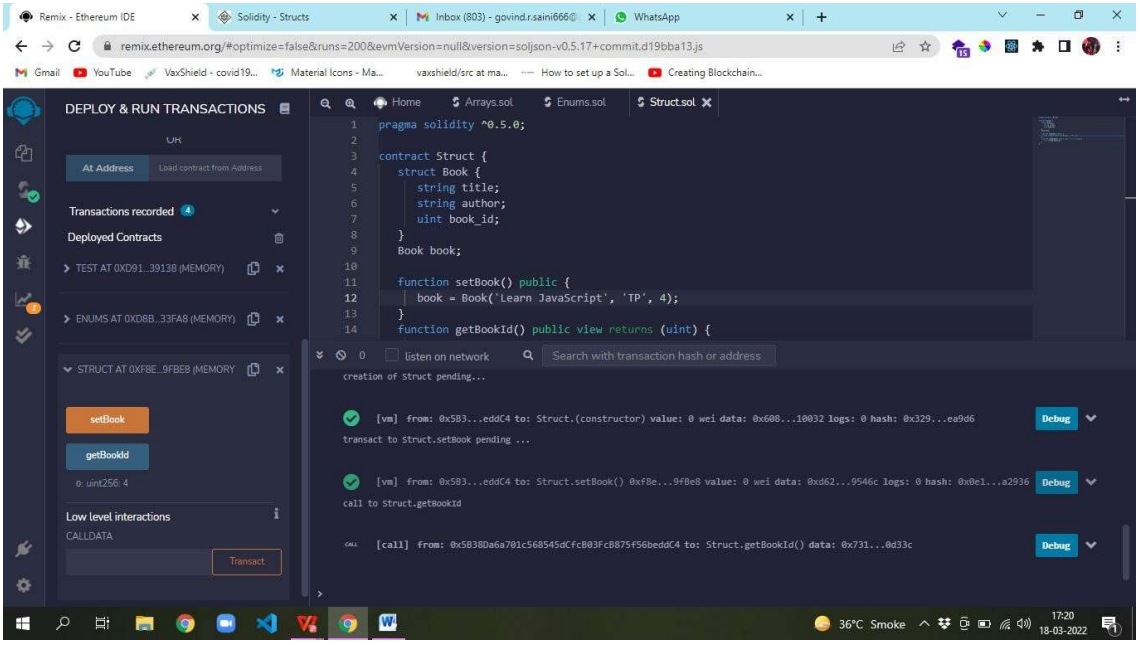


Struct

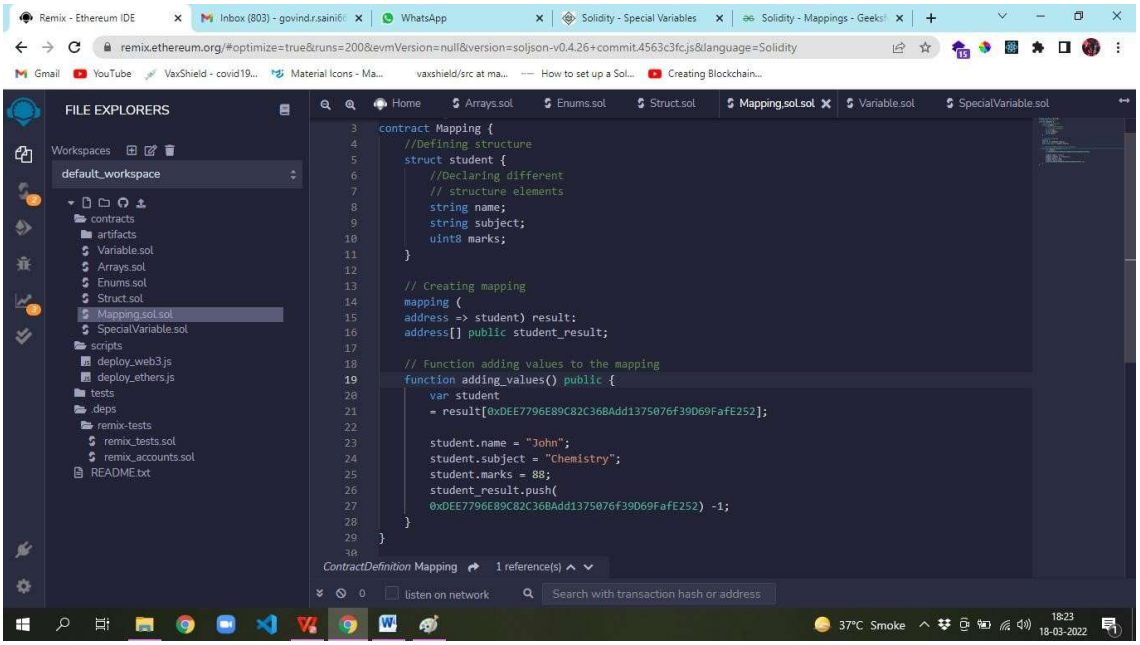
Code:



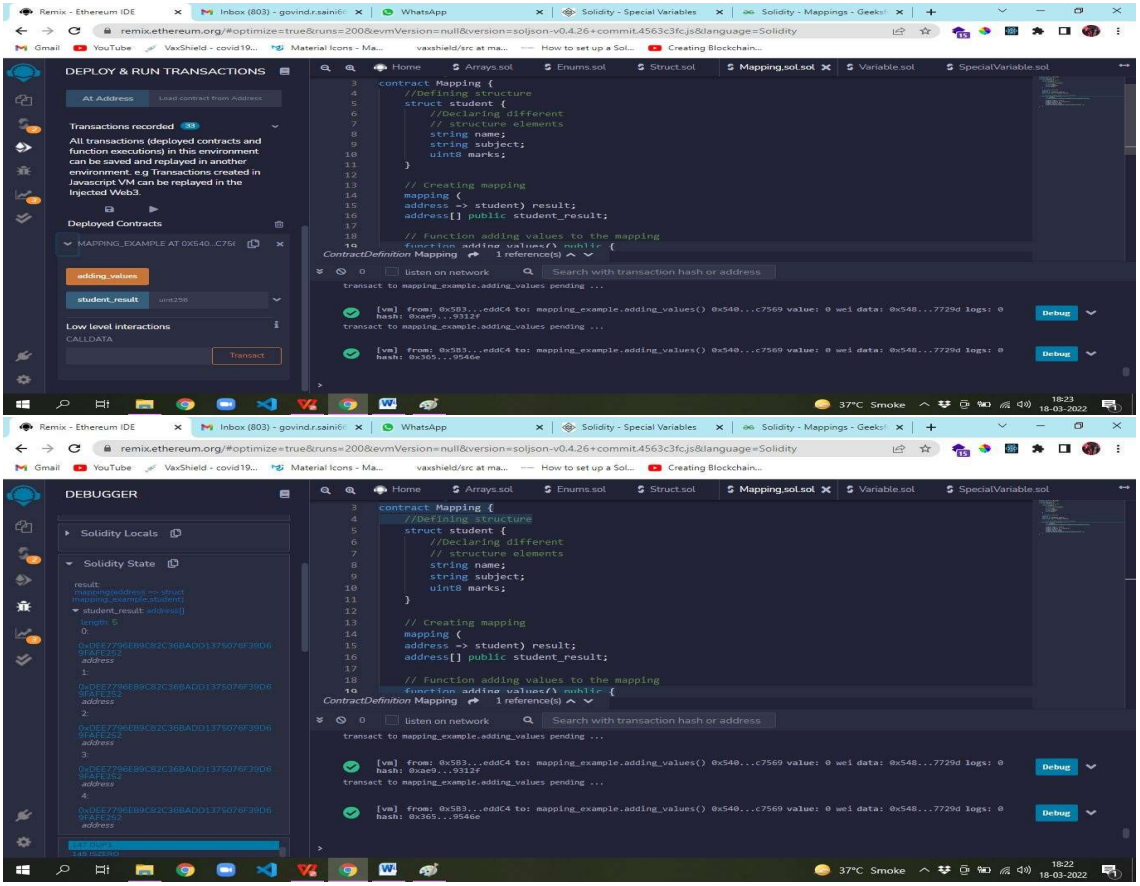
Output:



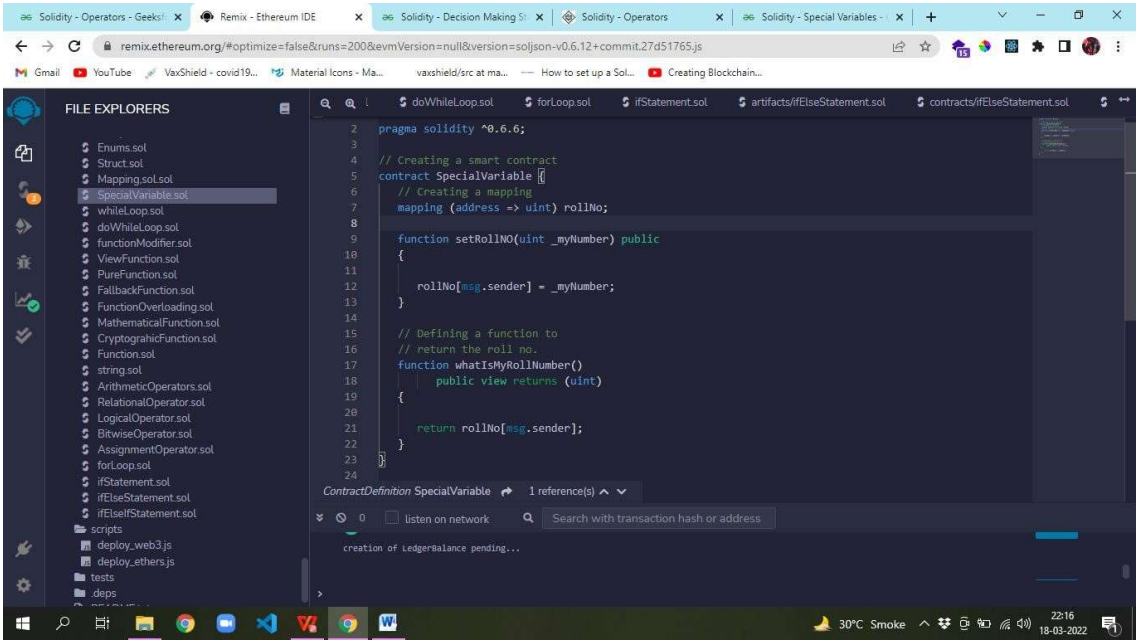
Mapping Code:



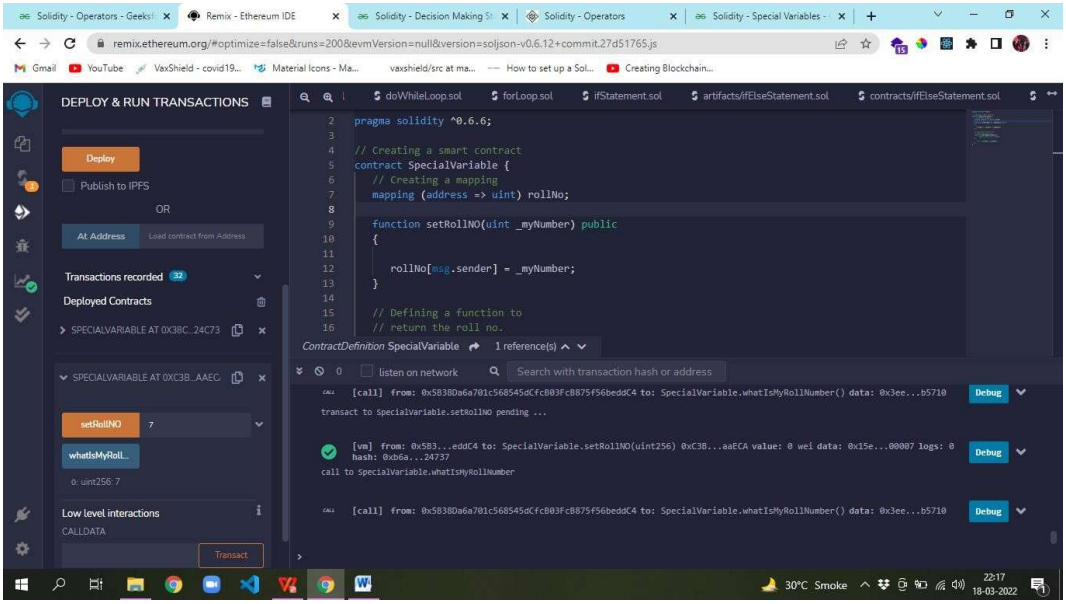
Output :



Special Variable Code:



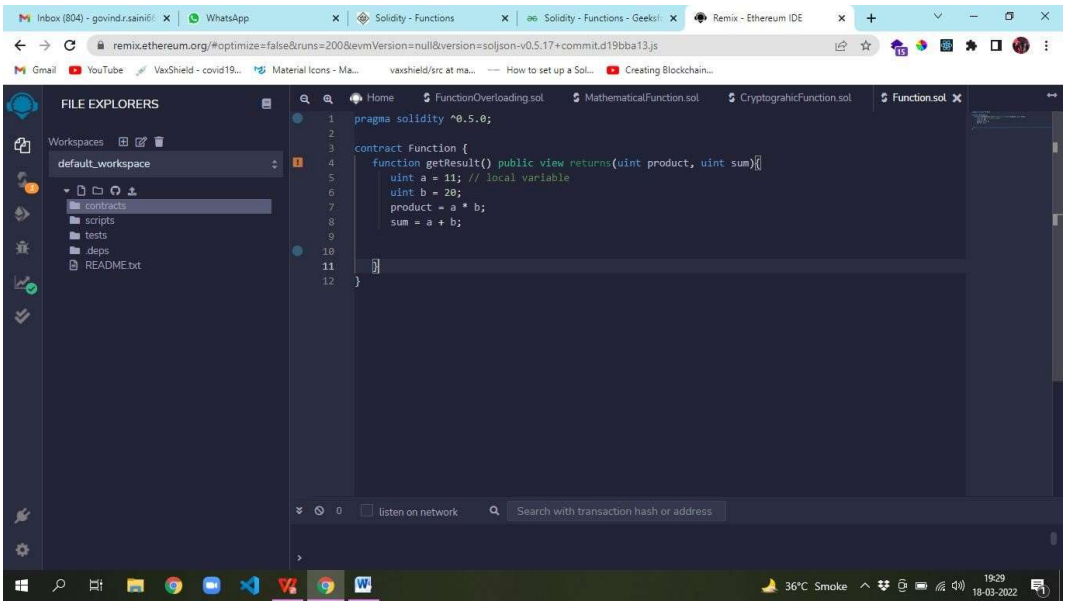
Output



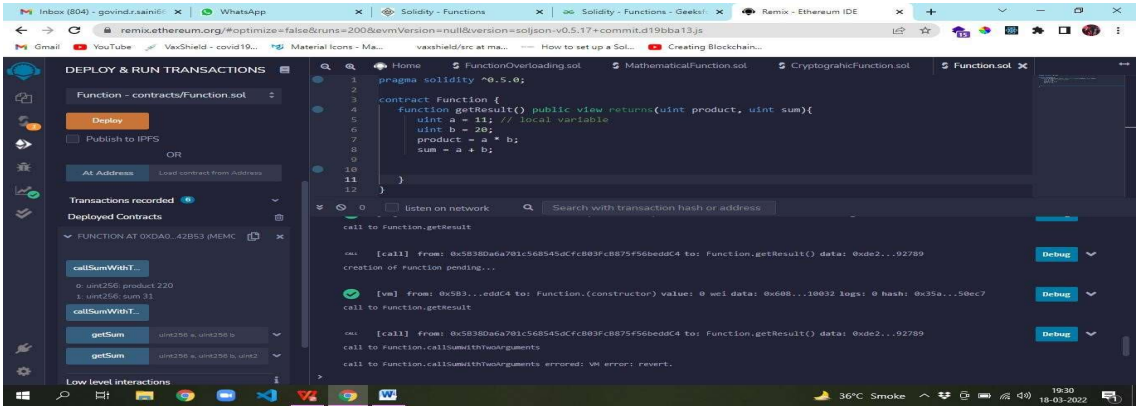
3b) Functions, Function Modifiers, View functions, Pure Functions, Fallback Function, Function Overloading, Mathematical functions, Cryptographic functions.

Function

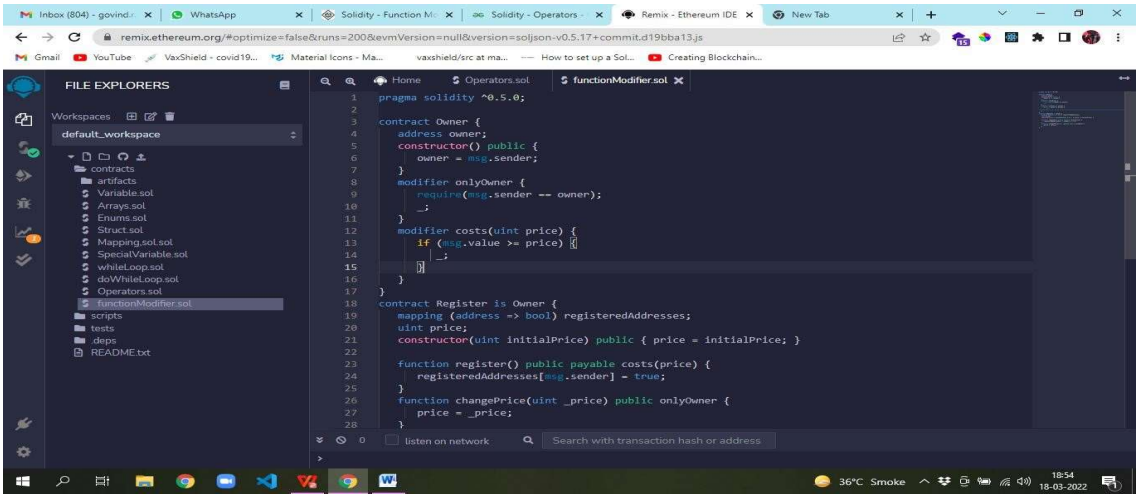
Code:



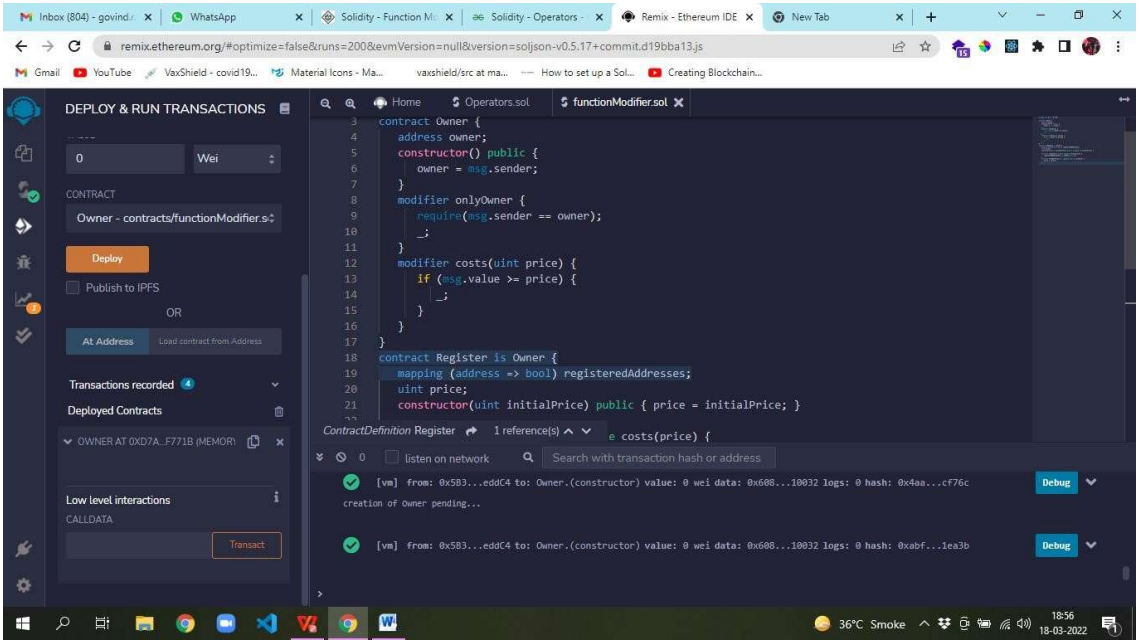
Output:



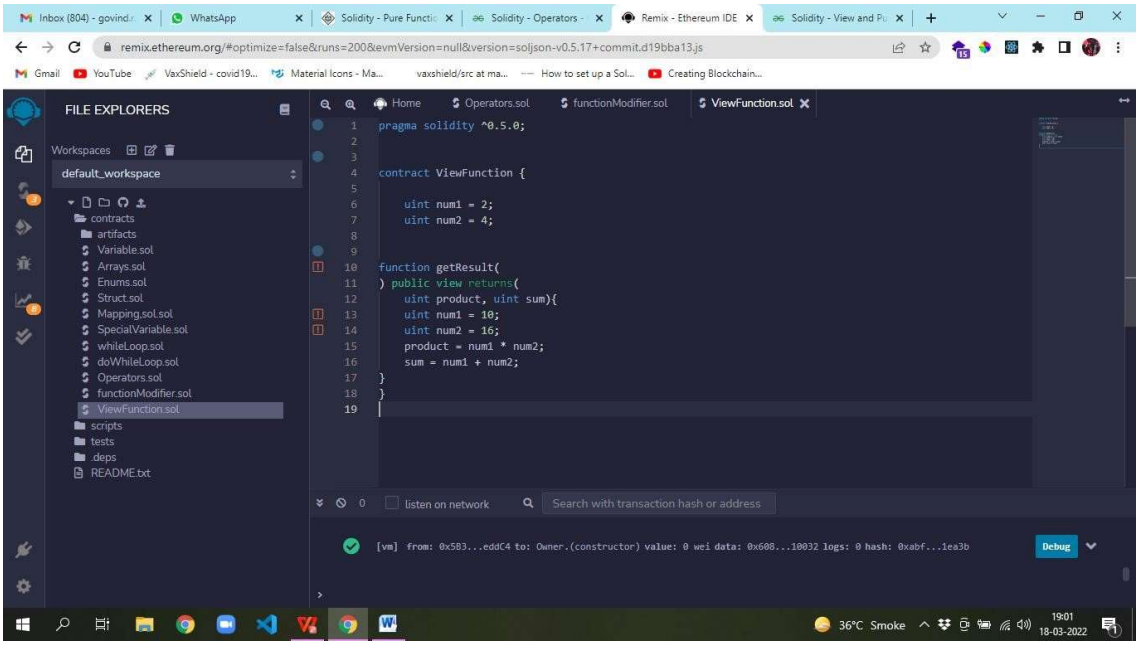
Functions Modifiers Code :



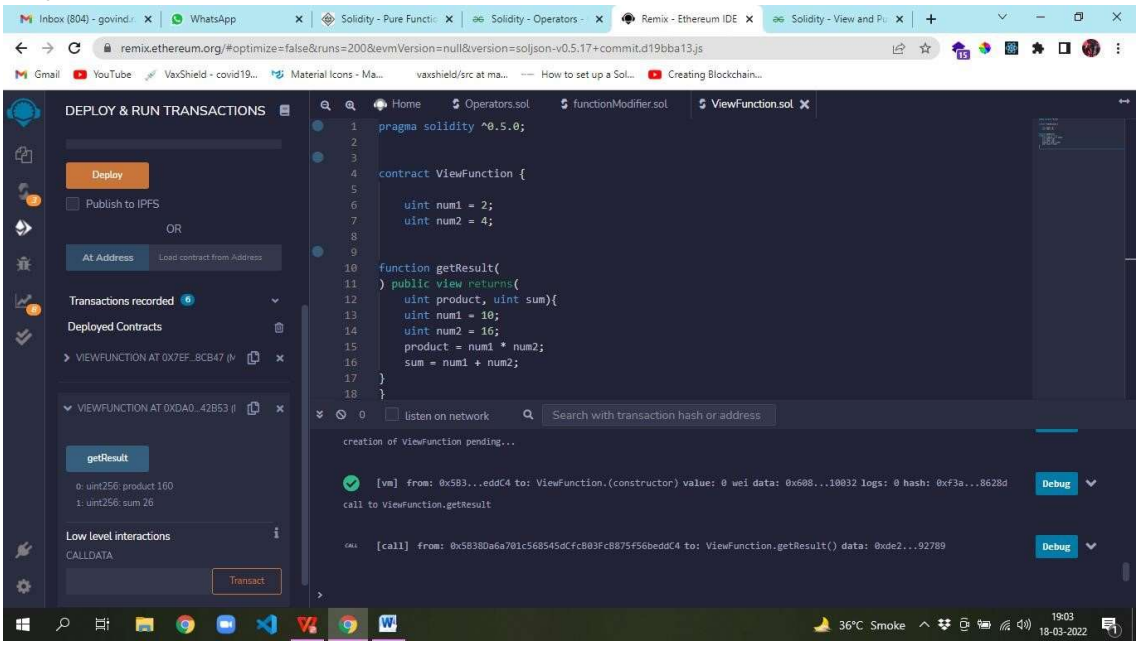
Output :



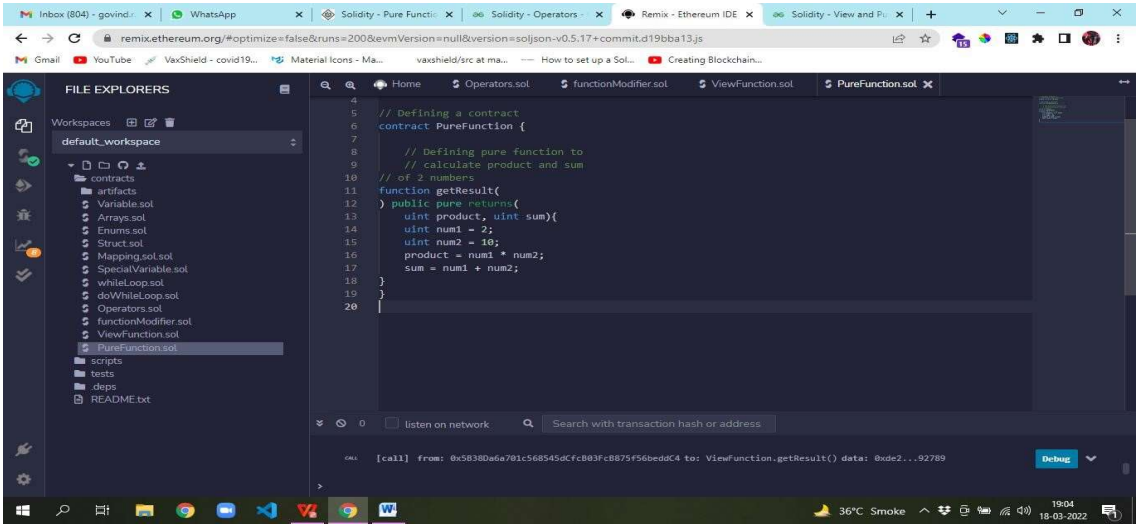
View function Code:



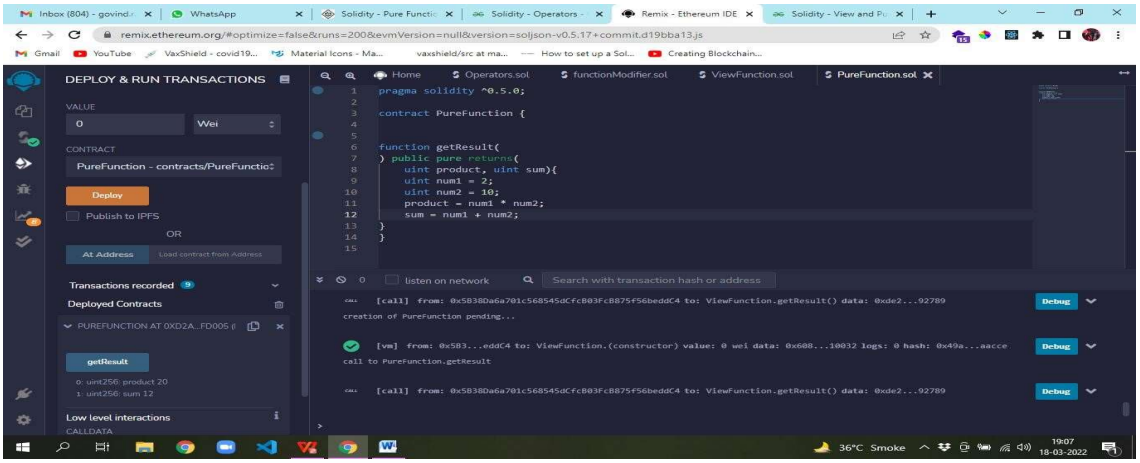
Output :



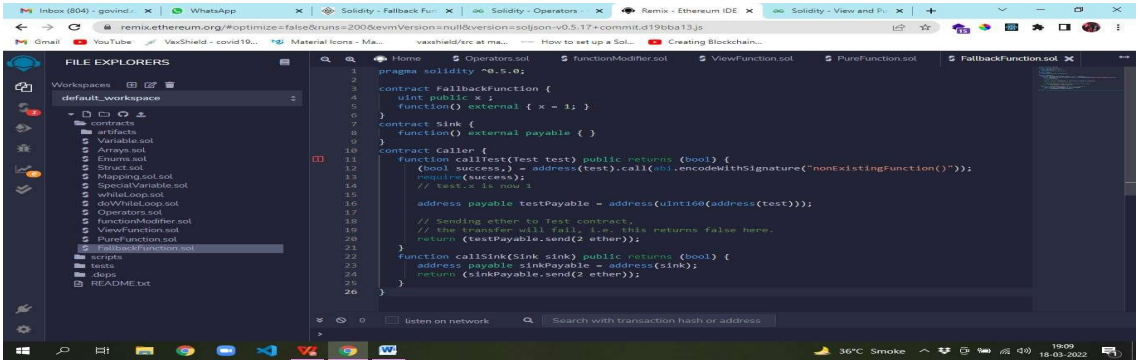
Pure function Code :



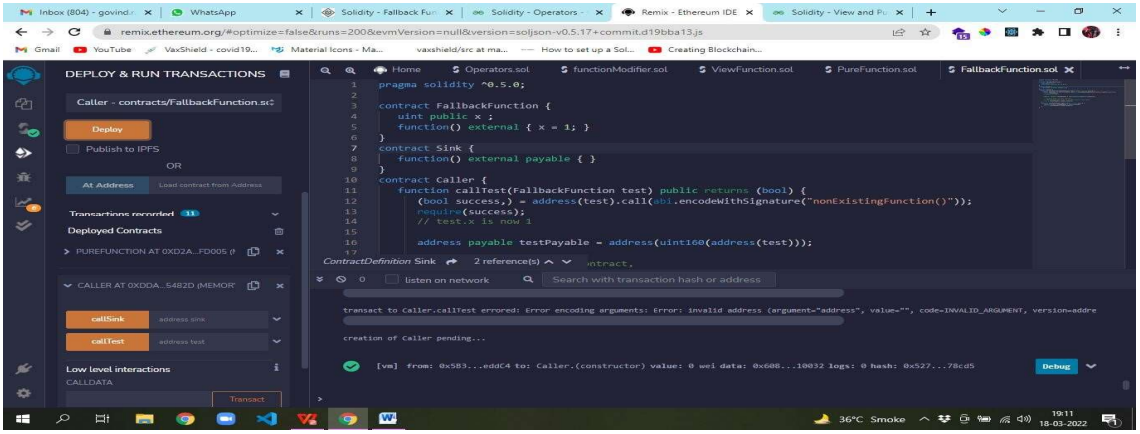
Output:



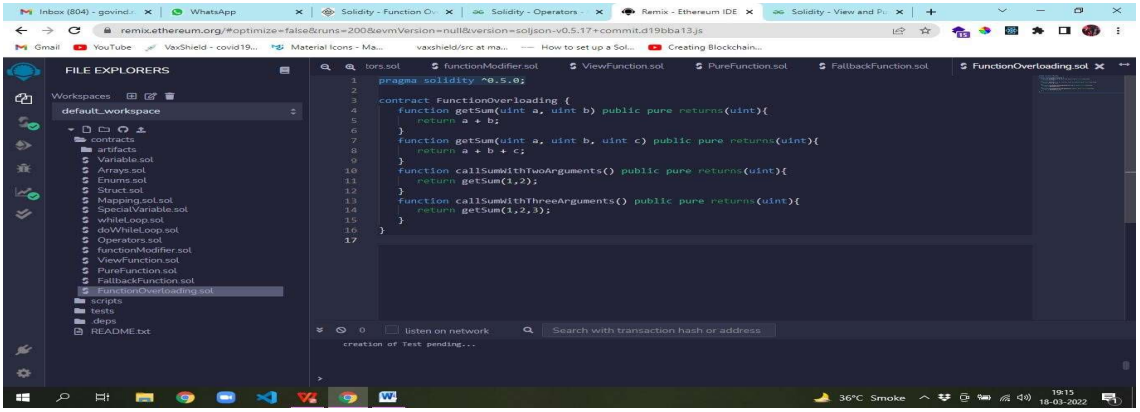
Fallback function



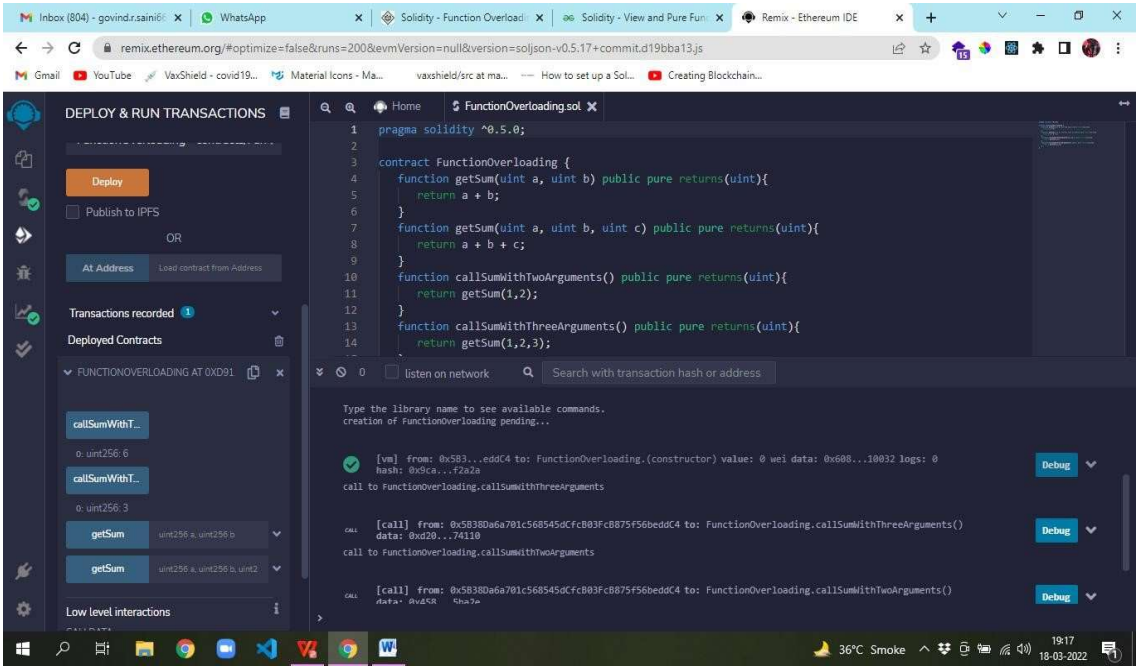
Output :



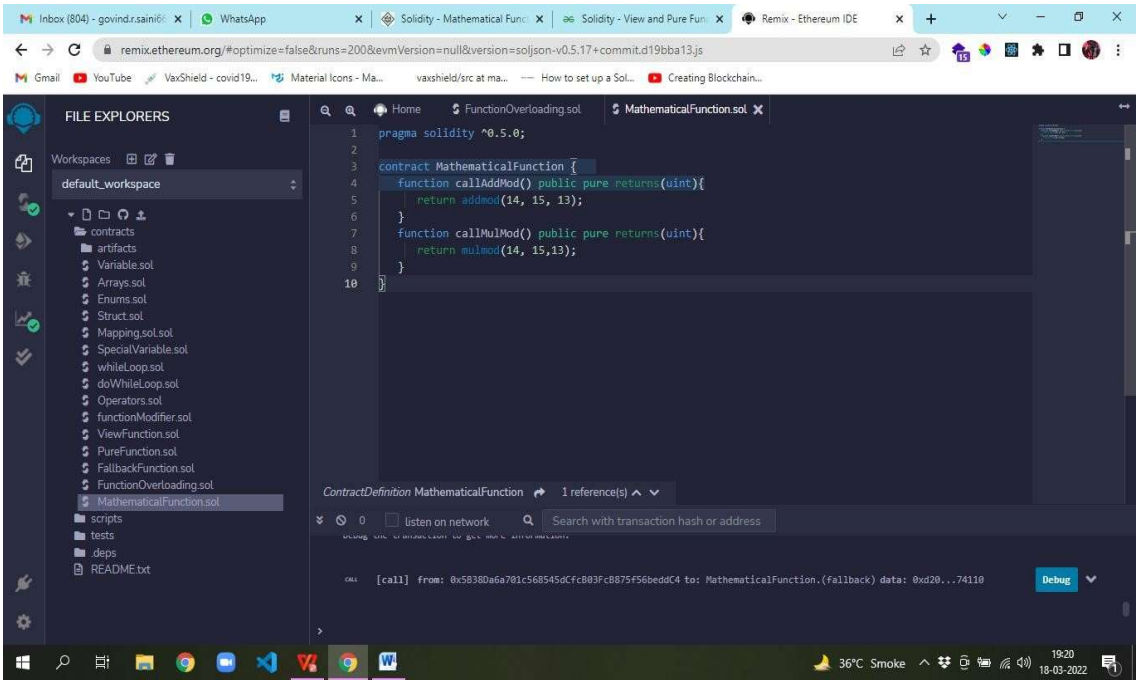
Function Overloading



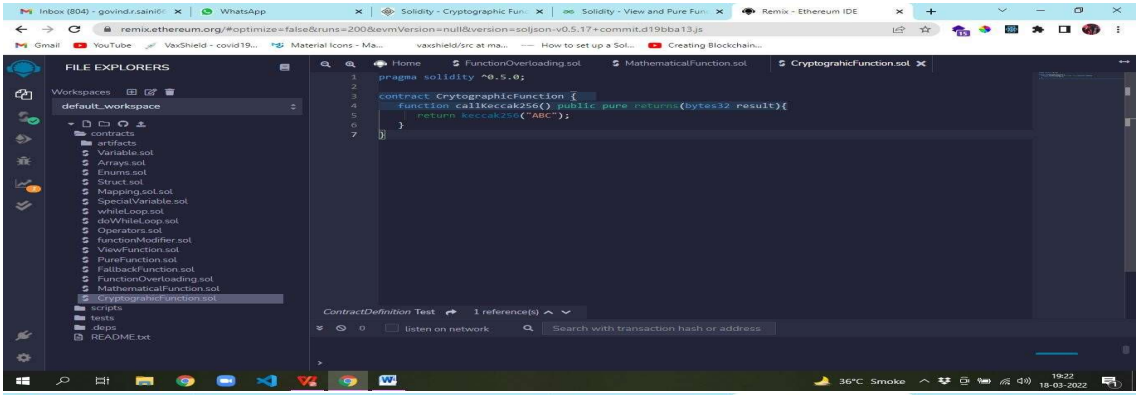
Output :

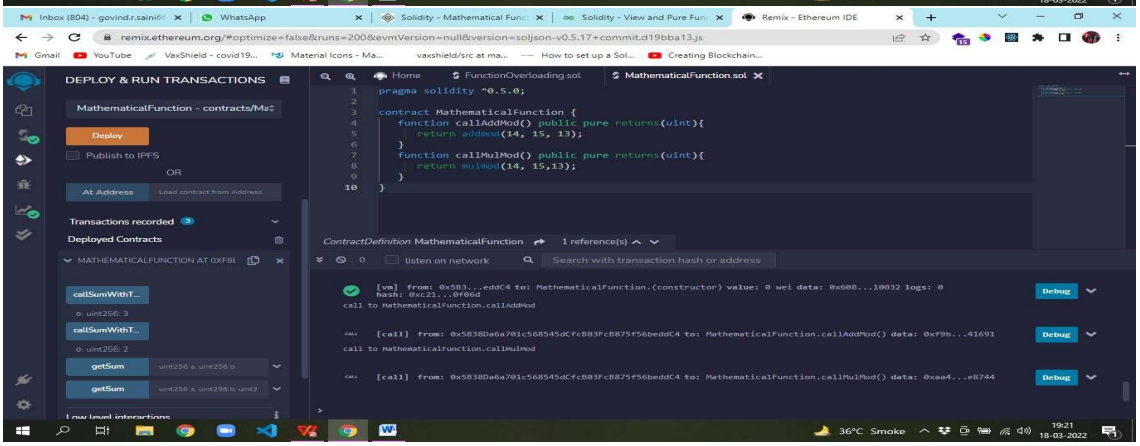


Mathematical Function

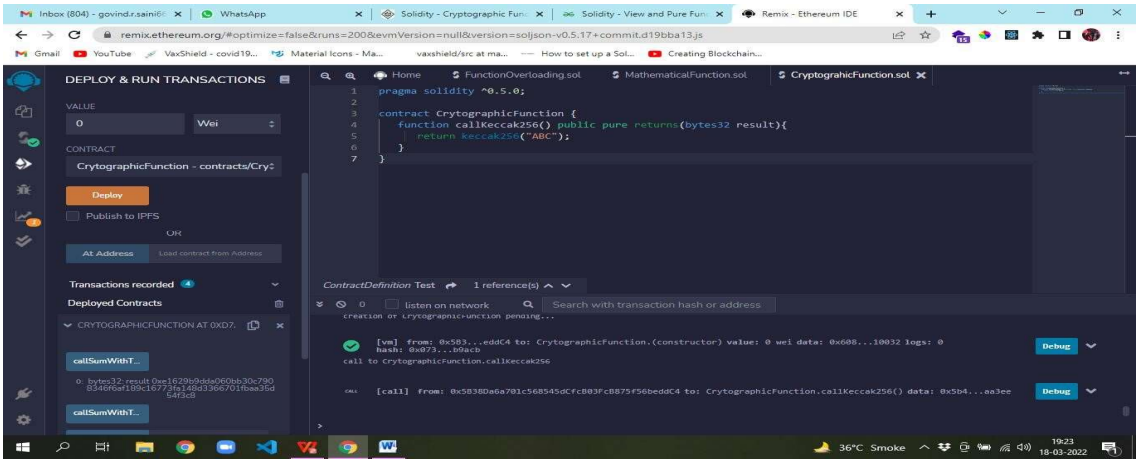


Output :


FILE EXPLORERS
Workspaces
default_workspace
contracts
artifacts
Variable.sol
Arrays.sol
Enums.sol
Struct.sol
Mapping.sol.sol
SpecialVariable.sol
whileLoop.sol
doWhileLoop.sol
Operators.sol
functionModifier.sol
ViewFunction.sol
PureFunction.sol
FallbackFunction.sol
FunctionOverloading.sol
MathematicalFunction.sol
CryptographicFunction.sol
scripts
tests
deps
README.txt
Home
FunctionOverloading.sol
MathematicalFunction.sol
CryptographicFunction.sol
pragma solidity ^0.5.0;
contract CrytographicFunction {
function callKeccak256() public pure returns(bytes32 result){
return keccak256("ABC");
}
}
ContractDefinition Test
1 reference(s)
listen on network
Search with transaction hash or address
36°C Smoke
19:22
18-03-2022


DEPLOY & RUN TRANSACTIONS
MathematicalFunction - contracts/Ma
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded
Deployed Contracts
MATHEMATICALFUNCTION AT 0XFBE
callSumWithT...
0: uint256: 3
callSumWithT...
0: uint256: 2
getSum
uint256 a, uint256 b
getSum
uint256 a, uint256 b, uint2
Low level interactions
Home
FunctionOverloading.sol
MathematicalFunction.sol
pragma solidity ^0.5.0;
contract MathematicalFunction {
function callAddMod() public pure returns(uint){
return addmod(14, 15, 13);
}
function callMulMod() public pure returns(uint){
return mulmod(14, 15,13);
}
}
ContractDefinition MathematicalFunction
1 reference(s)
listen on network
Search with transaction hash or address
[vm] from: 0x583...eddC4 to: MathematicalFunction.(constructor) value: 0 wei data: 0x608...10032 logs: 0
hash: 0xc21...0f06d
call to MathematicalFunction.callAddMod
[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: MathematicalFunction.callAddMod() data: 0xf9b...41691
call to MathematicalFunction.callMulMod
[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: MathematicalFunction.callMulMod() data: 0xea4...e8744
Debug
36°C Smoke
19:21
18-03-2022

Output :

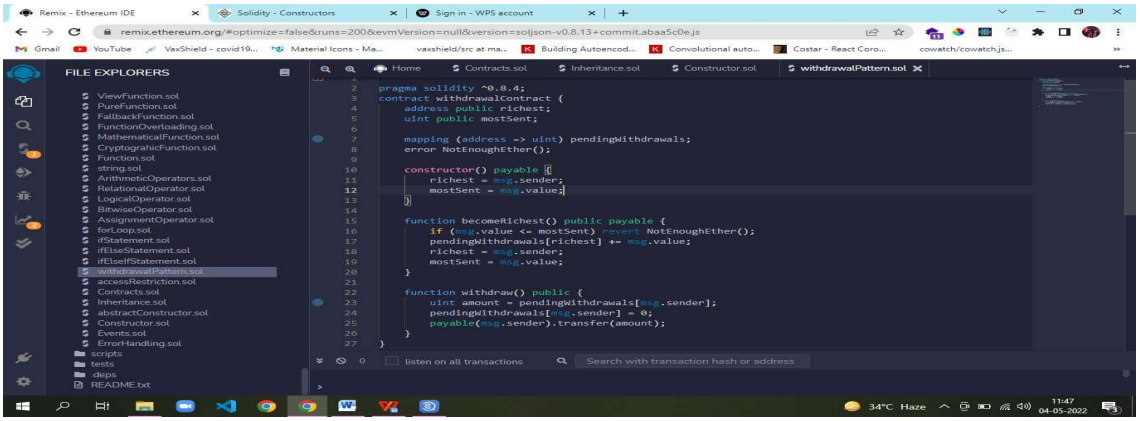

DEPLOY & RUN TRANSACTIONS
VALUE
0
Wei
CONTRACT
CrytographicFunction - contracts/Cry
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded
Deployed Contracts
CRYTOGRAPHICFUNCTION AT 0XD7...
callSumWithT...
0: bytes32: result 0xe16291b9d6e0604b30c7908346f6af189c16773fa148d3366701fbaa35d54f3c8
callSumWithT...
Home
FunctionOverloading.sol
MathematicalFunction.sol
CryptographicFunction.sol
pragma solidity ^0.5.0;
contract CrytographicFunction {
function callKeccak256() public pure returns(bytes32 result){
return keccak256("ABC");
}
}
ContractDefinition Test
1 reference(s)
listen on network
Search with transaction hash or address
creation of crytographicFunction pending...
[vm] from: 0x583...eddC4 to: CrytographicFunction.(constructor) value: 0 wei data: 0x608...10032 logs: 0
hash: 0x073...b9acb
call to CrytographicFunction.callKeccak256
[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CrytographicFunction.callKeccak256() data: 0x5b4...aa3ee
Debug
36°C Smoke
19:23
18-03-2022

Practical No : 4

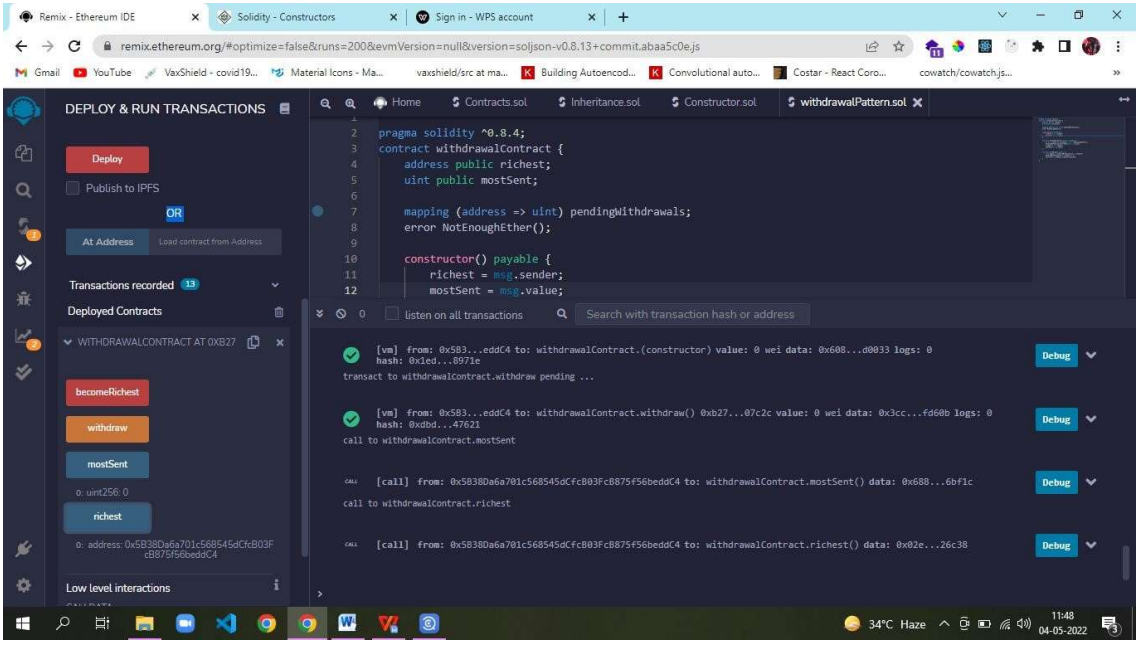
Aim : Implement and demonstrate the use of the following in Solidity :

4a) Withdrawal Pattern, Restricted Access. Withdrawal Pattern :

Code :

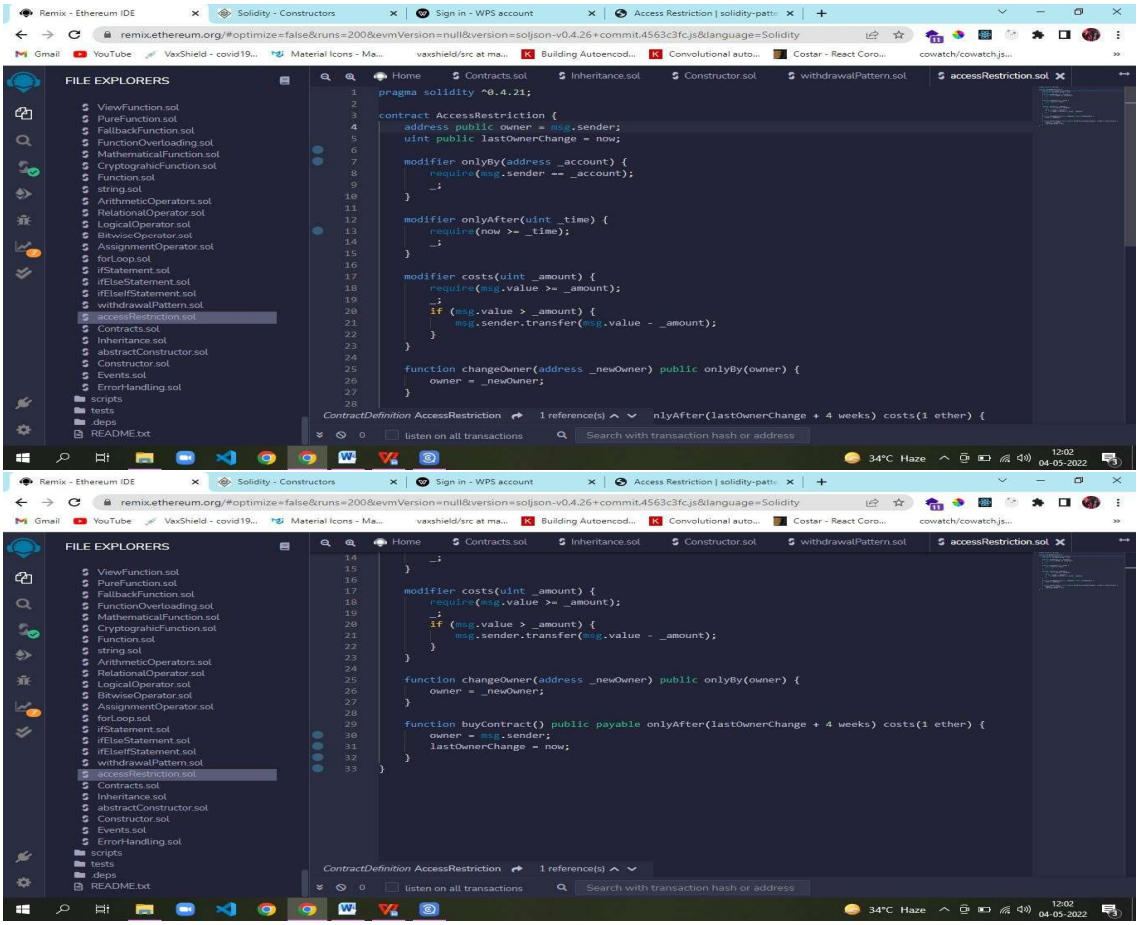


Ouput :

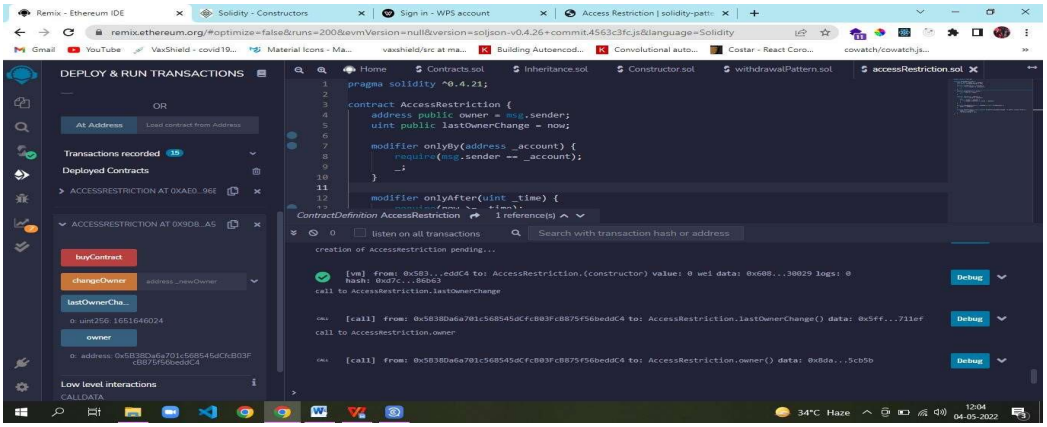


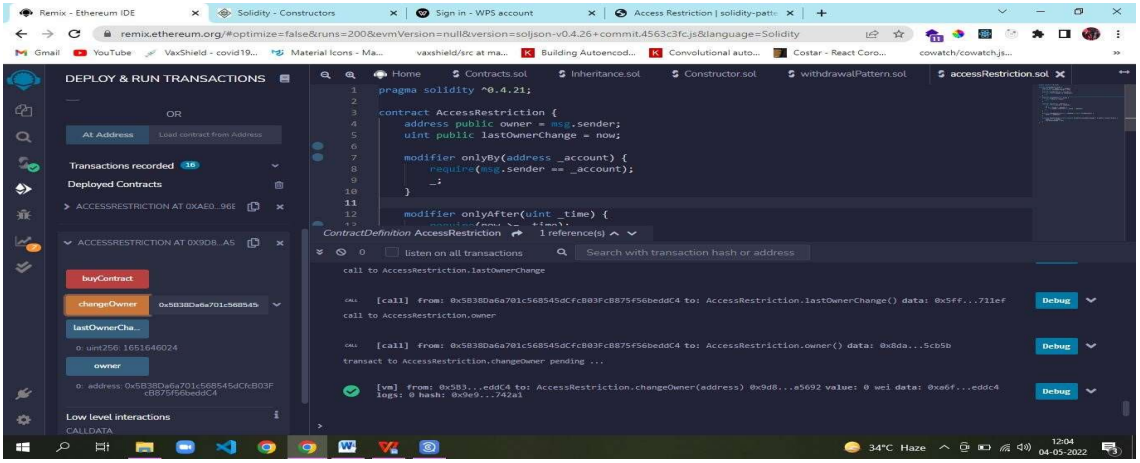
Restricted Access :

Code :



Ouput :

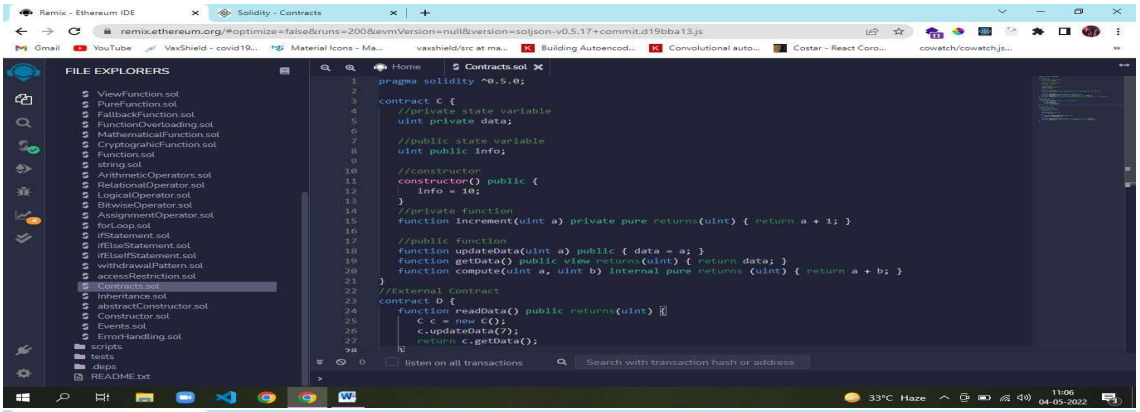


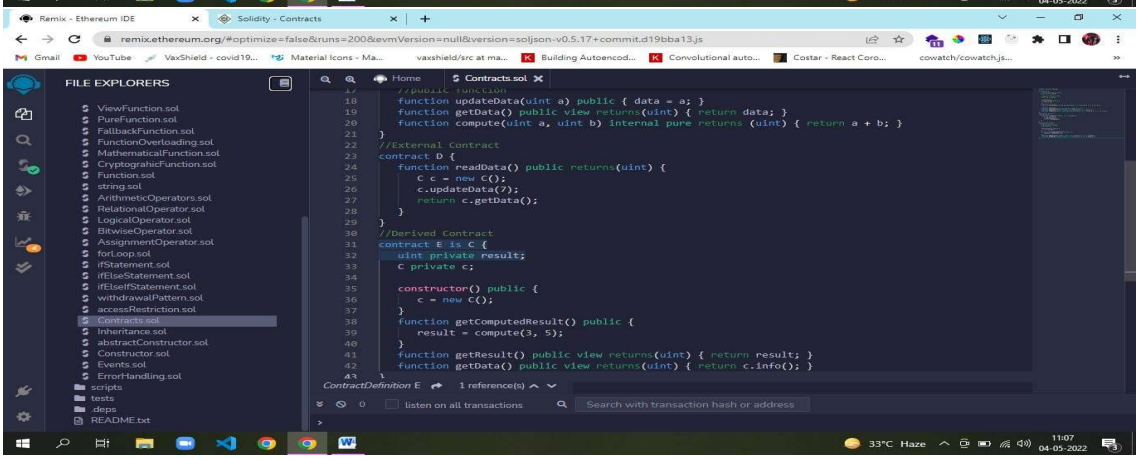


4b) Contracts, Inheritance, Constructors, Abstract Contracts, Interfaces.

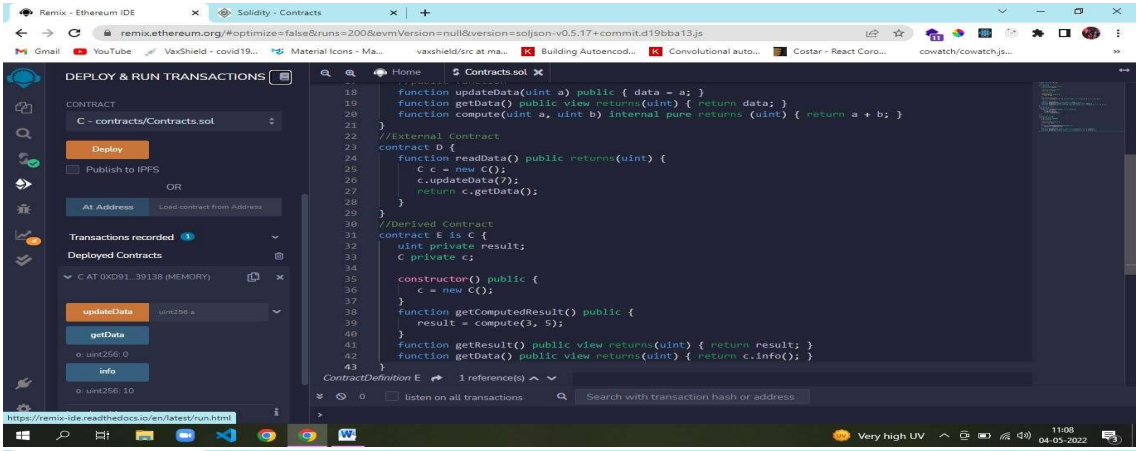
Contracts:

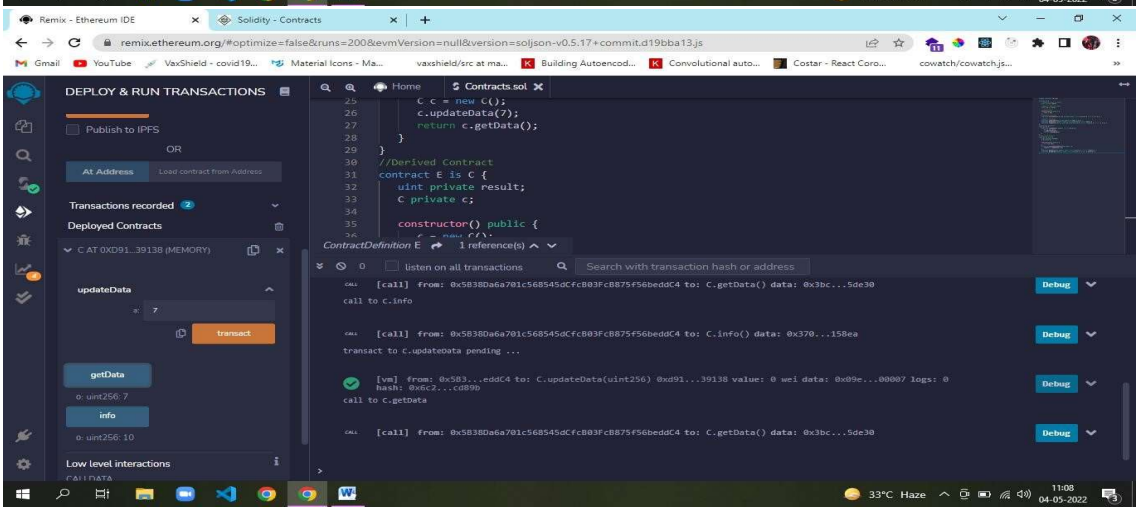
Code :



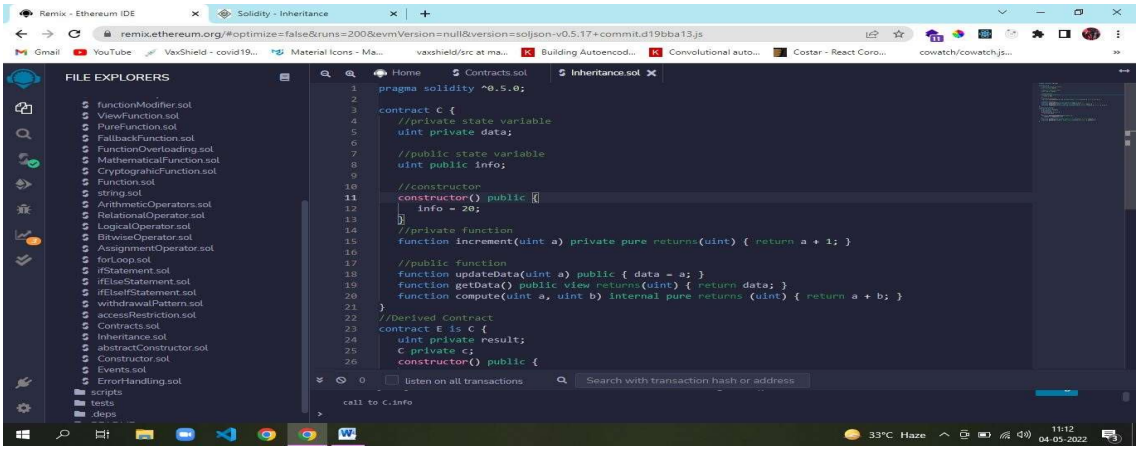


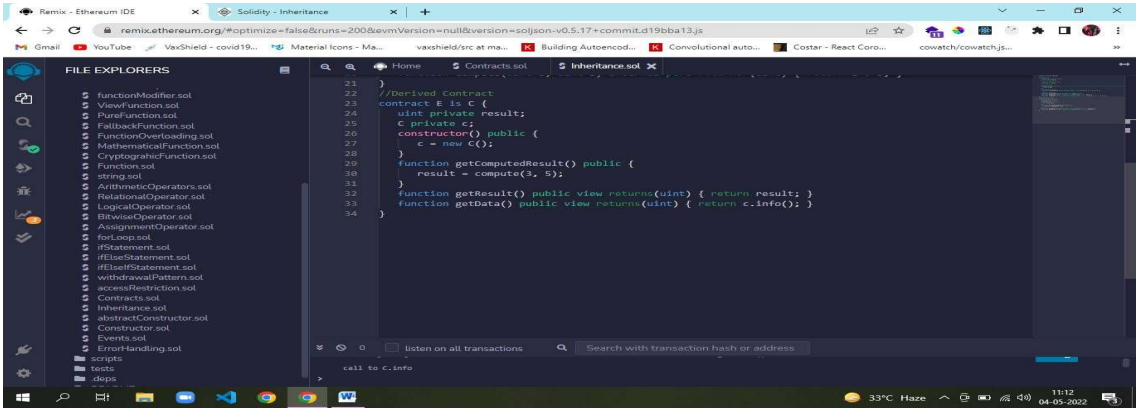
Output :



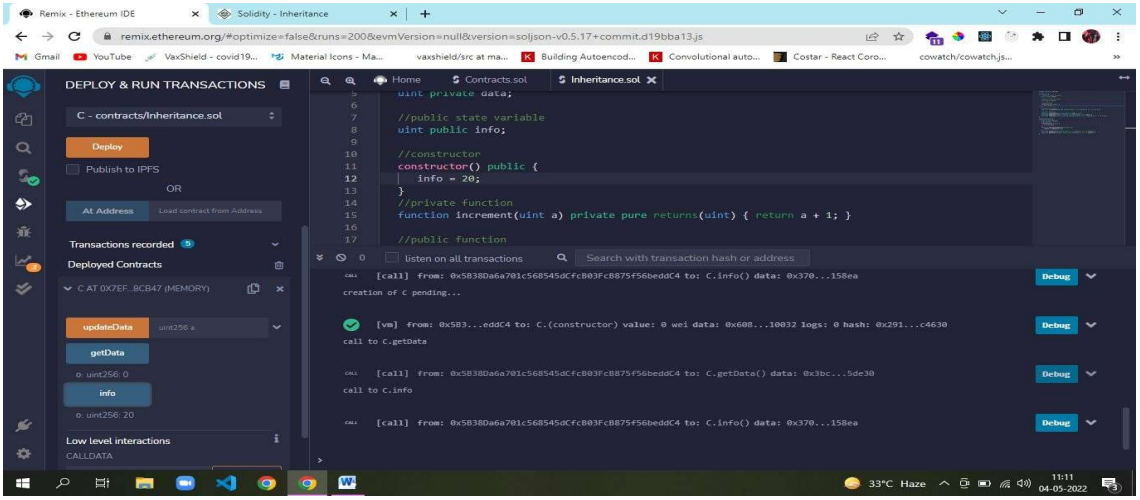


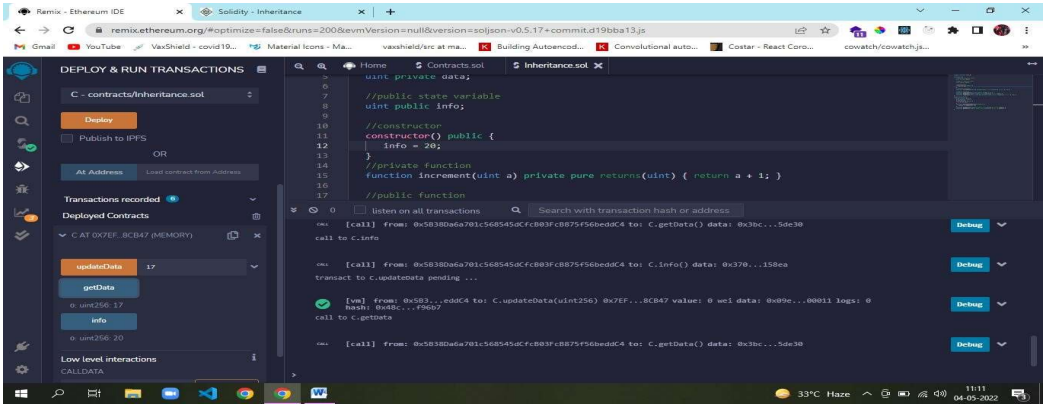
Inheritance Code :




Remix - Ethereum IDE
Solidity - Inheritance
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.17+commit.d19bba13.js
FILE EXPLORERS
Home
Contracts.sol
Inheritance.sol
//Derived Contract
contract E is C {
uint private result;
C private c;
constructor() public {
c = new C();
}
function getComputedResult() public {
result = compute(3, 5);
}
function getResult() public view returns(uint) { return result; }
function getData() public view returns(uint) { return c.info(); }
}
listen on all transactions
Search with transaction hash or address
call to C.info

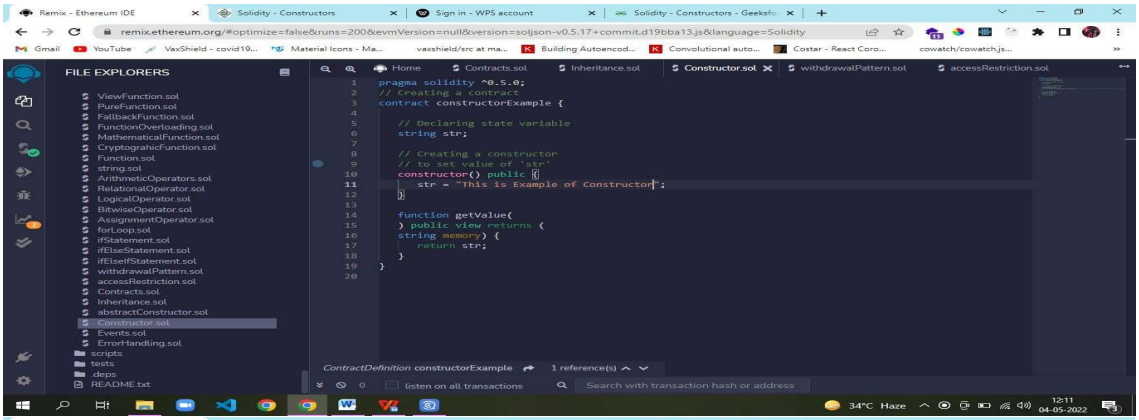
Output :

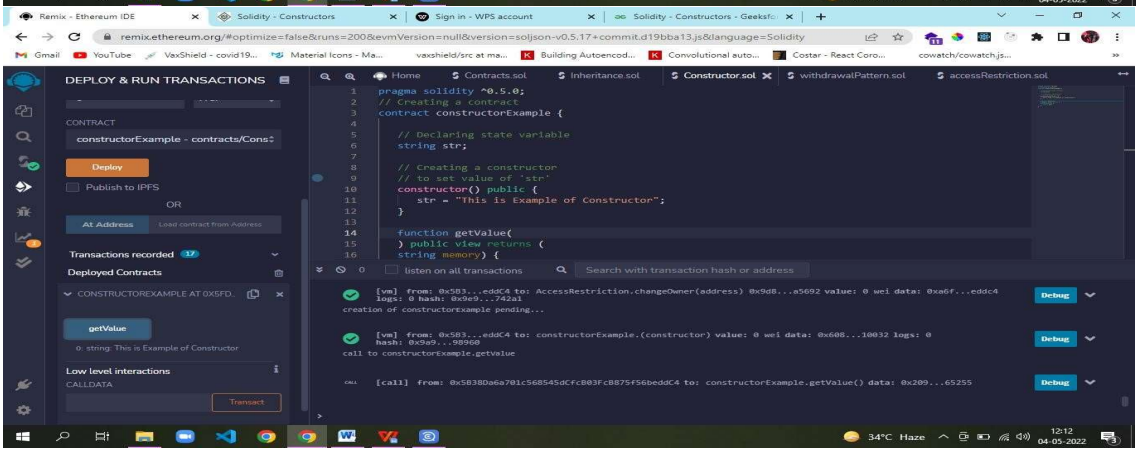

DEPLOY & RUN TRANSACTIONS
C - contracts/Inheritance.sol
Deploy
Publish to IPFS
OR
At Address
Transactions recorded
Deployed Contracts
updateData
getData
info
Low level interactions
CALLDATA
uint private data;
//public state variable
uint public info;
//constructor
constructor() public {
info = 20;
}
//private function
function increment(uint a) private pure returns(uint) { return a + 1; }
//public function
creation of C pending...
call to C.getData
call to C.info


DEPLOY & RUN TRANSACTIONS
C - contracts/Inheritance.sol
Deploy
Publish to IPFS
OR
At Address
Transactions recorded
Deployed Contracts
updateData
getData
info
Low level interactions
CALLDATA
uint private data;
//public state variable
uint public info;
//constructor
constructor() public {
info = 20;
}
//private function
function increment(uint a) private pure returns(uint) { return a + 1; }
//public function
call to C.info
transact to C.updateData pending ...
call to C.getData

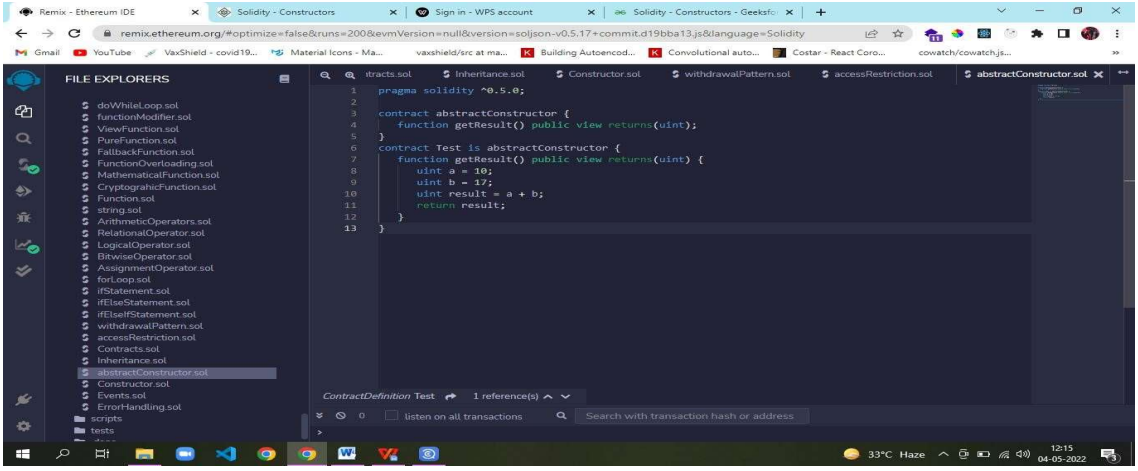
Constructors :

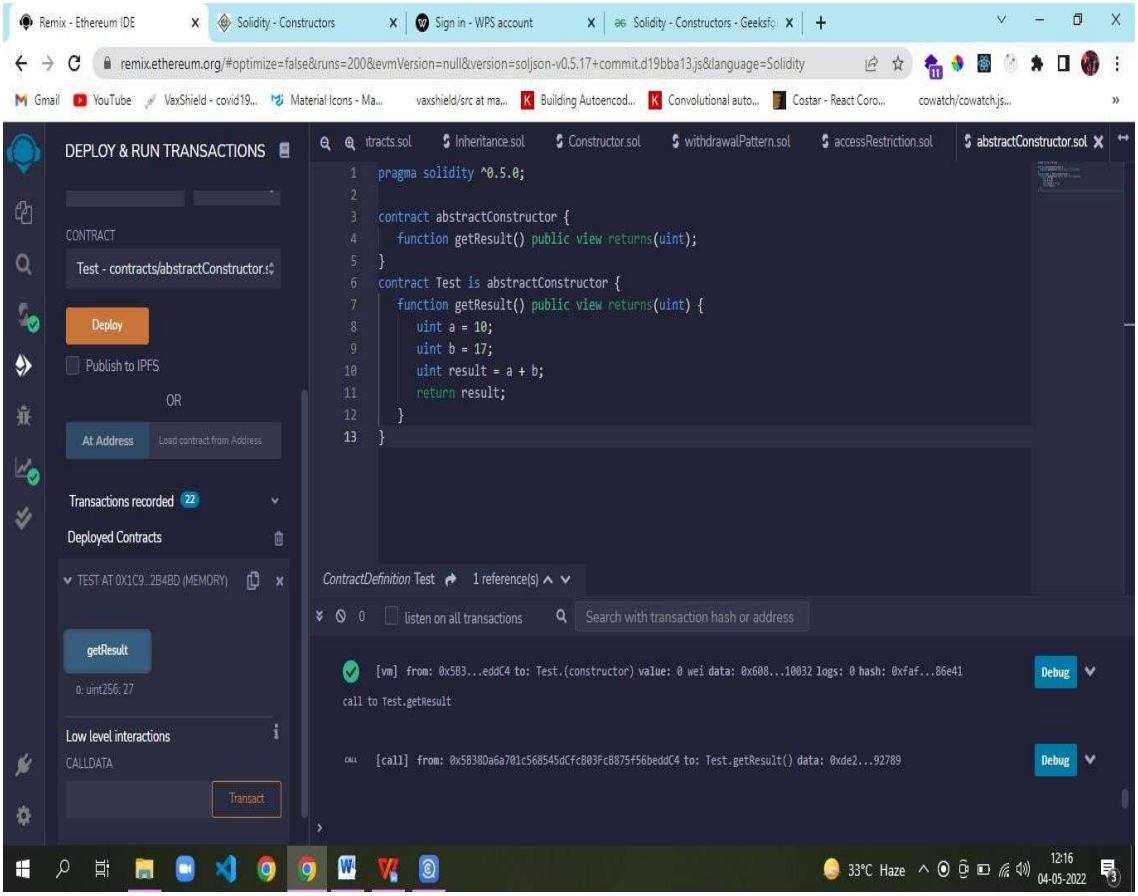
Code :





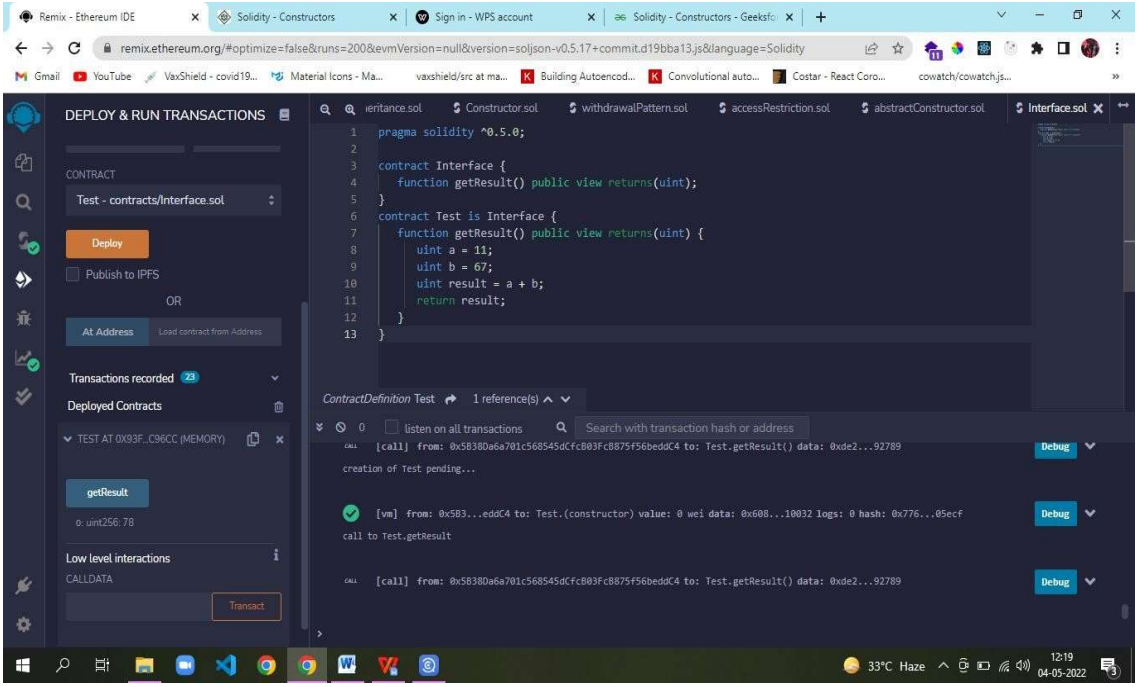
Abstract Contracts :Code :





Interfaces :

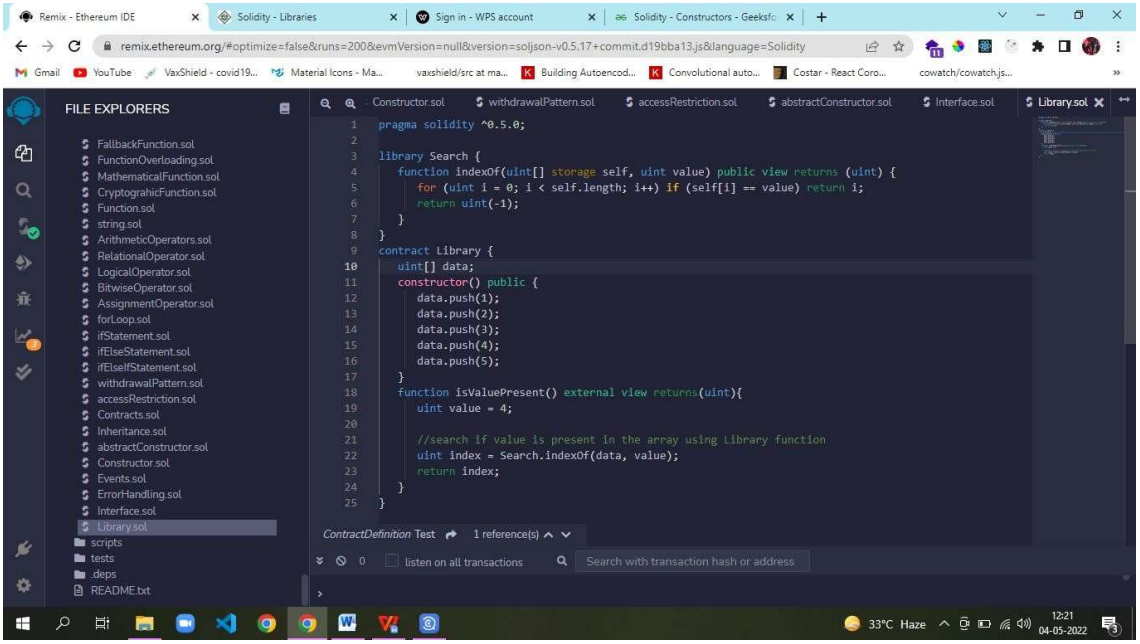
Code



4c) Libraries, Assembly, Events, Error handling.

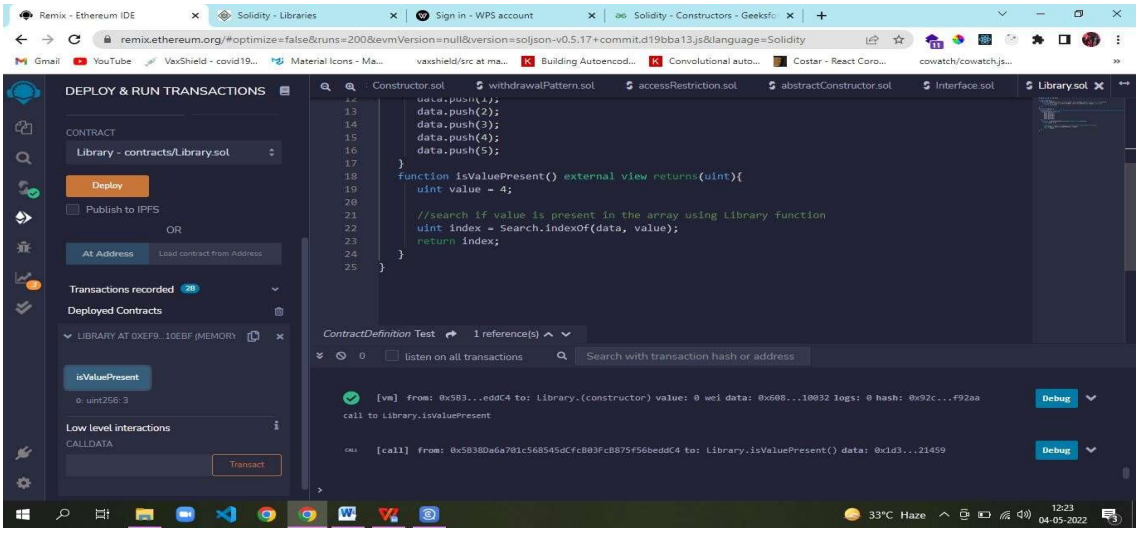
Libraries :

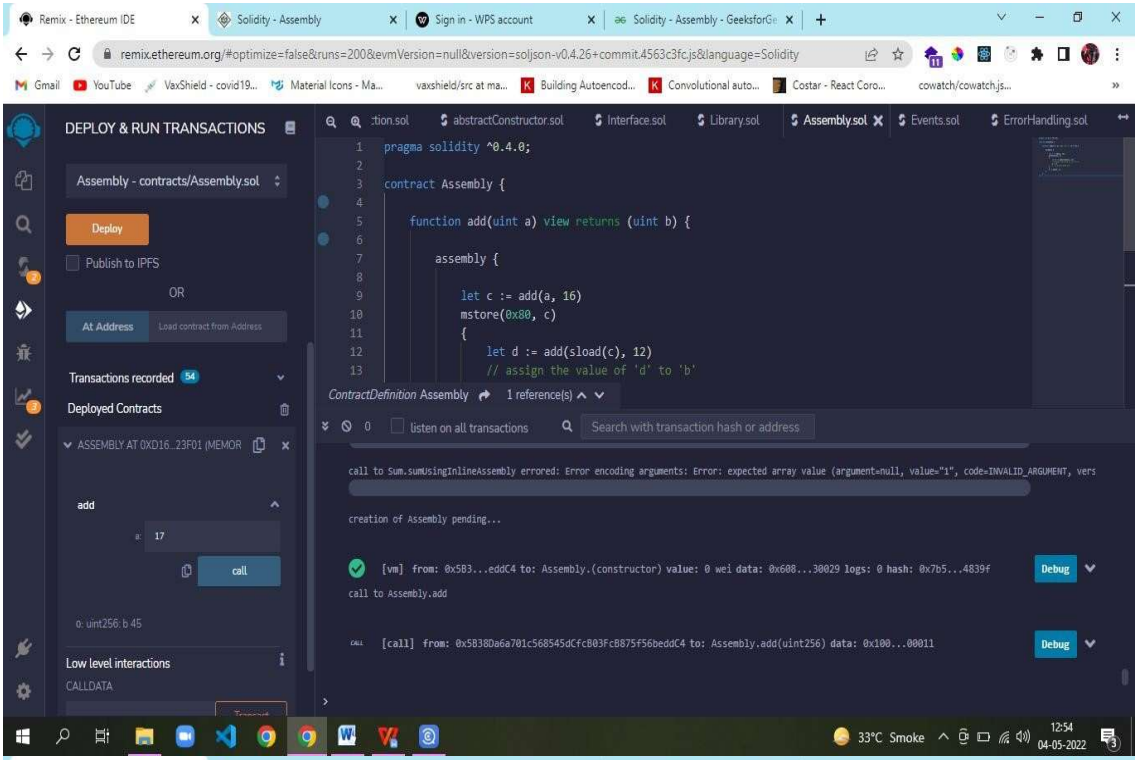
Code :

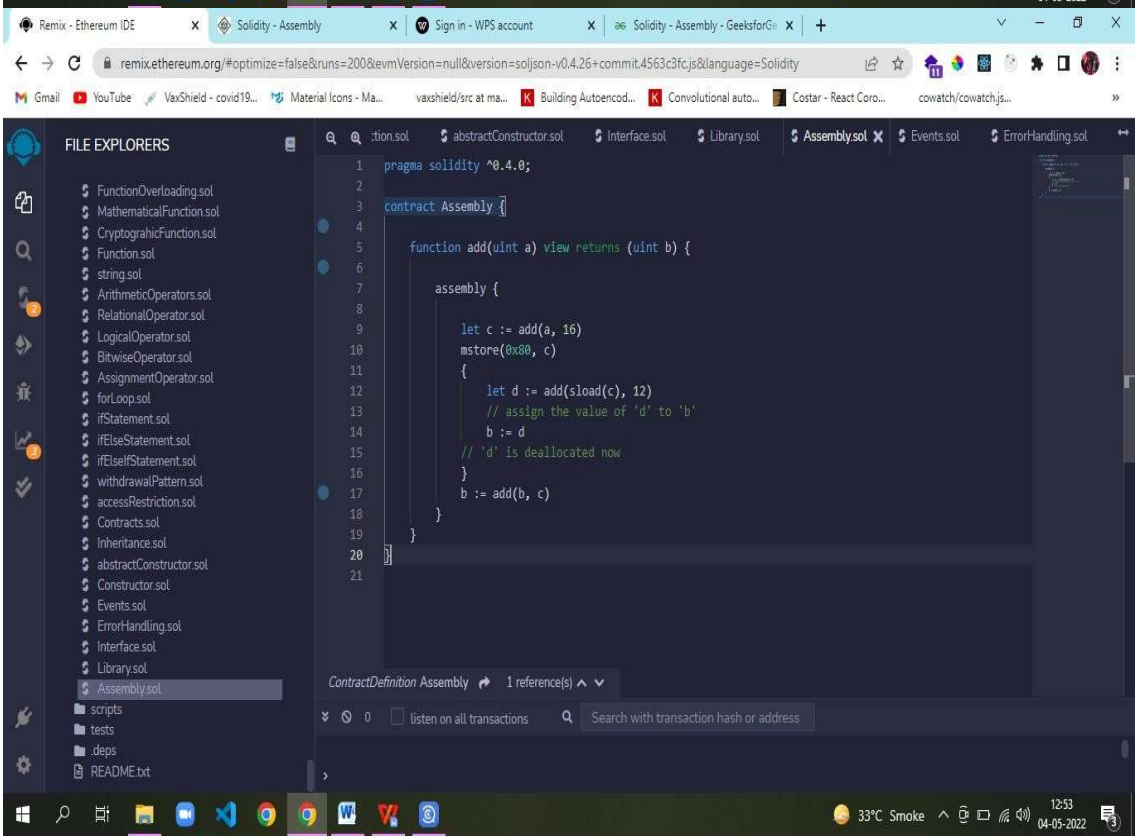


Output :

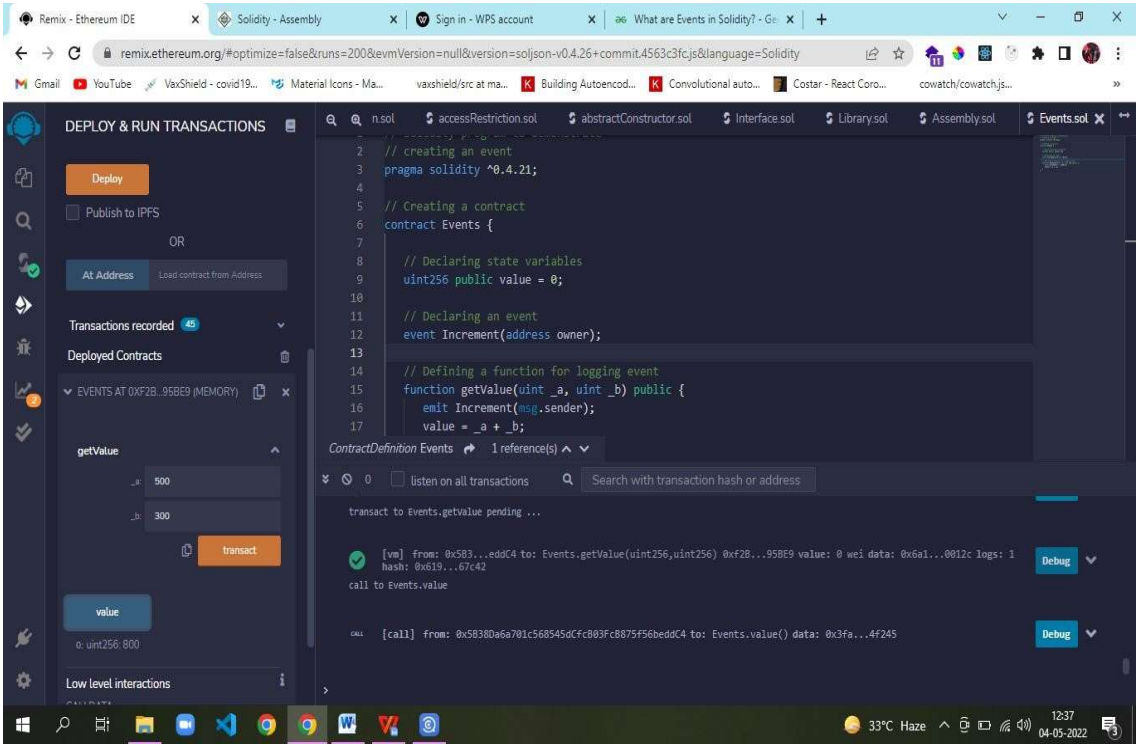
Assembly : Code :




Remix - Ethereum IDE
Solidity - Assembly
Sign in - WPS account
Solidity - Assembly - GeeksforGe
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.4.26+commit.4563c3fc.js&language=Solidity
DEPLOY & RUN TRANSACTIONS
Assembly - contracts/Assembly.sol
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 54
Deployed Contracts
ASSEMBLY AT 0XD16...23F01 (MEMOR
add
a: 17
call
0: uint256: b 45
Low level interactions
CALLDATA
Assembly.sol
Events.sol
ErrorHandling.sol
pragma solidity ^0.4.0;
contract Assembly {
function add(uint a) view returns (uint b) {
assembly {
let c := add(a, 16)
mstore(0x80, c)
let d := add(sload(c), 12)
// assign the value of 'd' to 'b'
ContractDefinition Assembly
1 reference(s)
listen on all transactions
Search with transaction hash or address
creation of Assembly pending...
[vm] from: 0x5B3...eddC4 to: Assembly.(constructor) value: 0 wei data: 0x608...30029 logs: 0 hash: 0x7b5...4839f
Debug
call to Assembly.add
[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: Assembly.add(uint256) data: 0x100...00011
Debug
33°C Smoke
12:54
04-05-2022

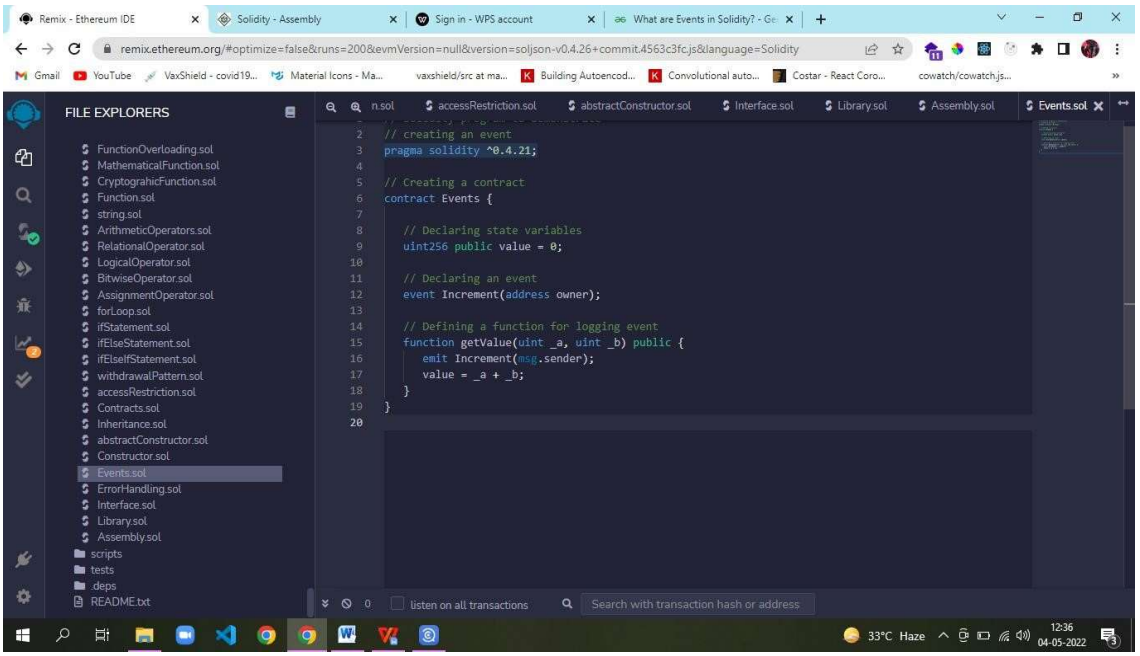

Remix - Ethereum IDE
Solidity - Assembly
Sign in - WPS account
Solidity - Assembly - GeeksforGe
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.4.26+commit.4563c3fc.js&language=Solidity
FILE EXPLORERS
FunctionOverloading.sol
MathematicalFunction.sol
CryptograhicFunction.sol
Function.sol
string.sol
ArithmeticOperators.sol
RelationalOperator.sol
LogicalOperator.sol
BitwiseOperator.sol
AssignmentOperator.sol
forLoop.sol
ifStatement.sol
ifElseStatement.sol
ifElseIfStatement.sol
withdrawalPattern.sol
accessRestriction.sol
Contracts.sol
Inheritance.sol
abstractConstructor.sol
Constructor.sol
Events.sol
ErrorHandling.sol
Interface.sol
Library.sol
Assembly.sol
scripts
tests
.deps
README.txt
pragma solidity ^0.4.0;
contract Assembly {
function add(uint a) view returns (uint b) {
assembly {
let c := add(a, 16)
mstore(0x80, c)
let d := add(sload(c), 12)
// assign the value of 'd' to 'b'
b := d
// 'd' is deallocated now
b := add(b, c)
ContractDefinition Assembly
1 reference(s)
listen on all transactions
Search with transaction hash or address
33°C Smoke
12:53
04-05-2022

Output :



Events:

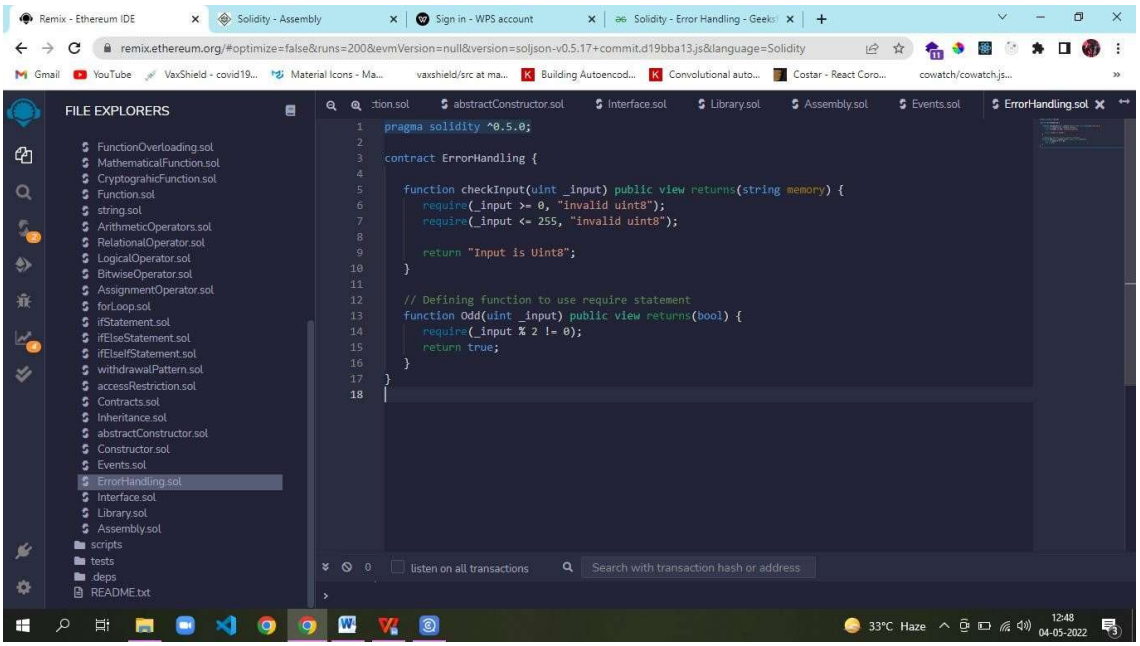
Code :



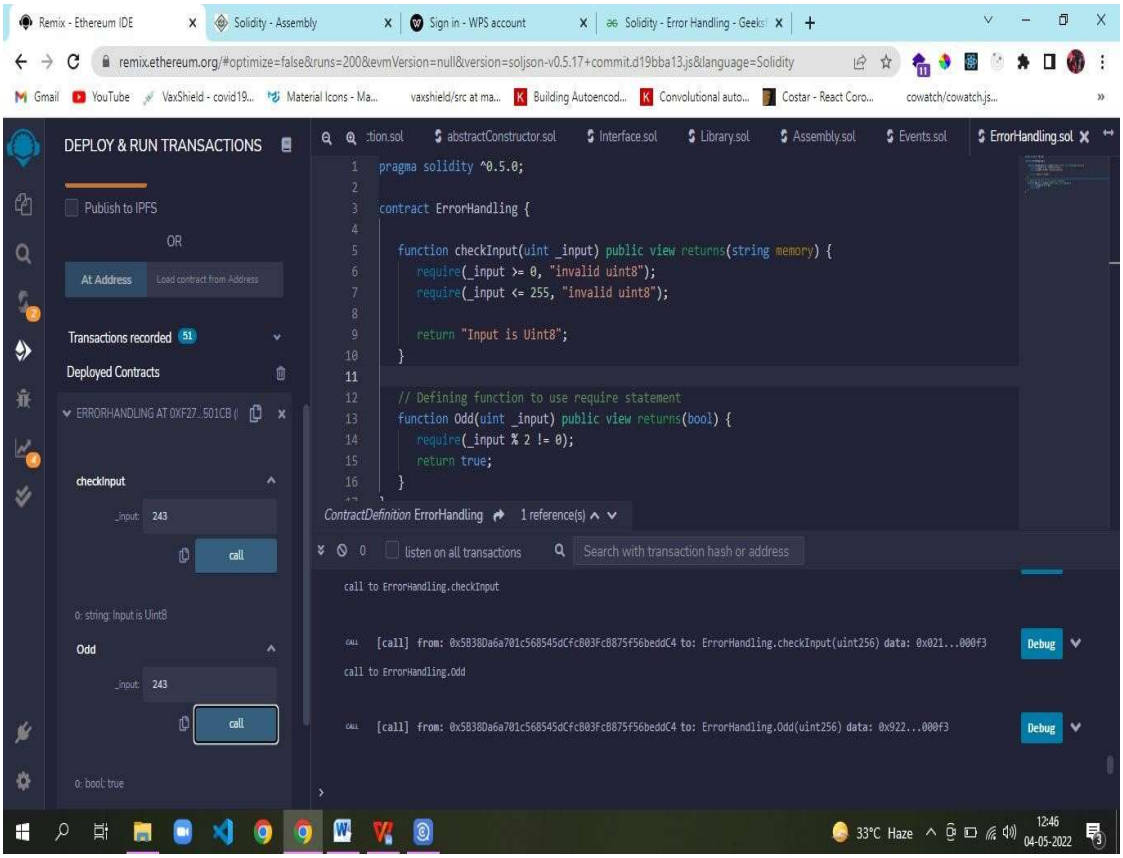
Outut :

Error handling :

Code :


Remix - Ethereum IDE
Solidity - Assembly
Sign in - WPS account
Solidity - Error Handling - Geeks
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.17+commit.d19bba13.js&language=Solidity
FILE EXPLORERS
FunctionOverloading.sol
MathematicalFunction.sol
CryptograhicFunction.sol
Function.sol
string.sol
ArithmeticOperators.sol
RelationalOperator.sol
LogicalOperator.sol
BitwiseOperator.sol
AssignmentOperator.sol
forLoop.sol
ifStatement.sol
ifElseStatement.sol
ifElseIfStatement.sol
withdrawalPattern.sol
accessRestriction.sol
Contracts.sol
Inheritance.sol
abstractConstructor.sol
Constructor.sol
Events.sol
ErrorHandling.sol
Interface.sol
Library.sol
Assembly.sol
scripts
tests
deps
README.txt
abstractConstructor.sol
Interface.sol
Library.sol
Assembly.sol
Events.sol
ErrorHandling.sol
pragma solidity ^0.5.0;
contract ErrorHandling {
function checkInput(uint _input) public view returns(string memory) {
require(_input >= 0, "invalid uint8");
require(_input <= 255, "invalid uint8");
return "Input is Uint8";
}
// Defining function to use require statement
function Odd(uint _input) public view returns(bool) {
require(_input % 2 != 0);
return true;
}
}
listen on all transactions
Search with transaction hash or address
33°C Haze
12:48
04-05-2022

Output :


Remix - Ethereum IDE
Solidity - Assembly
Sign in - WPS account
Solidity - Error Handling - Geeks
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.17+commit.d19bba13.js&language=Solidity
DEPLOY & RUN TRANSACTIONS
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 51
Deployed Contracts
ERRORHANDLING AT 0XF27...501CB
checkInput
_input: 243
call
0: string: Input is Uint8
Odd
_input: 243
call
0: bool: true
abstractConstructor.sol
Interface.sol
Library.sol
Assembly.sol
Events.sol
ErrorHandling.sol
pragma solidity ^0.5.0;
contract ErrorHandling {
function checkInput(uint _input) public view returns(string memory) {
require(_input >= 0, "invalid uint8");
require(_input <= 255, "invalid uint8");
return "Input is Uint8";
}
// Defining function to use require statement
function Odd(uint _input) public view returns(bool) {
require(_input % 2 != 0);
return true;
}
ContractDefinition ErrorHandling 1 reference(s)
listen on all transactions
Search with transaction hash or address
call to ErrorHandling.checkInput
call [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: ErrorHandling.checkInput(uint256) data: 0x021...000f3
Debug
call to ErrorHandling.Odd
call [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: ErrorHandling.Odd(uint256) data: 0x922...000f3
Debug
33°C Haze
12:46
04-05-2022