

**St. Francis Institute of Technology, Mumbai-400 103**  
**Department Of Information Technology**

A.Y. 2025-2026  
**Class: BE-ITA/B, Semester: VIII**  
Subject: BlockChain Lab

### **Experiment 4**

**1. Aim:** To implement smart contract using Remix IDE and deploy it using Ganache and MetaMask.

**2. Objective:** To ...

- explore working of Ganache ethernet network.
- test a smart contract by using Remix IDE

**3. Lab outcome:** After performing the experiment, the students will be able to **implement** smart contracts in Ethereum using different development frameworks (PO3, PSO2, BL3)

**4. Prerequisite:**

- Fundamental knowledge of blockchain
- Knowledge of the Ethereum platform and Remix IDE
- Familiarity with the Solidity programming language

**5. Requirements:** The following are the requirements –

Remix IDE, Ganache Provider, MetaMask wallet etc.

### **6. Pre-Experiment Theory:**

What is Ganache?

Ganache is a private Ethereum blockchain development environment that allows you to emulate the Ethereum blockchain so that you can interact with smart contracts in your own private blockchain. You can use Ganache across the entire development cycle; enabling you to develop, deploy, and test your dApps (decentralized Applications) in a safe and deterministic environment.

### **7. Laboratory Exercise**

#### **A. Steps to be implemented.**

To Follow the procedure given below to build smart contract in Remix IDE

- Download and install Ganache from <https://truffleframework.com/ganache>.
- Download and install the official MetaMask extension/ plugin.
- Open Remix IDE in Google Chrome.
- Write smart contract by creating new file under contract folder, with .sol extension.
- Write your contract code using solidity language.
- Click on compiler icon, choose compiler version or keep default and click on compile button.
- For deployment...

**Part A)** First start Ganache provider and then in Remix IDE, choose environment Ganache Provider and then deploy the said contract.

**Part B)** Set-up Ganache with MetaMask and then in Remix IDE choose environment Injected Provider- MetaMask and then deploy the said contract.

- Under deployed contract, get the output of your contract.

**B. Program Code**

Write a smart contract StudentMarks.sol to create student database with attributes, id, name, dept, sub1marks, sub2Marks etc. and deploy it using Ganache and MetaMask.

**8. Post Experimental Exercise-****A. Questions:**

1. List down the details of Ganache IDE
2. Create any smart contract that you have not done so far and deploy it.

**B. Results/Observations/Program output:**

Present the program input/output results if any and comment on the same.

**C. Conclusion:**

Write what was performed in the experiment.

Write which tools you used to perform the experiment

Write what you inferred from the output obtained.

**D. References:**

<https://www.geeksforgeeks.org/how-to-use-metamask-to-deploy-a-smart-contract-in-solidity-blockchain/>

<https://www.geeksforgeeks.org/how-to-set-up-ganache-with-metamask/>

Mastering Ethereum, Building Smart Contract and Dapps, Andreas M. Antonopoulos Dr. Gavin Wood, O'reilly

REMX 1.5.1

DEPLOY & RUN TRANSACTIONS

Transactions recorded 2 i

Deployed Contracts 1

STORAGE AT 0x8C...993E3 (B)

Balance: 0 ETH

STORE

num: 100

retrieveretrieve - call

e: uint256: 100

Low level interactions

CALLDATA

Transact

[block:1 txIndex:0] from: 0xfbe...a71e4 to: Storage.(constructor) value: 0 wei data: 0x60...f0833 logs: 0 hash: 0x240...c3307 [Verification] Contract deployed. Checking explorers for registration... Etherscan verification skipped: API key not provided. Please input the API Key in Remix Settings - Connected Services OR Contract Verification Plugin Settings. [Sourceify] Verification Error: Chain 1337 not found [Sourceify] Please open the "Contract Verification" plugin to retry. transact to Storage.store pending ...

[block:2 txIndex:0] from: 0xfbe...a71e4 to: Storage.store(uint256) 0x8C...993E3 value: 0 wei data: 0x60...00864 logs: 0 hash: 0x28e...0c749 call to storage.retrieve

[call] from: 0xfbe...a71e4 to: Storage.retrieve() data: 0x26...4ec1

Debug

REMIXAI ASSISTANT

RemixAI provides you personalized guidance as you build. It can break down concepts, answer questions about blockchain technology and assist you with your smart contracts.

List some gas saving techniques

Show a sybil-resistant voting contract

Select Context Ask Edit AI Beta

Select context and ask me anything!

MistralAI Audio Prompt Create new workspace with AI

Scan Alert Initialize as git repo Did you know? You can learn Solidity basics and more using the LearnETH plugin.

Search

ENG IN 08:41 04-02-2026

ENVIRONMENT

Dev - Ganache Provider

Geth Testnet (1337) network

ACCOUNT 0x277...a38cd (100.0 ETH)

GAS LIMIT

Estimated Gas

Custom 3000000

VALUE 0 Wei

CONTRACT Storage - contracts/1\_Storage.sol

evm version: london

Verify Contract on Explorers

Deploy & Verify

ENVIRONMENT

Dev - Ganache Provider

Geth Testnet (1337) network

ACCOUNT 0fbe...a71e4 (99.99981823235)

GAS LIMIT

Estimated Gas

Custom 3000000

ENVIRONMENT

Injected Provider - MetaMask

Geth Testnet (1337) network

ACCOUNT 0fbe...a71e4 (99.99643029235)

GAS LIMIT

Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS

CURRENT BLOCK 2 GAS PRICE 20000000000 GAS LIMIT 6721975 HARDFORK MERGE NETWORK ID 5777 RPC SERVER HTTP://127.0.0.1:7545 MINING STATUS AUTOMINING

BLOCK	MINED ON	GAS USED	
2	2026-02-04 09:38:11	43724	1 TRANSACTION
1	2026-02-04 09:37:54	125673	1 TRANSACTION
0	2026-02-04 09:29:03	0	NO TRANSACTIONS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

TX HASH 0x1bd43f69f779e8aeb6fa60ef63541c085e22a504905800b4daed29292ed9f7bd

FROM ADDRESS 0x277EDC51b3222D917271bf8cfB3E0dc3853A38cD TO CONTRACT ADDRESS 0x56f7c0Eed209F04D8346D0BD84d81970C316A206 GAS USED 43724 VALUE 0

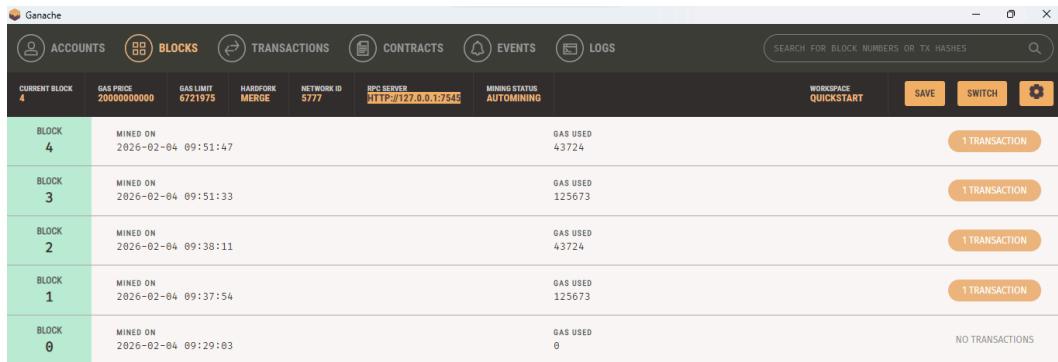
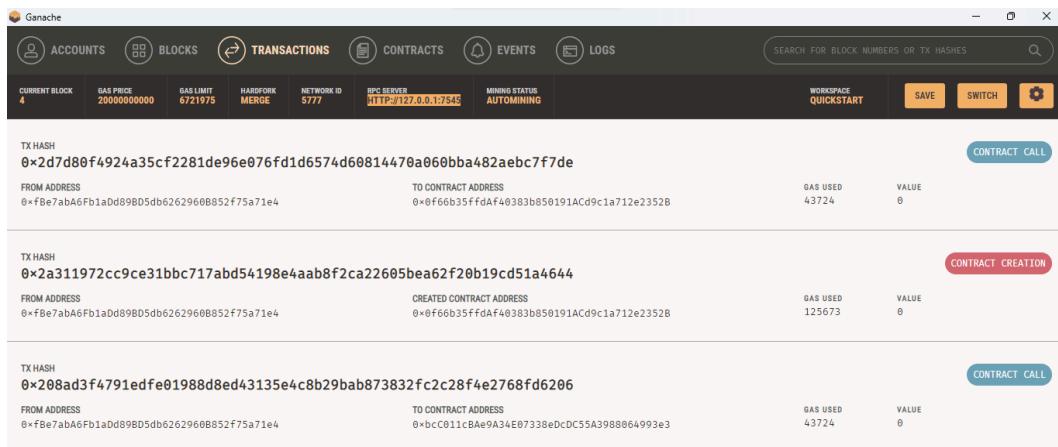
TX HASH 0x03345a0ae0d803378e0f033644c4f7f187a3edb270241a21cb9eab53f67d0cff

FROM ADDRESS 0x277EDC51b3222D917271bf8cfB3E0dc3853A38cD CREATED CONTRACT ADDRESS 0x56f7c0Eed209F04D8346D0BD84d81970C316A206 GAS USED 125673 VALUE 0

CONTRACT CALL

CONTRACT CREATION

## Metamask-ganache setup

## Program Code-

**Write a smart contract StudentMarks.sol to create student database with attributes, id,**

**name, dept, sub1marks, sub2Marks etc**

**Code-**

/ SPDX-License-Identifier: MIT	uint256 _sub1Marks,
--------------------------------	---------------------

```

pragma solidity ^0.8.0;
contract StudentMarks {
    struct Student {
        uint256 id;
        string name;
        string dept;
        uint256 sub1Marks;
        uint256 sub2Marks;
    }
    Student[] private students;
    mapping(uint256 => Student) private studentById;

    function addStudent(
        uint256 _id,
        string memory _name,
        string memory _dept,
        uint256 _sub1Marks,
        uint256 _sub2Marks
    ) public {
        require(_sub1Marks <= 100 && _sub2Marks <= 100, "Marks cannot exceed 100");
        Student memory newStudent = Student({
            id: _id,
            name: _name,
            dept: _dept,
            sub1Marks: _sub1Marks,
            sub2Marks: _sub2Marks
        });
        students.push(newStudent);
        studentById[_id] = newStudent;
    }

    function getStudent(uint256 _id) public view
    returns (Student memory) {
        return studentById[_id];
    }
}

```

The screenshot displays the development environment for Ethereum smart contracts:

- Remix IDE (Left):** Shows the Solidity code for the `StudentMarks` contract. It includes a struct `Student` with fields `id`, `name`, `dept`, `sub1Marks`, and `sub2Marks`. The contract has a private array `students` and a mapping `studentById`. A `function addStudent` is defined to add a new student to the array, validate marks (both must be ≤ 100), and update the mapping. A `getStudent` function is also provided.
- MetaMask Extension (Right):** Shows a `Transaction request` dialog. It indicates the transaction is from `remix.ethereum.org` to `0x469A3..cEfEf` on the `Ganache` network. The network fee is `0.0054 ETH`.
- Ganache (Bottom):** Shows the details of Block 6. It includes the block number (6), current block (6), gas price (20000000000), gas limit (6721975), hardfork merge (5777), network ID (5777), RPC server (HTTP://127.0.0.1:7545), mining status (AUTOMINING), workspace quickstart, save, switch, and settings buttons. The block details show `GAS USED: 270712`, `GAS LIMIT: 6721975`, `MINED ON: 2026-02-04 10:01:07`, and `BLOCK HASH: 0xca9957bc42799cb27d3fe60c89a9dd1544137433ef24b180b32951e9b9abfa8a`. The transaction hash is `0x15aa1f01ed72f758adff4b1e28b29241c8a3467593dd941b6c6e2d733f6d6e4f`.