

 Hello Solidity – Writing First Smart Contract

**Objective/Aim:**  
  
 To understand the basics of writing and deploying a first smart contract in Solidity using the Remix Ethereum IDE.

**Apparatus/Software Used:**

* Laptop
* Web browser
* Remix Ethereum IDE
* MetaMask Wallet

**Theory/Concept:**

**What is a Smart Contract?**

A **smart contract** is a **self-operating program** stored on a blockchain that runs exactly as programmed

without the need for intermediaries.

**Key Features of Smart Contracts**

1. **Automation** – Executes automatically when predefined conditions are met.
2. **Trustless** – No central authority or middleman is needed; trust is in the code.
3. **Immutable** – Cannot be altered after deployment.
4. **Transparent** – Code and transactions are publicly verifiable.
5. **Secure** – Protected by cryptographic principles and blockchain consensus.

**What is solidity?**

Solidity is a high-level, contract-oriented programming language used to write smart contracts.

**Structure of a Solidity Smart Contract:**

1. **SPDX License Identifier** – Declares licensing information.
2. **Pragma Directive** – Specifies the compiler version.
3. **Contract Definition** – The actual smart contract logic.
4. **State Variables** – Data stored on the blockchain.
5. **Functions** – Define behavior and actions.



**Procedure:**

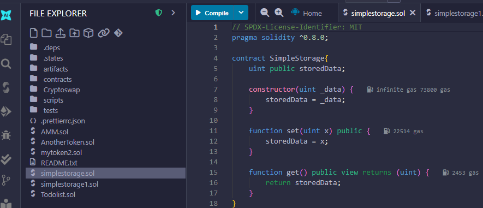
|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

A black screen with white text

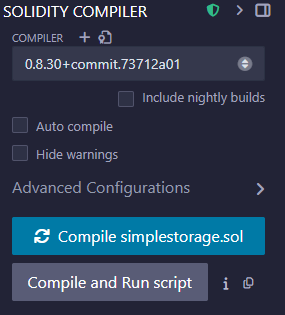
AI-generated content may be incorrect.**Step 1: Launch the Remix Ethereum IDE in your browser.**

**Step 2: In the “contracts” folder, create a new file named Simplestorage.sol**

**Step 3: Then write the code for the simple storage system**

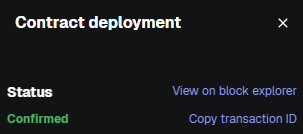
****

**Step 4: Compile the Contract:**

****

**Step 5: Deploy the Contract Using Injected Provider:**

**Step 6: Confirm the transaction and the first smart contract have been deployed**

****

**Observation Table:**



We have successfully taken a smart contract written in Solidity, compiled it into bytecode, and submitted it to the

blockchain network (testnet or mainnet) through a deployment transaction. Once the transaction was mined, the

contract became permanently stored on the blockchain, assigned a unique address, and is now ready for interaction

**Transaction**

nounce:36

Gas limit: 158793

Gas used: 157585

Base fee: 0.026

Total:0.00024