

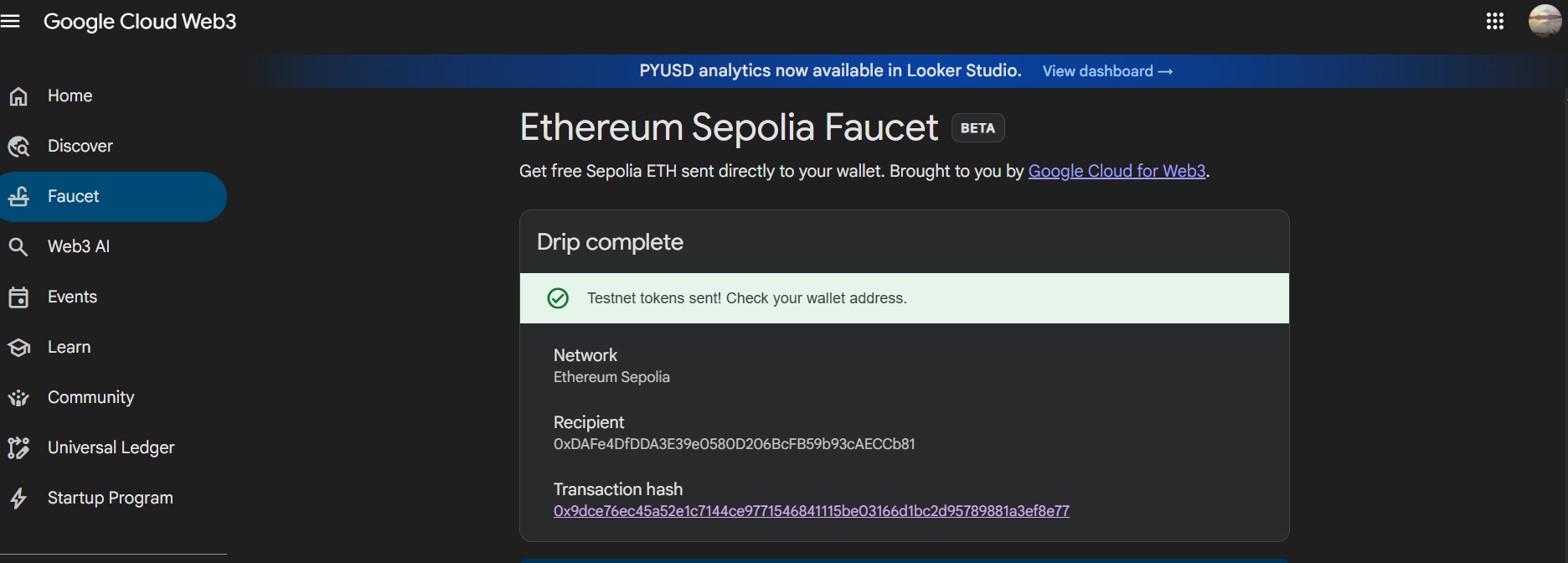
 Hello Solidity – Writing First Smart Contract  
  
**Objective/Aim:**

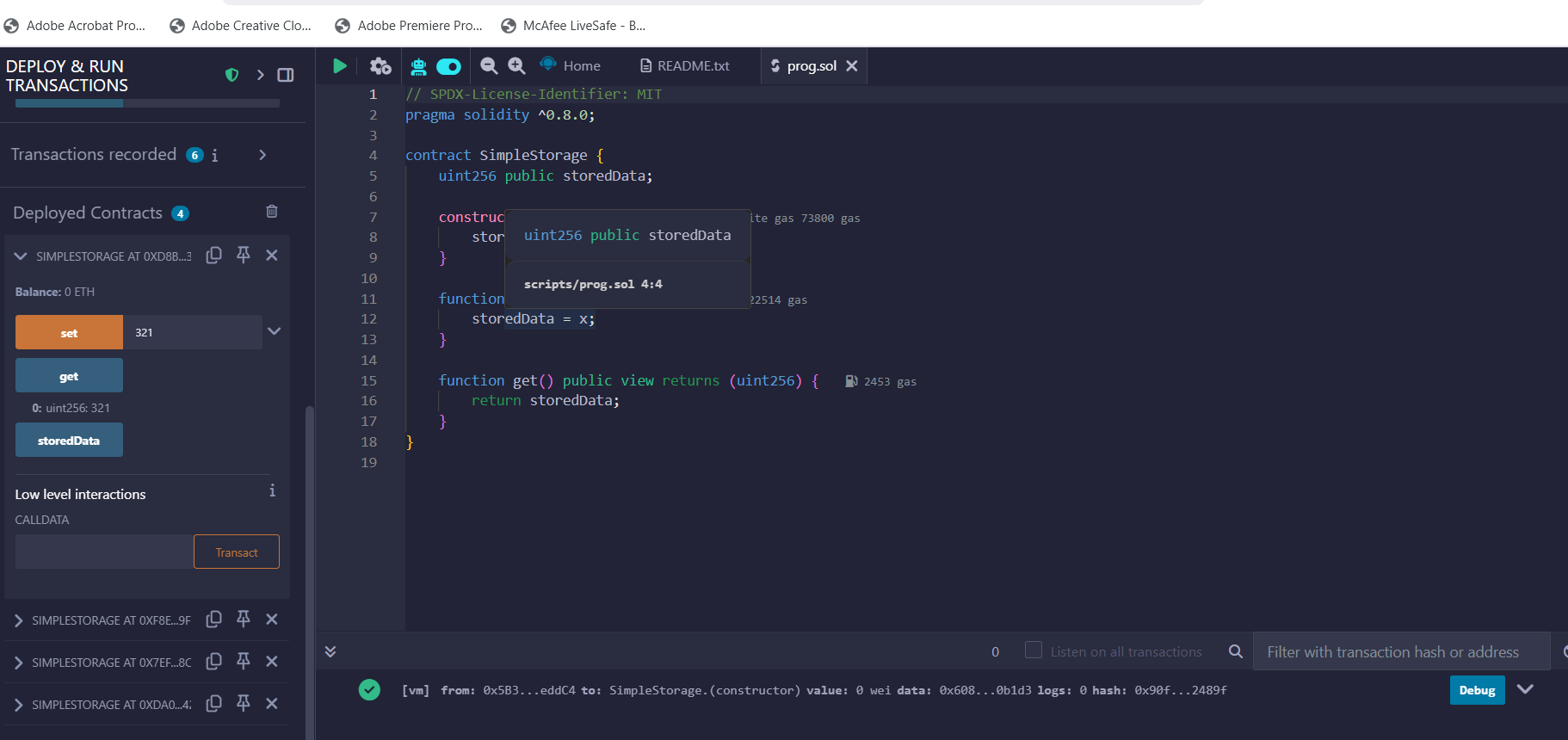
**Apparatus/Software Used:**

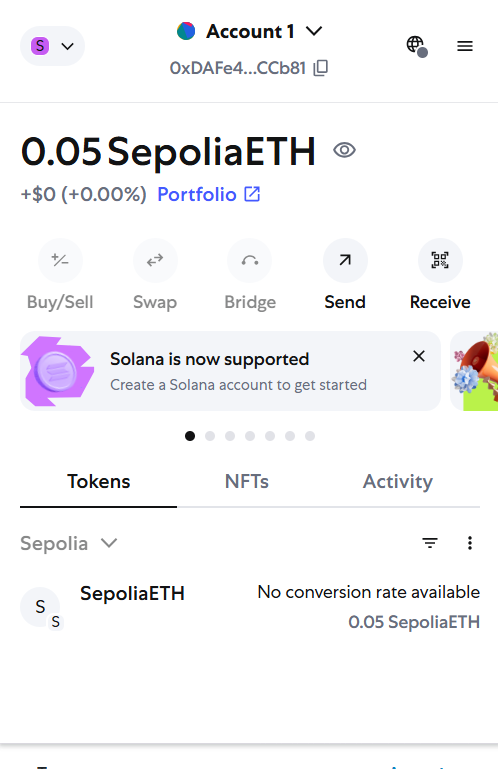
* Laptop/PC
* Remix IDE
* Ethereum cloud
* Sepolia tesnet
* Internet for research

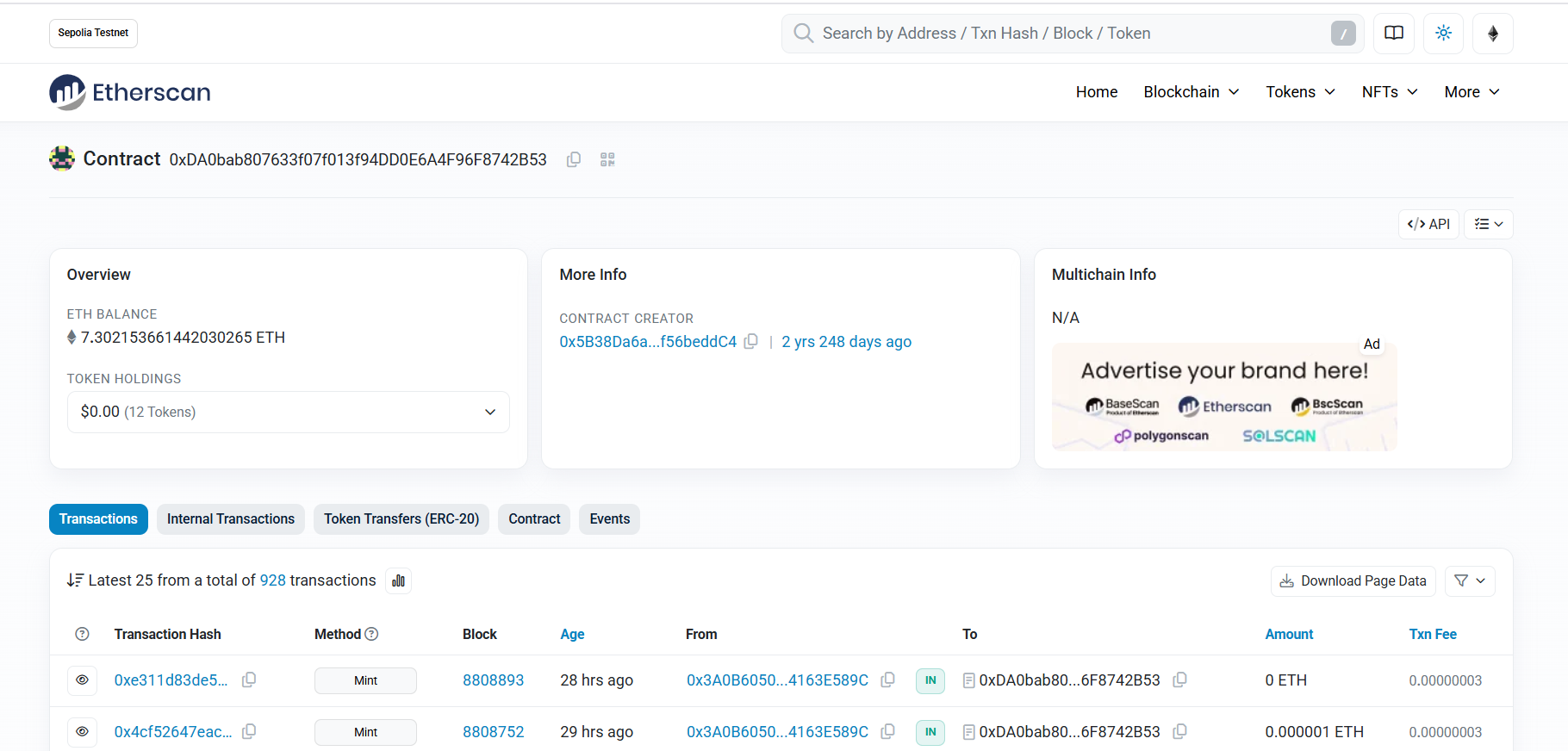
**Theory/Concept:**

1. **Solidity** is a programming language for implementing smart contracts on various blockchain platforms, most notably, Ethereum. Solidity is licensed under GNU General Public License v3.0. Solidity was designed by Gavin Woodand developed by Christian Reitwiessner, Alex Beregszaszi, and several former Ethereum core contributors. Programs in Solidity run on Ethereum Virtual Machine or on compatible virtual machines
2. A **smart contract** is a computer program or a transaction protocol that is intended to automatically execute, control or document events and actions according to the terms of a contract or an agreement.











**Procedure:**

1. Set up your development environment.
2. create a new Solidity file.
3. define the contract with a pragma statement.
4. add state variables and functions.
5. compile the contract, and deploy it to the blockchain.

**Observation Table:**

1. When writing your first smart contract, it's also important to consider best practices such as using SPDX license identifiers and specifying the compiler version to ensure compatibility. Furthermore, events can be used to log actions on the blockchain, making it easier to track changes and interactions.
2. Overall, writing your first smart contract involves a combination of understanding the language syntax, best practices, and the specific requirements of the contract you are building.



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



