

# Introduction to Solidity: Coding Ethereum Smart Contracts



# Session Four Scope

- **Scope**

Learn about ERC20 and understand additional programming nuances

- **What you will know**

What ERC20 is, know what constructors are, know what mappings are, know what events are

- **Next Steps: Session Five**

Using session four knowledge we write a small ERC 20 contract and deploy it on metamask test network

# Agenda



[Session Four Prerequisites](#)



[Additional Solidity Nuances](#)



[ERC20 Structure](#)



[ERC20 Code Review](#)



[ERC20 Builder Exercise](#)



[Appendix: Additional Resources](#)



[Solution Set](#)

# Session Four



## Session Four Prerequisites

- Google Chrome
- Remix - <https://remix.ethereum.org/>

# Additional Solidity Nuances

## Additional Solidity Nuances

- *Constructors*
- *Mappings*
- *Events*
- *Interface*

# ERC20 Structure



What is an ERC20 Token?

*ETHEREUM REQUEST FOR COMMENT 20*

## ERC20 Core Functions

- *transfer*
- *balanceOf*
- *transferFrom*
- *approve*
- *mappings*

## ERC20 Examples

- *AAVE*
- *USDT*
- *DAI*

## ERC20 v. ERC777 v. ERC677?

- *Core difference is that later ERC standards have additional functionalities but are still backwards compatible*

## Why ERC20?


- *Governance*
- *Secure a Network*
- *Synthetic Assets and Stable Coins*
- *Easy and uniform token creation*

# ERC20 Code Review

## Github Repository

- <https://github.com/PatrickAlphaC/erc20-brownie/blob/main/contracts/TokenERC20.sol>

*\* Code Review based on repository of Patrick Collins at Chainlink*



## Appendix: Additional Learning Resources

- *Tutorialspoint* -  
<https://www.tutorialspoint.com/solidity/index.htm>
- *ERC20 Tutorial* -  
<https://betterprogramming.pub/python-blockchain-token-deployment-tutorial-create-an-erc20-77a5fd2e1a58>