

Exercise – KSU Token

Build your ERC-20 Token Smart Contract

Here is the complete ERC-20 token smart contract Solidity code: [download](#)

Let's take a look at what this smart contract does, and how it implements the ERC-20 standard:

- It stores the token name `string public name = "KSU Token"`.
- It stores the token symbol for cryptocurrency exchanges `string public symbol = "KCoin"`.
- It stores the total supply of tokens in existence `uint256 public totalSupply`.
- It uses a Solidity mapping to store the balance of each account that owns tokens `mapping(address => uint256) public balanceOf`.
- It implements a `transfer` function to allow users to send tokens to another account.
- It implements an `approve` function that allows another account to spend tokens, like on a cryptocurrency exchange. This updates the `allowance` mapping to see how much the account is allowed to spend.
- It implements a `transferFrom` that allows another account to transfer tokens.

To do:

- 1) Use [Remix](#) to deploy your contract token to Ropsten network.
- 2) Initialize total supply token when you deploy your contract
- 3) Transfer tokens to a few accounts
- 4) Observe transfers and account holders on [Etherscan](#)
- 5) Get yourself familiar with all ERC-20 functions (if you've forgotten what they do—refer to this [guide](#))

*Feel free to rename the token