Hack the North Workshop

...

CONSENSYS

Everyone’s First Dapp Tutorial: Pet Shop

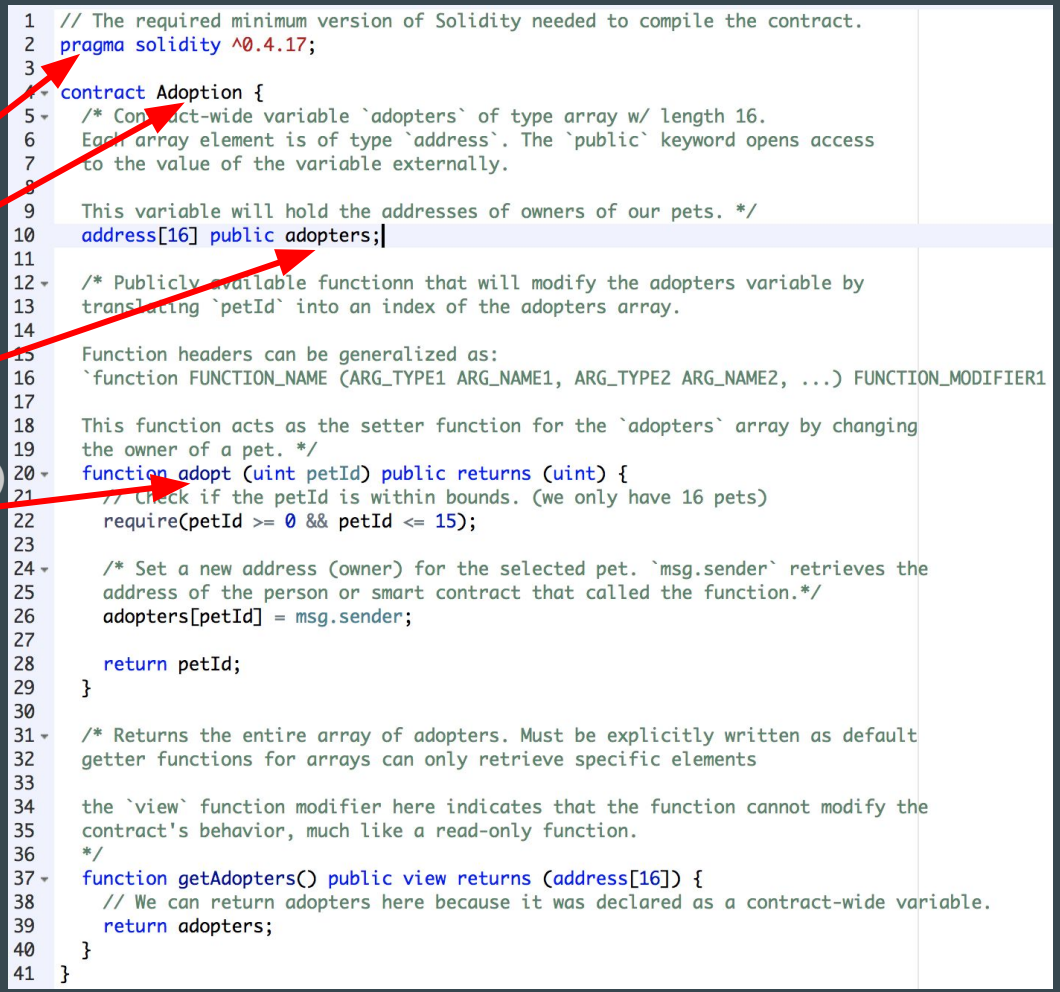
https://trufflframework.com/tutorials/pet-shop

Recommended Prereqs

- Basic knowledge about JS and React
  - https://github.com/workshopper/javascripting
- Basic knowledge about blockchains (preferrably Ethereum)
  - https://blockgeeks.com/guides/ethereum/
- Basic knowledge on command lines (no shame)
  - https://lifehacker.com/5633909/who-needs-a-mouse-learn-to-use-the-command-line-for-almost-anything

Smart Contracts (Solidity)

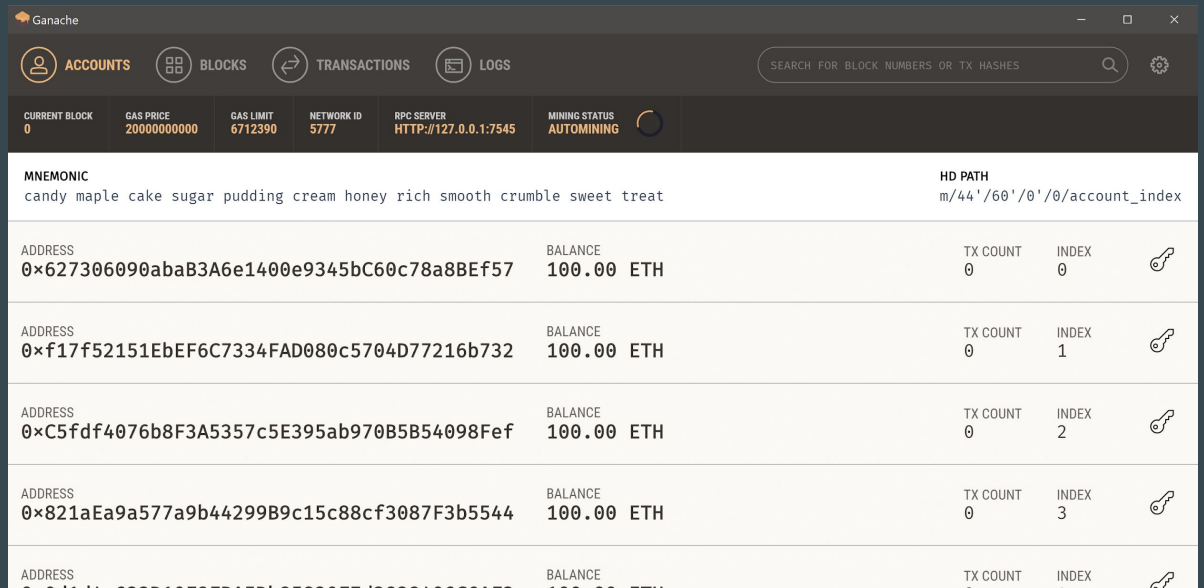
- Specify version of solidity
- Import other contracts (optional)
- Name your contract
- Generate your variables
- Write your constructor (optional)
- Write a function
- Use of modifiers (optional)
- User of events (optional)



```
1  // The required minimum version of Solidity needed to compile the contract.
2  pragma solidity ^0.4.17;
3
4  contract Adoption {
5    /* Contract-wide variable `adopters` of type array w/ length 16.
6    Each array element is of type `address`. The `public` keyword opens access
7    to the value of the variable externally.
8
9    This variable will hold the addresses of owners of our pets. */
10   address[16] public adopters;
11
12   /* Publicly available functionn that will modify the adopters variable by
13   translating `petId` into an index of the adopters array.
14
15   Function headers can be generalized as:
16   `function FUNCTION_NAME (ARG_TYPE1 ARG_NAME1, ARG_TYPE2 ARG_NAME2, ...) FUNCTION_MODIFIER1
17
18   This function acts as the setter function for the `adopters` array by changing
19   the owner of a pet. */
20   function adopt (uint petId) public returns (uint) {
21     // Check if the petId is within bounds. (we only have 16 pets)
22     require(petId >= 0 && petId <= 15);
23
24     /* Set a new address (owner) for the selected pet. `msg.sender` retrieves the
25     address of the person or smart contract that called the function.*/
26     adopters[petId] = msg.sender;
27
28     return petId;
29   }
30
31   /* Returns the entire array of adopters. Must be explicitly written as default
32   getter functions for arrays can only retrieve specific elements
33
34   the `view` function modifier here indicates that the function cannot modify the
35   contract's behavior, much like a read-only function.
36   */
37   function getAdopters() public view returns (address[16]) {
38     // We can return adopters here because it was declared as a contract-wide variable.
39     return adopters;
40   }
41 }
```

Ganache (Local Testnet)

- Launch Ganache (local JavaScript Ethereum test client)
- Block Explorer (Watch your transactions flow through)
- Automining (No need to wait for confirmation)
- Logs



Truffle Framework (Compile, Deploy, Test)

- Compile your smart contract (./build/contracts)
  - Compiles solidity code into bytecode and stores it in json format
- Deploy your smart contract
  - Using truffle.js, it deploys the smart contract to your specified blockchain (local/network)
  - Deploy scripts can be configured in ./migrations

```
Kevins-MacBook-Pro-2:annotated-pet-shop kevin$ truffle compile
Compiling ./contracts/Adoption.sol...
Compiling ./contracts/Migrations.sol...
Writing artifacts to ./build/contracts
```

```
Kevins-MacBook-Pro-2:annotated-pet-shop kevin$ truffle migrate --reset
Using network 'development'.

Running migration: 1_initial_migration.js
  Deploying Migrations...
  ... 0xc6a2122da1693e7f671d5baf05ae7b393588440672f06603d59c3415c7d0cc6f
  Migrations: 0x076feabc7e55f360c225ca9d16dbc2a1afbc0291
Saving successful migration to network...
  ... 0x3b5997a36fb5d4c6279f828ad09fff29de3149d4d0d330603ca1722a7fd3a4d5
Saving artifacts...
Running migration: 2_deploy_contracts.js
  Deploying Adoption...
  ... 0xd2c841f225d2b369d3a342dad286457662b9c2f330b6e9c1625b770d38822ed1
  Adoption: 0x32313ab4aee76a119ba12b2981996795bc0916e2
Saving successful migration to network...
  ... 0xadee2c2cdab4d389ce5ce87da1bff8ff497fec2c9d8de5706cda95eec96ebf12
Saving artifacts...
```

Connect your Frontend (using TruffleContract)

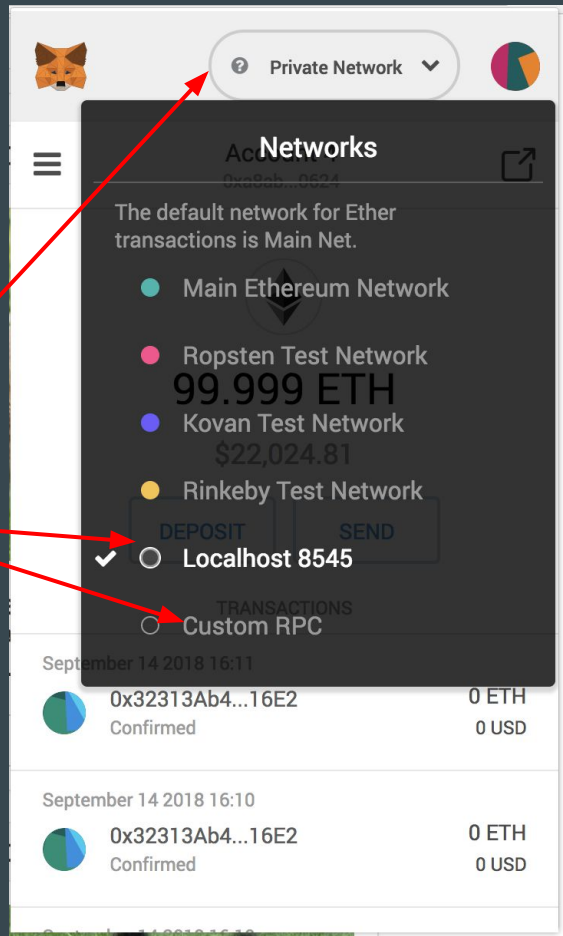
- Initialize a web3 instance
  - Either use MetaMask's window.web3 or
  - Use a local provider (localhost)
- Initialize the contract instance
  - Json file from "truffle compile"
  - Initialize the contract instance
  - Point the contract instance to web3
  - Call a function in the contract

```
initWeb3: function() {

  // WARNING: only usable until 11/2/18, by which developers will need to adhere to
  // new EIP-1102 standards:
  // https://our.status.im/breaking-change-to-the-status-browser/
  // https://medium.com/metamask/https-medium-com-metamask-breaking-change-injecting-web3-7722797916a8

  // Is there an injected web3 instance?
  if (typeof web3 !== 'undefined') {
    App.web3Provider = web3.currentProvider;
  } else {
    // If no injected web3 instance is detected, fall back to Ganache
    App.web3Provider = new Web3.providers.HttpProvider('http://localhost:7545');
  }

  // Regardless, set the web3 variable with what we have.
  web3 = new Web3(App.web3Provider);

  return App.initContract();
},

initContract: function() {
  $.getJSON('Adoption.json', function(data) {
    // Get the necessary contract artifact file and instantiate it with truffle-contract
    var AdoptionArtifact = data;
    App.contracts.Adoption = TruffleContract(AdoptionArtifact);

    // Set the provider for our contract
    App.contracts.Adoption.setProvider(App.web3Provider);

    // Use our contract to retrieve and mark the adopted pets
    return App.markAdopted();

  })

  return App.bindEvents();
},
```

Configuring MetaMask

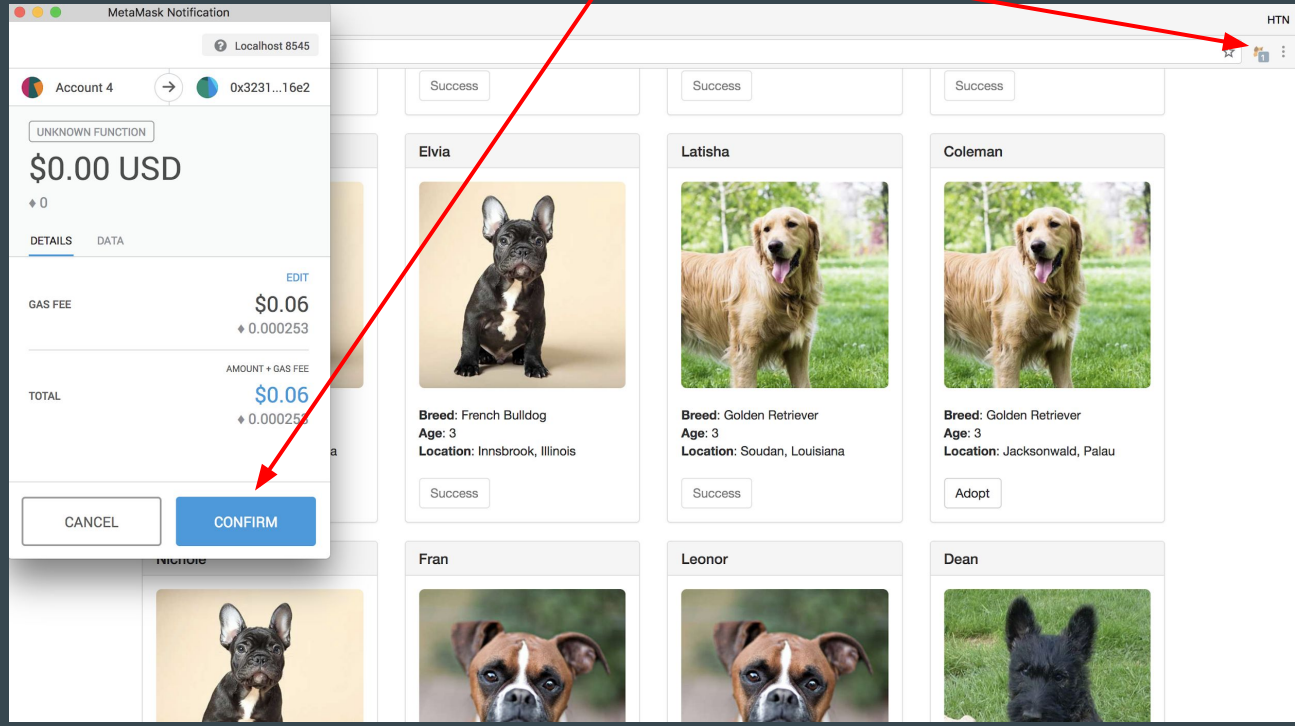
- Import your mnemonic from Ganache and import as seed

- Connect MetaMask to Ganache local dev environment
  - Select Private Network in the dropdown
  - Select Localhost 8545
    - Or Custom RPC if you used a different port for Ganache



Interacting with your Smart Contract (MetaMask)

- Use MetaMask to interface with your smart contract



Caveats

- When you change anything in your smart contract, you will need to redeploy it
- Be aware that documentation you are reading now might be outdated
- web3.js v0.x vs. web3.js v1.x
- Make sure you are on the right network when deploying and testing

Practice

https://truffleframework.com/boxes/pet-shop

https://truffleframework.com/tutorials/pet-shop

https://github.com/Zanibas/annotated-pet-shop

https://github.com/consensys