



Step1: Open Ganache

```

Microsoft Windows [Version 10.0.18363.1139]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\win>cd Desktop/token_sale

C:\Users\win\Desktop\token_sale>truffle migrate --reset

Compiling your contracts...
=====
> Compiling .\contracts\CryptoToken.sol
> Compiling .\contracts\Migrations.sol
> Artifacts written to C:\Users\win\Desktop\token_sale\build\contracts
> Compiled successfully using:
   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang

Starting migrations...
=====
> Network name:      'development'
> Network id:       5777
> Block gas limit: 6721975 (0x6691b7)

1_initial_migration.js
=====

   Deploying 'Migrations'
   -----
   > Blocks: 0           Seconds: 0
   > contract address:   0xEc2f97b3Cec336D998A3A2FEC55ffCDff8a3Af85
   > block number:      1
   > block timestamp:    1603002142
   > account:           0xbD8d88Bac5578EdcDC24cEb33B722Cacd2453380b
   > balance:           99.99616114
   > gas used:          191943 (0x2edc7)
   > gas price:          20 gwei
   > value sent:         0 ETH
   > total cost:         0.00383886 ETH

   > Saving migration to chain.
   > Saving artifacts
   -----
   > Total cost:         0.00383886 ETH

```

Step 2: Initial Migration

Command Prompt

2_deploy_contracts.js

=====

Replacing 'CryptoToken'

> transaction hash: 0x457adbc1b6e75067de1435346e27b104799e0b4e35cf75c629112480ab541b82
> Blocks: 0 Seconds: 0
> contract address: 0x4355f492e76332e3745FE896632f689d413c6bBC
> block number: 70
> block timestamp: 1603277902
> account: 0x1c30b97B3a163edd8E163f99cC3D30F2A8f4eaAD
> balance: 99.77379928
> gas used: 795083 (0xc21cb)
> gas price: 20 gwei
> value sent: 0 ETH
> total cost: 0.01590166 ETH

> Saving migration to chain.

> Saving artifacts

> Total cost: 0.01590166 ETH

Summary

=====

> Total deployments: 2
> Final cost: 0.01974052 ETH

C:\Users\win\Desktop\token_sale>

Step 3: Deploy contracts

Command Prompt - truffle console

```
truffle(development)> tokenInstance.name();
'Crypto Token'
truffle(development)> tokenInstance.symbol();
'Crypto'
truffle(development)> tokenInstance.standard();
'Crypto Token v1.0'
truffle(development)> tokenInstance.totalSupply();
BN {
  negative: 0,
  words: [ 1000000, <1 empty item> ],
  length: 1,
  red: null
}
truffle(development)> tokenInstance.totalSupply().then(function(s){supply=s;})
undefined
truffle(development)> supply
BN {
  negative: 0,
  words: [ 1000000, <1 empty item> ],
  length: 1,
  red: null
}
truffle(development)> supply.toNumber()
1000000
truffle(development)>
```

Step 4: token Instantiation

```
Command Prompt
C:\Users\win\Desktop\token_sale>truffle test
Using network 'development'.

Compiling your contracts...
=====
> Everything is up to date, there is nothing to compile.

Contract: CryptoToken
  ✓ initializes the contract with the correct values (304ms)
  ✓ allocates the initial supply upon deployment (174ms)
  ✓ transfers token ownership (490ms)
  ✓ approves tokens for delegated transfer (362ms)
  ✓ handles delegated token transfers (1363ms)

5 passing (3s)

C:\Users\win\Desktop\token_sale>_
```

Step 5: Smoke Tests for Crypto Token

```
Command Prompt
Using network 'development'.

Compiling your contracts...
=====
> Compiling .\contracts\CryptoToken.sol
> Compiling .\contracts\CryptoTokenSale.sol
> Artifacts written to C:\Users\win\AppData\Local\Temp\test--7964-n5oVt10UNDJY
> Compiled successfully using:
  - solc: 0.5.16+commit.9c3226ce.Emscripten.clang

Contract: CryptoToken
  ✓ initializes the contract with the correct values (254ms)
  ✓ allocates the initial supply upon deployment (160ms)
  ✓ transfers token ownership (461ms)
  ✓ approves tokens for delegated transfer (351ms)
  ✓ handles delegated token transfers (1391ms)

Contract: CryptoTokenSale
  ✓ initializes the contract with the correct values (176ms)
  ✓ facilitates token buying (942ms)
  ✓ ends token sale (659ms)

8 passing (5s)

C:\Users\win\Desktop\token_sale\test>_
```

Step 6: Smoke Tests for Crypto Token Sale