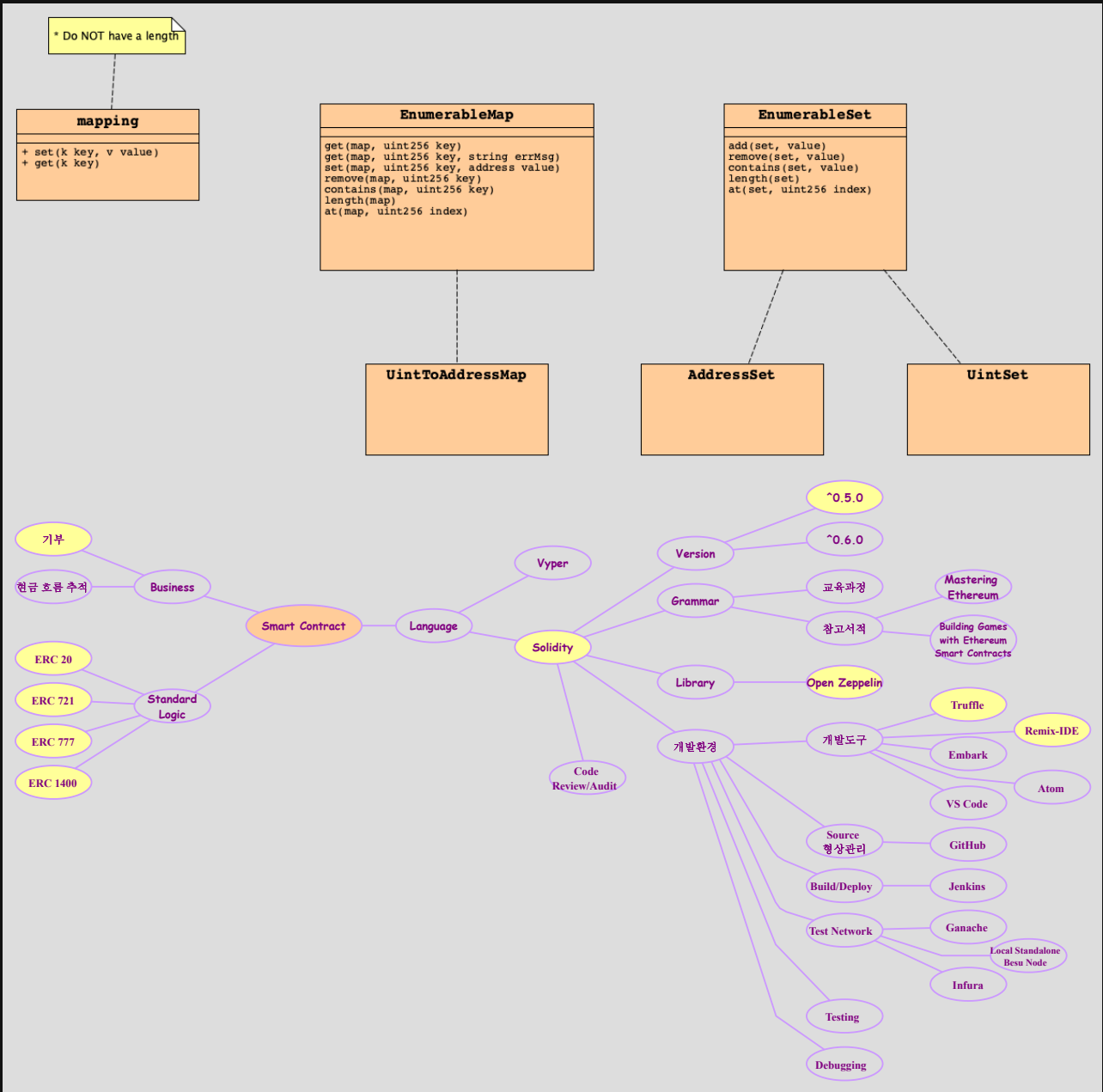


Smart Contract 개발 기본

Hello, World!

Images



DApp

DApp 개발 Lifecycle

- 응용 분석/설계
- Smart Contract 구현
- Smart Contract **Audit**
- Smart Contract **Local**배포/단위테스트
- Smart Contract **Testnet**배포/단위테스트
- 응용 구현
- 응용 배포/단위테스트

DApp 개발 환경/도구

Smart Contract

| Category | Tool/Service | Remarks |
|-------------------------|-------------------------------|------------------|
| Editing | <u>Remix IDE</u> | <u>Web based</u> |
| Build/Deploy | <u>Truffle</u> | JavaScript based |
| | <u>Browine</u> | Python based |
| Local Client | <u>Ganache</u> | Ganache CLI |
| Mainnet/Testnet Gateway | <u>Infura</u> | |
| Library | <u>OpenZeppelin Contracts</u> | |
| Block Explorer | <u>Etherscan</u> | Mainnet |
| | <u>Etherscan/Rinkeby</u> | |

Truffle

- Installing Node.js

```
$ node --version
```

- Installing Truffle

```
$ npm ls -g truffle      # check whether or not Truffle is installed in global scope
$ npm uninstall -g truffle # uninstalled currently installed Truffle
$ npm install -g truffle
```

- Creating Truffle Project

```
$ mkdir smart-contract-101 && cd smart-contract-101
$ truffle init
..
$ ls
/      ../      contracts/  migrations/  test/      truffle-config.js
$ cat truffle-config.js | less
..
```

Truffle Commands

```
$ truffle help
Truffle v5.4.22 - a development framework for Ethereum

Usage: truffle <command> [options]

Commands:
  build      Execute build pipeline (if configuration present)
  compile    Compile contract source files
  ...
  console    Run a console with contract abstractions and commands available
  ...
  deploy     (alias for migrate)
  ...
  networks   Show addresses for deployed contracts on each network
  ...
  test       Run JavaScript and Solidity tests
  ...
See more at http://trufflesuite.com/docs
```

```
$ truffle help migrate
..
$ truffle help test
..
```

- Resources
 - Truffle documentation
 - Truffle commands

Truffle Configuration

Prerequisite

- Setup **Mnemonic**

- macOS

```
$ env | grep BIP39_MNEMONIC
.. .
$ echo 'export BIP39_MNEMONIC=" ... "' >> ~/.profile
$ cat ~/.profile
.. .
```

- Windows

```
> setx BIP39_MNEMONIC=" ... "
```

- Setup **Infura Project**

- macOS

```
$ env | grep INFURA_PROJECT_ID
.. .

$ echo 'export INFURA_PROJECT_ID= ... ' >> ~/.profile
$ cat ~/.profile
.. .
```

- Windows

```
> setx INFURA_PROJECT_ID= ...
```

- Setup **Node.js** project

```
$ npm init -y
$ npm install -D @truffle/hdwallet-provider
.. .
$ cat package.json
```

Truffle Configuration

- `truffle-config.js`

```
const HDWalletProvider = require("@truffle/hdwallet-provider");
const mnemonic = process.env.BIP39_MNEMONIC;

module.exports = {

  networks: {
    development: {
      host: '127.0.0.1',
      port: 8545,
      network_id: 2016,
      gas: 3E8,
      gasPrice: 0,
      websockets: false
    },

    mainnet: {
      provider: () => new HDWalletProvider(
        mnemonic, "https://mainnet.infura.io/v3/" + process.env.INFURA_PROJECT_ID),
      network_id: '1'
    },

    rinkeby: {
      provider: () => new HDWalletProvider(
        mnemonic, "https://rinkeby.infura.io/v3/" + process.env.INFURA_PROJECT_ID),
      network_id: '4',
    }
  }
}
```

Testnets

Truffle Console / Rinkeby

JSON-RPC and Web3.js

Ganache (Ganache CLI)

- Local standalone client

