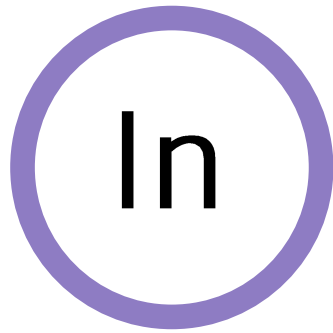


이더리움 DApp

Solidity 언어 실습





Solidity

07 DApp의

보안

대칭키, 비대칭키(암호화,
서명)

해시

PKI, SSL, TLS

네트워크

p2p

서버-클라이언트

DB

파일db

JSON

웹서비스

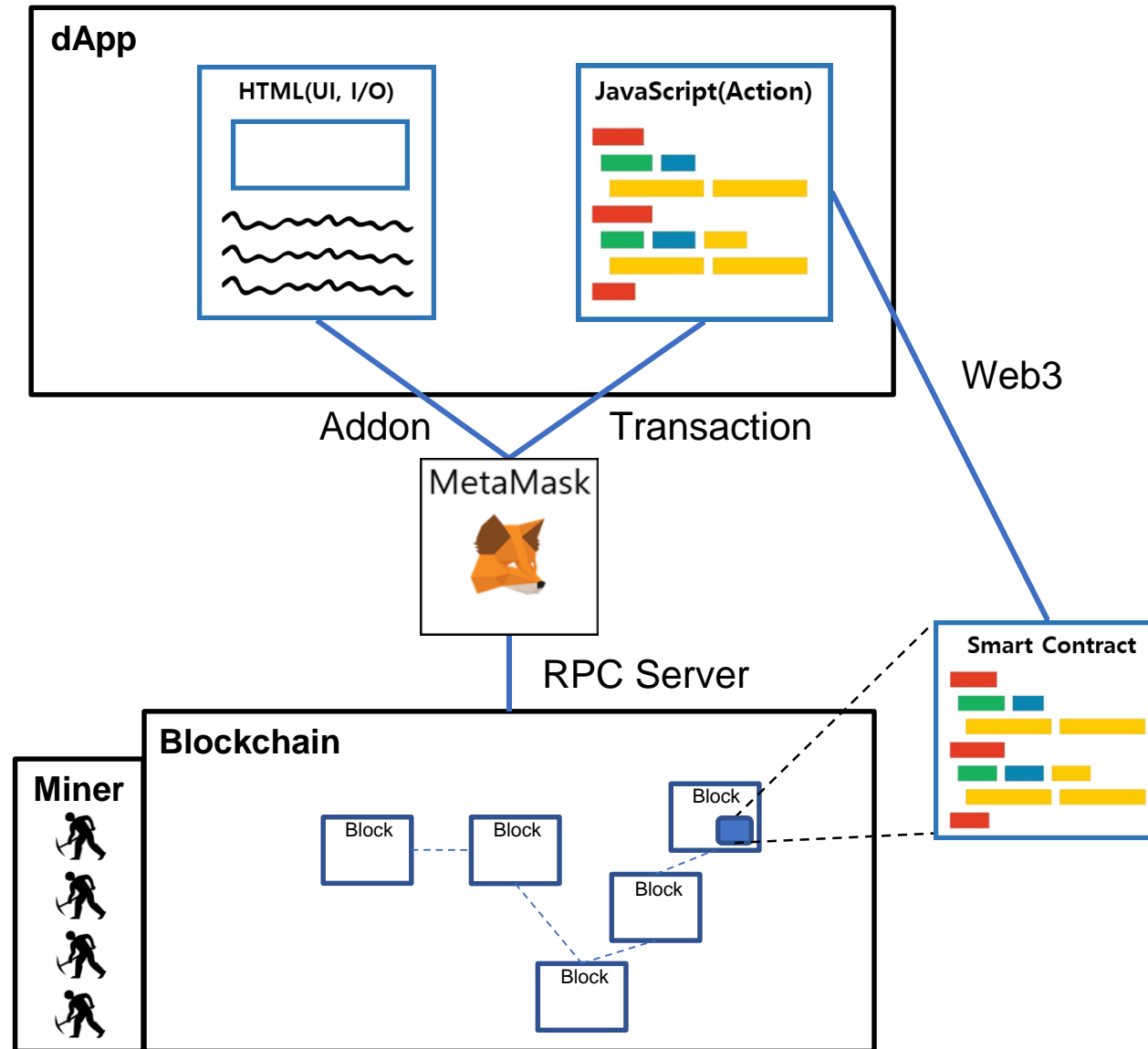
nodejs

반응형 웹페이지

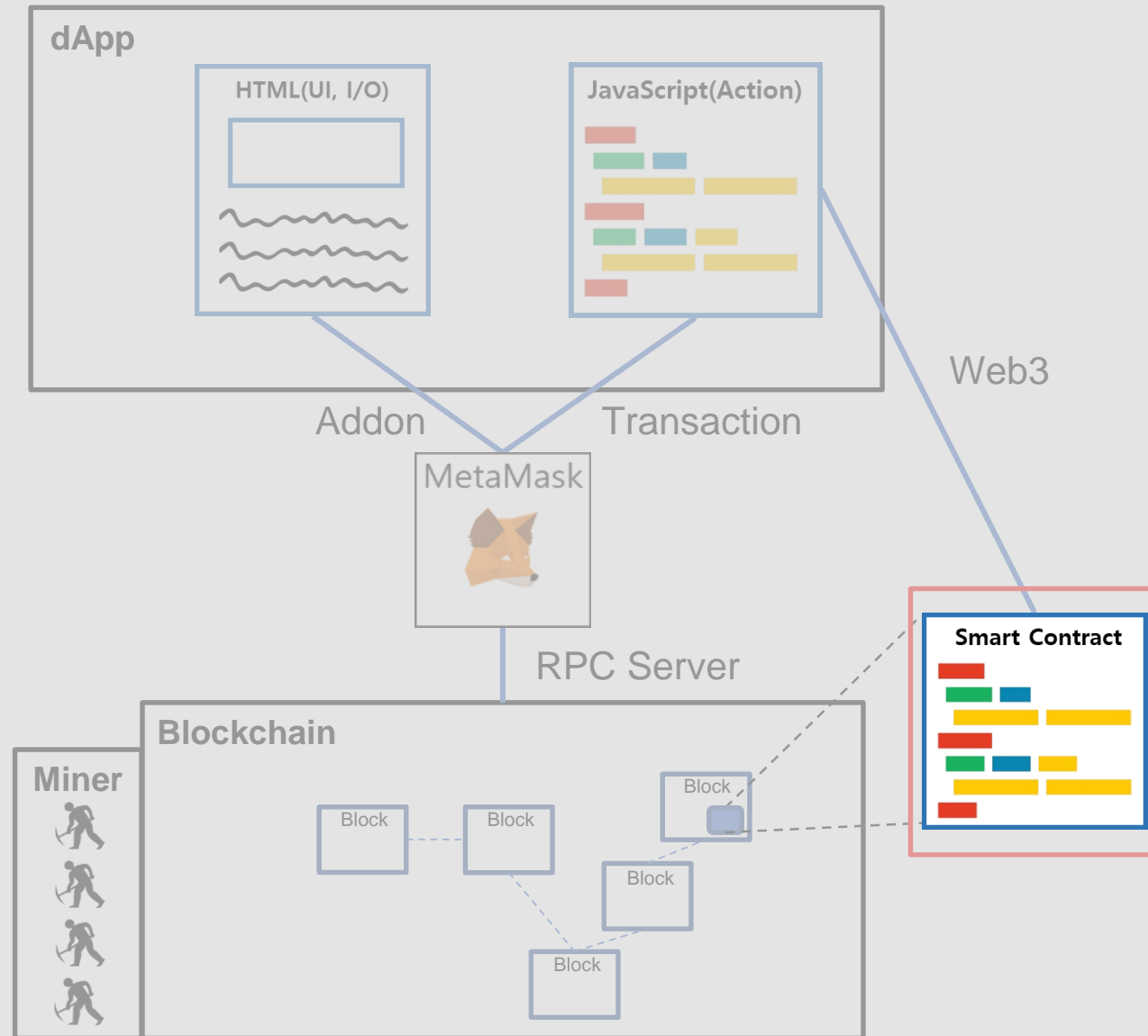
프로그래밍

스마트컨트랙트

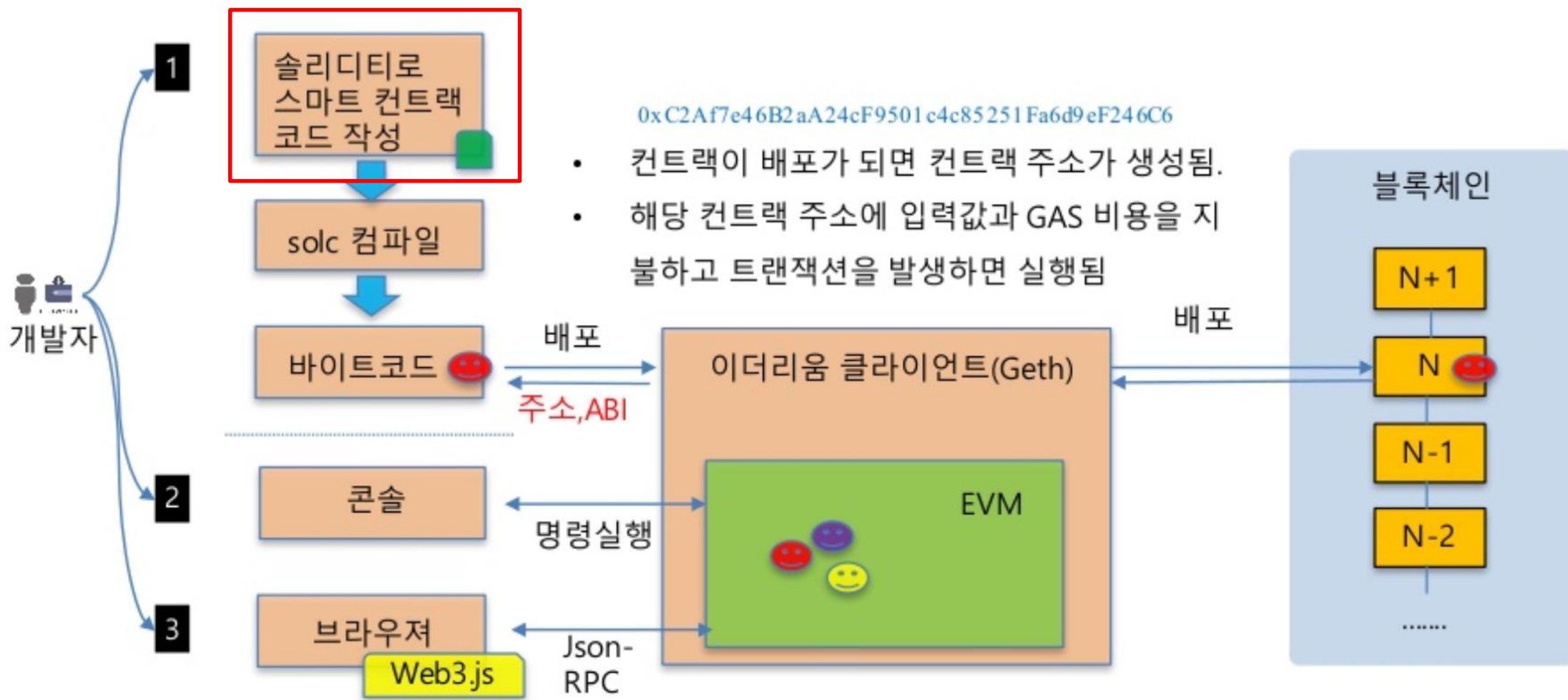
01 dApp Architecture



01 dApp Architecture



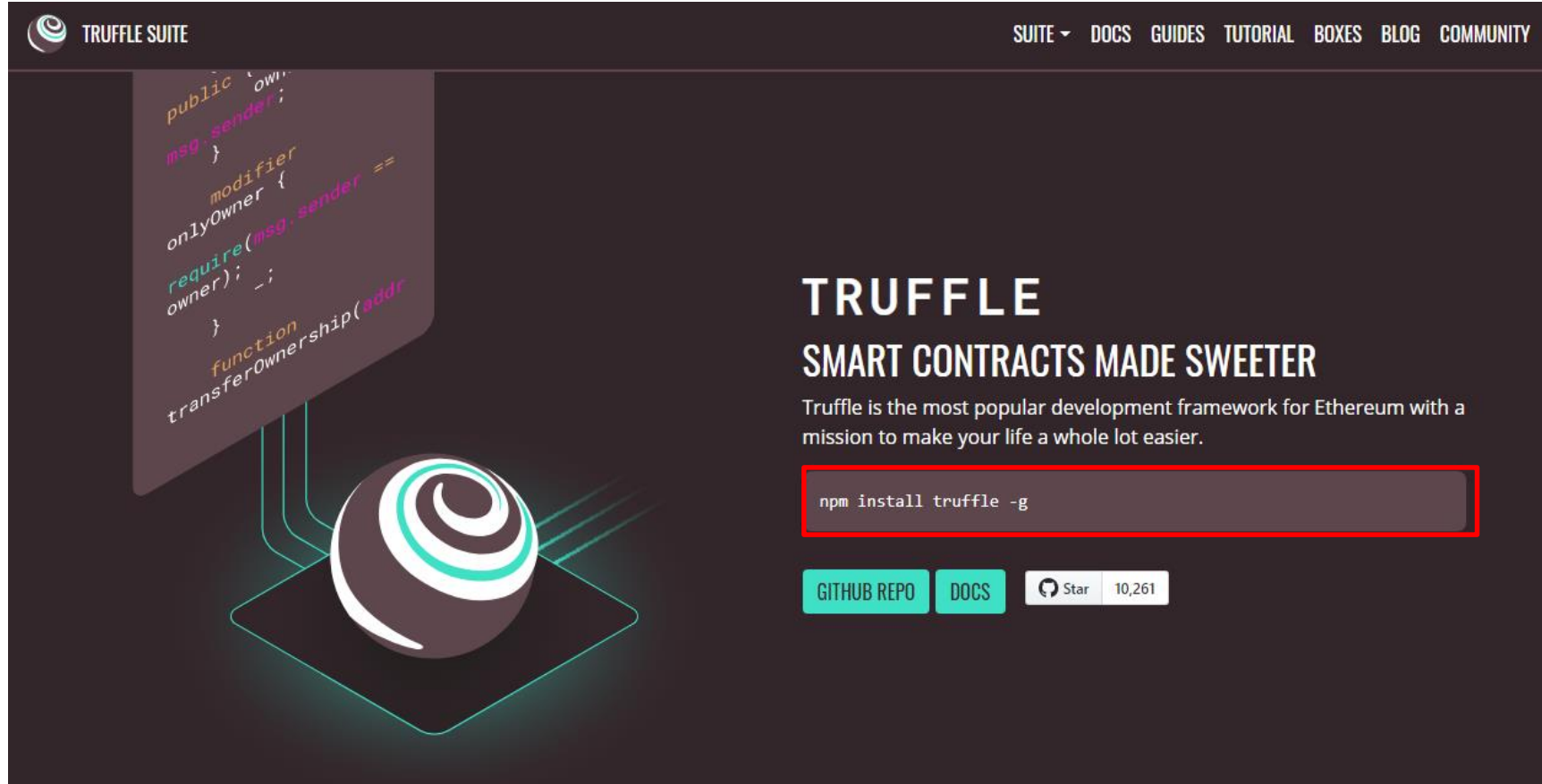
02 스마트 컨트랙트 구성요소



01

Truffle를 이용한 DApp

01 Truffle이란?



<https://www.trufflesuite.com/>

01 Truffle이란?

```
npm install truffle -g
```

npm?

npm: Node Packaged Manager

Node?

Node.js: Javascript runtime built on Chrome's V8 JavaScript engine

Javascript?

Javascript: 자바스크립트는 '웹페이지에 생동감을 불어넣기 위해' 만들어진 프로그래밍 언어. 동적인 DOM(HTML – 웹사이트의 표현을 위한 언어.)

01 Truffle이란?

HTML?

HTML(Hyper Text Markup Language):

웹페이지를 만들기 위한 언어로 웹브라우저 위에서 동작하는 언어다.

```
<!DOCTYPE html>
<html>
<head>...</head>
<body>
  <style type="text/css">...</style>
  <!-- Google Tag Manager (noscript) -->
  <noscript>...</noscript>
  <!-- End Google Tag Manager (noscript) -->
  <nav class="navbar navbar-expand-md navbar-dark fixed-top" id="primaryNav">...</nav>
  <main role="main" class="container">
    <div class="row align-items-center truffle-heading-row">
      <div class="col-md-6">
        
      </div>
      <div class="col-md-6">
        <!--<p class="truffle-down-arrow d-none d-md-block" id="headerArrow"><i class="fas fa-chevron-down"></i></p-->
      </div>
    </div>
    <div class="row mt-5 mb-6" id="contentBegins">...</div>
    <div class="row align-items-center truffle-feature-row mb-6">...</div>
    <div class="row align-items-center truffle-feature-row mb-6">...</div>
    <div class="row align-items-center truffle-feature-row mb-7">...</div>
  </main>
  <div class="banner-taste truffle">...</div>
  <script src="https://code.jquery.com/jquery-3.2.1.min.js"></script>
  <script>...</script>
  <script src="js/bootstrap.min.js"></script>
  <script src="js/scroll-arrow.js"></script>
  <script async defer src="https://buttons.github.io/buttons.js"></script>
  <footer>...</footer>
</body>
</html> == $0
```

Styles Computed Layout Event Listeners >>

Filter :hov .cls + [4]

element.style { }

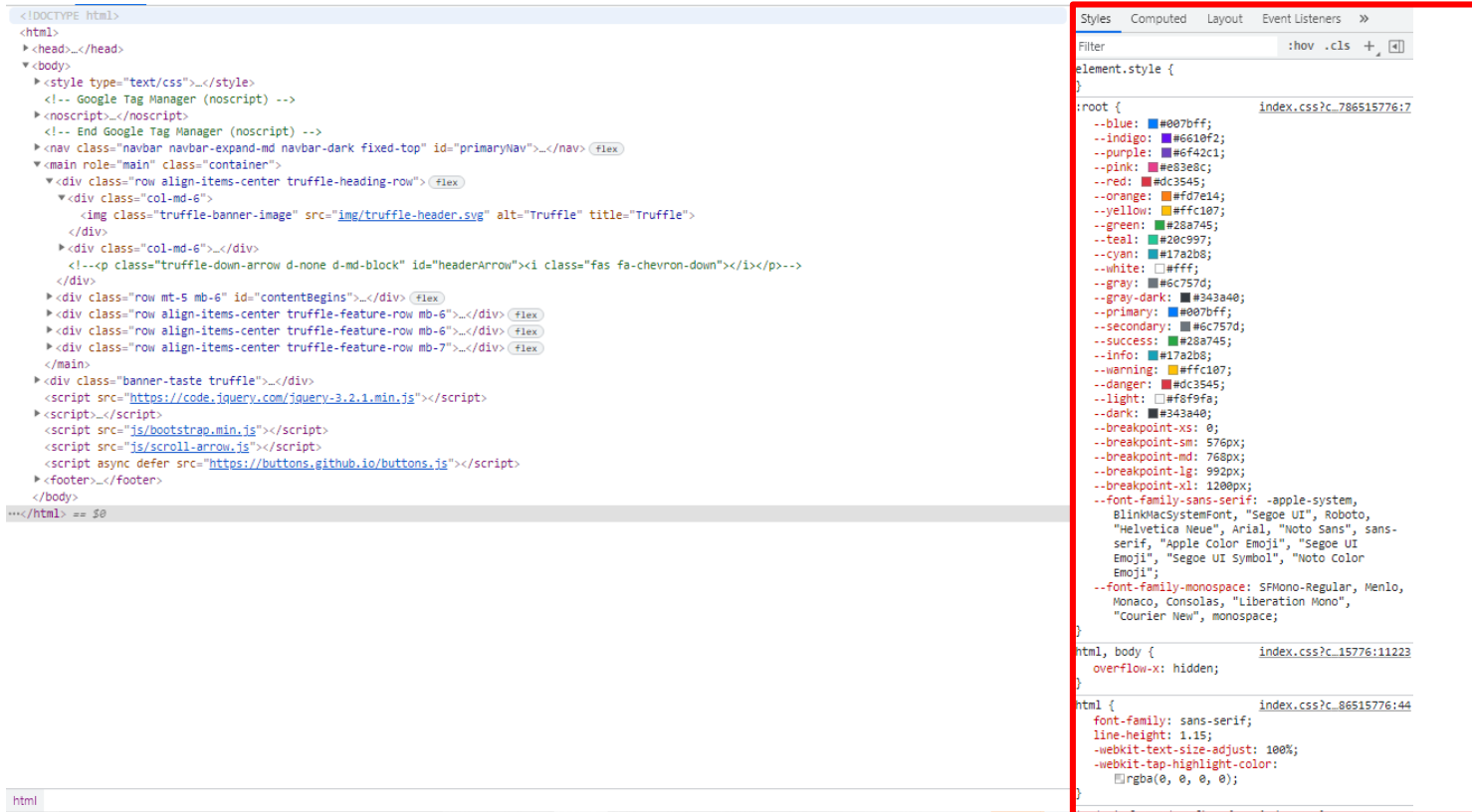
:root { index.css?c_786515776:17

```
--blue: #007bff;
--indigo: #6610f2;
--purple: #6f42c1;
--pink: #e83e8c;
--red: #dc3545;
--orange: #fd7e14;
--yellow: #ffc107;
--green: #28a745;
--teal: #20c997;
--cyan: #17a2b8;
--white: #fff;
--gray: #6c757d;
--gray-dark: #343a40;
--primary: #007bff;
--secondary: #6c757d;
--success: #28a745;
--info: #17a2b8;
--warning: #ffc107;
--danger: #dc3545;
--light: #f8f9fa;
--dark: #343a40;
--breakpoint-xs: 0;
--breakpoint-sm: 576px;
--breakpoint-md: 768px;
--breakpoint-lg: 992px;
--breakpoint-xl: 1200px;
--font-family-sans-serif: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, Helvetica Neue, Arial, "Noto Sans", sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto Color Emoji";
--font-family-monospace: SFMono-Regular, Menlo, Monaco, Consolas, "Liberation Mono", "Courier New", monospace;
}

html, body {
  overflow-x: hidden;
}

html {
  font-family: sans-serif;
  line-height: 1.15;
  -webkit-text-size-adjust: 100%;
  -webkit-tap-highlight-color: rgba(0, 0, 0, 0);
}
```

01 Truffle이란?



CSS(Cascading Style Sheets: 상속적인 스타일 시트)
HTML의 스타일(어떻게 보이는지)를 지정해줌.

01 Truffle이란?

우리가 Truffle을 이용해 Dapp을 만든다는건?

- 블록체인과 통신하는 웹 application을 만든다는 것
그럼 알아야 하는 것은?
- Javascript + Node.js
- HTML
- CSS

01 Truffle이란?

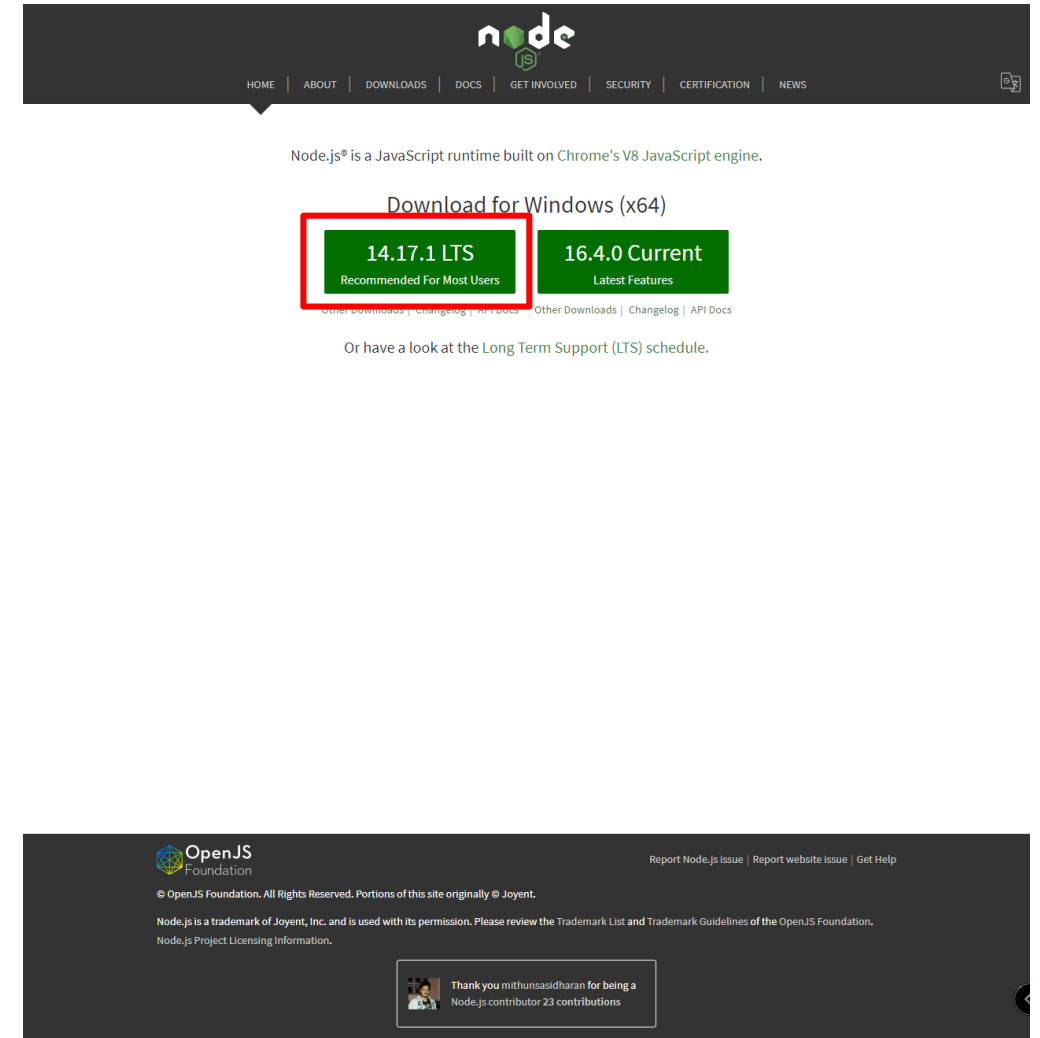
우선 개발환경을 설정해보자.

1. Node.js 설치
2. VS-Code Extension 설치
3. npm을 통한 truffle 전역설치
4. Metamask 설치 및 설정
5. ganache 설치

02 개발환경 설정

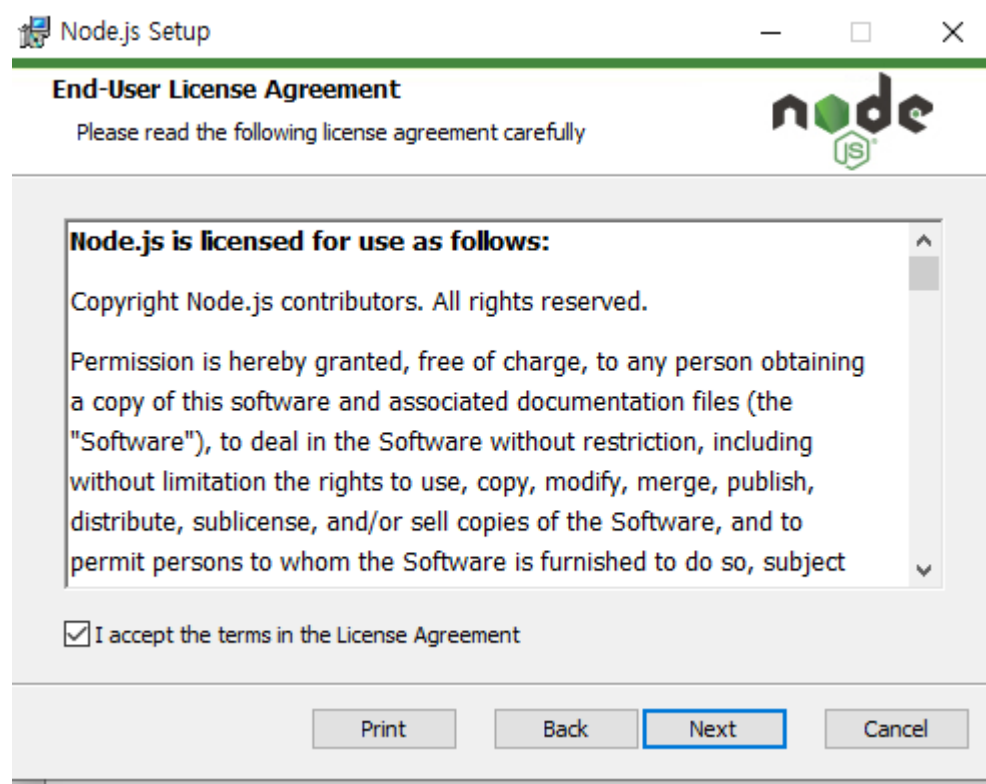
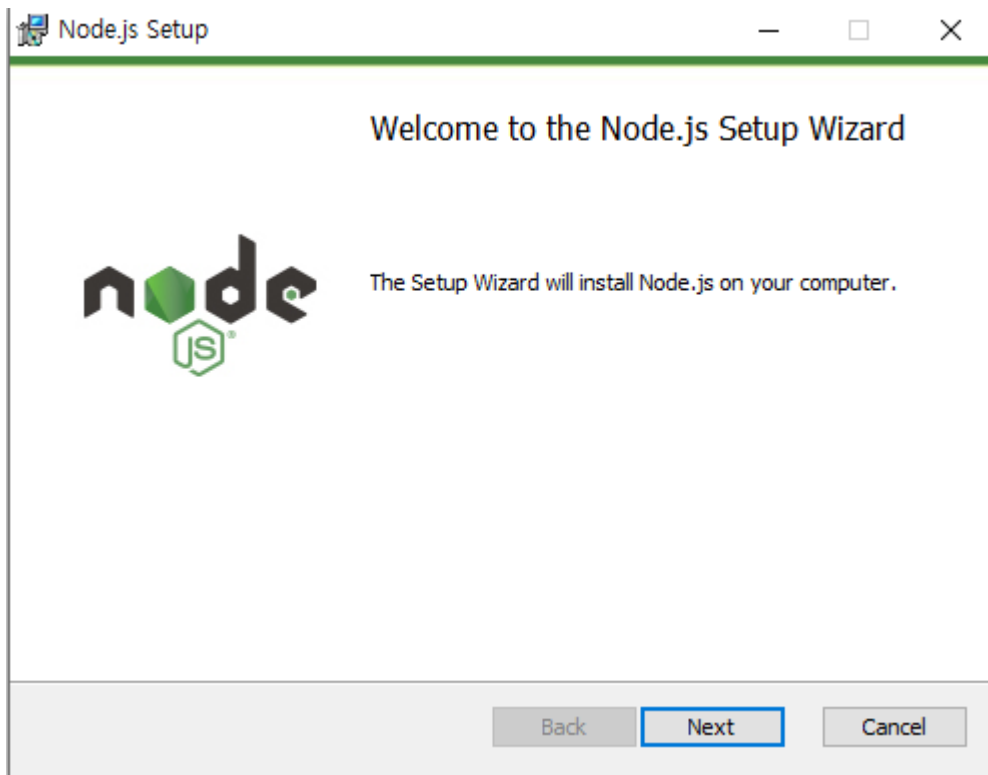
Node.js: Javascript runtime
built on Chrome's V8 JavaScript engine

- <https://nodejs.org/>



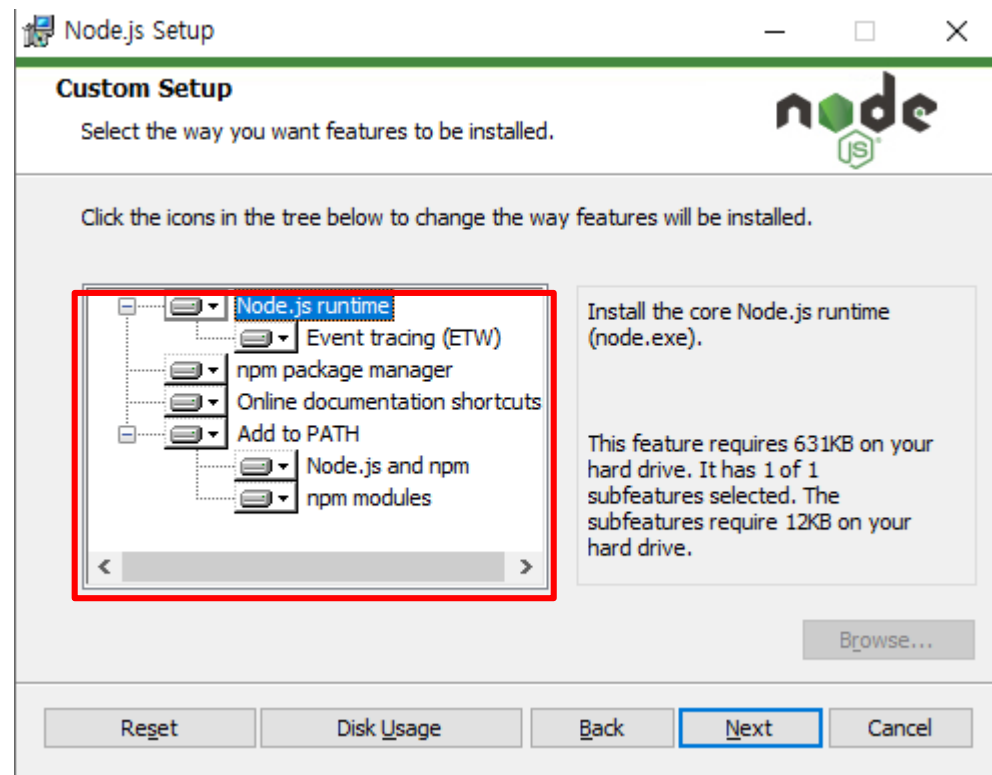
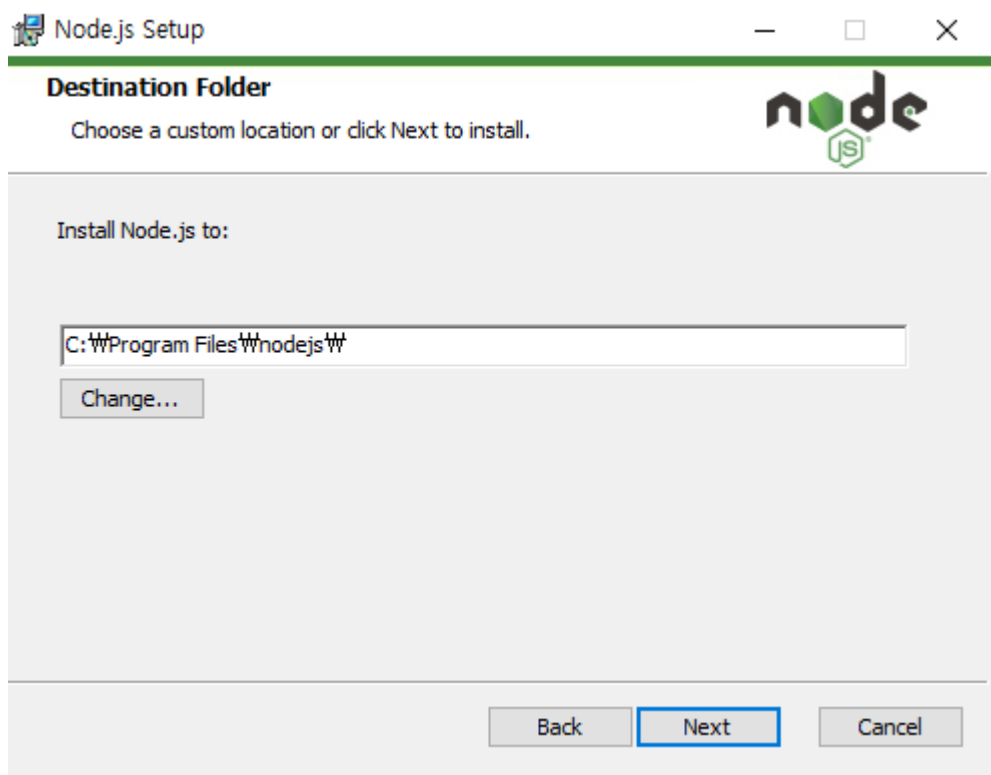
02 개발환경 설정

Node.js



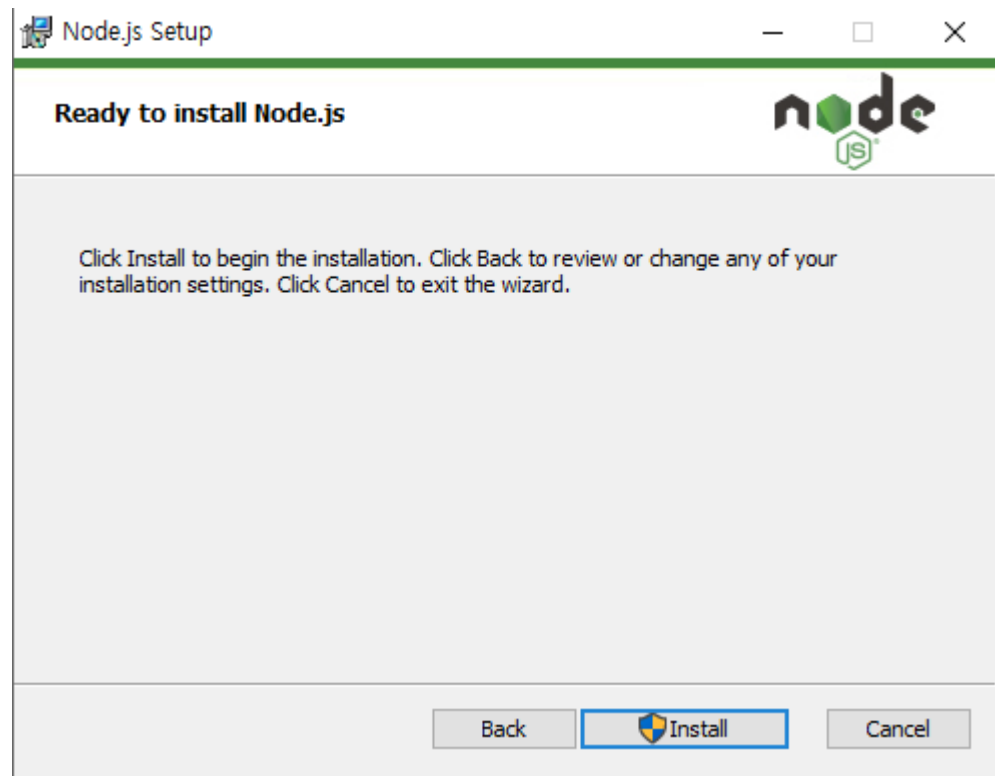
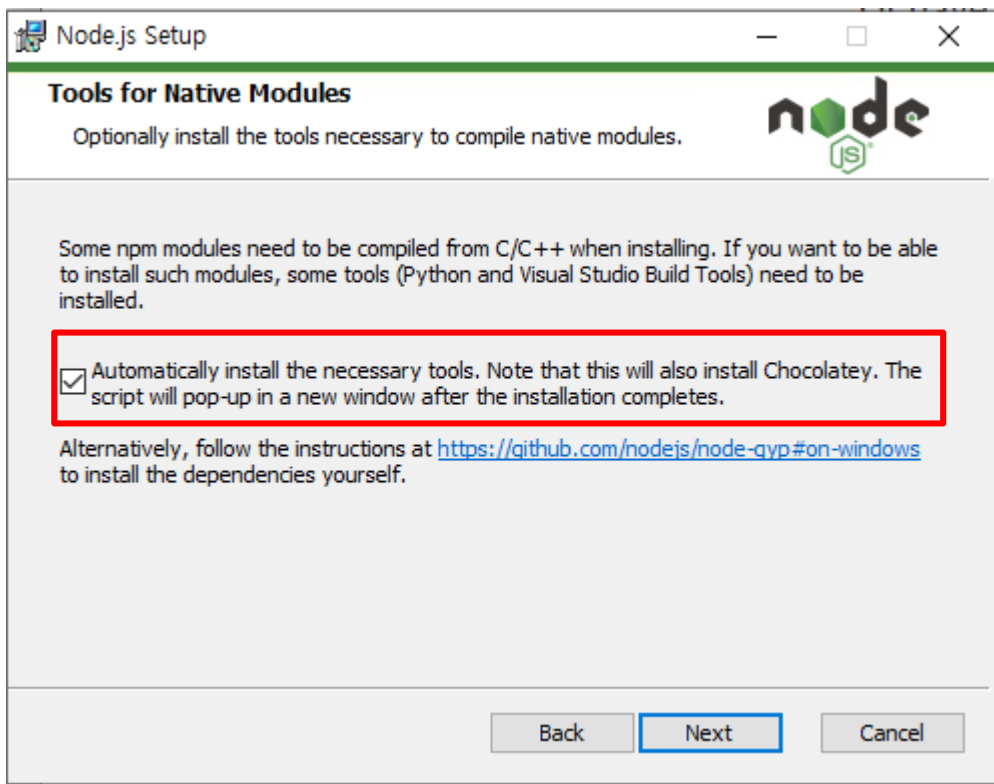
02 개발환경 설정

Node.js



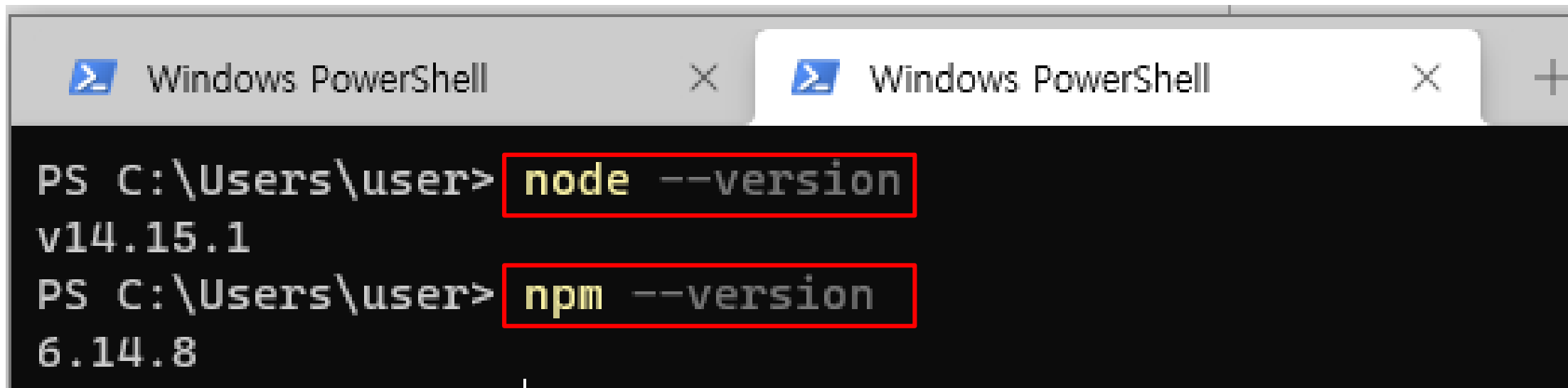
02 개발환경 설정

Node.js



02 개발환경 설정

Node.js



A screenshot of a Windows PowerShell terminal window. The window has two tabs, both labeled 'Windows PowerShell'. The terminal shows two commands being executed: 'node --version' and 'npm --version'. The output of the first command is 'v14.15.1' and the output of the second command is '6.14.8'. The command text 'node --version' and 'npm --version' are highlighted with red rectangular boxes.

```
PS C:\Users\user> node --version
v14.15.1
PS C:\Users\user> npm --version
6.14.8
```

02 개발환경 설정

VS-Code Extension 설치



solidity `juanblanco.solidity`

Juan Blanco

330,338



Repository

v0.0.121

Ethereum Solidity Language for Visual Studio Code

Reload Required

Disable



Uninstall



Please reload Visual Studio Code to enable the updated extension.



ESLint `dbaeumer.vscode-eslint`

Dirk Baeumer

15,045,599



Repository

License

v2.1.16

Integrates ESLint JavaScript into VS Code.

Disable



Uninstall




This extension is enabled globally.

This extension is recommended based on the files you recently opened.

02 개발환경 설정

VS-Code Extension 설치




Bracket Pair Colorizer 2

coenraads.bracket-pair-colorizer-2

CoenraadS | 2,774,578 | ★★★★★ | Repository | License | v0.2.1

A customizable extension for coloring matching brackets

[Disable](#) [Uninstall](#) ⚙️ This extension is enabled globally.



Bracket Highlighter

durzn.brackethighlighter


Durzn | 21,544 | ★★★★★ | Repository | License | v2.2.4

Decorates text inbetween symbols.

[Disable](#) [Uninstall](#) ⚙️ This extension is enabled globally.

02 개발환경 설정

VS-Code Extension 설치




Auto Close Tag

formulahendry.auto-close-tag

Jun Han | 4,897,276 | ★★★★★ | Repository | License | v0.5.10

Automatically add HTML/XML close tag, same as Visual Studio IDE or Sublime Text

[Disable](#) [Uninstall](#) ⚙️ This extension is enabled globally.



Auto Rename Tag

formulahendry.auto-rename-tag


Jun Han | 5,765,705 | ★★★★★ | Repository | v0.1.6

Auto rename paired HTML/XML tag

[Disable](#) [Uninstall](#) ⚙️ This extension is enabled globally.

02 개발환경 설정

VS-Code Extension 설치




Path Intellisense

christian-kohler.path-intellisense

Christian Kohler | 4,905,789 | ★★★★★ | Repository | License | v2.3.0

Visual Studio Code plugin that autocompletes filenames

[Disable](#) [Uninstall](#) ⚙️ *This extension is enabled globally.*



npm Intellisense

christian-kohler.npm-intellisense

Christian Kohler | 3,177,162 | ★★★★★ | Repository | License | v1.3.1

Visual Studio Code plugin that autocompletes npm modules in import statements

[Disable](#) [Uninstall](#) ⚙️ *This extension is enabled globally.*

This extension is recommended based on the files you recently opened.

02 개발환경 설정

npm을 이용한 truffle 설치(전역 - global)

npm으로 패키지를 설치하려면?

→ `npm install <패키지명>`

npm으로 패키지를 전역으로 설치하려면?

→ `npm install -g <패키지명>`

```
PS C:\Users\user> npm install -g truffle
```

02 개발환경 설정

npm을 이용한 truffle 설치(전역 - global)

설치가 되면 truffle 명령어를 이용할 수 있다.

[주요 truffle 명령어]

- truffle init : truffle 프로젝트 생성
- truffle create : contract를 만들기 위한 헬퍼 명령어
- truffle compile : solidity 컴파일
- truffle migrate : solidity deploy.
- truffle develop : develop version **ethereum client** 실행.
- truffle install : solidity 외부 라이브러리 설치.

02 개발환경 설정

Metamask 설치

<https://chrome.google.com/webstore/search/metamask?hl=ko>

홈 > 확장 프로그램 > MetaMask



MetaMask

제공업체: <https://metamask.io>

★★★★★ 2,018 | 생산성 | 👤 사용자 5,000,000+명

Chrome에 추가

02 개발환경 설정

Metamask 설치

MetaMask가 처음이세요?



아니요. 이미 시드 구문이 있습니다.

12단어 시드 구문을 사용하여 지갑 가져오기

지갑 가져오기



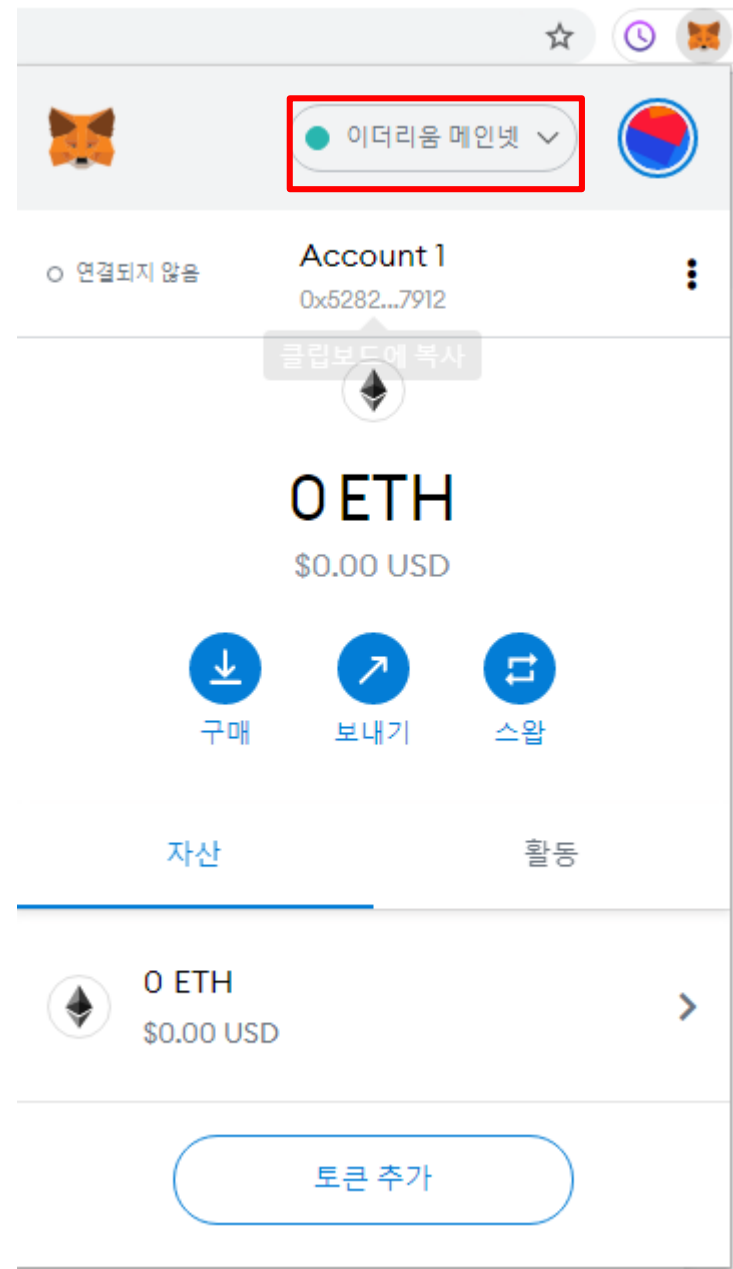
설정을 시작하죠!

새 지갑과 시드 구문을 만듭니다.

지갑 생성

02 개발환경 설정

Metamask 설치



02 개발환경 설정

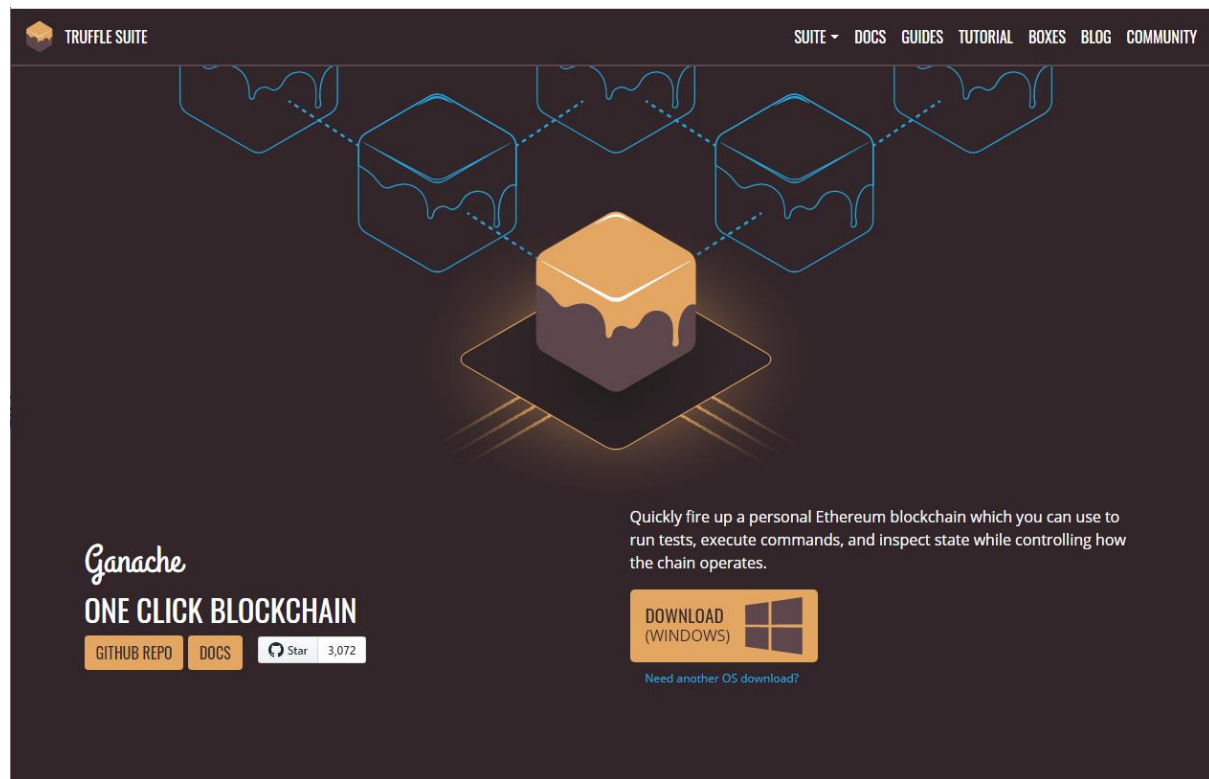
ganache 설치

<https://www.trufflesuite.com/ganache>

Quickly fire up a personal Ethereum blockchain which you can use to run tests, execute commands, and inspect state while controlling how the chain operates.

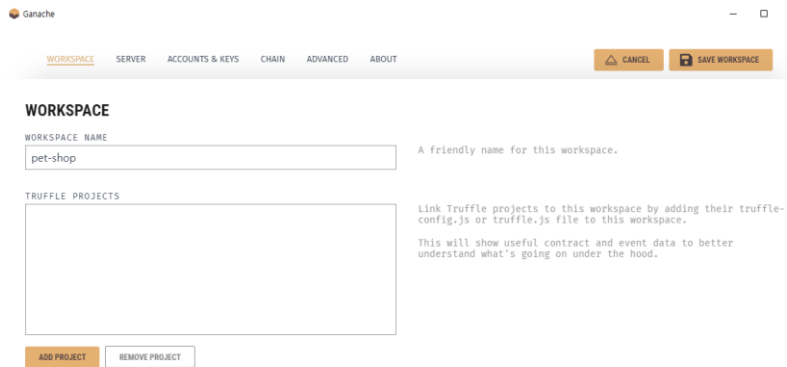
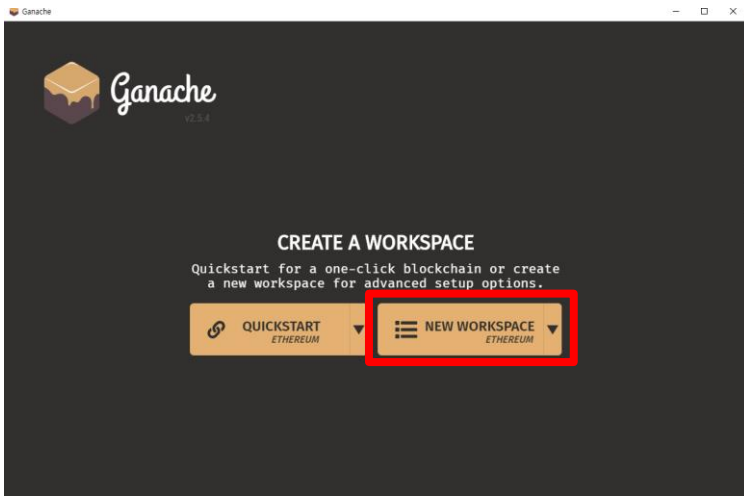
빠르게 개인 이더리움 블록체인을 시작하는 방법.

- contract를 테스트 할 수 있는 gui 툴.



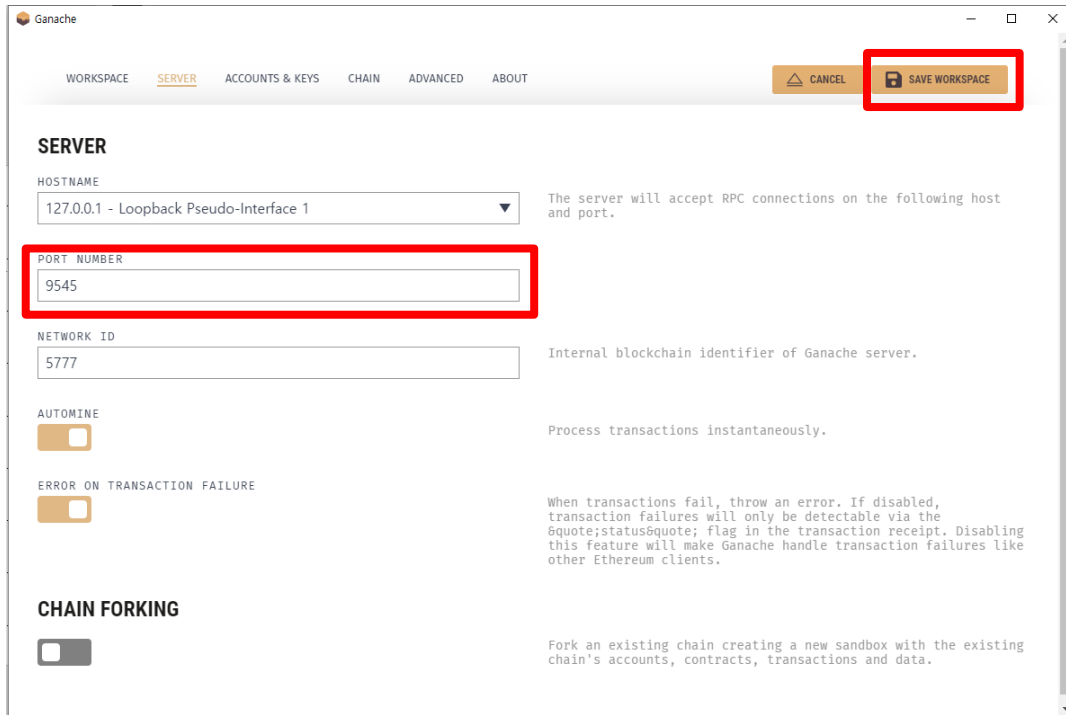
03 Solidity 개발 시작하기

시작전에 ganache를 실행해 주세요.
- personal network를 실행하는 것



03 Solidity 개발 시작하기

시작전에 ganache를 실행해 주세요.
- personal network를 실행하는 것



03 Solidity 개발 시작하기

metamask personal network로 설정

metamask personal network로 설정

약성 네트워크 공급업체는 블록체인 상태를 거짓으로 보고하고 네트워크 활동을 기록할 수 있습니다. 신뢰하는 맞춤형 네트워크만 추가하세요.

네트워크 이름

pet-shop

새 RPC URL

http://localhost:9545

체인 ID ⓘ

1337

통화 기호(선택 사항)

블록 탐색기 URL(선택 사항)

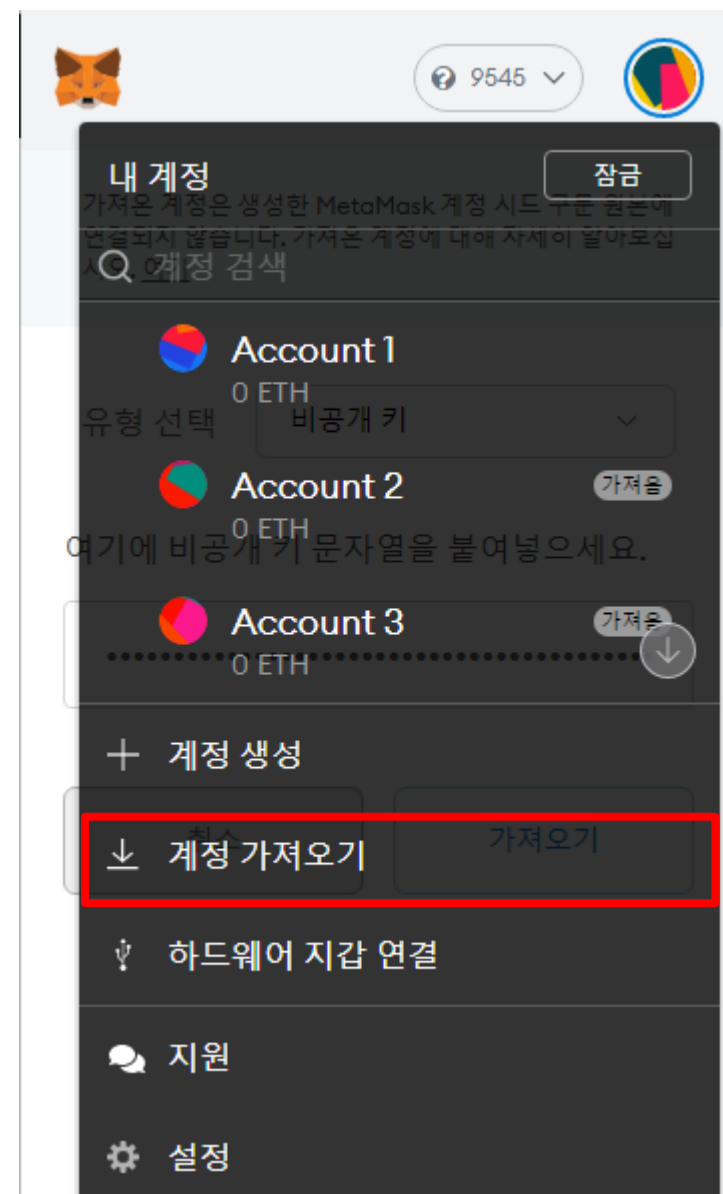
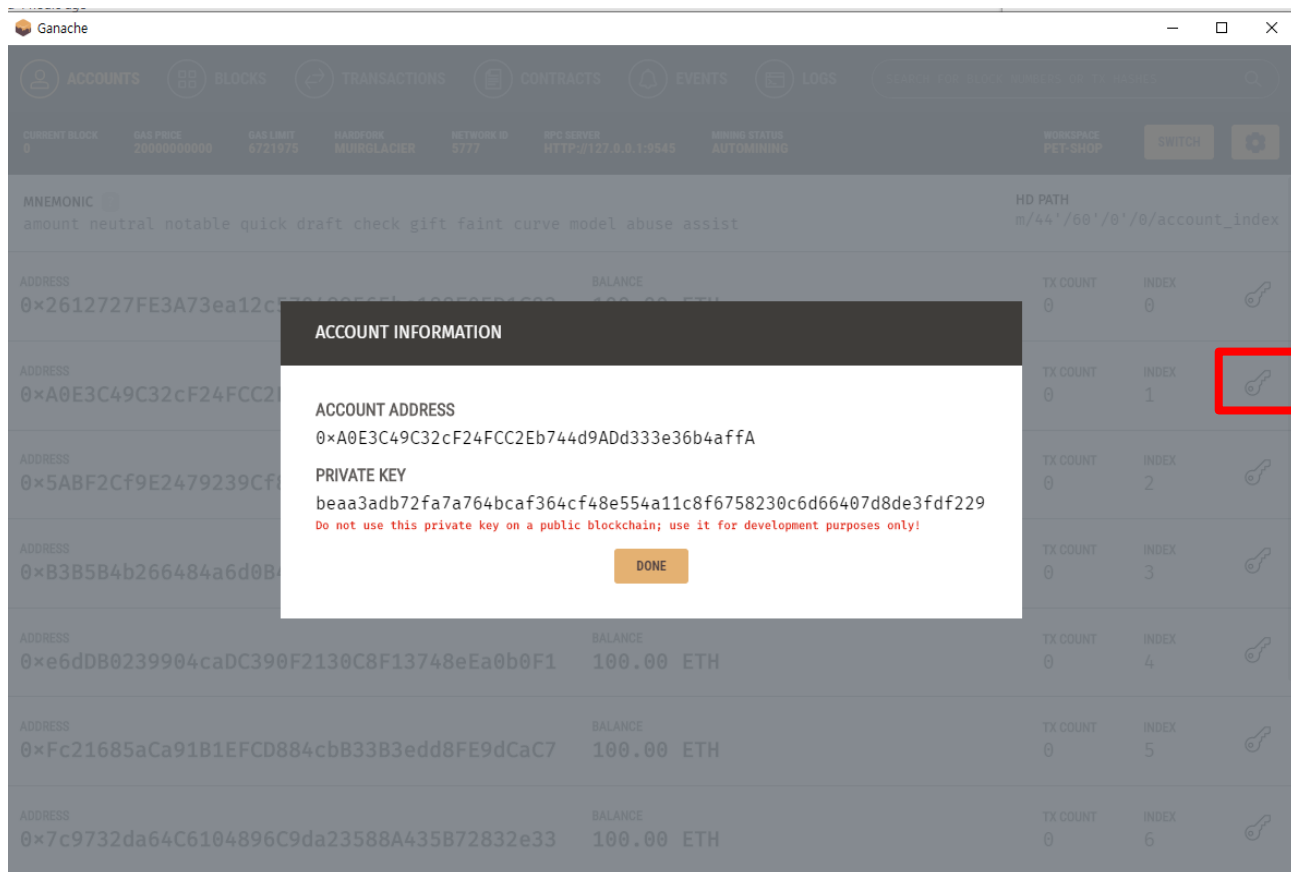
03 Solidity 개발 시작하기

이더 보내기 실습

Ganache									
ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS									
CURRENT BLOCK 8 GAS PRICE 20000000000 GAS LIMIT 6721975 HARDFORK MUIRGLACIER NETWORK ID 5777 RPC SERVER HTTP://127.0.0.1:8545 MINING STATUS AUTOMINING WORKSPACE PET-SHOP SWITCH									
ADDRESS	BALANCE	TX COUNT	INDEX						
0x5ABF2Cf9E2479239Cf8D449Adfa476a595C1Ae39	100.00 ETH	0	2						
ADDRESS	BALANCE	TX COUNT	INDEX						
0xB3B5B4b266484a6d0B464b0bACdC926B174fB6D1	100.00 ETH	0	3						
ADDRESS	BALANCE	TX COUNT	INDEX						
0xe6dDB0239904caDC390F2130C8F13748eEa0b0F1	100.00 ETH	0	4						
ADDRESS	BALANCE	TX COUNT	INDEX						
0xFc21685aCa91B1EFCD884cbB33B3edd8FE9dCaC7	100.00 ETH	0	5						
ADDRESS	BALANCE	TX COUNT	INDEX						
0x7c9732da64C6104896C9da23588A435B72832e33	100.00 ETH	0	6						
ADDRESS	BALANCE	TX COUNT	INDEX						
0xEAFC4fBB92b917d7CcdaEa848795855c88a7e581	100.00 ETH	1	7						
ADDRESS	BALANCE	TX COUNT	INDEX						
0x2eD14F2baC77605F2D796593092098946cDE9A65	100.00 ETH	0	8						
ADDRESS	BALANCE	TX COUNT	INDEX						
0xF204B4a9E34170A56381689de2ee6A1B23E62143	90.00 ETH	1	9						

03 Solidity 개발 시작하기

Metamask 계정추가



03 Solidity 개발 시작하기

실습)

Metamask로 Personal Network
송금을 해보세요!

03 Solidity 개발 시작하기

truffle 환경에서 Contract 개발시작하기

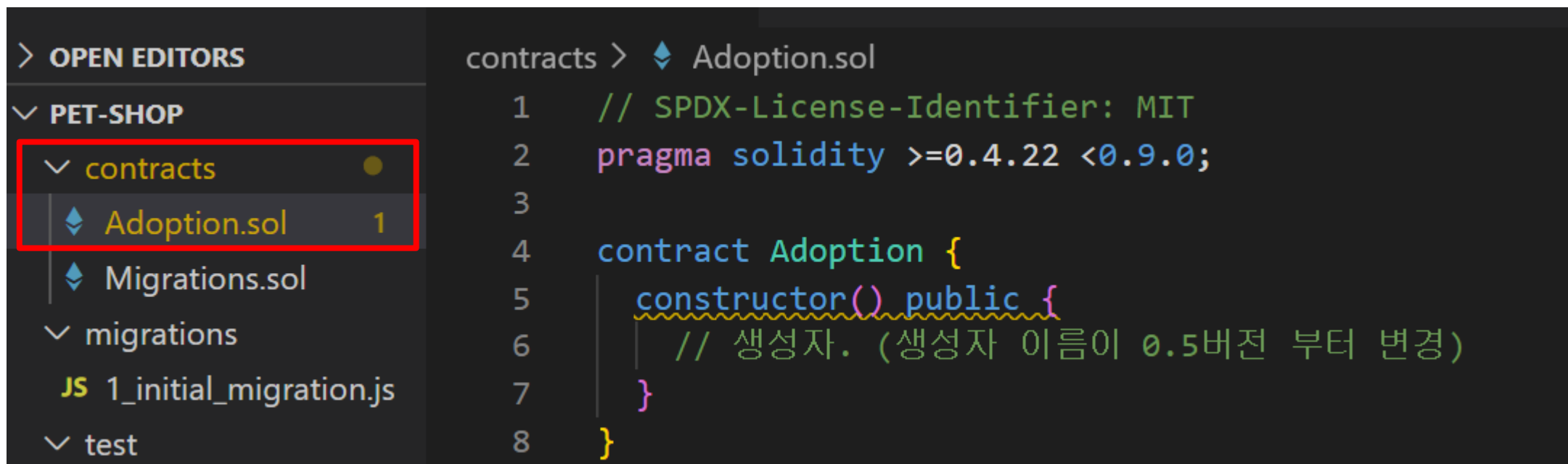
```
> mkdir <project>  
> cd <project>  
> truffle init
```

```
✓ SOLIDITY_SAMPLE  
  ✓ contracts  
    ◆ Migrations.sol  
  ✓ migrations  
    JS 1_initial_migration.js  
  > test  
    JS truffle-config.js
```

03 Solidity 개발 시작하기

truffle 환경에서 Contract 개발시작하기

> truffle create contract <Contract 이름>



The screenshot shows the VS Code interface with the 'contracts' folder expanded in the left sidebar. The file 'Adoption.sol' is selected and highlighted with a red box. The main editor displays the content of 'Adoption.sol', which includes a license header, a Solidity pragma statement, and a contract definition with a constructor.

```
contracts > Adoption.sol
1  // SPDX-License-Identifier: MIT
2  pragma solidity >=0.4.22 <0.9.0;
3
4  contract Adoption {
5      constructor() public {
6          // 생성자. (생성자 이름이 0.5버전 부터 변경)
7      }
8  }
```

03 Solidity 개발 시작하기

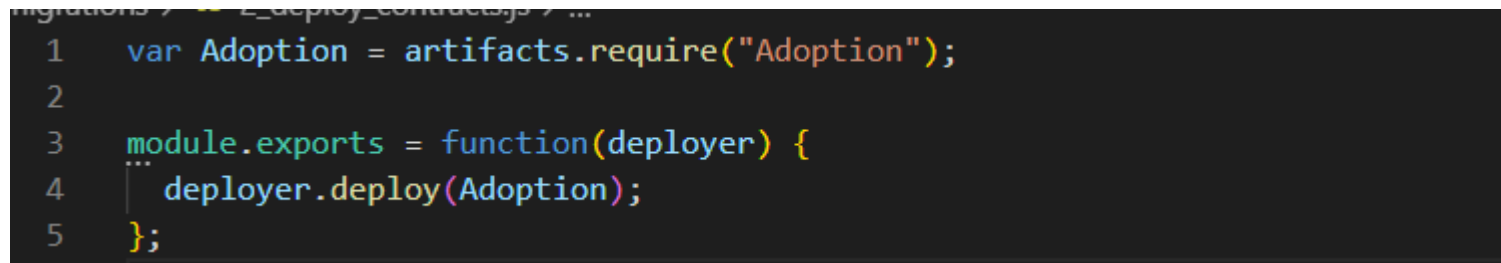
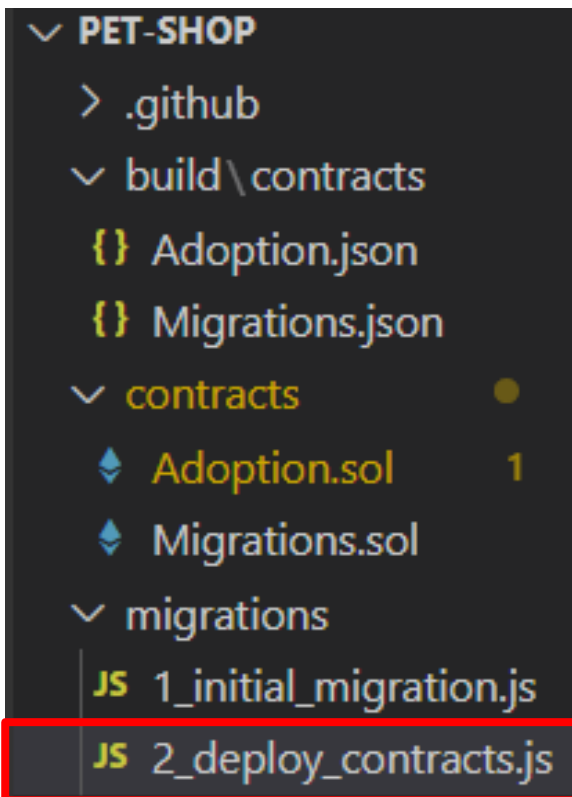
Contract 만들기

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity >=0.4.22 <0.9.0;
3
4 contract Adoption {
5     address[16] public adopters;
6
7     constructor() public {
8         // 생성자. (생성자 이름이 0.5버전 부터 변경)
9     }
10
11     function adopt(uint petId) public returns (uint){
12         require(petId>=0 && petId <=15);
13
14         adopters[petId] = msg.sender;
15         return petId;
16     }
17
18     function getAdopters() external view returns (address[16] memory){
19         return adopters;
20     }
21 }
```

03 Solidity 개발 시작하기

Contract 만들기

migrations/2_deploy_contracts.js 생성



03 Solidity 개발 시작하기

Contract 만들기

> truffle migrate

```
truffle(develop)> migrate

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

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

2_deploy_contracts.js
=====

Deploying 'Adoption'
-----
> transaction hash: 0x3446203c5018bb0c518294f1b205c13474d857b6c20b850cc11819593a71469d
> Blocks: 0        Seconds: 0
> contract address: 0xdCABcece7D4E40153ad5Fd29feC5d3bC6dC93FB8
> block number:     3
> block timestamp:  1609384909
> account:          0x0113ED56b67fa01cE4eA08067700E0Bf7D1C964F
> balance:          99.99123808
> gas used:         203815 (0x31c27)
> gas price:        20 gwei
> value sent:       0 ETH
> total cost:       0.0040763 ETH

> Saving migration to chain.
> Saving artifacts
-----
> Total cost:       0.0040763 ETH

Summary
=====
> Total deployments: 1
> Final cost:       0.0040763 ETH
```

04 Solidity 개발 맛보기 (pet-shop)

truffle pet-shop 실습하기

> **truffle unbox pet-shop**

```
Starting unbox...
=====

✓ Preparing to download box
✓ Downloading
npm WARN pet-shop@1.0.0 No description
npm WARN pet-shop@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.4 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.4: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

✓ Cleaning up temporary files
✓ Setting up box

Unbox successful, sweet!

Commands:
  Compile:      truffle compile
  Migrate:      truffle migrate
  Test contracts: truffle test
  Run dev server: npm run dev
```


04 Solidity 개발 맛보기 (pet-shop)

truffle pet-shop 프로젝트 살펴보기

contract 파일

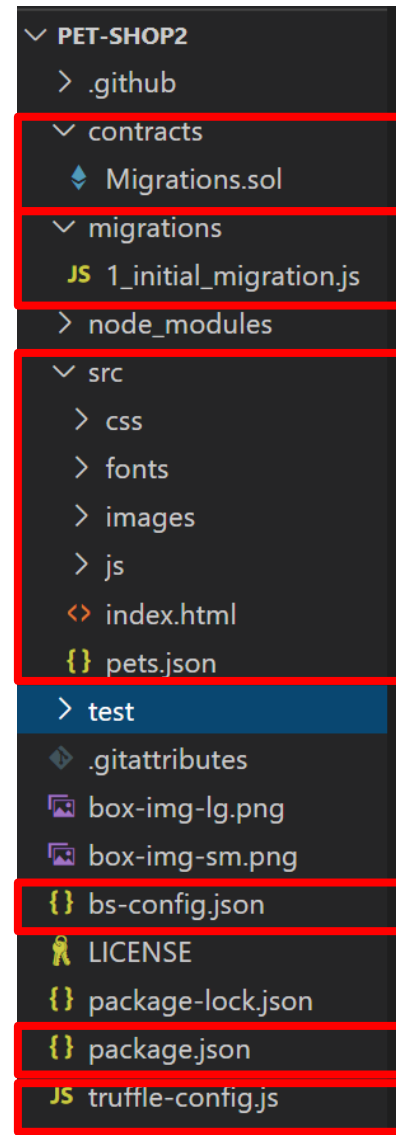
migration 파일

웹 client(front-end)
동작 파일

웹 서버 파일

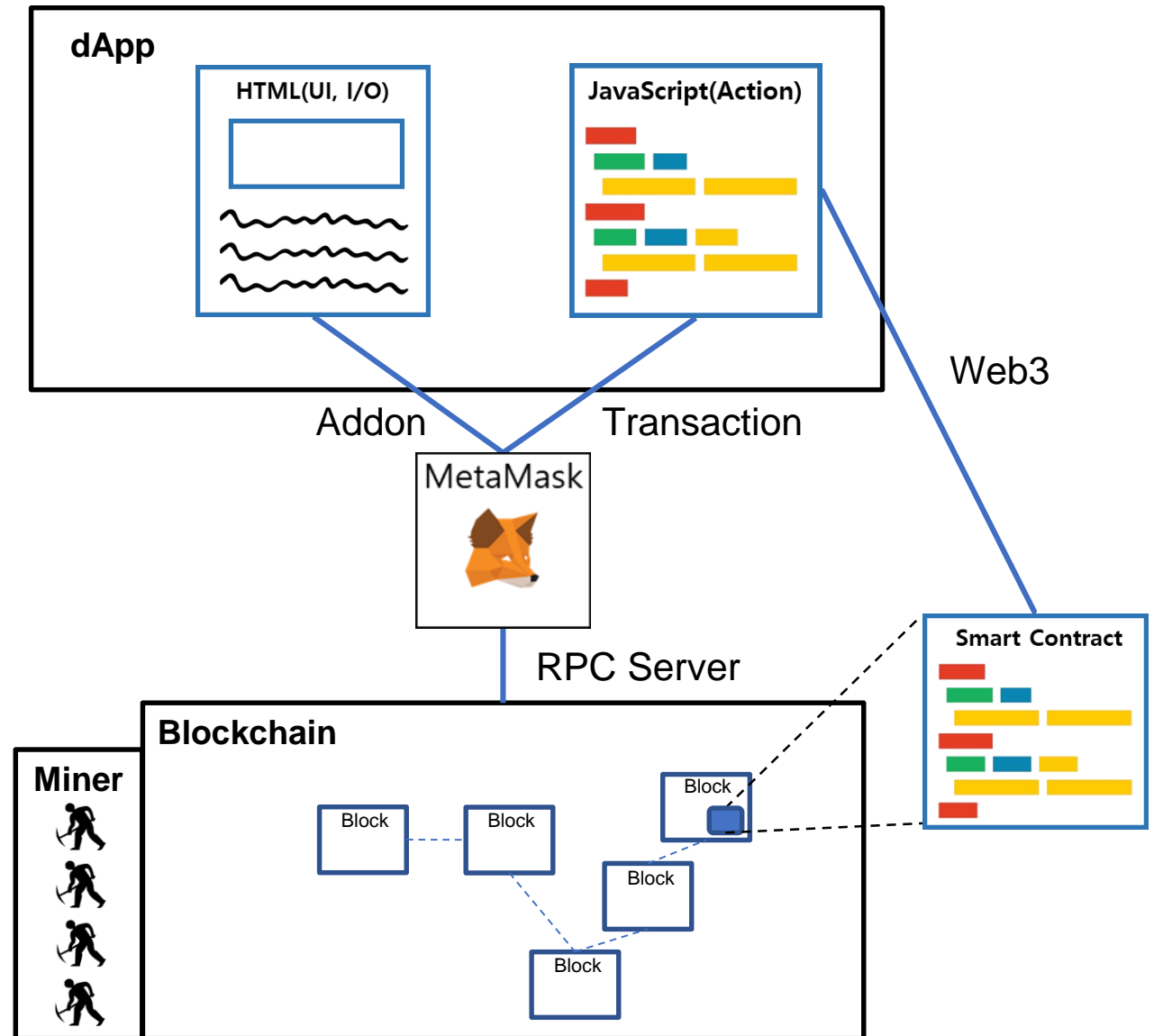
npm file

truffle configuration



04 Solidity 개발 맛보기 (pet-shop)

Dapp Architecture



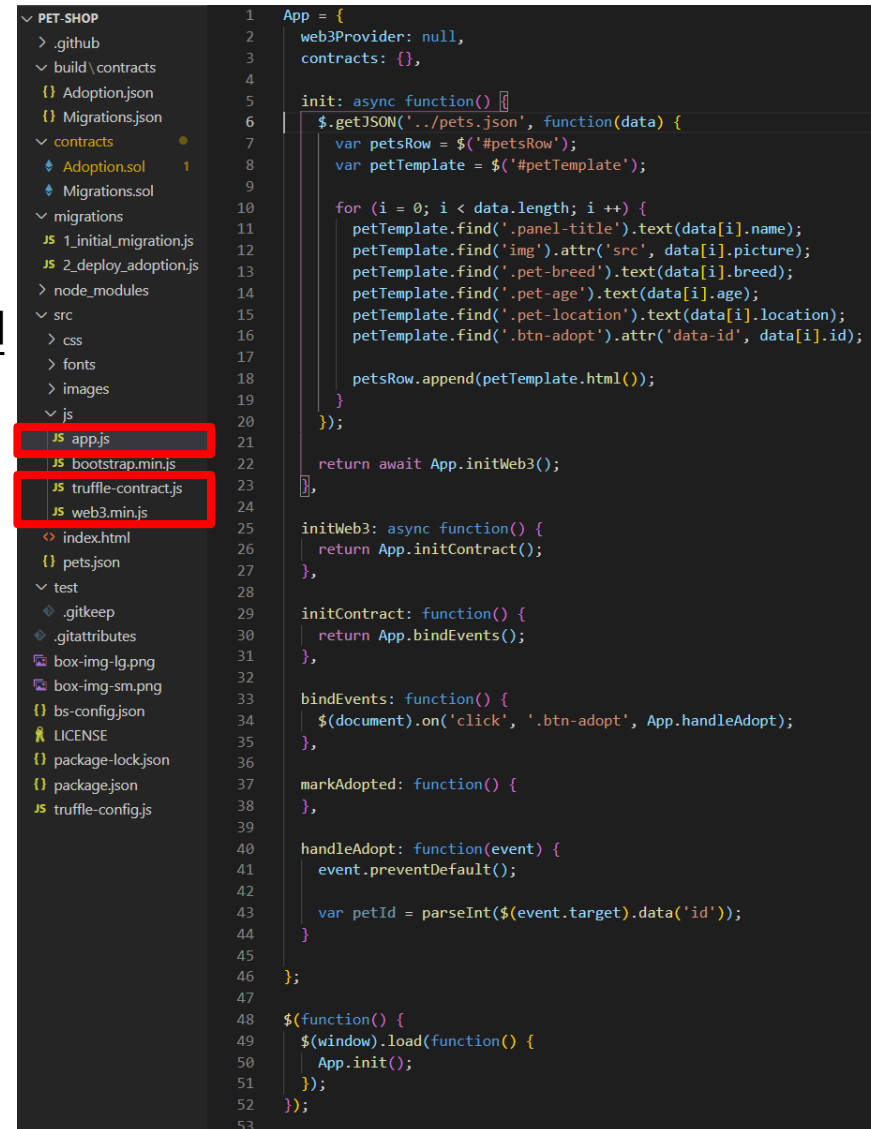
04 Solidity 개발 맛보기 (pet-shop)

app.js

우리는 우리의 Client(Front-End Application)이 블록체인 네트워크와 통신하기를 원한다.

➔ 그걸 도와주는 Front-End Library = Web3.js

truffle-contract.js도 web3.js 를 이용하고 있음.



```
1 App = {
2   web3Provider: null,
3   contracts: {},
4
5   init: async function() {
6     $.getJSON('../pets.json', function(data) {
7       var petsRow = $('#petsRow');
8       var petTemplate = $('#petTemplate');
9
10      for (i = 0; i < data.length; i++) {
11        petTemplate.find('.panel-title').text(data[i].name);
12        petTemplate.find('img').attr('src', data[i].picture);
13        petTemplate.find('.pet-breed').text(data[i].breed);
14        petTemplate.find('.pet-age').text(data[i].age);
15        petTemplate.find('.pet-location').text(data[i].location);
16        petTemplate.find('.btn-adopt').attr('data-id', data[i].id);
17
18        petsRow.append(petTemplate.html());
19      }
20    });
21
22    return await App.initWeb3();
23  },
24
25  initWeb3: async function() {
26    return App.initContract();
27  },
28
29  initContract: function() {
30    return App.bindEvents();
31  },
32
33  bindEvents: function() {
34    $(document).on('click', '.btn-adopt', App.handleAdopt);
35  },
36
37  markAdopted: function() {
38  },
39
40  handleAdopt: function(event) {
41    event.preventDefault();
42
43    var petId = parseInt($(event.target).data('id'));
44  }
45
46 };
47
48 $(function() {
49   $(window).load(function() {
50     App.init();
51   });
52 });
53
```

04 Solidity 개발 맛보기 (pet-shop)

index.html

```
<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
<!-- Include all compiled plugins (below), or include individual files as needed -->
<script src="js/bootstrap.min.js"></script>
<script src="js/web3.min.js"></script>
<script src="js/truffle-contract.js"></script>
<script src="js/app.js"></script>
</body>
```

HTML 파일

➔ 해당 html파일이 서버를 실행시켰을 때 나오는 파일.

- jquery: js를 쉽게 사용하도록 도와주는 라이브러리
- bootstrap: css 라이브러리(스타일시트)

➔ 해당 파일에서 web3와 app.js 를 불러오는 것을 알 수 있음.

04 Solidity 개발 맛보기 (pet-shop)

서버 시작하기

> npm run dev

[package.json]

node.js라이브러리를 관리하기 위한 패키지 정의 파일

- scripts: "npm run <명령어>" 로 실행 가능함
- dependencies: 의존하는 라이브러리.
- devDependencies: 개발에서만 의존 라이브러리.

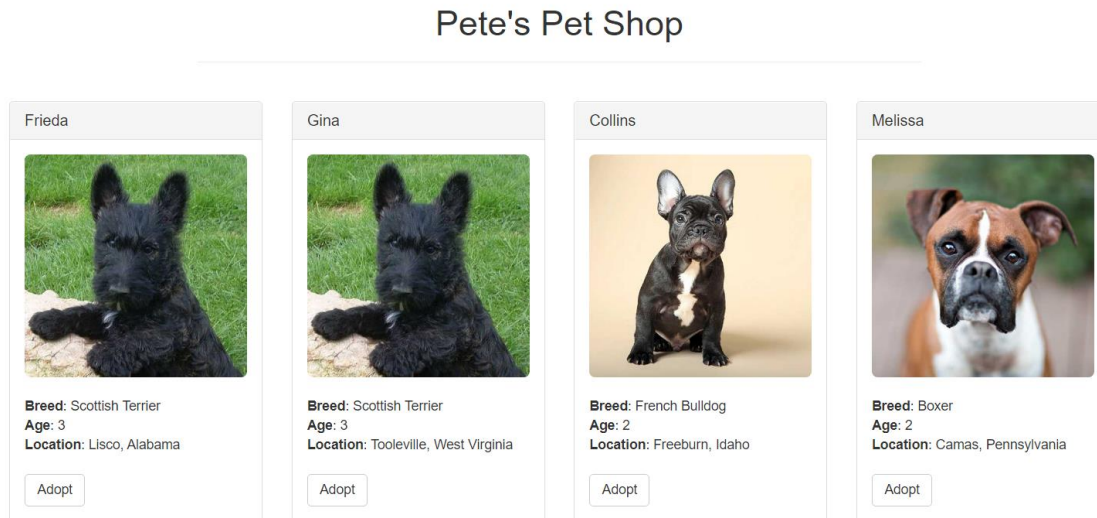
```
{ package.json > ...
1  {
2    "name": "pet-shop",
3    "version": "1.0.0",
4    "description": "",
5    "main": "truffle.js",
6    "directories": {
7      "test": "test"
8    },
9    "scripts": {
10     "dev": "lite-server",
11     "test": "echo \"Error: no test specified\" && exit 1"
12   },
13   "author": "",
14   "license": "ISC",
15   "devDependencies": {
16     "lite-server": "^2.3.0"
17   }
18 }
```

package.json

04 Solidity 개발 맛보기 (pet-shop)

서버 시작하기

> npm run dev



아직 어떤 것도 작동되지 않음.

➔ 아직 우리 js에서 통신하는 작업을 하지 않았기 때문.

04 Solidity 개발 맛보기 (pet-shop)

web3.js 적용하여 블록체인에 SmartContract 요청하기

[app.js] 수정

```
initWeb3: async function() {  
  if (window.ethereum) {  
    // metamask가 최근 설치된 Browser(크롬, firefox, edge ..)  
    App.web3Provider = window.ethereum;  
    try {  
      // metamask 요청  
      await window.ethereum.enable();  
    } catch (error) {  
      // account 요청 접근 제한  
      alert("account 접근 제한")  
    }  
  }  
  else if (window.web3) {  
    // 구버전 Metamask  
    App.web3Provider = window.web3.currentProvider;  
  }  
  else {  
    // web3 인스턴스가 탐지가 안되었을 때..  
    App.web3Provider = new Web3.providers.HttpProvider('http://localhost:8545');  
  }  
  web3 = new Web3(App.web3Provider);  
  return App.initContract();  
},
```

04 Solidity 개발 맛보기 (pet-shop)

web3.js 적용하여 블록체인에 SmartContract 요청하기

[app.js] 수정

```
49  initContract: function() {
50    $.getJSON('Adoption.json', function(data) {
51      // jquery 문법.
52      // Adoption.json을 가져 온다 (abi가 포함된 artifact 파일)
53
54      const AdoptionArtifact = data; // 객체화
55
56      // truffle-contract.js 이용.
57      App.contracts.Adoption = TruffleContract(AdoptionArtifact);
58
59      // Web3 Provider 셋팅
60      App.contracts.Adoption.setProvider(App.web3Provider);
61
62      // 컨트랙트를 call 한다음. adopted된걸 마킹하기 위해 호출.
63      return App.markAdopted();
64    });
65
66    return App.bindEvents();
67  },
```


04 Solidity 개발 맛보기 (pet-shop)

web3.js 적용하여 블록체인에 SmartContract 요청하기

[app.js] 수정

```
73 markAdopted: function() {
74     let adoptionInstance;
75
76     // deploy 되어 있는지 확인한 후.
77     App.contracts.Adoption.deployed().then(function(instance) {
78         adoptionInstance = instance;
79
80         // Adoption Contract에서 getAdopters 를 호출. (조회하기.)
81         return adoptionInstance.getAdopters.call();
82     }).then(function(adopters) {
83         for (i = 0; i < adopters.length; i++) {
84             if (adopters[i] !== '0x0000000000000000000000000000000000000000') {
85                 // 입양 된것은 입양 되었다고 표기하기
86                 $('#.panel-pet').eq(i).find('button').text('Success').attr('disabled', true);
87             }
88         }
89     }).catch(function(err) {
90         alert(err.message);
91     });
92 },
```

04 Solidity 개발 맛보기 (pet-shop)

web3.js 적용하여 블록체인에 SmartContract 요청하기

[app.js] 수정

```
handleAdopt: function(event) {
  event.preventDefault();

  const petId = parseInt($(event.target).data('id'));

  let adoptionInstance;

  // metmask에 등록된 account 가져오기
  web3.eth.getAccounts(function(error, accounts) {
    if (error) {
      console.log(error);
    }

    const account = accounts[0];

    App.contracts.Adoption.deployed().then(function(instance) {
      adoptionInstance = instance;


      // adopt contract function 호출.
      return adoptionInstance.adopt(petId, {from: account});
    }).then(function(result) {
      // 성공시 markAdopted 호출
      return App.markAdopted();
    }).catch(function(err) {
      // 에러시 알람.
      alert(err.message);
    });
  });
}
```

04 Solidity 개발 맛보기 (pet-shop)

web3.js 적용하여 블록체인에 SmartContract 요청하기

Pete's Pet Shop


Frieda



Breed: Scottish Terrier
Age: 3
Location: Lisco, Alabama

Success


Gina



Breed: Scottish Terrier
Age: 3
Location: Tooleville, West Virginia

Success

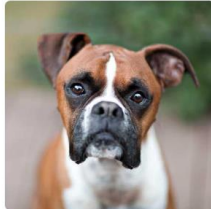
Collins



Breed: French Bulldog
Age: 2
Location: Freeburn, Idaho

Adopt

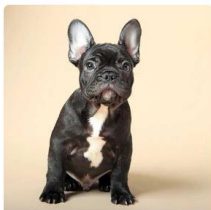
Melissa



Breed: Boxer
Age: 2
Location: Camas, Pennsylvania

Success

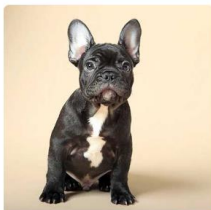
Jeanine



Breed: French Bulldog
Age: 2
Location: Gerber, South Dakota

Adopt


Elvia



Breed: French Bulldog
Age: 3
Location: Innsbrook, Illinois

Adopt


Latisha



Breed: Golden Retriever
Age: 3
Location: Soudan, Louisiana

Adopt

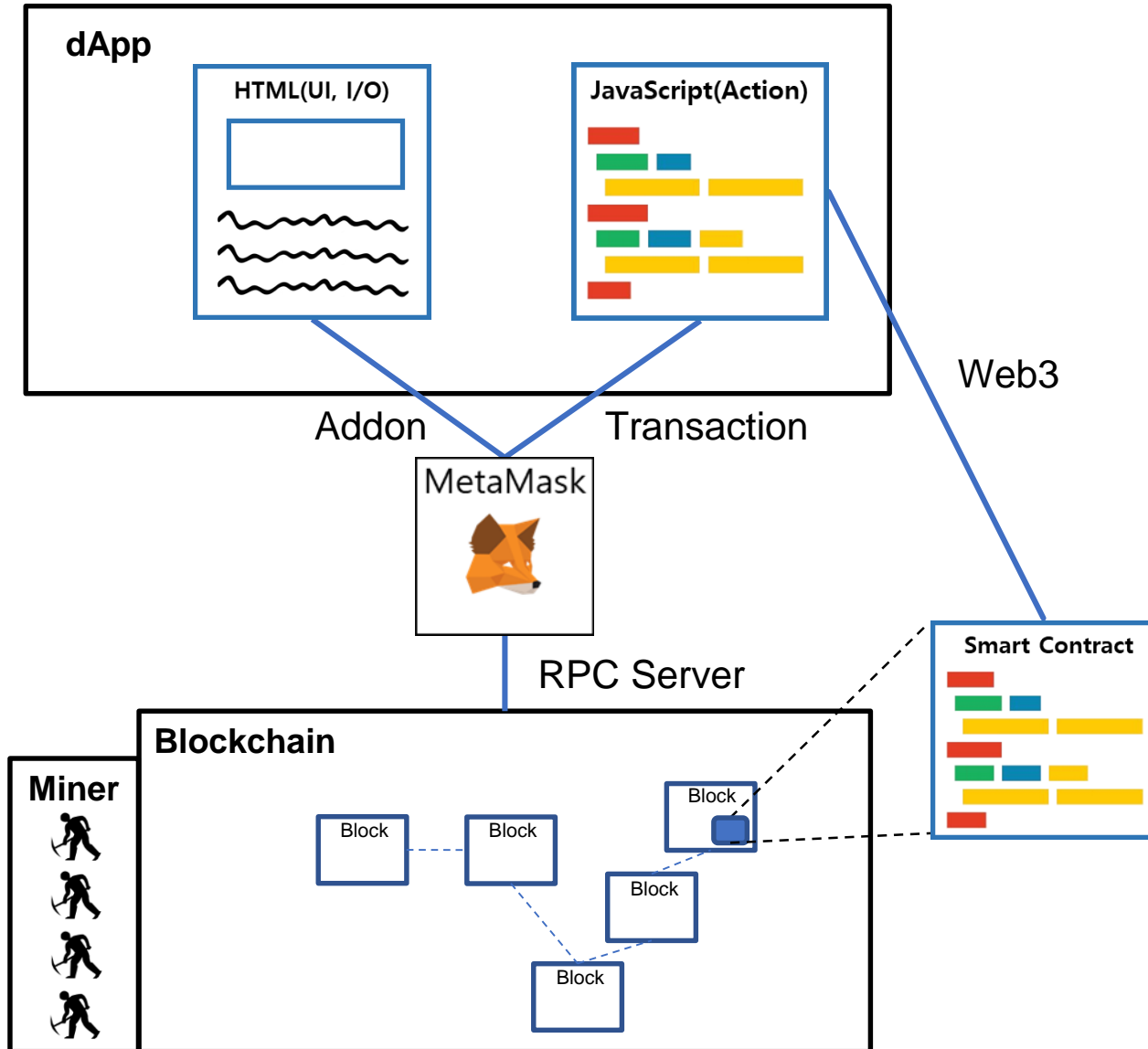
Coleman



Breed: Golden Retriever
Age: 3
Location: Jacksonwald, Palau

Success

04 Solidity 개발 맛보기 (pet-shop)



이더리움 dApp을 만드려면?

- HTML/CSS (보여지는 화면)
- javascript + Web3.js
- Solidity