



BLOCKCHAIN



1. Geth (go Ethereum)
2. Node.js
3. Git
4. vscode
5. node-gyp
6. browserify
7. http-server
8. windows-build-tools
9. web3
10. ipfs
11. buffer



1. Download geth from

<https://geth.ethereum.org/downloads/>

2. Click installation file



Downloads | Go Ethereum

https://geth.ethereum.org/downloads/

Go Ethereum Install Downloads Documentation

Download Geth – Lucky Leprechaun (v1.9.1) – Release Notes

You can download the latest 64-bit stable release of Geth for our primary platforms below. Packages for all supported platforms, as well as develop builds, can be found further down the page. If you're looking to install Geth and/or associated tools via your favorite package manager, please check our [installation guide](#).

[Geth 1.9.1 for Linux](#) [Geth 1.9.1 for macOS](#) **Geth 1.9.1 for Windows** [Geth 1.9.1 sources](#)

Specific Versions

If you're looking for a specific release, operating system or architecture, below you will find:

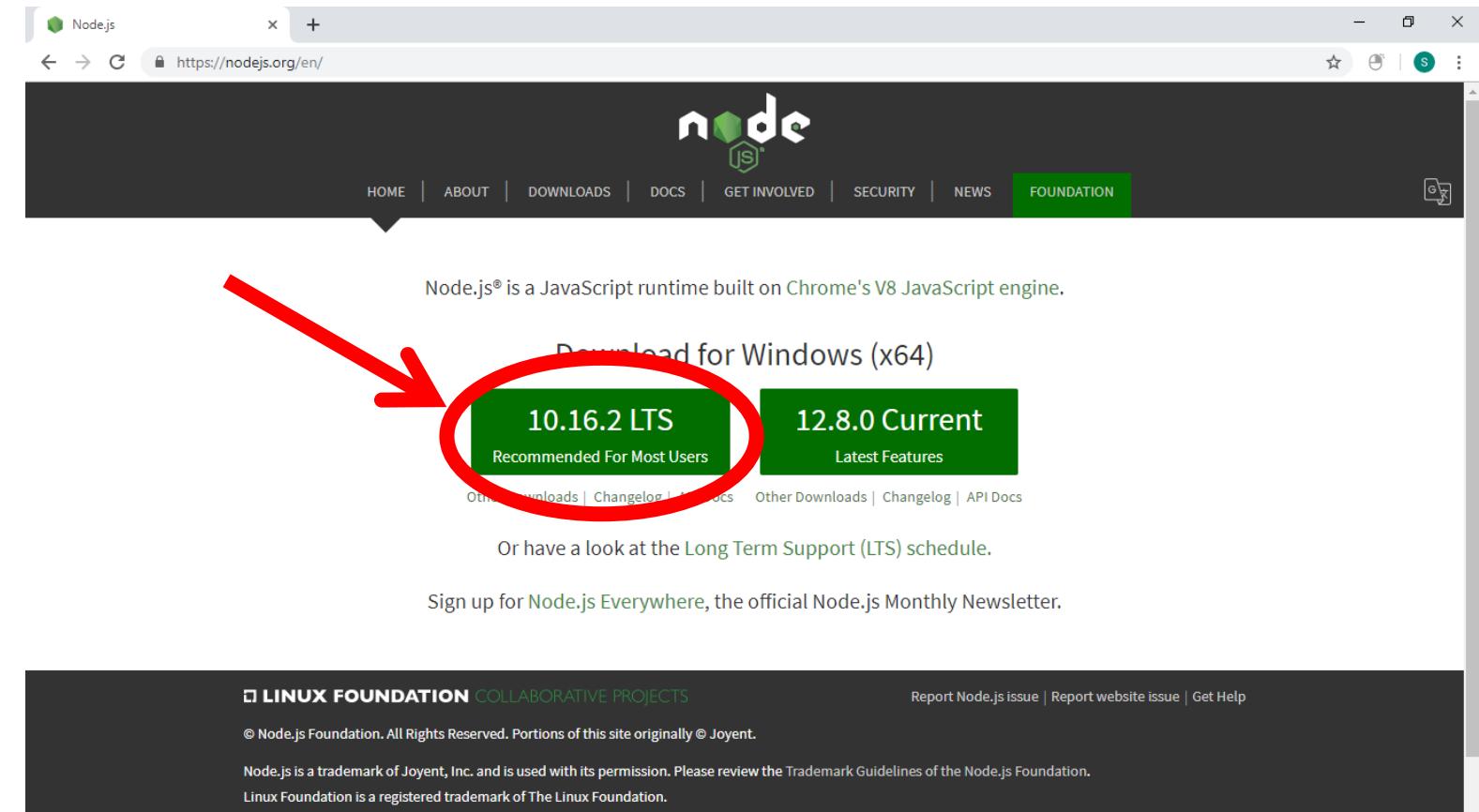
- All stable and develop builds of Geth and tools
- Archives for non-primary processor architectures
- Android library archives and iOS XCode frameworks

Please select your desired platform from the lists below and download your bundle of choice. Please be aware that the [MD5](#) checksums are provided by our binary hosting platform (Azure Blobstore) to help check for download errors. For security guarantees please verify any downloads via the attached PGP signature files (see [OpenPGP Signatures](#) for details).

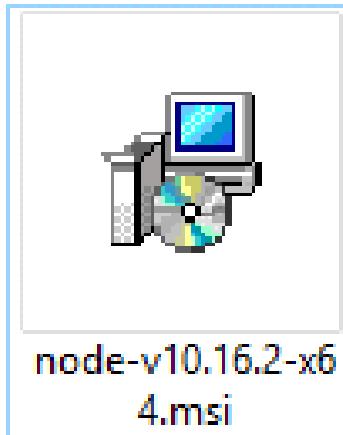
[Stable releases](#)



1. Download Nodejs from
<https://nodejs.org/en/>



The screenshot shows the official Node.js website at <https://nodejs.org/en/>. The page features a dark header with the Node.js logo and navigation links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, NEWS, and FOUNDATION. The main content area is titled "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine." It displays two prominent green buttons: "Download for Windows (x64)" and "10.16.2 LTS Recommended For Most Users". A large red arrow points to the "10.16.2 LTS" button, which is circled in red. Below these buttons, there are links for "Other Downloads | Changelog | API Docs" and "Latest Features". Further down, there's a link to the "Long Term Support (LTS) schedule" and a call to "Sign up for Node.js Everywhere, the official Node.js Monthly Newsletter". The footer contains links for "Report Node.js issue | Report website issue | Get Help", copyright information, and trademarks.



2. Click installation file



1. Download Git from

<https://git-scm.com/downloads>

2. Click installation file



The screenshot shows a web browser window with the URL git-scm.com/downloads. The page features the Git logo and the tagline "git --everything-is-local". On the left, there's a sidebar with links for About, Documentation, Downloads (which is currently selected), GUI Clients, Logos, and Community. Below the sidebar, there's a snippet from the "Pro Git" book. The main content area is titled "Downloads" and shows sections for Mac OS X, Windows, and Linux/Unix. A large image of a computer monitor displays the "Latest source Release 2.22.0" and a "Download 2.22.0 for Windows" button, which is circled in red. A red arrow points from the "Downloads" section towards this button. To the right of the monitor image, there are sections for "GUI Clients" (with a link to "View GUI Clients") and "Logos" (with a link to "View Logos"). At the bottom, there's a section titled "Git via Git" with a link to "git clone https://github.com/git/git".

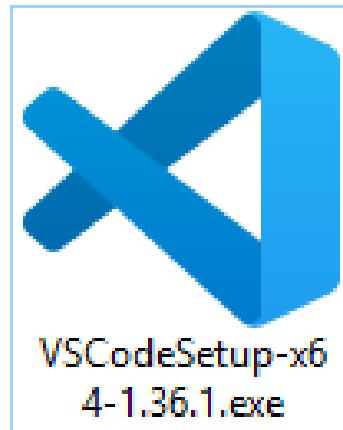


1. Download vscode from

[https://code.visualstudio.com/
/download](https://code.visualstudio.com/download)

1.1 select System Installer

2. Click installation file



The screenshot shows a web browser displaying the Visual Studio Code download page at <https://code.visualstudio.com/download>. The page header says "Version 1.36 is now available! Read about the new features and fixes from June." Below the header, there's a main title "Download Visual Studio Code" and a subtitle "Free and built on open source. Integrated Git, debugging and extensions." On the right side of the page, there are download links for different operating systems: Windows, Linux (Ubuntu), and macOS. A red arrow points to the "Windows" section, specifically to the "System Installer" link, which is circled in red. The "System Installer" link is followed by "64 bit" and "32 bit".



1. Download Bracket Pair Colorize Extension

The screenshot shows the Visual Studio Code interface with the Extensions Marketplace open. The sidebar on the left has icons for File, Edit, Selection, View, Go, Debug, Terminal, and Help. A red circle labeled '1' highlights the Extensions icon. The main area shows the 'EXTENSIONS: MARKETPLACE' tab selected, with the search bar containing 'bracket pair colorize'. A red circle labeled '2' highlights the search result for 'Bracket Pair Colo...'. The right side shows the details for the 'Bracket Pair Colorizer' extension by CoenraadS, with a red circle labeled '3' highlighting the green 'Install' button.

File Edit Selection View Go Debug Terminal Help

Extension: Bracket Pair Colorizer - Visual Studio Code

EXTENSIONS: MARKETPLACE 2

bracket pair colorize

Bracket Pair Colo... 1.0.61 ⚡ 4M ★ 4.5
A customizable extension for coloriz...
CoenraadS Install

Bracket Pair Col... 0.0.28 ⚡ 422K ★ 4.5
A customizable extension for coloriz...
CoenraadS Install

colorize 0.8.11 ⚡ 311K ★ 4
A vscode extension to help visualize ...
kamikillerto Install

bracket-padder 0.1.1 ⚡ 3K
Smart whitespace padding & auto-c...
Viable lab Install

Bracket Jumper 1.1.8 ⚡ 8K ★ 5
Bracket-based editor navigation and...
sashawei Install

Extension: Bracket Pair Colorizer

Bracket Pair Colorizer coenraads.bracket-pair-colorizer

CoenraadS | ⚡ 4,068,227 | ★★★★☆ | Repository | License

A customizable extension for colorizing matching brackets

Install 3

Details Contributions Changelog

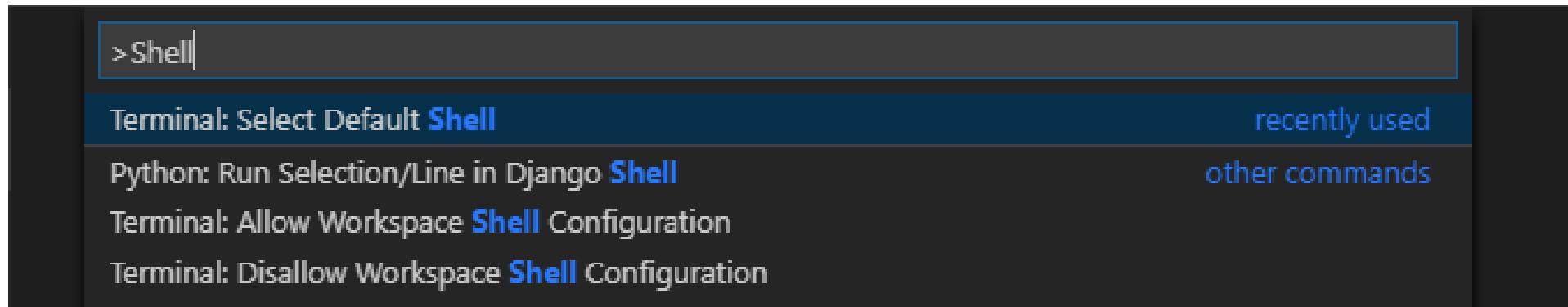
Bracket Pair Colorizer

Announcement: A new version is being developed at <https://github.com/CoenraadS/Bracket-Pair-Colorizer-2>

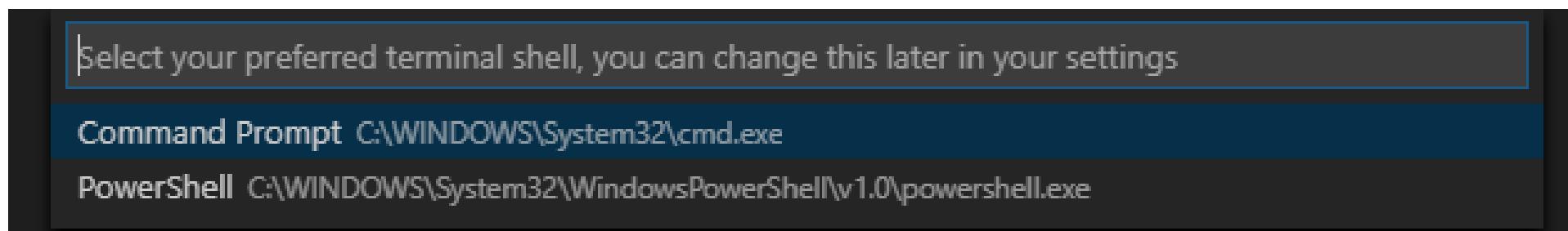


2. Set default shell

1. press **F1** or **Ctrl+Shift+P** and typing/selecting **Terminal Select Default Shell**



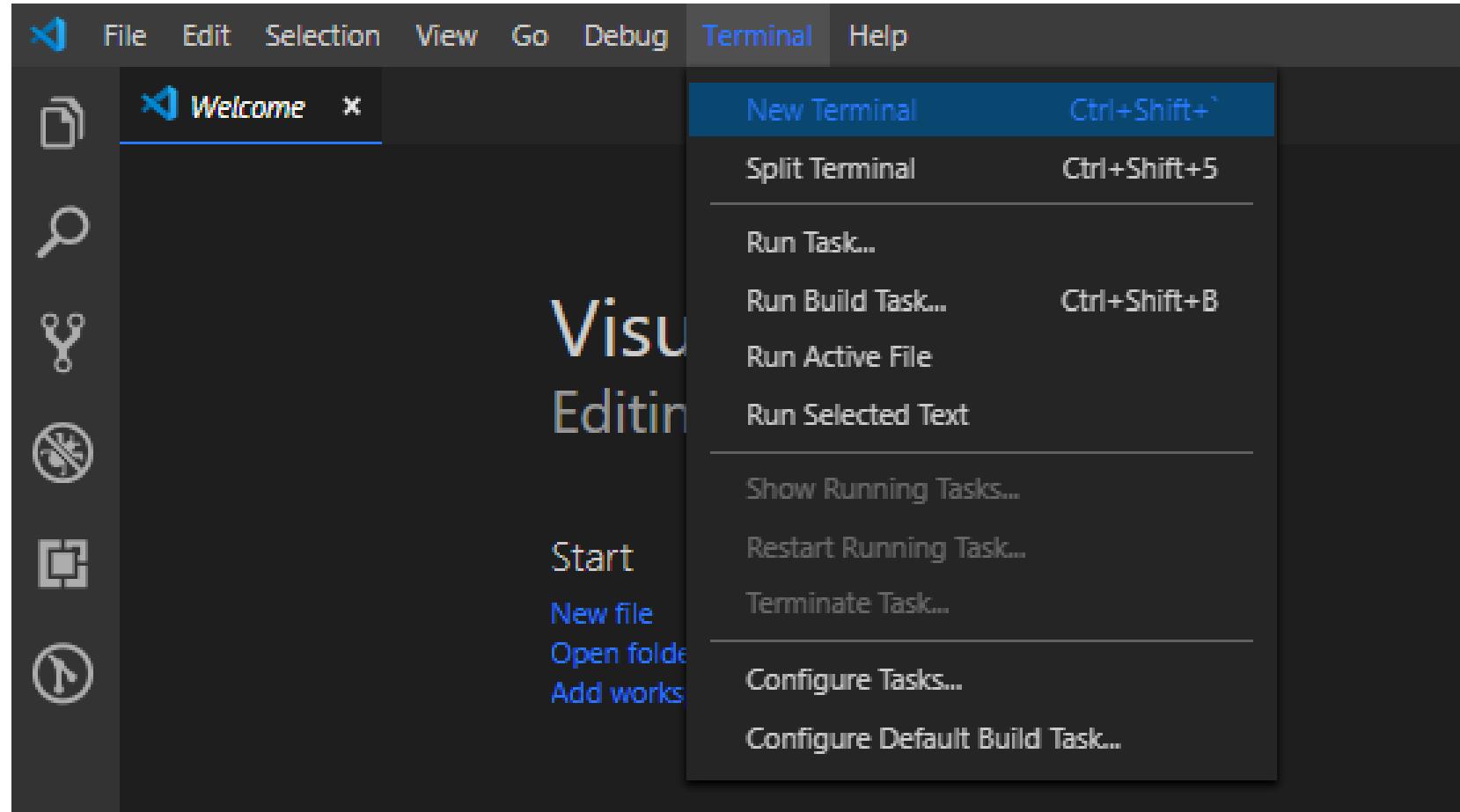
2. Select Command Prompt





install node-gyp

1. Open Terminal





install node-gyp

2. Run command

C:\User> **npm i -g node-gyp**

The screenshot shows a terminal window with a dark background and white text. At the top, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL, with TERMINAL being the active tab. To the right of the tabs is a toolbar with icons for file operations. The main area of the terminal displays the following text:

```
Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Suppakorn>npm i -g node-gyp
C:\Users\Suppakorn\AppData\Roaming\npm\node-gyp -> C:\Users\Suppakorn\AppData\Roaming\npm\node_modules\node-gyp\bin\node-gyp.js
+ node-gyp@5.0.3
updated 1 package in 4.093s
```



install browserify

1. Run command

C:\User> **npm i -g browserify**

The screenshot shows a terminal window with a dark background and light-colored text. At the top, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL, with TERMINAL being the active tab. To the right of the tabs is a tab bar labeled '1: node' with icons for new, close, and others. The main area of the terminal displays the following text:

```
Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Suppakorn>npm i -g browserify
C:\Users\Suppakorn\AppData\Roaming\npm\browserify -> C:\Users\Suppakorn\AppData\Roaming\npm\node_modules\browserify\bin\cmd.js
+ browserify@16.5.0
added 137 packages from 109 contributors in 18.188s
```



1. Run command

```
C:\User> npm i -g http-server
```

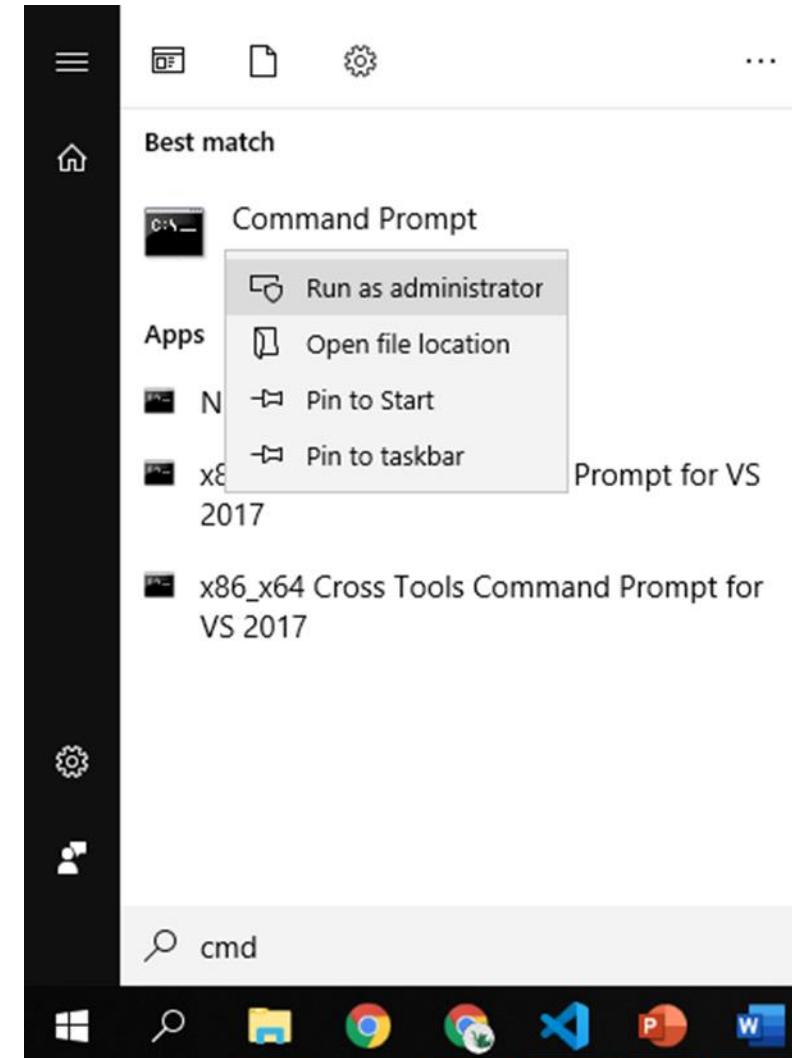
The screenshot shows a terminal window with the following details:

- Header: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (underlined), 1: cmd, +, □, △, ▲, ▾, X.
- Text area:
 - Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.
 - C:\Users\Suppakorn>npm i -g http-server
 - C:\Users\Suppakorn\AppData\Roaming\npm\http-server -> C:\Users\Suppakorn\AppData\Roaming\npm\node_modules\http-server\bin\http-server
 - C:\Users\Suppakorn\AppData\Roaming\npm\hs -> C:\Users\Suppakorn\AppData\Roaming\npm\node_modules\http-server\bin\http-server
 - + http-server@0.11.1
 - added 26 packages from 28 contributors in 6.142s
- Bottom line: C:\frontend>



install windows-build-tools

1. Open Command prompt as administrator





install windows-build-tools

2. Run command

C:\User>npm i -g windows-build-tools

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>npm i -g windows-build-tools

> windows-build-tools@5.2.2 postinstall C:\Users\Suppakorn\AppData\Roaming\npm\node_modules\windows-build-tools
> node ./dist/index.js

Downloading python-2.7.15.amd64.msi
[> ] 0.0% (0 B/s)
Downloaded python-2.7.15.amd64.msi. Saved to C:\Users\Suppakorn\.windows-build-tools\python-2.7.15.amd64.msi.
Downloading vs_BuildTools.exe
[> ] 0.0% (0 B/s)
Downloaded vs_BuildTools.exe. Saved to C:\Users\Suppakorn\.windows-build-tools\vs_BuildTools.exe.

Starting installation...
Launched installers, now waiting for them to finish.
This will likely take some time - please be patient!

Status from the installers:
----- Visual Studio Build Tools -----
Successfully installed Visual Studio Build Tools.
----- Python -----
Successfully installed Python 2.7

Now configuring the Visual Studio Build Tools and Python...

All done!

+ windows-build-tools@5.2.2
added 145 packages from 99 contributors in 1424.029s

C:\WINDOWS\system32>
```

อาจใช้เวลาติดตั้งเกิน 20 นาที
ห้ามปิด
ห้ามกด Ctrl+C



1. make 2 folders

 **getlab**

 **frontend**

2. back to the Terminal

3. run command

C:\User> cd C:\frontend

```
C:\Users\Suppakorn>cd C:\frontend
```

```
C:\frontend>
```



1. run command

C:\frontend> **npm init**

```
C:\frontend>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (frontend)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to C:\frontend\package.json:

{
  "name": "frontend",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}

Is this OK? (yes)
C:\frontend>
```



2. Run command

C:\frontend>npm i web3

```
C:\frontend>npm i web3

> sha3@1.2.3 install C:\frontend\node_modules\sha3
> node-gyp rebuild

C:\frontend\node_modules\sha3>if not defined npm_config_node_gyp (node "C:\Program Files\nodejs\node_modules\npm\node_modules\npm-lifecycle\node-gyp-bin\\..\..\node_modules\node-gyp\bin\n
(node "C:\Program Files\nodejs\node_modules\npm\node_modules\node-gyp\bin\node-gyp.js" rebuild )
Building the projects in this solution one at a time. To enable parallel build, please add the "/m" switch.
  addon.cpp
  displayIntermediateValues.cpp
  KeccakF-1600-reference.cpp
  KeccakNISTInterface.cpp
  KeccakSponge.cpp
  win delay load hook.cc
c:\frontend\node_modules\websocket\src\validation.cc(133): warning C4996: 'v8::Value::ToObject': was declared deprecated [C:\frontend\node_modules\websocket\build\validation.vcxproj]
  c:\users\suppakorn\.node-gyp\10.16.2\include\node\v8.h(10046): note: see declaration of 'v8::Value::ToObject'
    Creating library C:\frontend\node_modules\websocket\build\Release\Validation.lib and object C:\frontend\node_modules\websocket\build\Release\Validation.exp
  Generating code
All 121 functions were compiled because no usable IPDB/IOBJ from previous compilation was found.
  Finished generating code
  validation.vcxproj -> C:\frontend\node_modules\websocket\build\Release\Validation.node
npm notice created a lockfile as package-lock.json. You should commit this file.
npm warn frontend@1.0.0 No description
npm warn frontend@1.0.0 No repository field.
npm warn optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\fsevents):
npm warn notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ web3@1.2.1
added 623 packages from 406 contributors and audited 117991 packages in 122.727s
found 1 low severity vulnerability
  run `npm audit fix` to fix them, or `npm audit` for details

C:\frontend>
```



1. Run command

C:\frontend>**npm i ipfs**

```
C:\frontend>npm i ipfs

> gc-stats@1.4.0 install C:\frontend\node_modules\gc-stats
> node-pre-gyp install --fallback-to-build

node-pre-gyp [WARN] Using request for node-pre-gyp https download
[gc-stats] Success: "C:\frontend\node_modules\gc-stats\build\gcstats\v1.4.0\Release\node-v64-win32-x64\gcstats.node" is installed via remote

> keccak@1.4.0 install C:\frontend\node_modules\keccak
> npm run rebuild || echo "Keccak bindings compilation fail. Pure JS implementation will be used."

> keccak@1.4.0 rebuild C:\frontend\node_modules\keccak
> node-gyp rebuild

> assemblyscript@0.6.0 postinstall C:\frontend\node_modules\assemblyscript
> opencollective-postinstall || exit 0

Thank you for using assemblyscript!
If you rely on this package, please consider supporting our open collective:
> https://opencollective.com/assemblyscript/donate

npm [WARN] frontend@1.0.0 No description
npm [WARN] frontend@1.0.0 No repository field.
npm [WARN] optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\fsevents):
npm [WARN] notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ ipfs@0.37.0
added 782 packages from 683 contributors and audited 135454 packages in 237.214s
found 4 vulnerabilities (2 low, 2 high)
  run `npm audit fix` to fix them, or `npm audit` for details

C:\frontend>
```



install buffer

1. Run command

C:\frontend>**npm i buffer**

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3: cmd + - ⌂ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉ ×

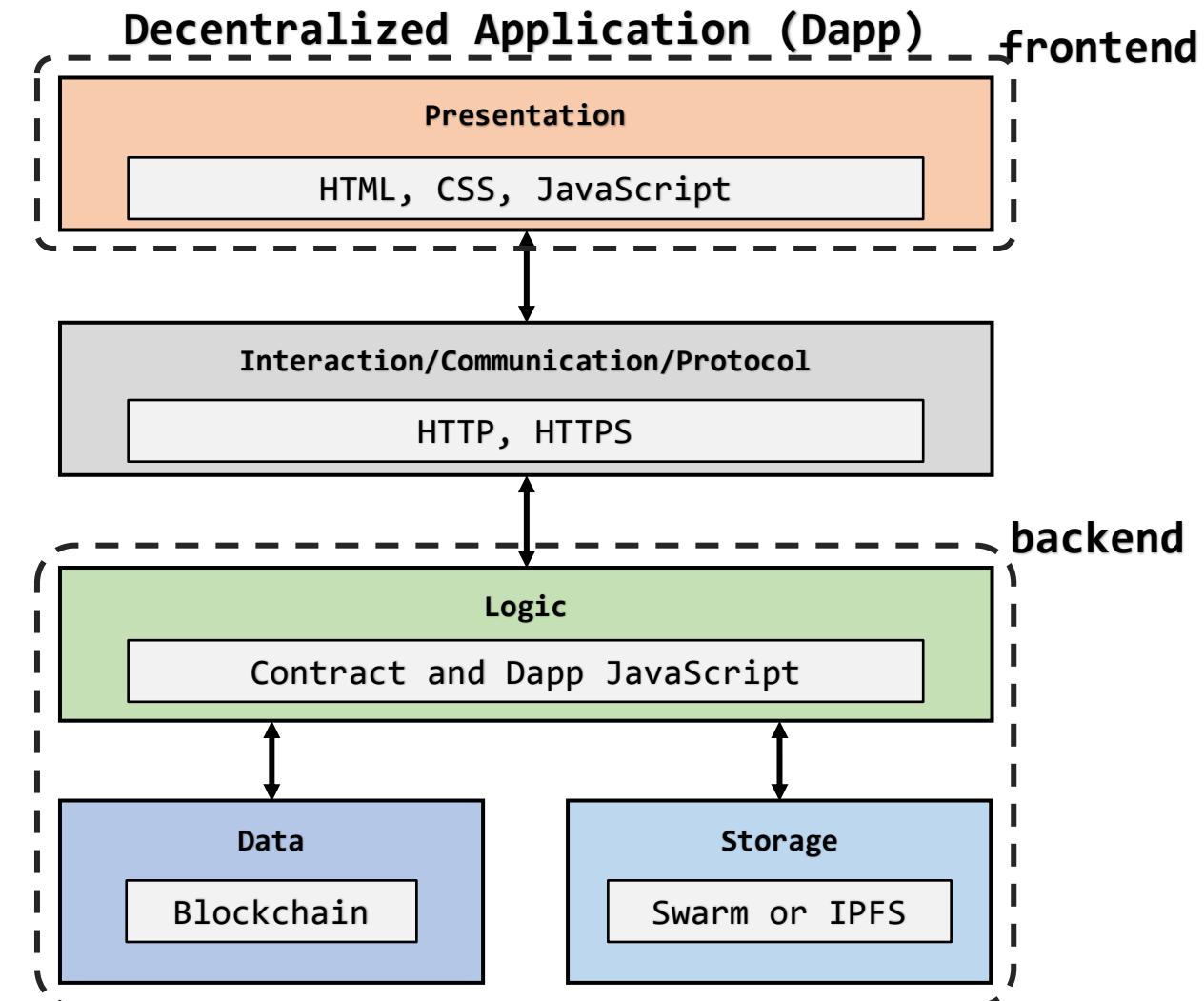
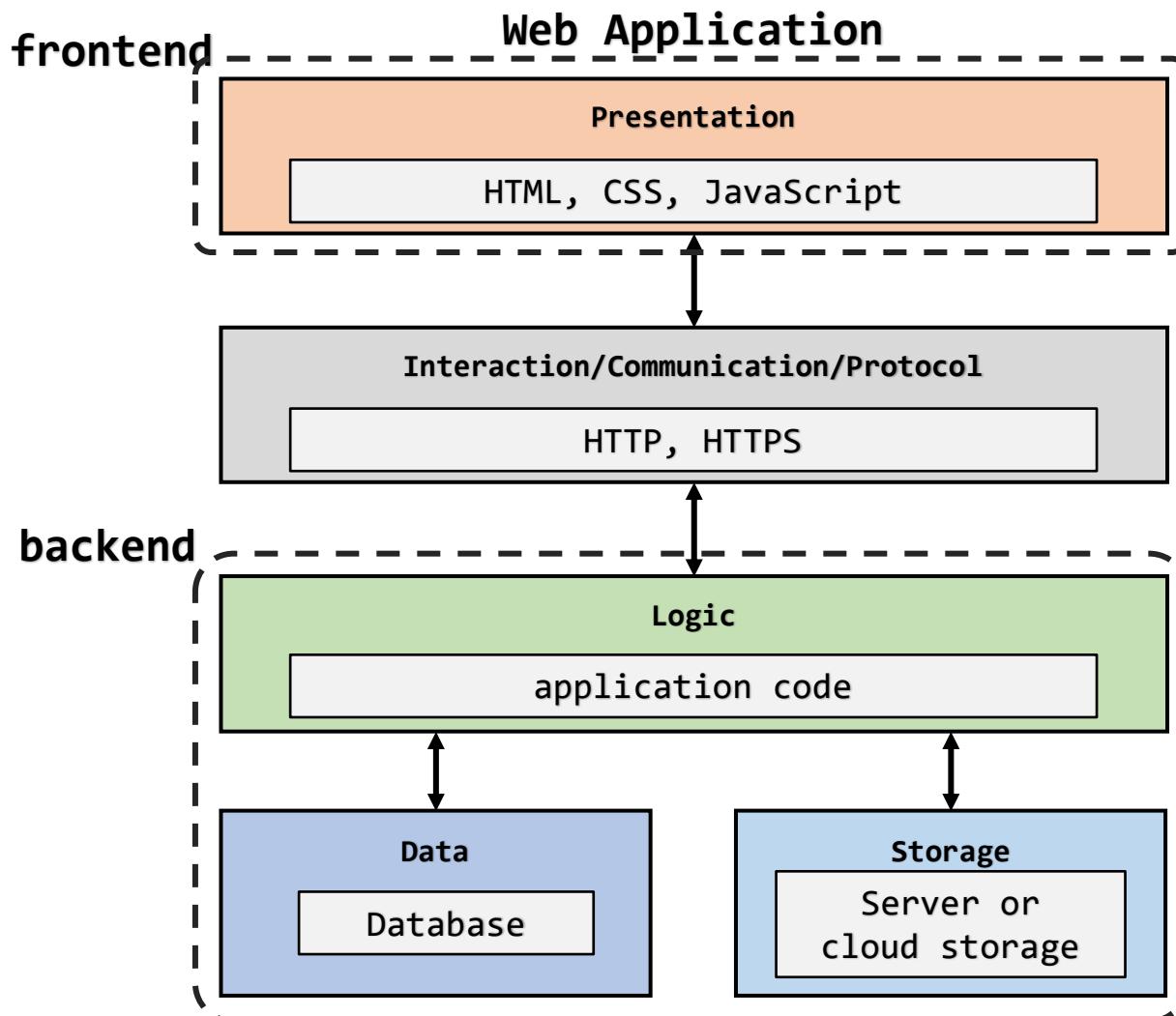
Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.

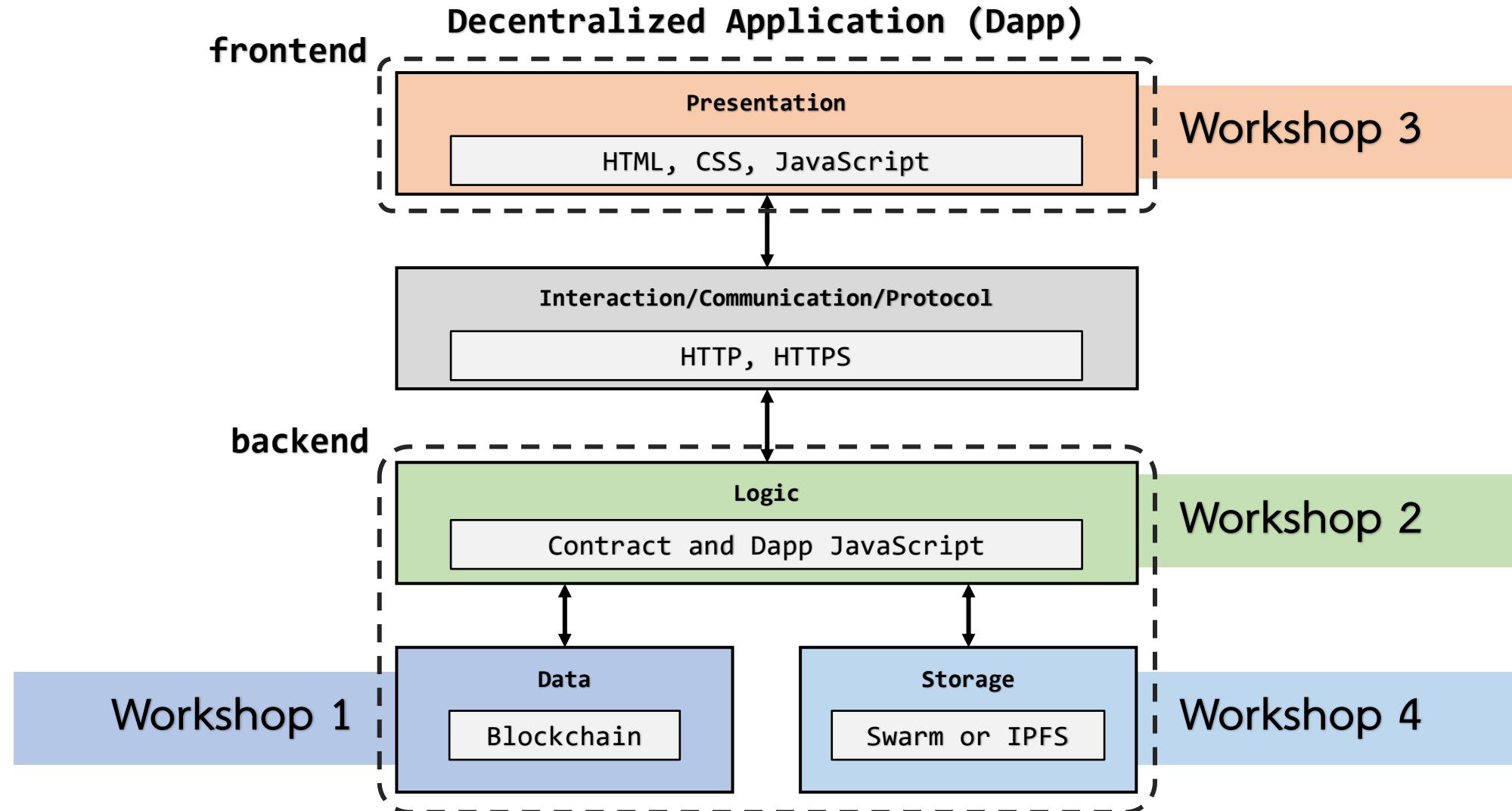
C:\frontend>npm i buffer
npm WARN frontend@1.0.0 No description
npm WARN frontend@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ buffer@5.4.0
updated 1 package and audited 135457 packages in 37.694s
found 4 vulnerabilities (2 low, 2 high)
  run `npm audit fix` to fix them, or `npm audit` for details

C:\frontend>
```

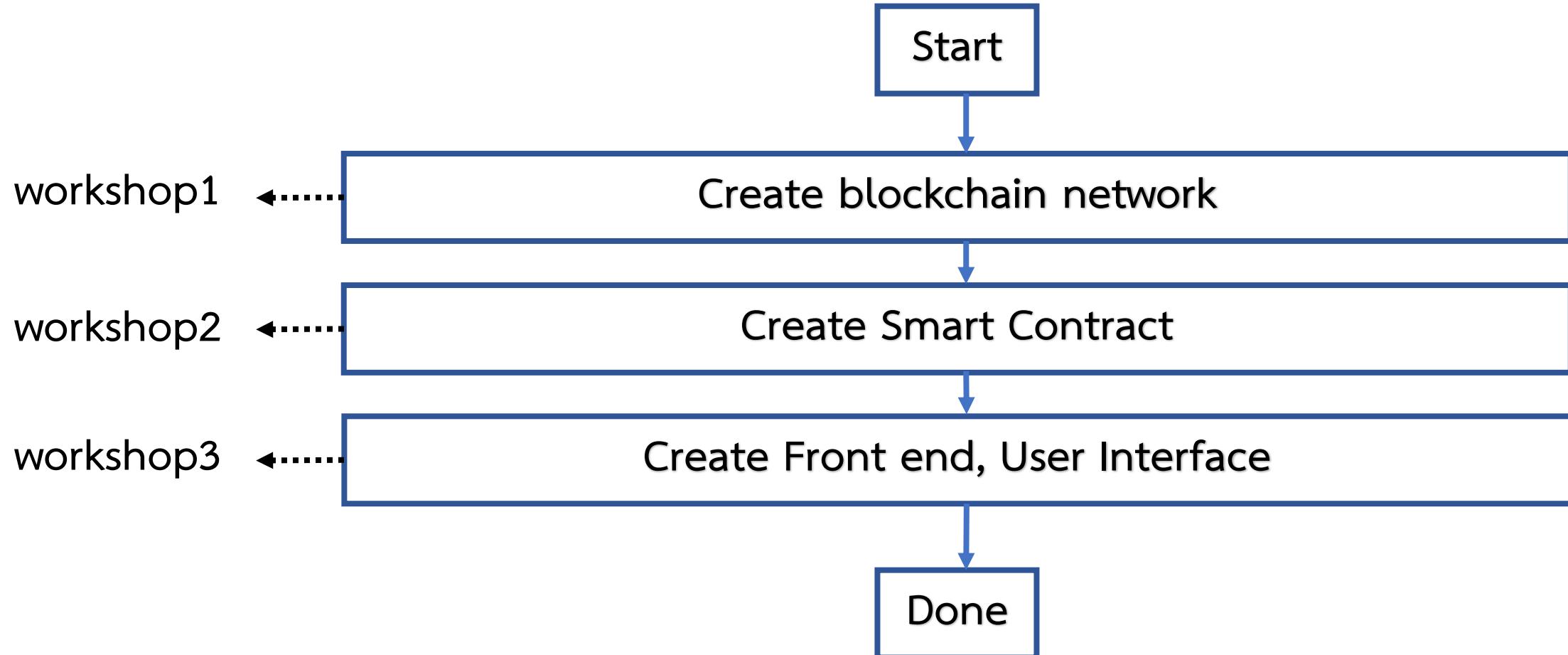
Ln 109, Col 9 Spaces: 4 UTF-8 CRLF HTML 🎧 1







Course Outline



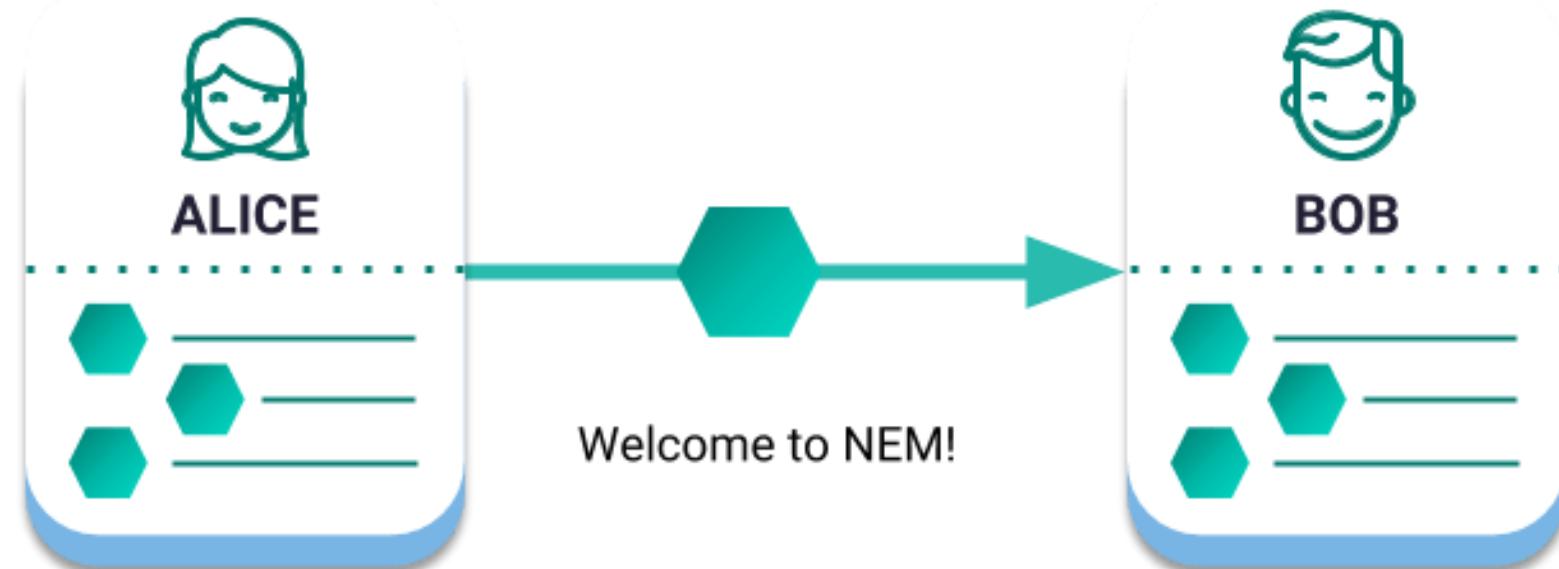




Transaction

Blockchain transaction can be defined as a small unit of task that is stored in public records.

- send transaction (โอนเหรียญ)
- send data
- query data
- edit data
- deploy smart contract





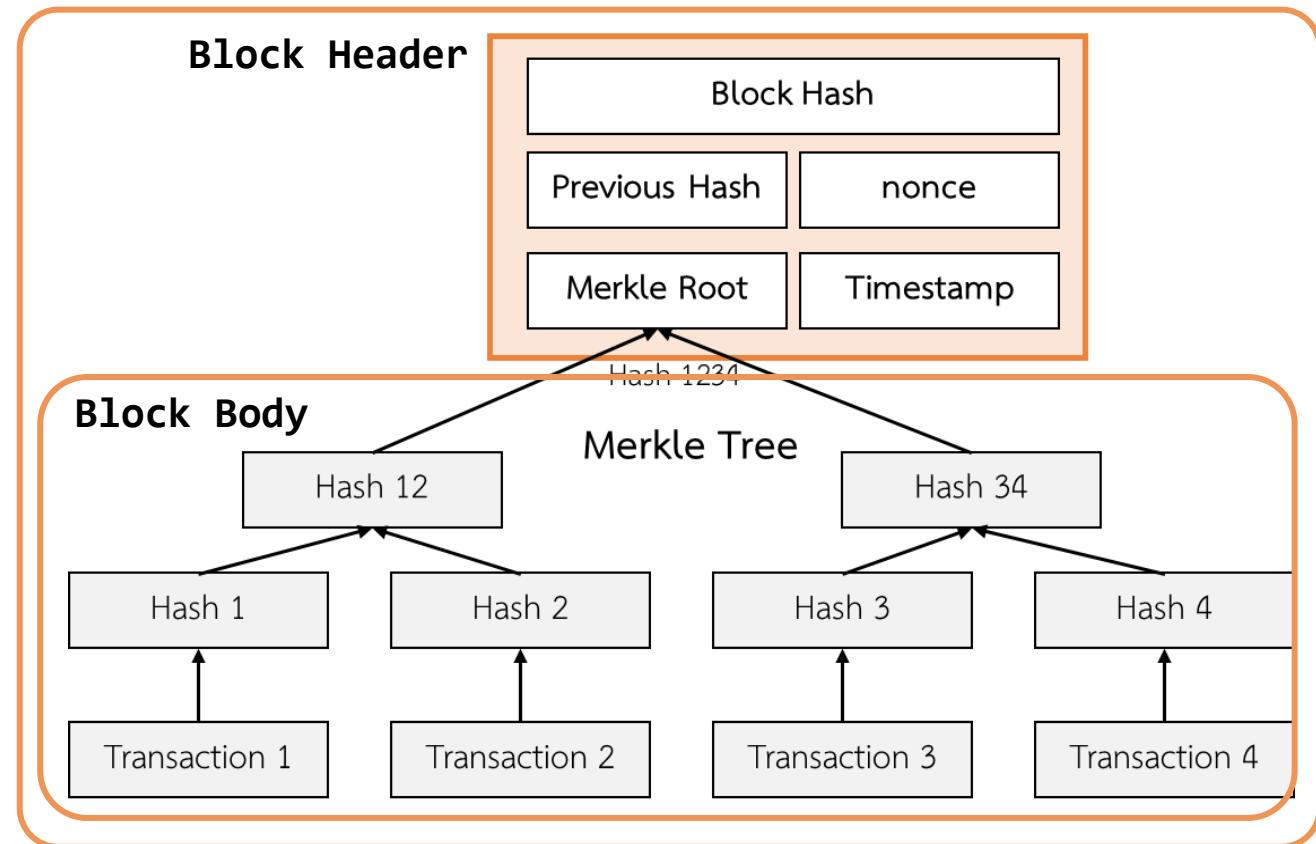
Block header

- 1.the hash of this block
- 2.the hash of the previous block
- 3.the root hash of the Merkle tree (Merkle Root)
- 4.the time in seconds since 1970-01-01 T00: 00 UTC
- 5.the nonce

Block Body

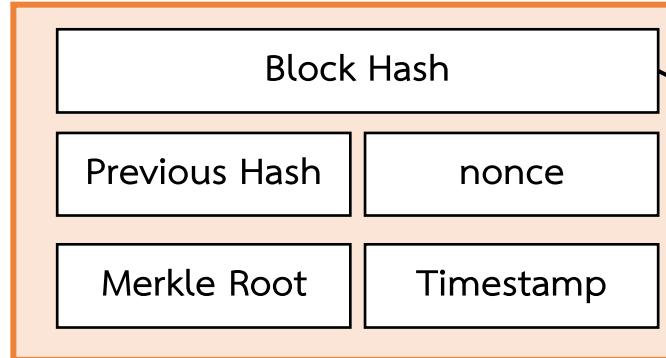
The block body is conceivable as the loading space of a truck. It contains all transactions that are confirmed with the block. The transactions in a block are stored in Merkle Tree.

Block

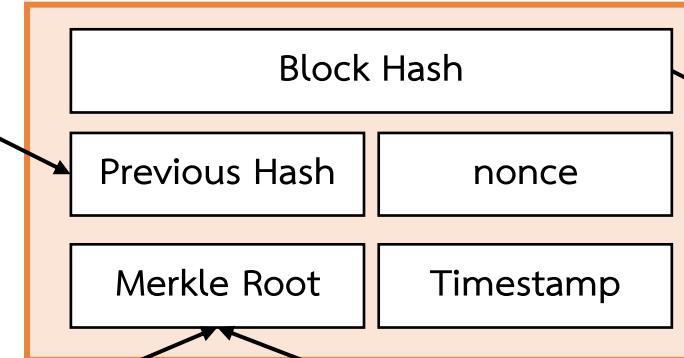




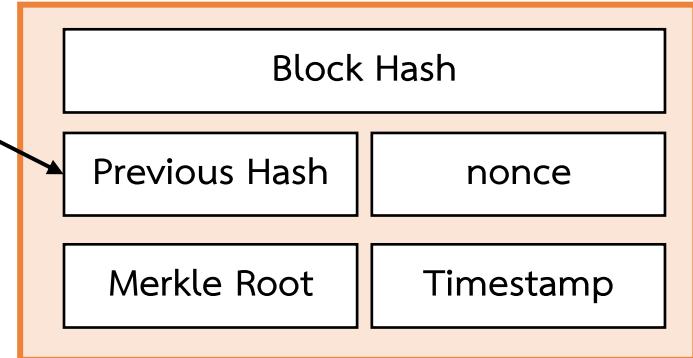
Block 1



Block 2

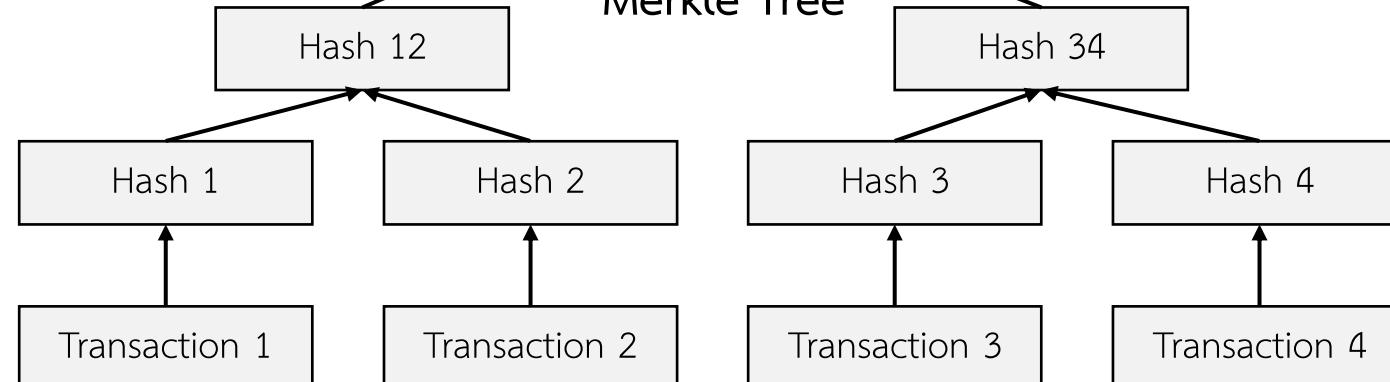


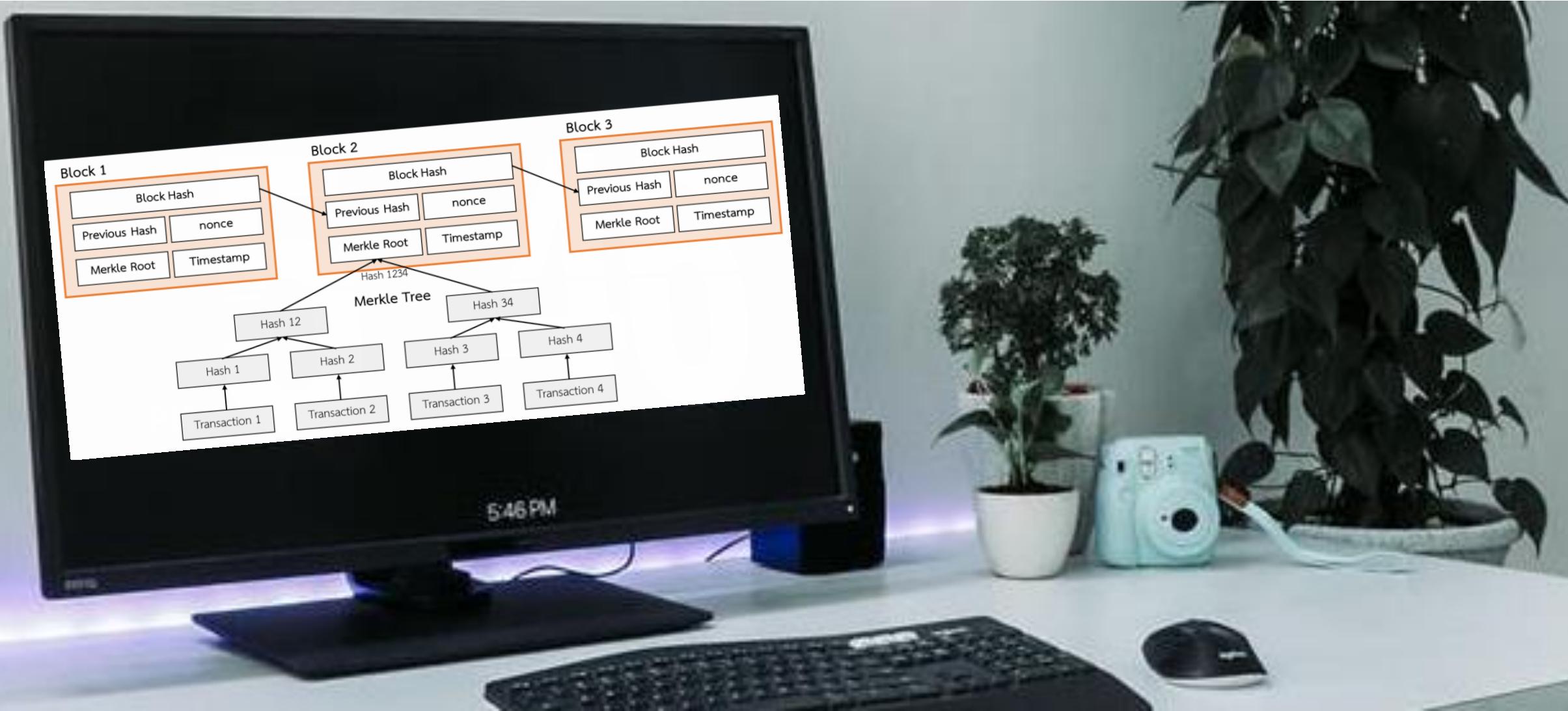
Block 3

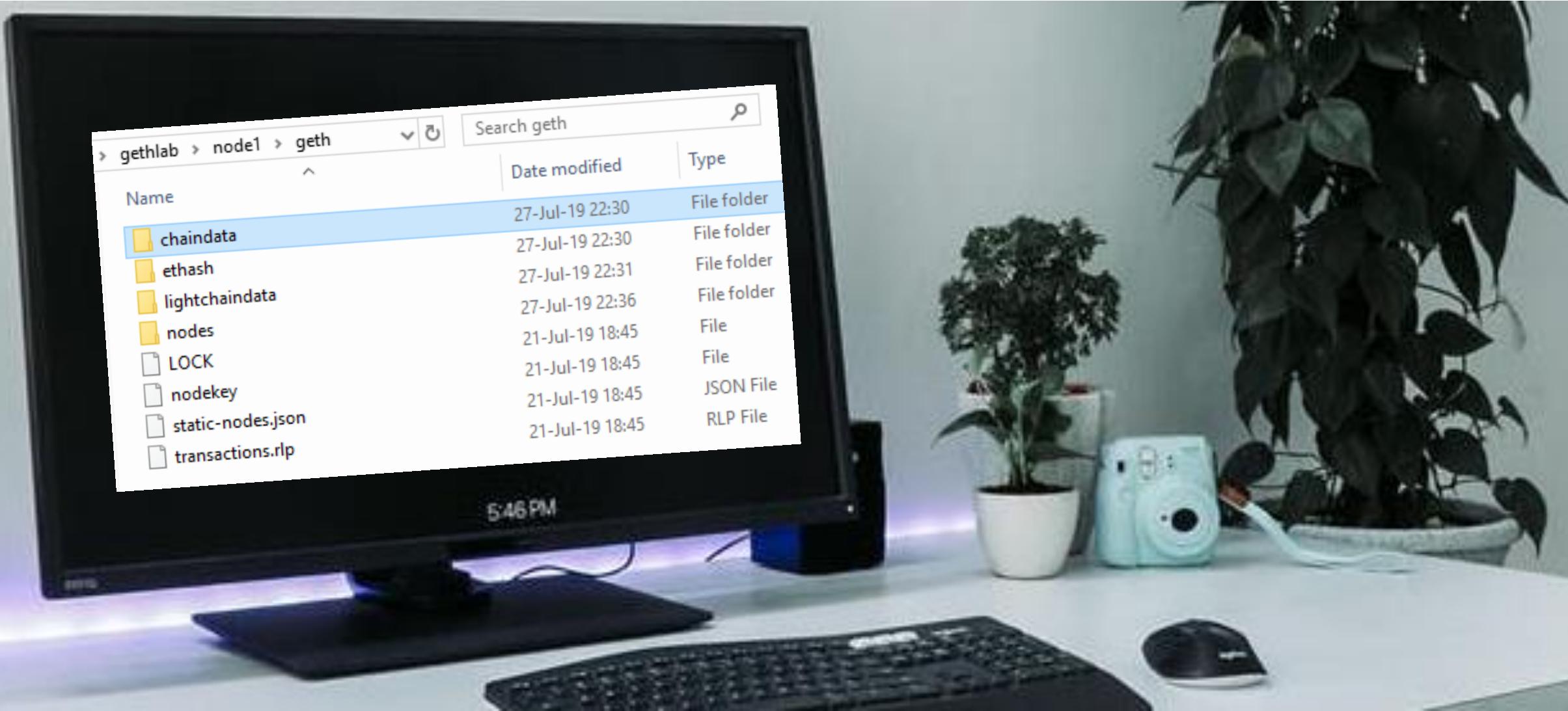


Hash 1234

Merkle Tree



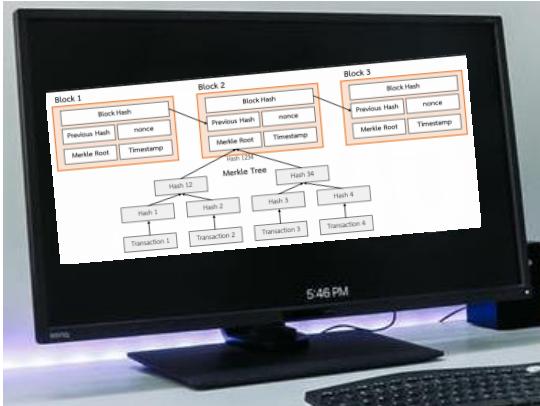




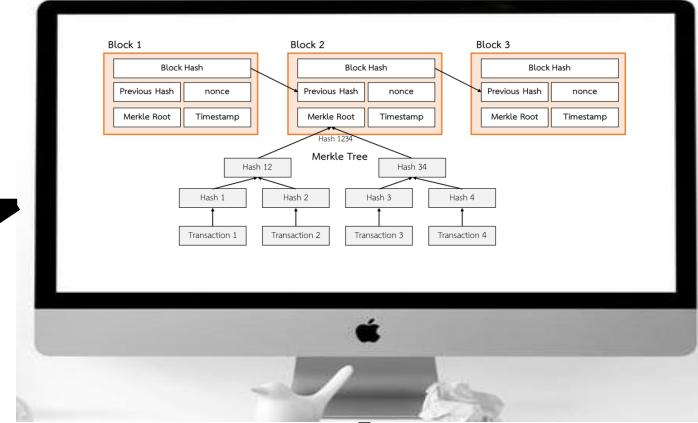


Blockchain Network

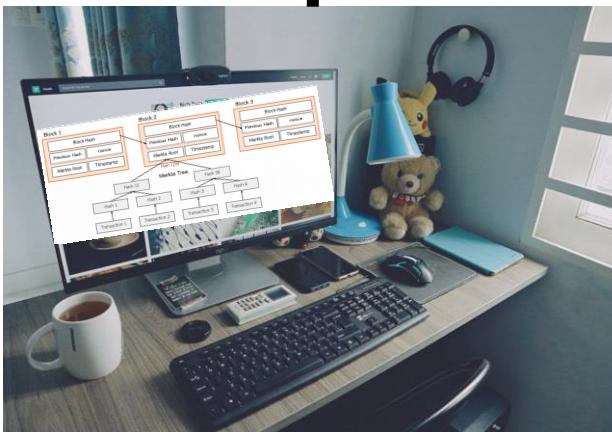
Node



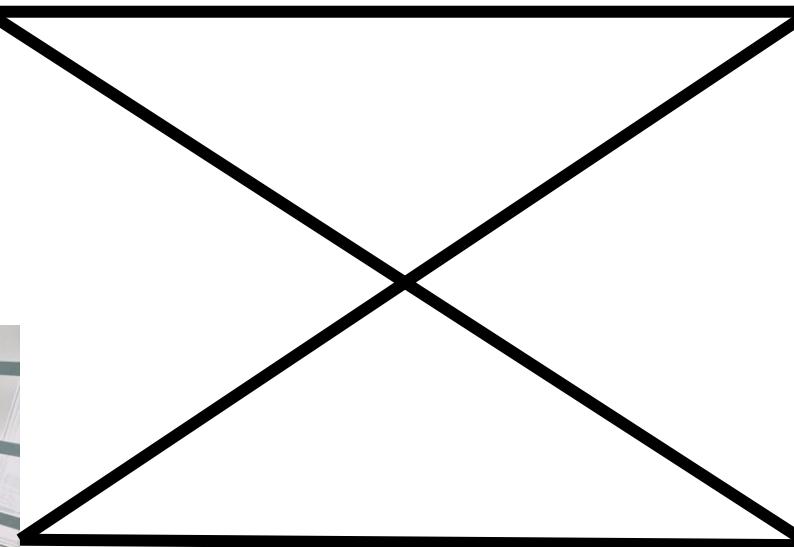
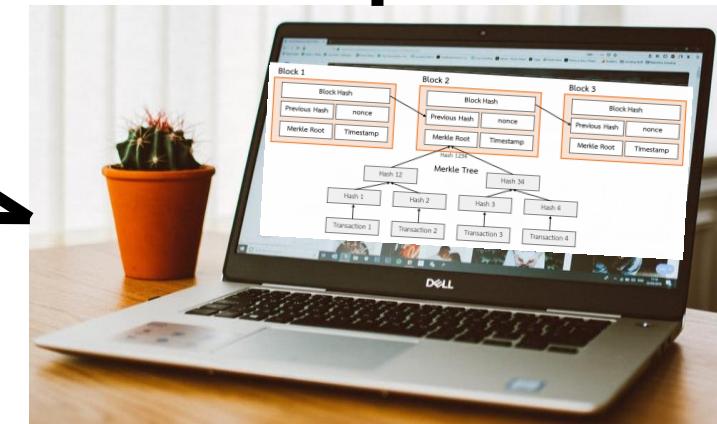
Node

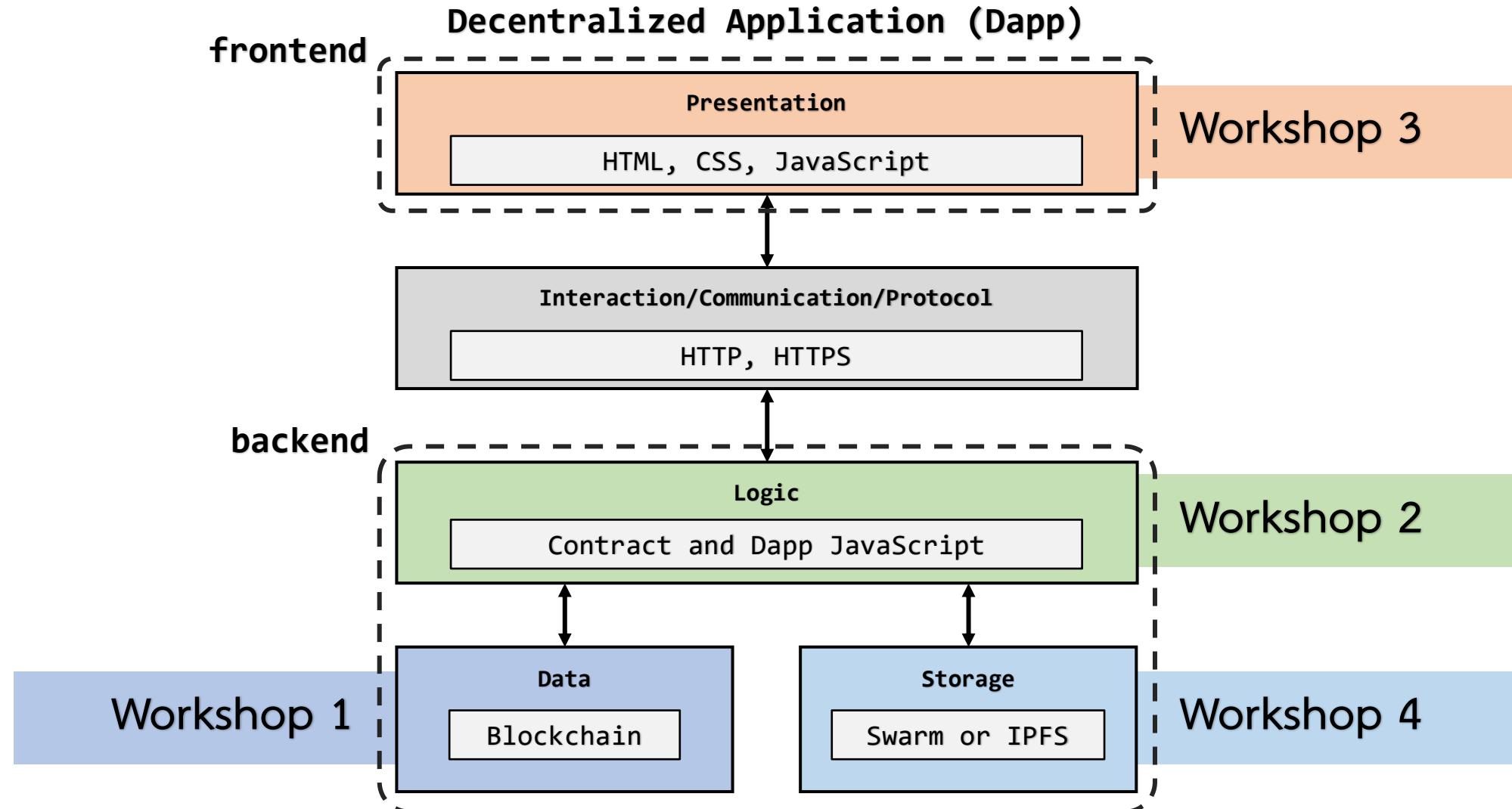


Node



Node





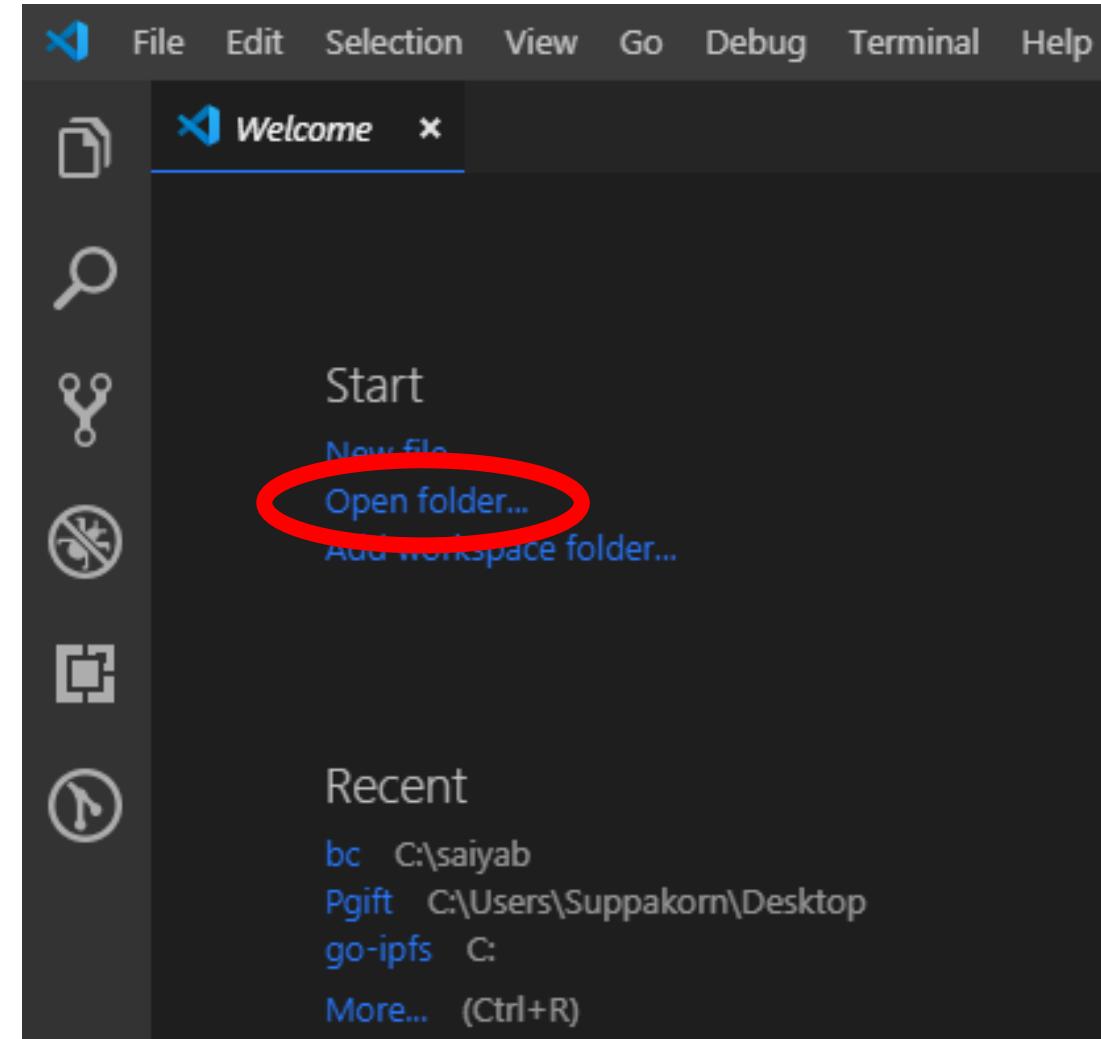
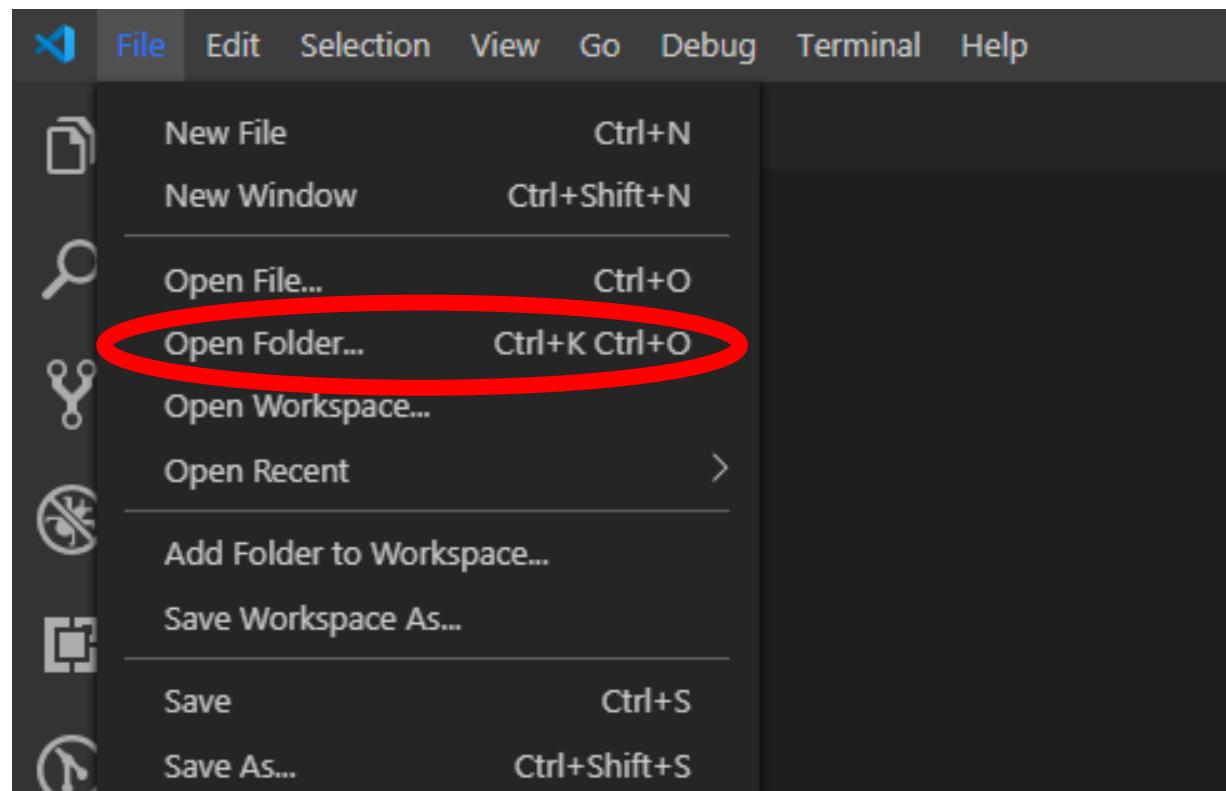


workshop นี้ทำอะไร

1. create new local blockchain network (1 node) on personal laptop using Geth (open source)
2. Blockchain ที่สร้างได้เป็น blockchain เปล่าๆ ไม่มีอะไรเลย
 - 2.1 เราจะมาเริ่มกันที่สร้างบัญชี 2 บัญชีภายใน 1 node โดยหมายเลขบัญชีจะเรียกว่า Wallet Address
 - 2.2 แล้วเราจะขุดหรือยืนกัน เป็นหรือยูที่ไม่มีค่า แต่ผมอยากรู้ว่าทุกคนมองว่าหรือยูนี้คือข้อมูล
 - 2.2.1 ตอนขุดจะกินเวลาค่อนข้างนาน ผมจะใช้เวลาช่วงนี้อธิบายว่าการขุดคืออะไร และสัมพันธ์กับด้านsecurity ของblockchain ยังไง
3. พอมีบัญชี และมีหรือยูแล้ว เราจะมาโอนหรือยู (ส่งข้อมูล) จากบัญชีที่หนึ่งไปยังบัญชีที่สอง
4. เราจะสร้าง node ขึ้นมาอีกหนึ่งอัน แล้วก็จะเอามันมาเชื่อมต่อกับโนนดแรก ก็จะได้ blockchain network(2 nodes)
5. แล้วเราจะโอนหรือยู(ส่งข้อมูล)ข้ามโนนดกัน



1. Open vscode > File > Open Folder > C:\gethlab





The screenshot shows the Visual Studio Code interface with the 'Welcome' tab selected in the top bar. The left sidebar includes the Explorer, Open Editors (with 'Welcome' listed), and Gethlab sections. The main area displays the 'Welcome' page with sections for Start, Recent, Help, and Learn. The 'Start' section contains links for 'New file', 'Open folder...', and 'Add workspace folder...'. The 'Recent' section lists recent projects like 'bc C:\saiyab', 'P gift C:\Users\Suppakorn\Desktop', and 'go-ipfs C:'. The 'Help' section links to 'Printable keyboard cheatsheet', 'Introductory videos', 'Tips and Tricks', 'Product documentation', 'GitHub repository', 'Stack Overflow', and 'Join our Newsletter'. The 'Learn' section links to 'Find and run all commands', 'Interface overview', and 'Interactive playground'. At the bottom, there is a checkbox for 'Show welcome page on startup' and a status bar with icons for battery, signal, and notifications.

File Edit Selection View Go Debug Terminal Help

Welcome - gethlab - Visual Studio Code

EXPLORER

OPEN EDITORS

Welcome

GETHLAB

Start

New file

Open folder...

Add workspace folder...

Recent

bc C:\saiyab

P gift C:\Users\Suppakorn\Desktop

go-ipfs C:

More... (Ctrl+R)

OUTLINE

The active editor cannot provide outline information.

Help

Printable keyboard cheatsheet

Introductory videos

Tips and Tricks

Product documentation

GitHub repository

Stack Overflow

Join our Newsletter

Show welcome page on startup

Customize

Tools and languages

Install support for JavaScript, TypeScript, Python, PHP, Azure, Docker and ...

Settings and keybindings

Install the settings and keyboard shortcuts of Vim, Sublime, Atom and oth...

Color theme

Make the editor and your code look the way you love

Learn

Find and run all commands

Rapidly access and search commands from the Command Palette (Ctrl+Sh...

Interface overview

Get a visual overlay highlighting the major components of the UI

Interactive playground

Try essential editor features out in a short walkthrough

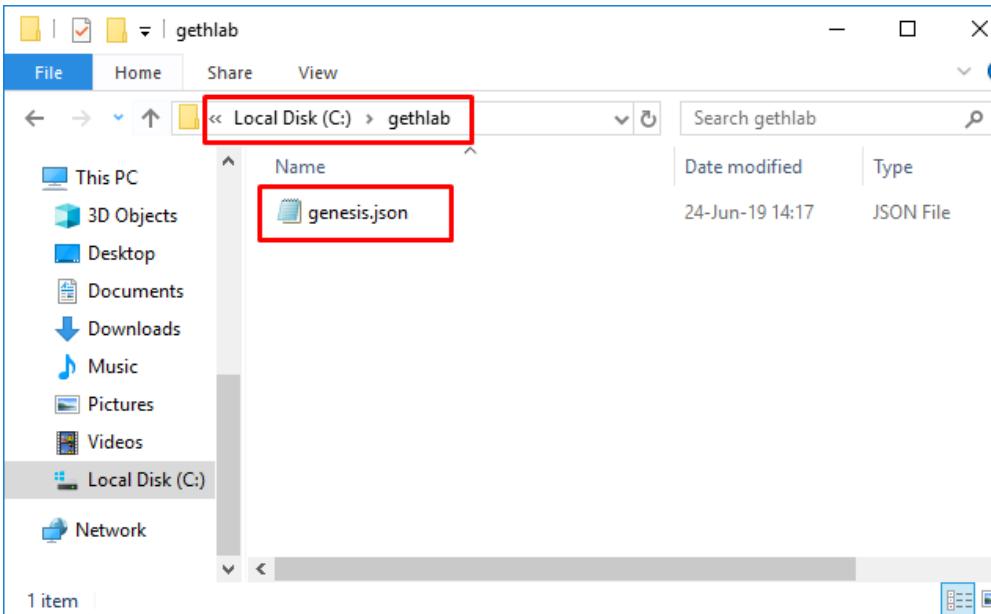
0 0 0

1



2. File > New File > paste code

3. Save file as **genesis.json** in folder **C:\gethlab**



```
{  
  "config": {  
    "ChainID": 10,  
    "HomesteadBlock": 0,  
    "DAOForkBlock": 0,  
    "EIP150Block": 0,  
    "EIP155Block": 0,  
    "EIP158Block": 0,  
    "ByzantiumBlock": 0,  
    "ConstantinopleBlock": 0,  
    "PetersburgBlock": 0,  
    "IstanbulBlock": 0  
  },  
  "alloc" : {},  
  "coinbase" : "0x0000000000000000000000000000000000000000000000000000000000000000",  
  "difficulty" : "0x0010",  
  "extraData" : "",  
  "gasLimit" : "0x5F5E100",  
  "nonce" : "0x0000000000000000",  
  "mixhash" : "0x0000000000000000000000000000000000000000000000000000000000000000",  
  "parentHash" : "0x0000000000000000000000000000000000000000000000000000000000000000",  
  "timestamp" : "0x00"  
}
```

<https://medium.com/coinmonks/ethereum-setting-up-a-private-blockchain-67bbb96cf4f1>

<https://github.com/ethereum/go-ethereum/blob/master/params/config.go>



The screenshot shows a Visual Studio Code interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Debug, Terminal, Help.
- Title Bar:** genesis.json - gethlab - Visual Studio Code.
- Explorer:** Shows "OPEN EDITORS" with "genesis.json" selected, and "GETHLAB" with "genesis.json" also listed.
- Editor:** Displays the JSON content of the genesis.json file. The code is as follows:

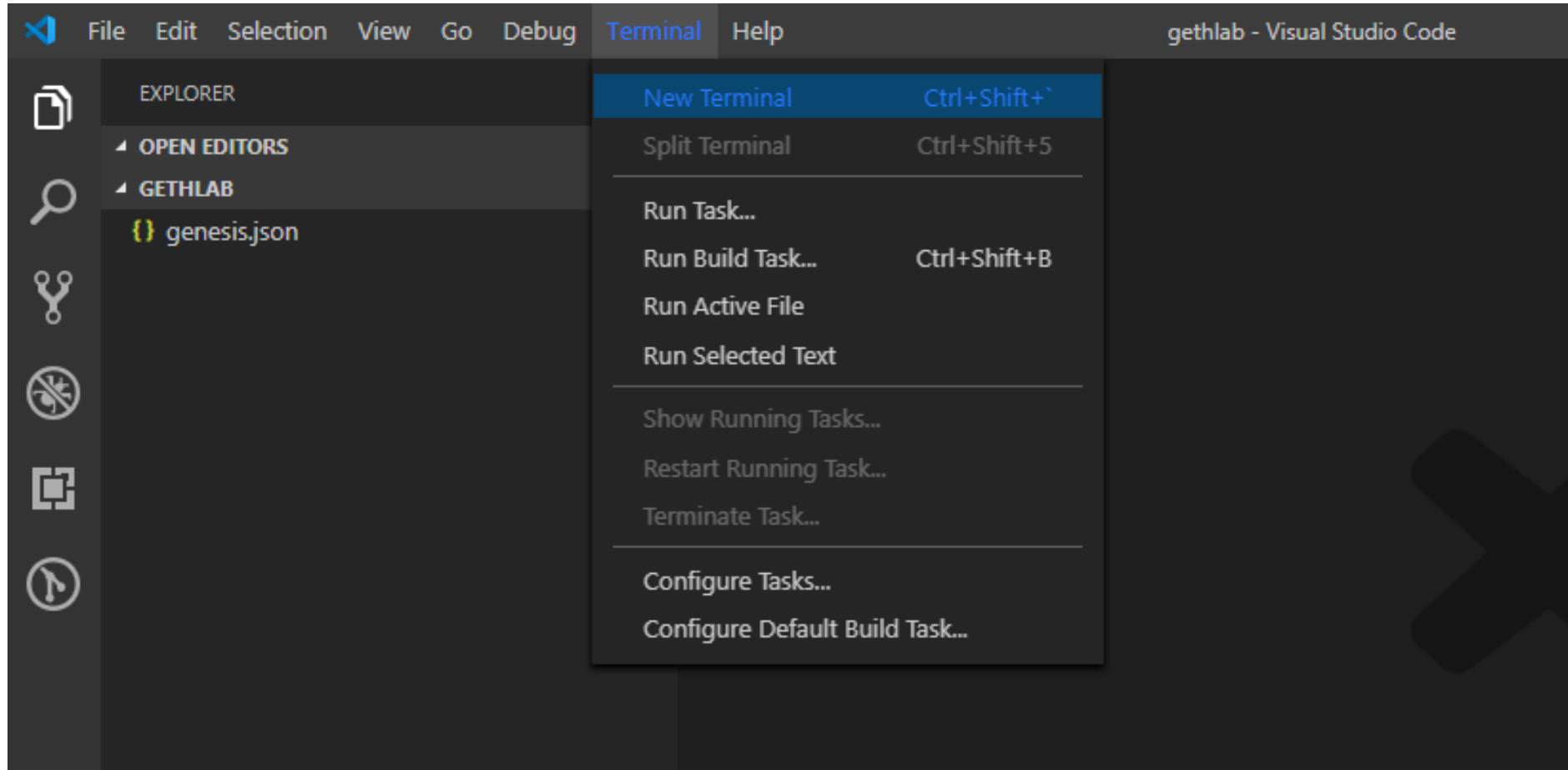
```
1 {
2   "config": {
3     "ChainID": 10,
4     "HomesteadBlock": 0,
5     "DAOForkBlock": 0,
6     "EIP150Block": 0,
7     "EIP155Block": 0,
8     "EIP158Block": 0,
9     "ByzantiumBlock": 0,
10    "ConstantinopleBlock": 0,
11    "PetersburgBlock": 0,
12    "IstanbulBlock": 0
13  },
14  "alloc": {},
15  "coinbase": "0x0000000000000000000000000000000000000000",
16  "difficulty": "0x00010",
17  "extraData": "",
18  "gasLimit": "0x000000",
19  "nonce": "0x0000000000000000",
20  "mixhash": "0x0000000000000000000000000000000000000000",
21  "parentHash": "0x0000000000000000000000000000000000000000",
22  "timestamp": "0x00"
23 }
```

- Outline:** Shows the structure of the JSON object, including "alloc", "config" (with sub-items like "chainid", "eip155Block", etc.), and "abc" (with sub-items like "coinbase", "difficulty", etc.).
- Status Bar:** Ln 16, Col 1 | Spaces: 4 | UTF-8 | CRLF | JSON | 1



Create local node

1. Open Command prompt





Create local node

2. Run command

C:\gethlab>**geth --datadir ./node1 init ./genesis.json**

The terminal window shows the command being run and its output. The output details the initialization process, including cache bumping, trie persisting, and genesis state writing. It also lists the configuration for the new node, including the database path, size, and hash.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
C:\gethlab>geth --datadir ./node1 init ./genesis.json
INFO [08-09|20:16:10.832] Bumping default cache on mainnet
WARN [08-09|20:16:10.839] Sanitizing cache to Go's GC limits
INFO [08-09|20:16:10.909] Maximum peer count
INFO [08-09|20:16:10.971] Allocated cache and file handles
INFO [08-09|20:16:11.125] Writing custom genesis block
INFO [08-09|20:16:11.145] Persisted trie from memory database
ivesize=0.00B
INFO [08-09|20:16:11.153] Successfully wrote genesis state
INFO [08-09|20:16:11.161] Allocated cache and file handles
16
INFO [08-09|20:16:11.281] Writing custom genesis block
INFO [08-09|20:16:11.286] Persisted trie from memory database
ivesize=0.00B
INFO [08-09|20:16:11.298] Successfully wrote genesis state
C:\gethlab>
```

1: cmd

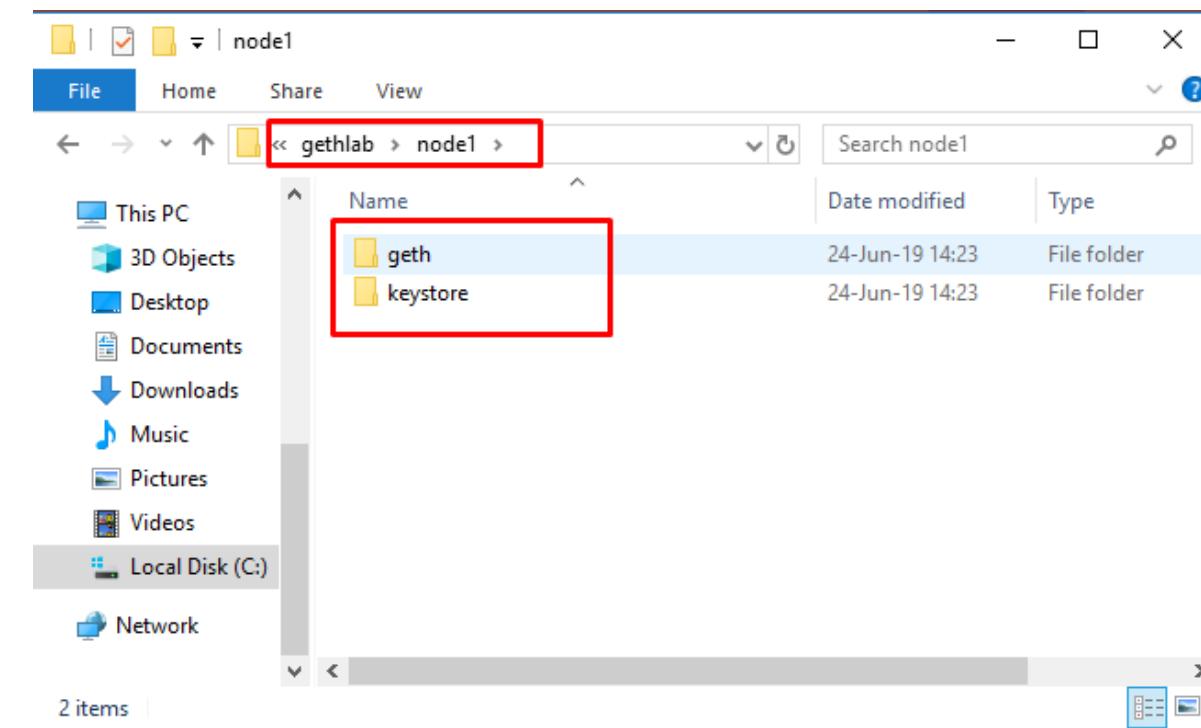
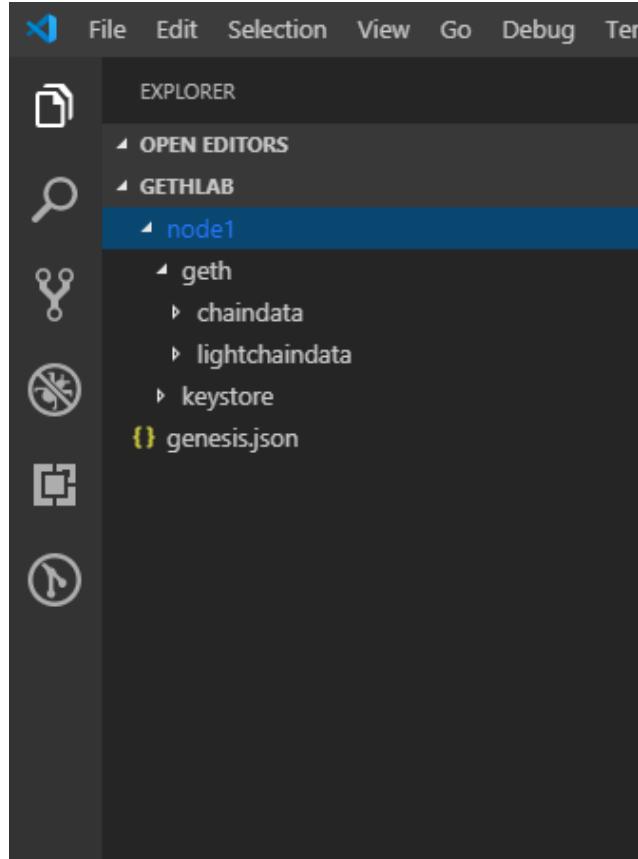
```
provided=1024 updated=4096
provided=4096 updated=4055
ETH=50 LES=0 total=50
database=C:\\gethlab\\node1\\geth\\chaindata cache=16.00MiB handles=16
nodes=0 size=0.00B time=0s gcnodes=0 gcsize=0.00B gctime=0s livenodes=1 1
database=chaindata hash=51238a..d772f1
database=C:\\gethlab\\node1\\geth\\lightchaindata cache=16.00MiB handles=
nodes=0 size=0.00B time=0s gcsize=0.00B gctime=0s livenodes=1 1
database=lightchaindata hash=51238a..d772f1
```

1: cmd



Create local node

3. Check your file and folder





1. Run command

```
C:\gethlab>geth --datadir ./node1 --networkid 10 --verbosity 3 console 2>console.log
```

```
C:\gethlab>geth --datadir ./node1 --networkid 10 --verbosity 3 console
INFO [08-09|20:20:00.501] Maximum peer count
INFO [08-09|20:20:00.571] Starting peer-to-peer node
INFO [08-09|20:20:00.578] Allocated trie memory caches
INFO [08-09|20:20:00.583] Allocated cache and file handles
INFO [08-09|20:20:00.938] Regenerated local transaction journal
INFO [08-09|20:20:00.958] Allocated fast sync bloom
INFO [08-09|20:20:00.963] Initialized fast sync bloom
INFO [08-09|20:20:01.130] New local node record
INFO [08-09|20:20:01.139] Started P2P networking
91092e15dc4106c83182be1f569ebcfe8d416393af3ae58ba7403351a40990ba8270@127.0.0.1:30303
INFO [08-09|20:20:01.142] IPC endpoint opened
WARN [08-09|20:20:01.291] Served eth_coinbase
Welcomme to the Geth JavaScript console!

instance: Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7
at block: 0 (Thu, 01 Jan 1970 07:00:00 +07)
datadir: C:\gethlab\node1
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
```

>





geth	the go-ethereum command line interface
--datadir ./node1	Data directory for the databases and keystore
--port 30303	Network listening port (default: 30303)
--networkid 10	Network identifier (integer, 1=Frontier, 2=Morden(disused), 3=Ropsten, 4=Rinkeby, 5=Goerli) (default: 1)
--verbosity 3	Logging verbosity: 0=silent, 1=error, 2=warn, 3=info, 4=debug, 5=detail (default: 3)
console	Start an interactive JavaScript environment
2>console.log	write STDERR (standard error) to console.log (0=STDIN, 1=STDOUT, 2=STDERR)



personal.newAccount()

```
> personal.newAccount()
```

Passphrase:

Repeat passphrase:

```
INFO [08-10|11:30:27.045] Your new key was generated
WARN [08-10|11:30:27.050] Please backup your key file!
7b2ca8af32603fc2b1337690a6d3a5938013bc3
WARN [08-10|11:30:27.056] Please remember your password!
"0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3"
```

```
address=0xF7b2CA8af32603fc2B1337690A6d3A5938013Bc3
path=C:\\gethlab\\\\node1\\\\keystore\\\\UTC--2019-08-10T04-30-24.308487100Z--f
```

```
> personal.newAccount("1234")
```

```
INFO [08-10|11:30:53.351] Your new key was generated
WARN [08-10|11:30:53.356] Please backup your key file!
0418c6eb01705adabada1348d7e892cc07bb741
WARN [08-10|11:30:53.363] Please remember your password!
"0x70418c6eb01705adabada1348d7e892cc07bb741"
```

```
address=0x70418C6EB01705AdaBadA1348d7E892Cc07BB741
path=C:\\gethlab\\\\node1\\\\keystore\\\\UTC--2019-08-10T04-30-51.239425800Z--7
```



```
> eth.accounts
["0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3", "0x70418c6eb01705adabada1348d7e892cc07bb741"]
> eth.accounts[0]
"0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3"
> eth.accounts[1]
"0x70418c6eb01705adabada1348d7e892cc07bb741"
> eth.accounts[2]
undefined
>
```



```
> eth.coinbase
```

```
INFO [08-10|11:33:22.171] Etherbase automatically configured
```

```
"0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3"
```

```
> eth.coinbase
```

```
"0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3"
```

```
> |
```



miner.setEtherbase(account)

```
> miner.setEtherbase("0x70418c6eb01705adabada1348d7e892cc07bb741")
true
> eth.coinbase
"0x70418c6eb01705adabada1348d7e892cc07bb741"
> miner.setEtherbase(eth.accounts[0])
true
> eth.coinbase
"0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3"
>
```



miner.start(thread)

> miner.start(1)

```
> miner.start(1)
INFO [08-10|11:36:38.681] Updated mining threads           threads=1
INFO [08-10|11:36:38.705] Transaction pool price threshold updated price=1000000000
null
> INFO [08-10|11:36:38.747] Commit new mining work          number=1 sealhash=5b9512..a9b6e7 uncles=0 txs=0 gas=0 fees=0 elapsed=36.
623ms
```

```
INFO [08-10|12:29:46.035] Successfully sealed new block      number=967 sealhash=3a0ae4..0bb26f hash=b2a06c..b0315e elapsed=4.722s
INFO [08-10|12:29:46.040] ⚡block reached canonical chain    number=960 hash=32df66..8cecc6
INFO [08-10|12:29:46.045] ↴mined potential block           number=967 hash=b2a06c..b0315e
INFO [08-10|12:29:46.049] Commit new mining work            number=968 sealhash=52ad76..6b4418 uncles=0 txs=0 gas=0 fees=0 elapsed=8.9
```

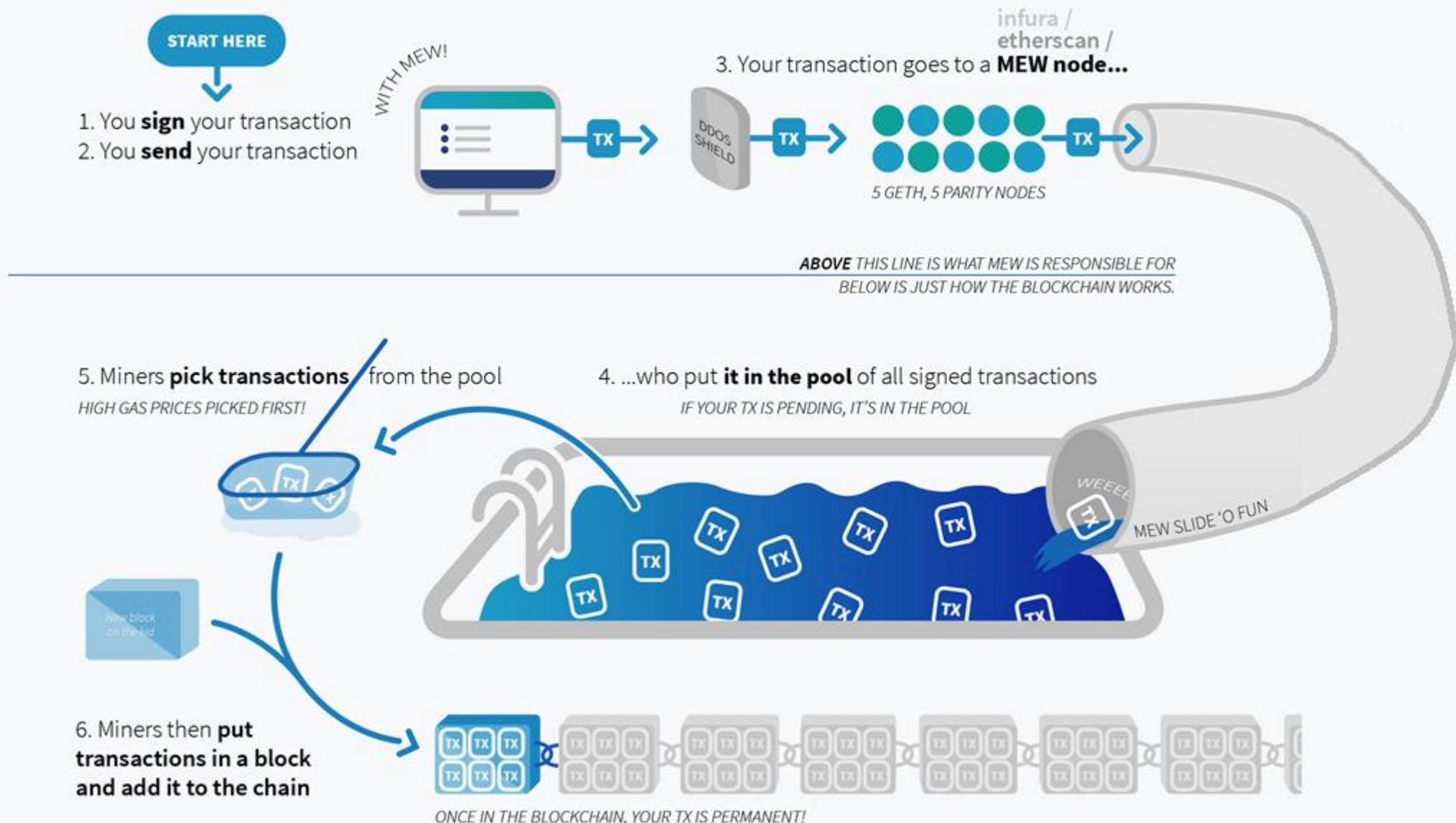


```
> miner.stop()  
null
```



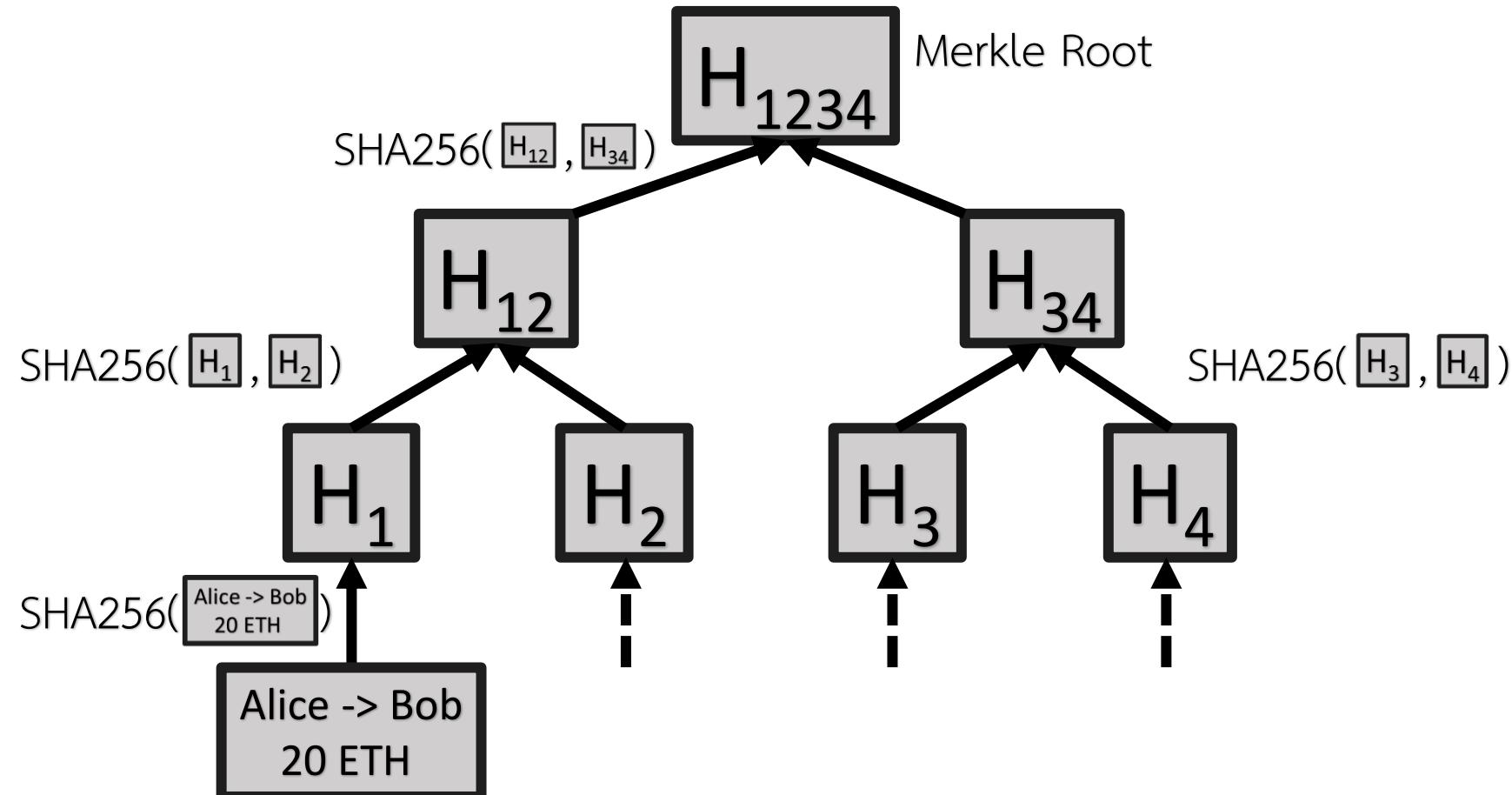
MyEtherWallet Behind-The-Scenes

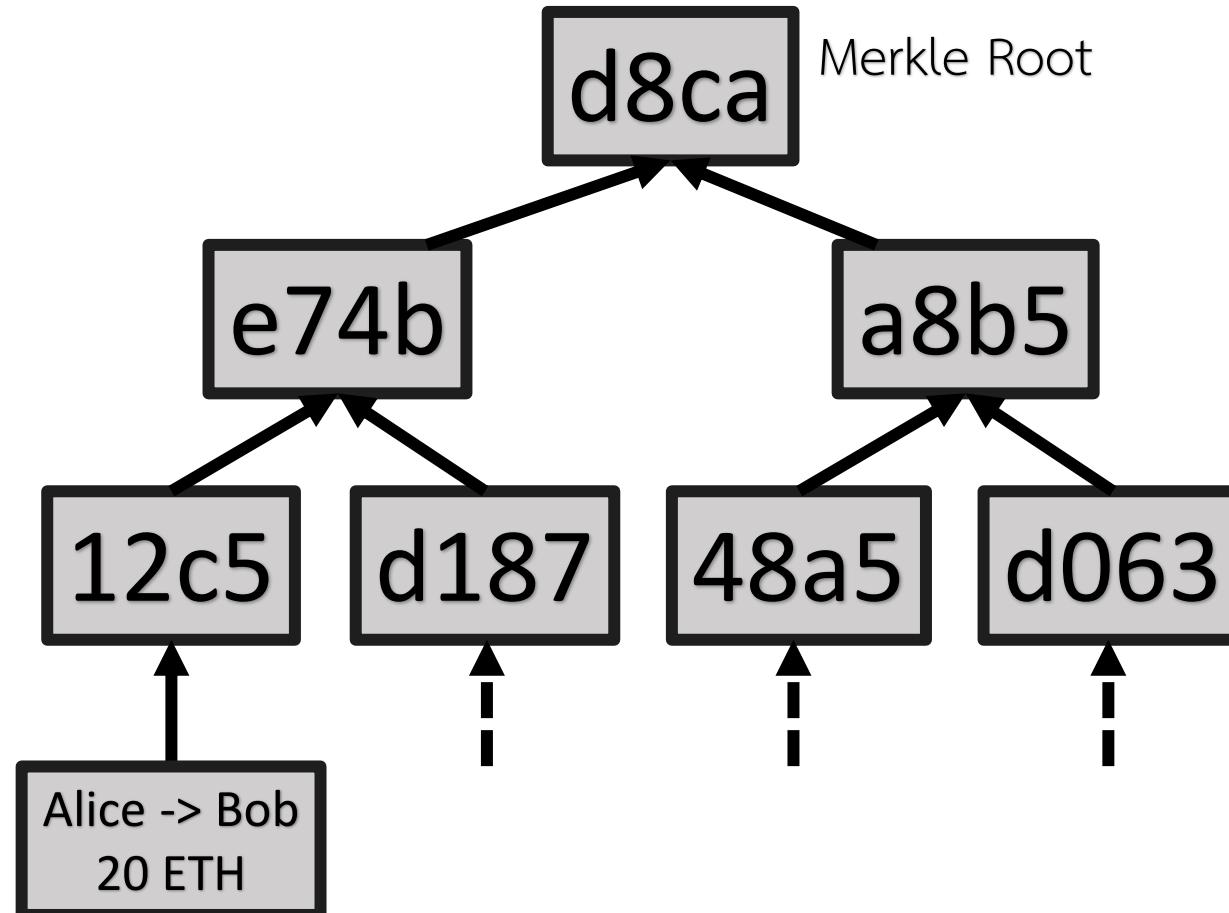
Made by @veenspace





Merkle Tree



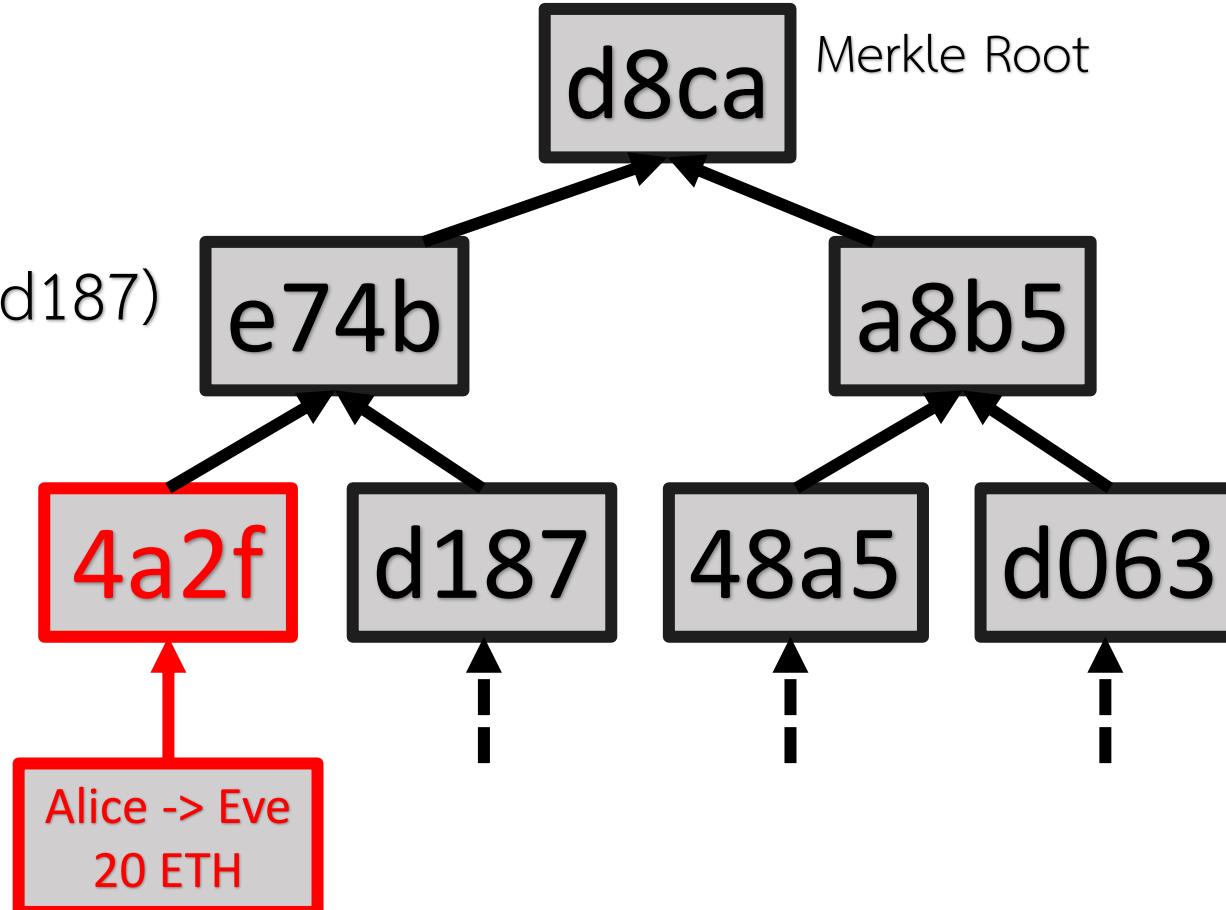




ERROR:

SHA256(4a2f+d187)

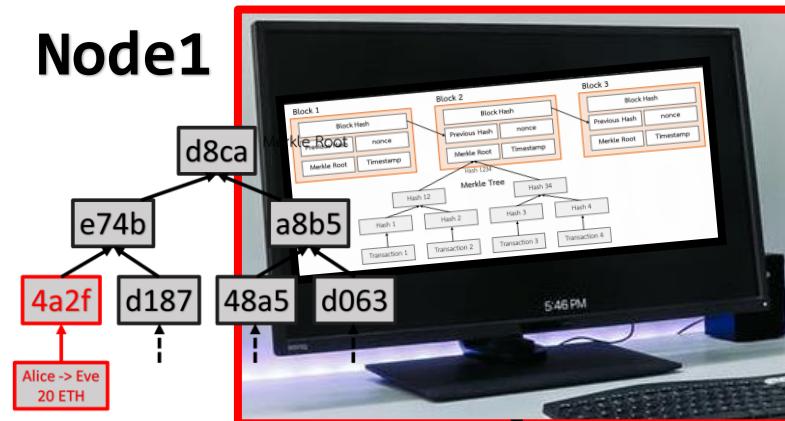
$\neq e74b$



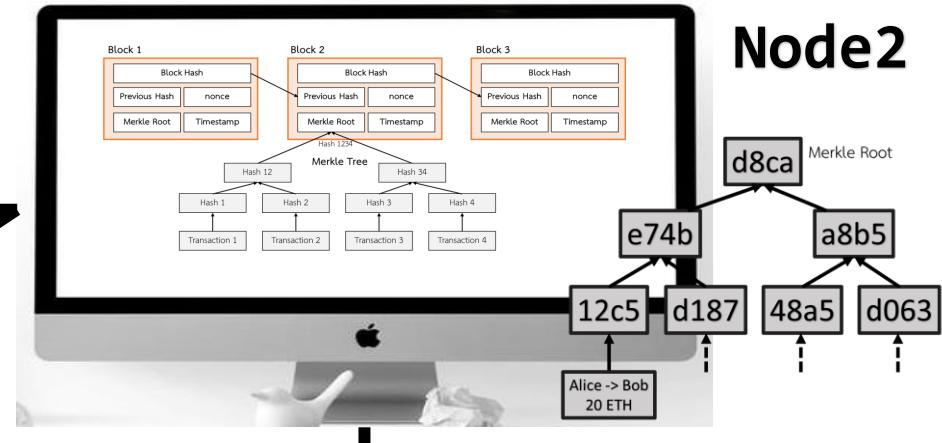


Attack Transaction

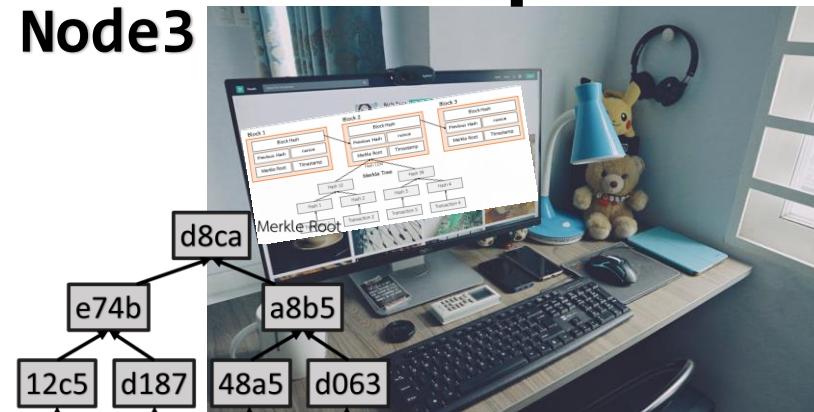
Node1



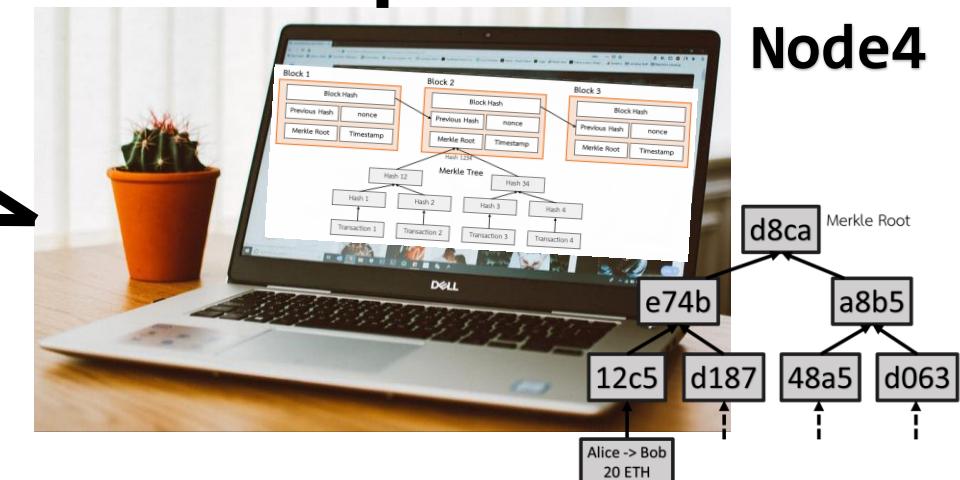
Node2



Node3



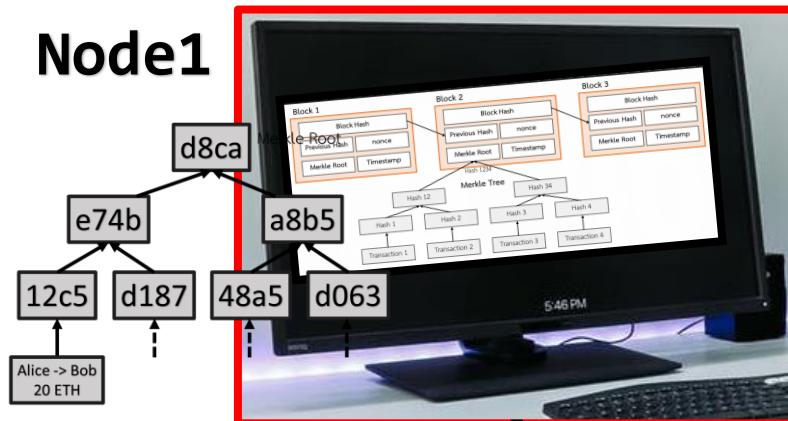
Node4



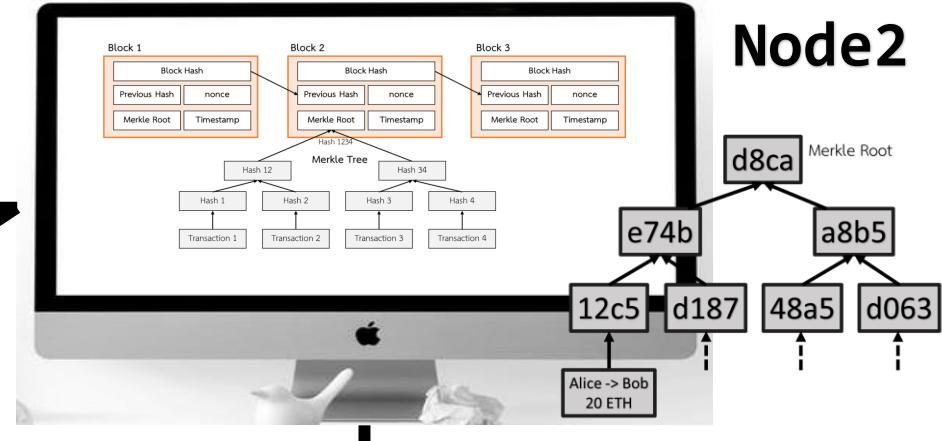


Attack Transaction

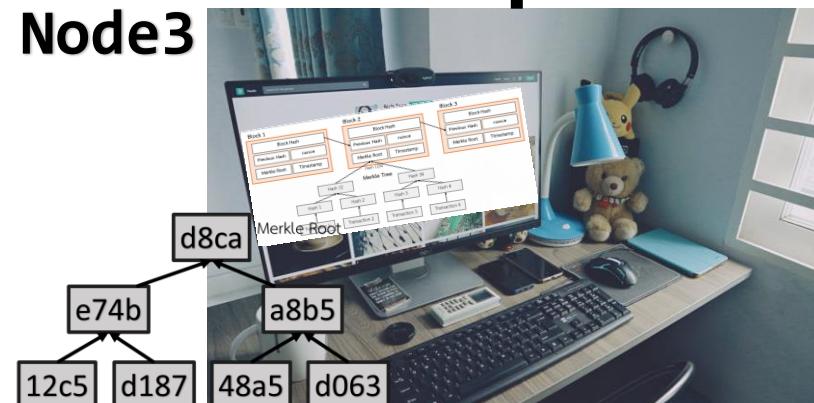
Node1



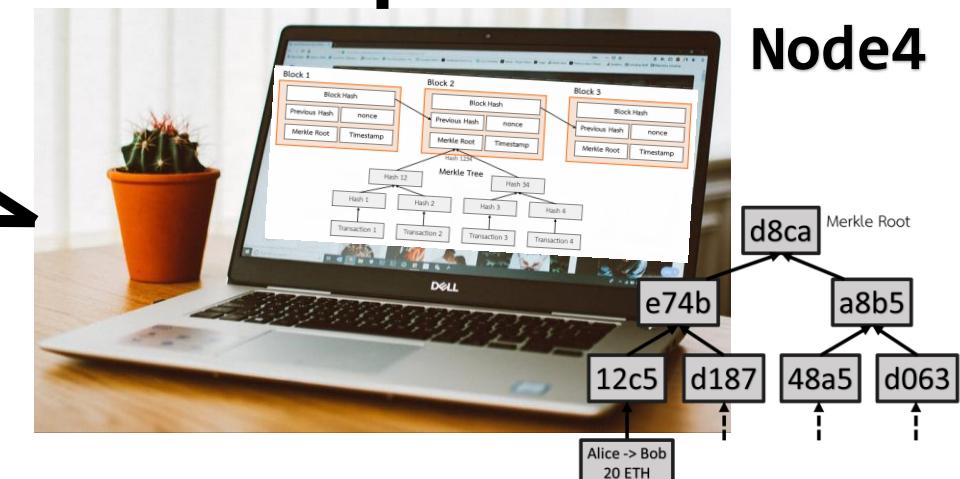
Node2



Node3

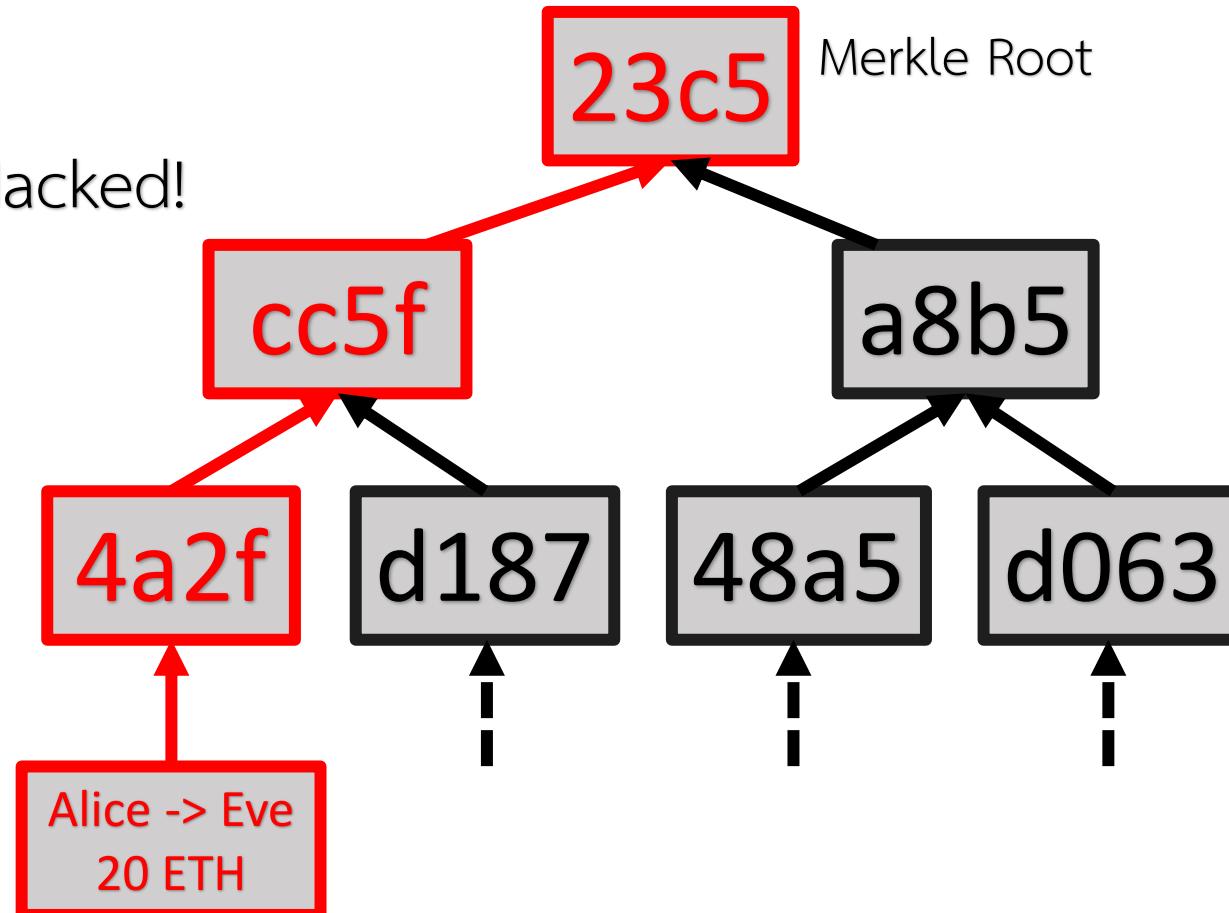


Node4





Merkle Tree Hacked!





$\text{SHA256}(\text{SHA256}(\text{All hashes in block header +nonce})) \leq \text{target}$

$$\text{target} = (10^{60})_{16} = 2^{240}$$

nonce	hashในบล็อก+nonce	SHA256(SHA256(ข้อมูลในบล็อก +nonce)) [เลขฐาน 16]	[เลขฐาน 10]
0	"5a3b1a0"	1312af178c253f84028d480a6adc1e25e81caa44c749ec81976192e2ec934c64	$2^{252.253458683}$
1	"5a3b1a1"	e9afc424b79e4f6ab42d99c81156d3a17228d6e1eef4139be78e948a9332a7d8	$2^{255.868431117}$
2	"5a3b1a2"	ae37343a357a8297591625e7134cbea22f5928be8ca2a32aa475cf05fd4266b7	$2^{255.444730341}$
...
4248	"5a3b1a4248"	6e110d98b388e77e9c6f042ac6b497cec46660deef75a55ebc7cfdf65cc0b965	$2^{254.782233115}$
4249	"5a3b1a4249"	c004190b822f1669cac8dc37e761cb73652e7832fb814565702245cf26ebb9e6	$2^{255.585082774}$
4250	"5a3b1a4250"	0000c3af42fc31103f1fdc0151fa747ff87349a4714df7cc52ea464e12dcd4e9	$2^{239.61238653}$



Target = $\frac{0x00000000FFFFFFFFFFFFFFFFFFF...FFFF}{difficulty}$

why 10 minutes? -> moore's law

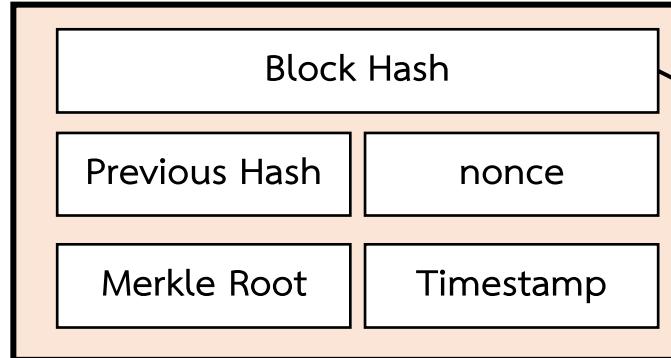
A block header with no transactions would be about 80 bytes. If we suppose blocks are generated every 10 minutes, $80 \text{ bytes} * 6 * 24 * 365 = 4.2\text{MB per year}$. With computer systems typically selling with 2GB of RAM as of 2008, and Moore's Law predicting current growth of 1.2GB per year, storage should not be a problem even if the block headers must be kept in memory.

Update every 2 weeks (2016 blocks)

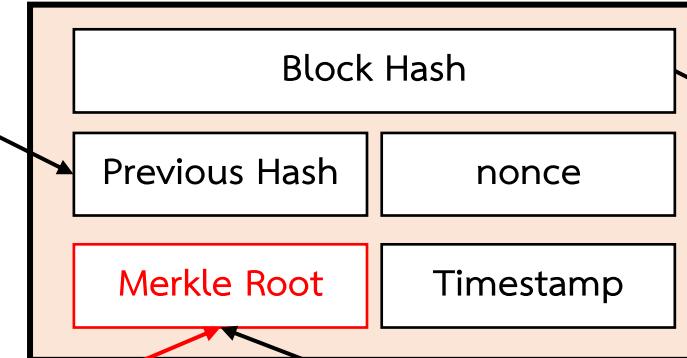
$$2 \text{ weeks} \times \frac{7 \text{ days}}{1 \text{ week}} \times \frac{24 \text{ hours}}{1 \text{ day}} \times \frac{60 \text{ mins}}{1 \text{ hour}} \times \frac{1 \text{ blocks}}{10 \text{ mins}} = 2016 \text{ blocks}$$



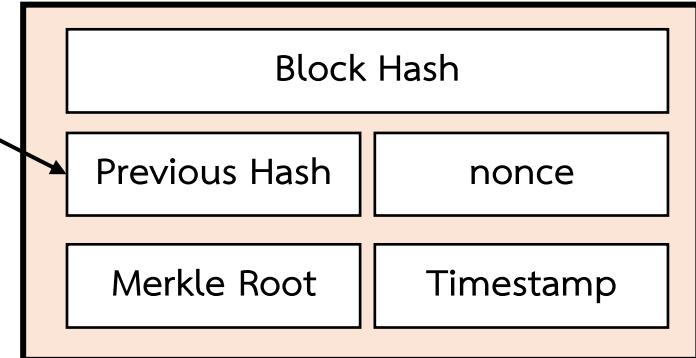
Block 1



Block 2

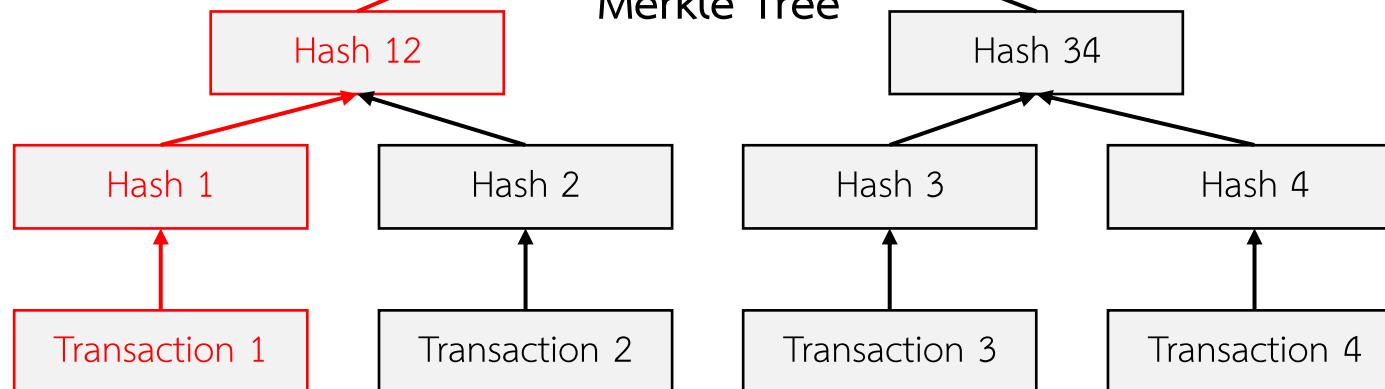


Block 3



Hash 1234

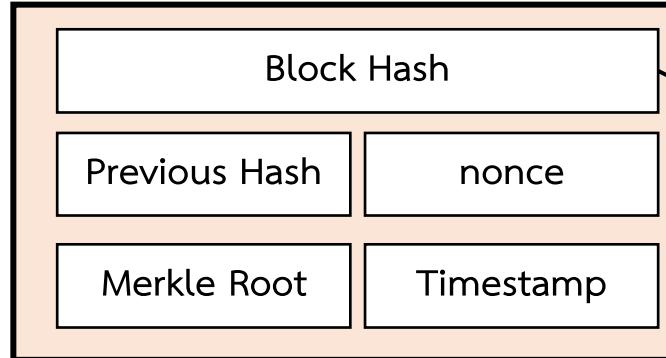
Merkle Tree



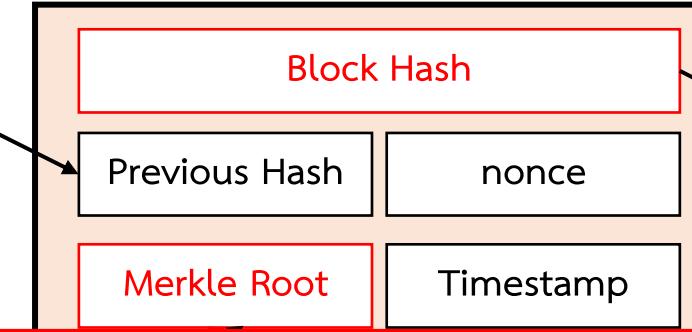


Attack Node

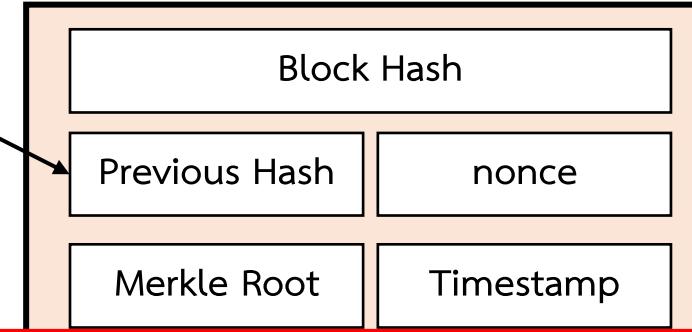
Block 1



Block 2



Block 3

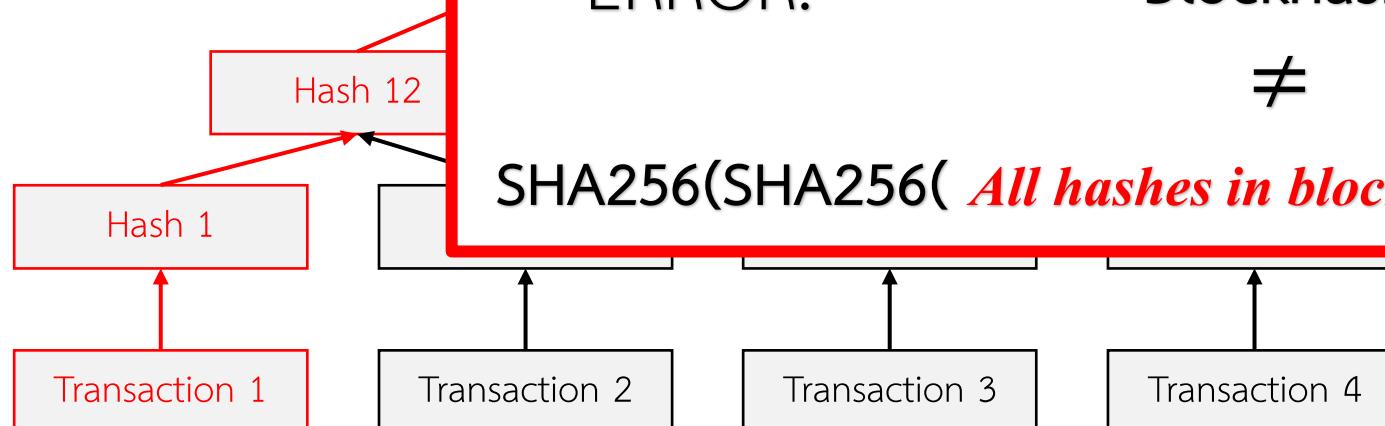


ERROR:

Blockhash

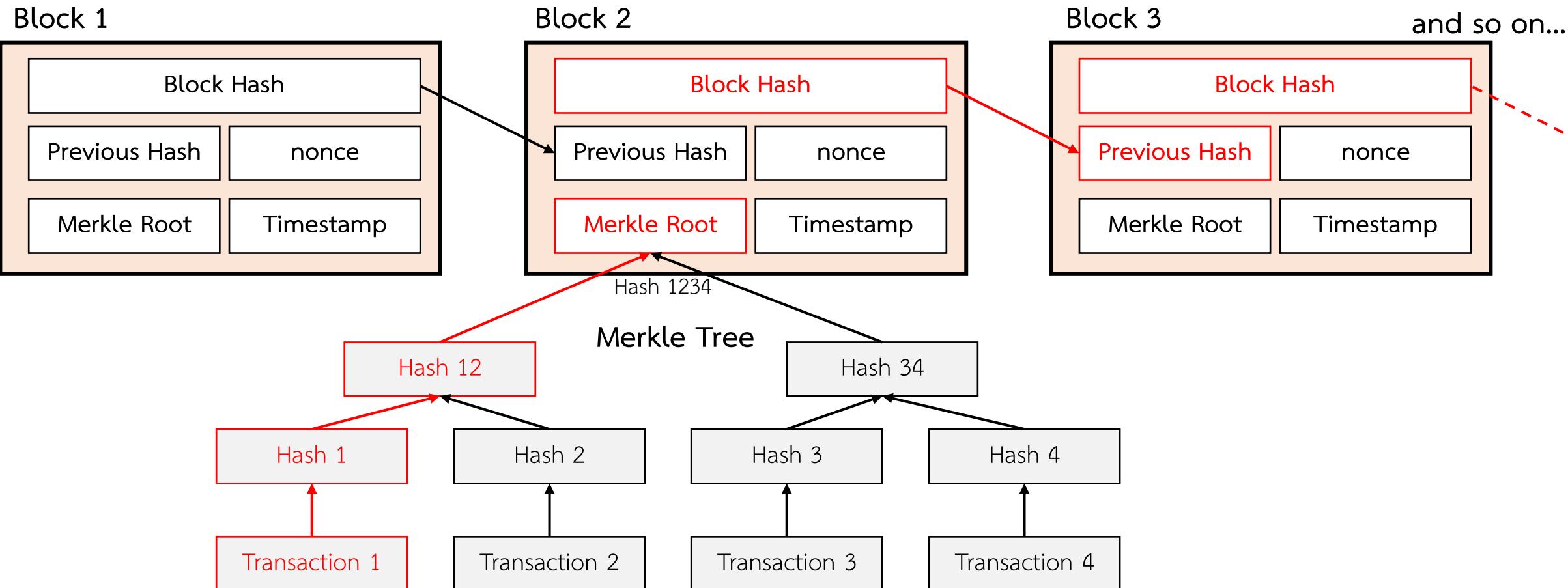
≠

$\text{SHA256}(\text{SHA256}(\text{All hashes in block header} + \text{nonce})) \leq \text{target}$





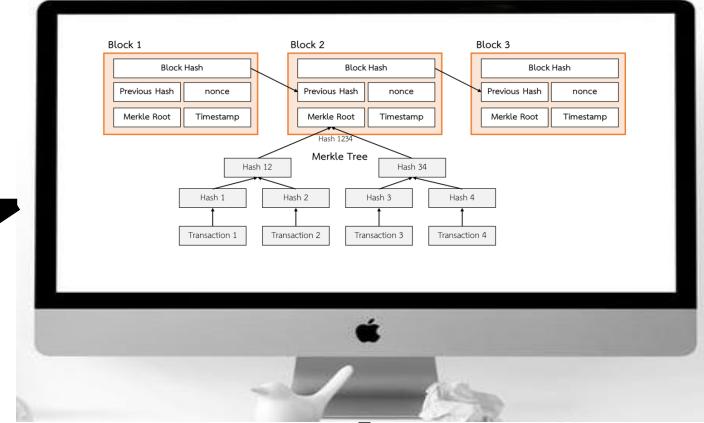
Attack Node





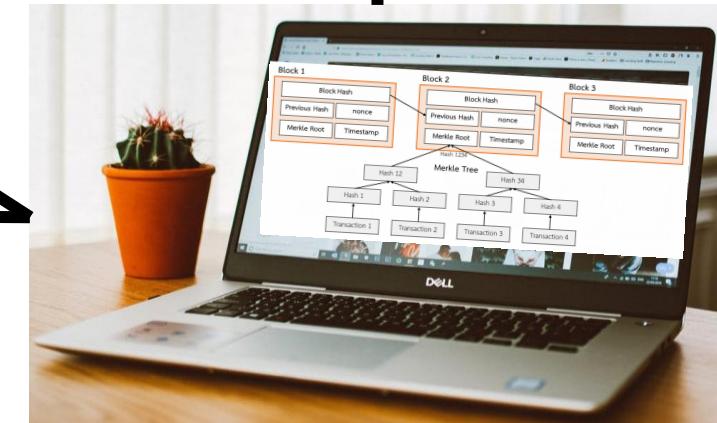
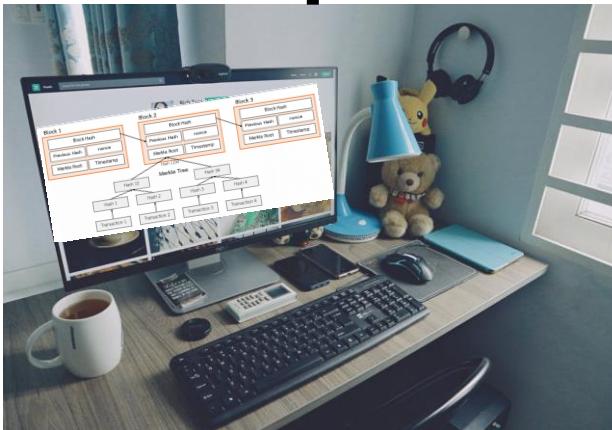
Attack Node < 50%

Node1



Node2

Node3

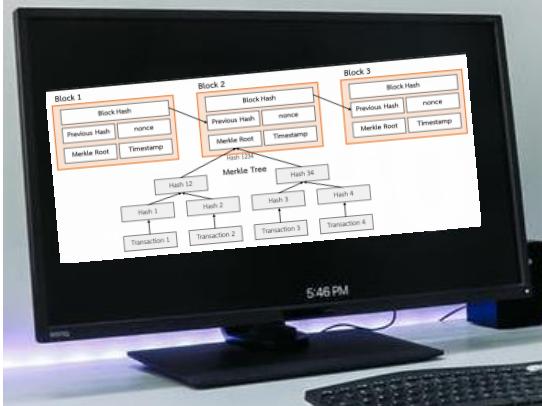


Node4

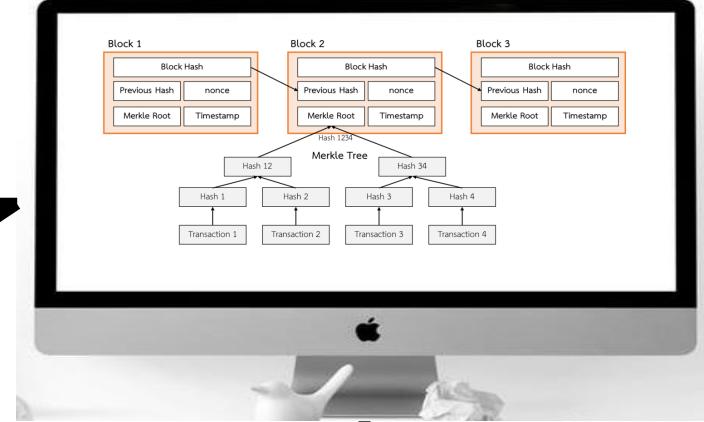


Attack Node < 50%

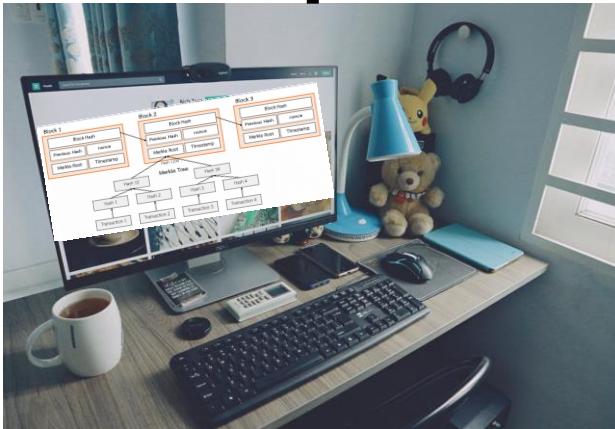
Node1



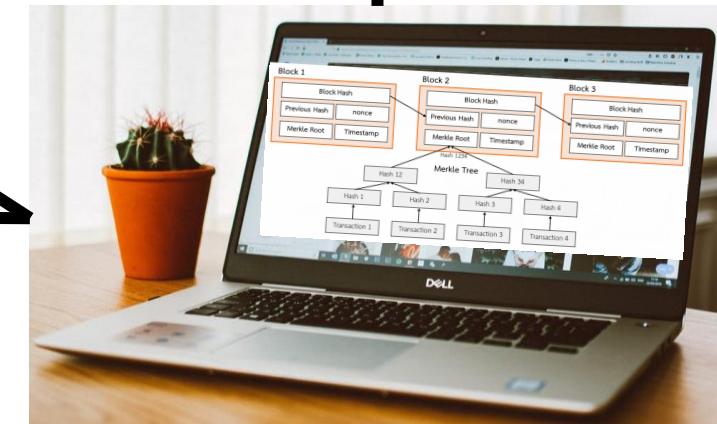
Node2



Node3



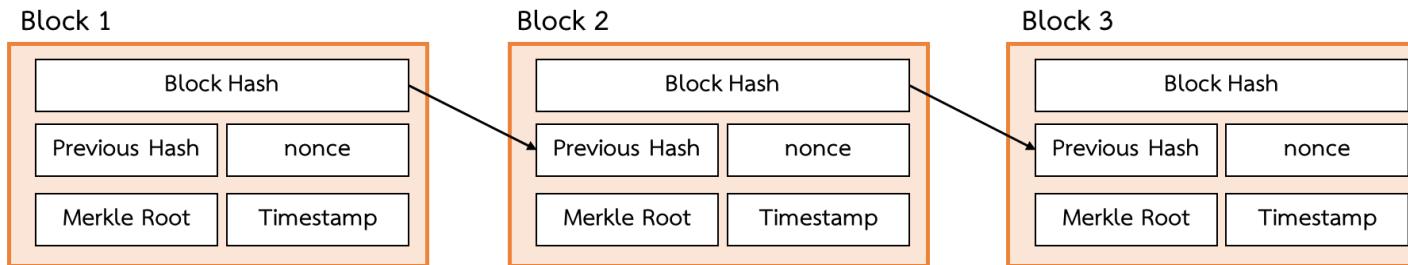
Node4





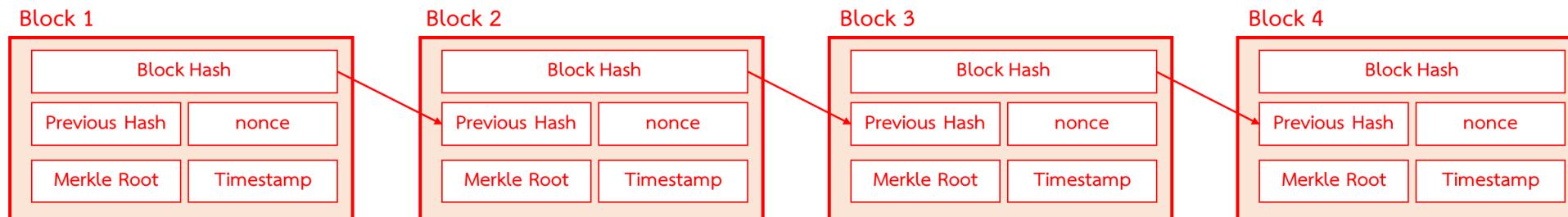
มีโอกาสเกิดกับ public blockchain ที่เปิดให้บุคคลภายนอก join node เข้ามาได้เท่านั้น

Node 1 3 blocks



Join Node

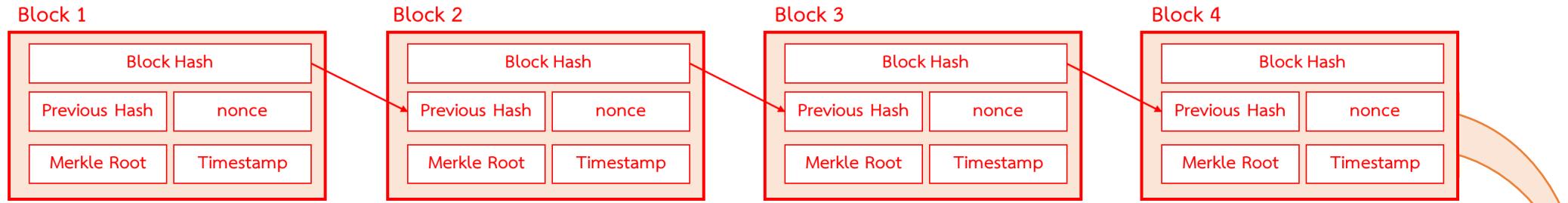
Node 2 Attacker 4 blocks



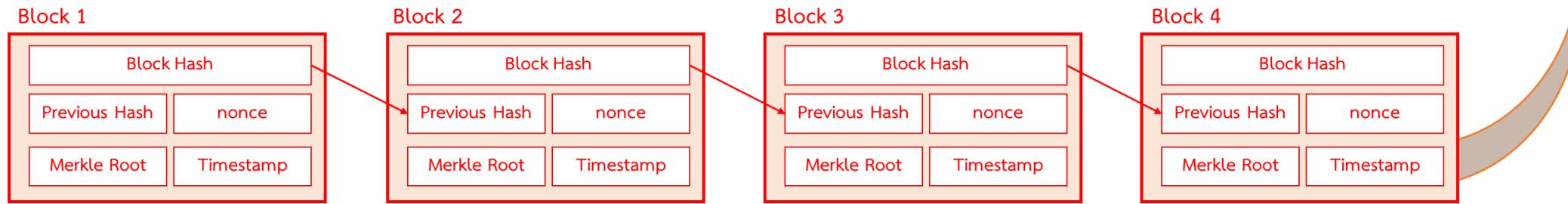


มีโอกาสเกิดกับ public blockchain ที่เปิดให้บุคคลภายนอก join node เข้ามาได้เท่านั้น

Node 1 4 blocks



Node 2 Attacker 4 blocks





IT Management
Faculty of Engineering



eth.blockNumber

» eth.blockNumber
2028



eth.getBlock(number)

```
>eth.getBlock(1)
```

```
>eth.getBlock(2028)
```



eth.getBalance(account)

```
> eth.getBalance("0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3")
1.014e+22
> eth.getBalance(eth.accounts[0])
1.014e+22
> eth.getBalance(eth.coinbase)
1.014e+22
> |
```



personal.unlockAccount(account)

```
> personal.unlockAccount("0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3")
Unlock account 0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3
Passphrase:
true
> personal.unlockAccount(eth.accounts[0])
Unlock account 0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3
Passphrase:
true
```

```
> personal.unlockAccount("0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3")
Unlock account 0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3
Passphrase:
WARN [08-10|14:31:08.496] Failed account unlock attempt           addr
  key with given passphrase"
WARN [08-10|14:31:08.505] Served personal_unlockAccount           requi
Error: could not decrypt key with given passphrase
>
```



eth.sendTransaction({from: "0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3" to:
"0x70418c6eb01705adabada1348d7e892cc07bb741" value: 2e+18})

$$2e+18 = 2 \times 10^{18}$$

```
> eth.sendTransaction({from: "0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3" to: "0x70418c6eb01705adabada1348d7e892cc07bb741" value: 2e+18})
INFO [08-10|14:33:36.079] Submitted transaction
be recipient=0x70418C6EB01705AdaBadA1348d7E892Cc07BB741
"0x6fbce8ef88cb30919a44f340a3c9a8df611da80b4ae247d0f571b6c97dfa20be"
```

eth.sendTransaction({from: eth.accounts[0] to: eth.accounts[1] value: 2e+18})

```
> eth.sendTransaction({from: eth.accounts[0] to: eth.accounts[1] value: 2e+18})
INFO [08-10|14:33:40.395] Submitted transaction
fullhash=0xf47bd9e197587849fd394138ef9f6abb57c52a8498bc7ad7ae886324f1572be3
recipient=0x70418C6EB01705AdaBadA1348d7E892Cc07BB741
"0xf47bd9e197587849fd394138ef9f6abb57c52a8498bc7ad7ae886324f1572be3"
```



Unit	Wei Value	Wei
wei	1 wei	1
Kwei (babbage)	1e3 wei	1,000
Mwei (lovelace)	1e6 wei	1,000,000
Gwei (shannon)	1e9 wei	1,000,000,000
microether (szabo)	1e12 wei	1,000,000,000,000
milliether (finney)	1e15 wei	1,000,000,000,000,000
ether	1e18 wei	1,000,000,000,000,000,000



Add Peer/ Join Node

1. สร้างโหนดขึ้นมาอีกอัน

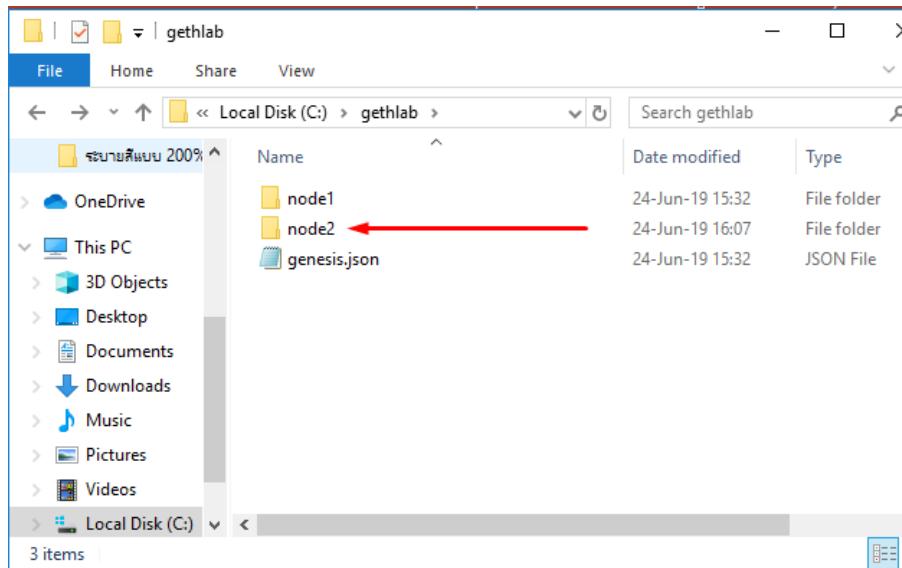
1.1 เปิด command prompt อีกหนึ่งอัน

1.2 สร้างโหนดสอง run command

```
C:\gethlab>geth --datadir ./node2 init ./genesis.json
```

1.3 เข้าคอนโซลของโหนดสอง run command

```
C:\gethlab>geth --datadir ./node2 --port 30304 --networkid 10 --ipcdisable console 2>console2.log
```



```
C:\gethlab>geth --datadir ./node2 init ./genesis.json
INFO [08-10|14:36:03.708] Bumping default cache on mainnet
WARN [08-10|14:36:03.718] Sanitizing cache to Go's GC limits
INFO [08-10|14:36:03.839] Maximum peer count
INFO [08-10|14:36:03.907] Allocated cache and file handles
INFO [08-10|14:36:04.022] Writing custom genesis block
INFO [08-10|14:36:04.039] Persisted trie from memory database
ivesize=0.00B
INFO [08-10|14:36:04.047] Successfully wrote genesis state
INFO [08-10|14:36:04.053] Allocated cache and file handles
16
INFO [08-10|14:36:04.189] Writing custom genesis block
INFO [08-10|14:36:04.194] Persisted trie from memory database
ivesize=0.00B
INFO [08-10|14:36:04.202] Successfully wrote genesis state

C:\gethlab>
```



Add Peer/ Join Node

****ขอเรียกค่อนโฉลแรกว่า cmd#1 และ ค่อนโฉลของโนนดที่สร้างใหมว่า cmd#2****

2. Get Enode

cmd#2

2.1 run command

>admin.nodeInfo

2.2 File > New File

2.3 paste enode

```
> admin.nodeInfo
{
  enode: "enode://edbf10f2cae81f0ff19e4003c85b4fc2734d298a5c61ad
fdfaf59@49.229.210.152:30304",
  enr: "enr:-Je4QPGRIXzsmV4Hco5UEe42qxFeLTmcI8-RgYQqG19lN9_GYX4E
c2VjcDI1NmsxoQPtvxDyyugfD_GeQAPIW0_Cc00pilxhreg0cHaUjVUVRIN0Y3C0
  id: "9e0db447c3e3375f6430daf4fcab9741bfdbab388641f9ba9579146b
  ip: "49.229.210.152",
  listenAddr: "[::]:30304",
  name: "Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7",
  ports: {
    discovery: 30304,
    listener: 30304
  },
  protocol: "Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7"
}
```

COPY



Add Peer/ Join Node

3. get IPv4 address

3.1 เปิด command prompt อีกหนึ่งอัน (cmd#3)

3.2 run command

C:\gethlab>**ipconfig**

3.3 แก้ไข ip ใน enode ให้เป็น IPv4

```
C:\gethlab>ipconfig
```

```
Windows IP Configuration
```

```
Wireless LAN adapter Wi-Fi:
```

```
Connection-specific DNS Suffix . :  
Link-local IPv6 Address . . . . . : fe80::5cf7:23d:9698:b566%15  
IPv4 Address . . . . . : 10.226.160.128  
Subnet Mask . . . . . : 255.255.240.0  
Default Gateway . . . . . : 10.226.160.1
```

Change
to
IPv4

Untitled-1

```
1 "enode://edbf10f2cae81f0ff19e4003c85b4fc2734d298a5c61ade8347076948d5515449f6e7ca3309ca0a5714256e450da8347d8c2db0978eb9bf5ad686641c4fdaf59@49.229.210.152:30304"
```



Untitled-1

```
1 "enode://edbf10f2cae81f0ff19e4003c85b4fc2734d298a5c61ade8347076948d5515449f6e7ca3309ca0a5714256e450da8347d8c2db0978eb9bf5ad686641c4fdaf59@10.226.160.128:30304"
```



4. addPeer

cmd#1

4.1 run command format: admin.addPeer(**enode**) // อย่าลืมใส่ " "

```
>admin.addPeer("enode://edbf10f2cae81f0ff19e4003c85b4fc2734d298a5c61ade8347076948d5515449f6e7ca3309ca0a57142  
56e450da8347d8c2db0978eb9bf5ad686641c4fdaf59@10.226.160.128:30304")
```

```
> admin.addPeer("enode://edbf10f2cae81f0ff19e4003c85b4fc2734d298a5c61ade8347076948d5515449f  
86641c4fdaf59@10.226.160.128:30304")
```

```
true
```



5. Check peers info

5.1 run command

>admin.peers

ถ้ารันใน node1 จะเห็นข้อมูลของ node2

node1

```
> admin.peers
[{
  caps: ["eth/63"],
  enode: "enode://edbf10f2cae81f0ff19e4003c85b4fc2734d298a5c61ade8347076948d5515449fc4fdaf59@10.226.160.128:30304",
  id: "9e0db447c3e3375f6430daf4fcab9741bfdbab388641f9ba9579146bde44dfc",
  name: "Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7",
  network: {
    inbound: false,
    localAddress: "10.226.160.128:55053",
    remoteAddress: "10.226.160.128:30304",
    static: true,
    trusted: false
  },
  protocols: {
    eth: {
      difficulty: 16,
      head: "0x51238aa1ec08fc8e7ac48c8bf29cdb4043a2a34a830de0ebdf5168ec0cd772f1",
      version: 63
    }
  }
}]
```

ถ้ารันใน node2 ก็จะเห็นข้อมูลของ node1

node2

```
> admin.peers
[{
  caps: ["eth/63"],
  enode: "enode://d66ed9d9ecfd36ea7630489cac8f664462f4a65659b727ab297ec179e34f9109290ba8270@10.226.160.128:55053",
  id: "e8d2f7018b518169989e9d375b626e00f71004c631bddf42a6936496370175c3",
  name: "Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7",
  network: {
    inbound: true,
    localAddress: "10.226.160.128:30304",
    remoteAddress: "10.226.160.128:55053",
    static: false,
    trusted: false
  },
  protocols: {
    eth: {
      difficulty: 435961351,
      head: "0x1955d16b60f1261d41f22ee6c58a356ea725e45e6414cb14a152837b93bee896",
      version: 63
    }
  }
}]
```

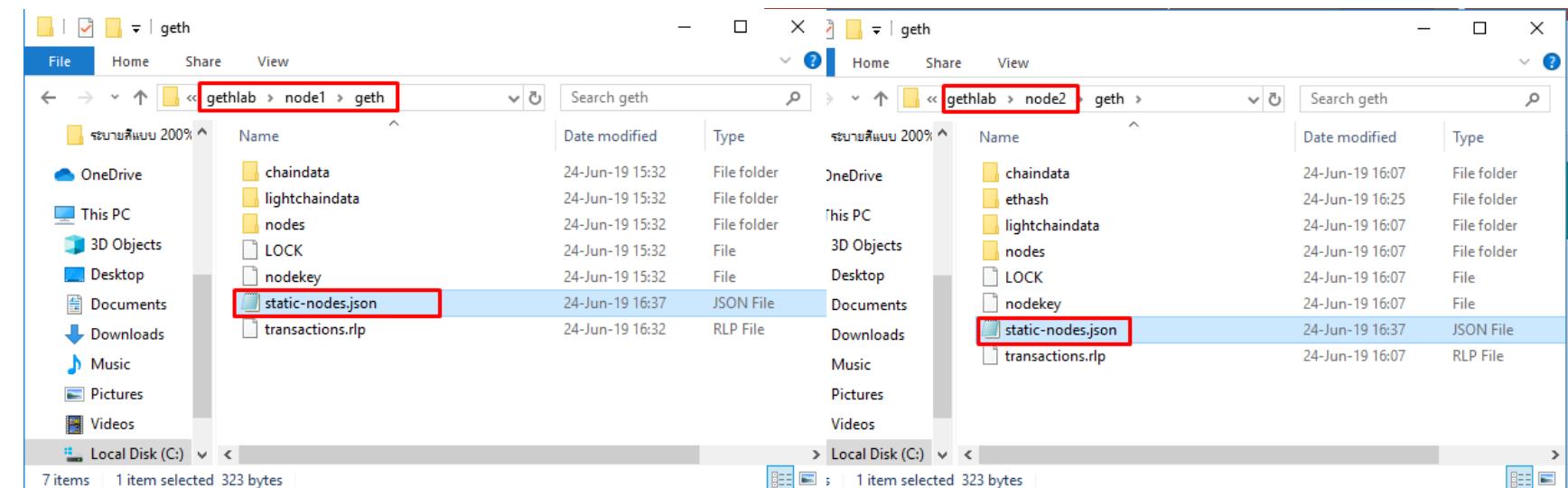


static-nodes.json

1. File > New File > paste code
2. Save file as **static-nodes.json**
3. Copy file to Data folder **C:\gethlab\node1\geth** และ **C:\gethlab\node2\geth**

static-nodes.json format

```
[  
  "enode://pubkey@ip:port",  
  "enode://pubkey@ip:port"  
]
```



```
{ static-nodes.json x }  
{ static-nodes.json ... }  
1 [  
2   "enode://d66ed9d9ecfd36ea7630489cac8f664462f4a92e15dc4106c83182be1f569ebcf8d416393af3ae58ba7403351a40990ba8270@49.229.210.152:30303",  
3   "enode://edbfb10f2cae81f0ff19e4003c85b4fc2734d249f6e7ca3309ca0a5714256e450da8347d8c2db0978eb9bf5ad686641c4fdaf59@10.226.160.128:30304"  
4 ]
```



explorer.html

File | C:/saiyab/bc/bc1%20รุน%206/getlab/explorer.html

Number	Tx#	Size	Timestamp
2027	0	540	1565420050
2026	0	540	1565420049
2025	0	540	1565420047
2024	0	540	1565420029
2023	0	540	1565420028
2022	0	540	1565420026
2021	0	540	1565420025
2020	0	540	1565420023
2019	0	540	1565420020
2018	0	540	1565420019
2017	0	540	1565420018
2016	0	540	1565420017
2015	0	540	1565420014
2014	0	540	1565420003
2013	0	540	1565420000
2012	0	540	1565419996
2011	0	540	1565419992



1. exit console (Ctrl+D or type **exit** in console)
2. enter console with command

```
C:\gethlab>geth --datadir ./node1 --networkid 10 --verbosity 3  
--rpc --rpcapi "eth,web3,personal,net,db" --rpccorsdomain "*"  
console 2>console.log
```

```
C:\gethlab>geth --datadir ./node1 --networkid 10 --verbosity 3 --rpc --rpcapi "eth,web3,personal,net,db" --rpccorsdomain "*" console 2>console.log  
Welcome to the Geth JavaScript console!
```

```
instance: Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7  
coinbase: 0x10effd5845381002f566e10e9371ff00d8767f41  
at block: 725 (Thu, 15 Aug 2019 00:28:49 +07)  
datadir: C:\gethlab\node1  
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
```

```
>
```



accounts, cmd, internal: disable unlock account as default #17037

Merged

karalabe merged 4 commits into `ethereum:master` from `rjl493456442:disable_unlock_account_asdefault` on Apr 4



holiman commented on Mar 29

Contributor

+ ...

As it now works, I think it's fine.

- If you call `unlock` via CLI, it fails the unlock if `ws` or `rpc` is enabled,
- If you call `personal.unlock` via rpc, it fails it if `ws` or `http-rpc` is enabled

It is still possible to shoot yourself in the foot if you really try hard, via

```
> build/bin/geth --unlock 0 --password password.txt --nodiscover --maxpeers 0 console
[...]
> admin.startRPC()
INFO [03-29|11:32:57.269] HTTP endpoint opened                               url=http://localhost:8545
true
```



1. with allow-insecure-unlock option

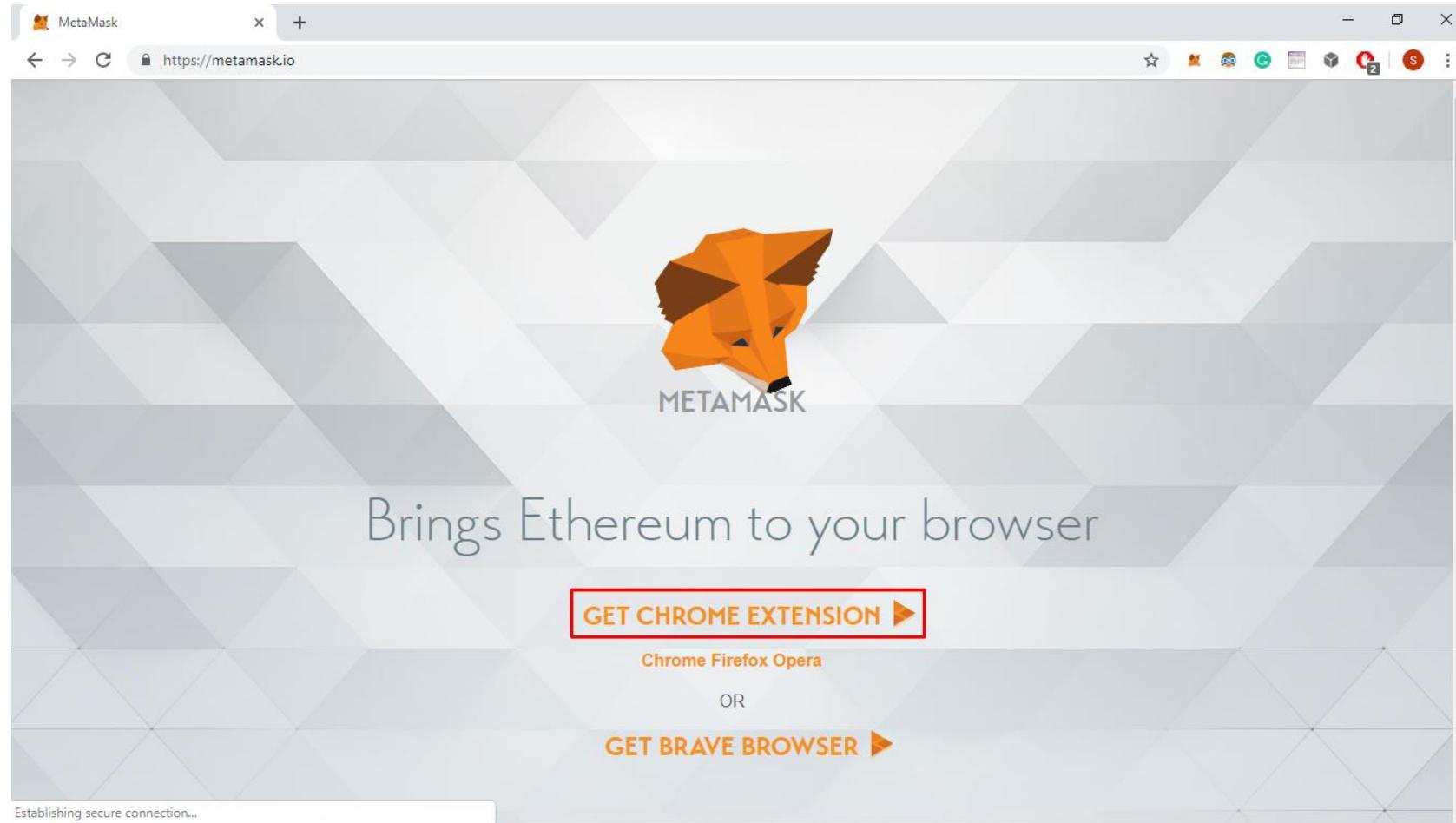
```
C:\gethlab>geth --datadir ./node1 --networkid 10 --verbosity 3  
--rpc --rpcapi "eth,web3,personal,net,db" --rpccorsdomain "*"  
--allow-insecure-unlock --unlock 0 --password password.txt  
console 2>console.log
```

```
C:\gethlab>geth --datadir ./node1 --networkid 10 --verbosity 3 --rpc --rpcapi "eth,web3,personal,net,db" --rpccorsdomain "*" --allow-insecure-unlock --unlock 0 --password password.txt console 2>console.log  
Welcome to the Geth JavaScript console!  
  
instance: Geth/v1.9.1-stable-b7b2f60f/windows-amd64/go1.12.7  
coinbase: 0xf7b2ca8af32603fc2b1337690a6d3a5938013bc3  
at block: 2028 (Sat, 10 Aug 2019 13:54:11 +07)  
datadir: C:\gethlab\node1  
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0  
> []
```



How to Unlock

2. with metamask





Command	Description
personal.newAccount()	Create new account
eth.accounts	View all accounts in the node
eth.coinbase	View coinbase
miner.setEtherbase(account)	Change coinbase to account
miner.start(thread)	Start mining with thread thread(s)
miner.stop()	Stop mining
eth.getBalance(account)	Get balance of that account



Command	Description
eth.blockNumber	View current block high
eth.getBlock(number)	Data of blok at that number
personal.unlockAccount(account)	Unlock the account
eth.sendtransaction({from: sender to: receiver value: amount })	Send amount wei(s) from sender to receiever
admin.nodeInfo	View node's info
admin.addPeer(enode_URL)	adding Peer using enode_URL
admin.peers	View peers' info



Install Ganache-cli

1. Open Terminal
2. Run command

C:\web3lab>**npm i -g ganache-cli**

The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: cmd + - ×
Microsoft Windows [Version 10.0.17134.829]
(c) 2018 Microsoft Corporation. All rights reserved.

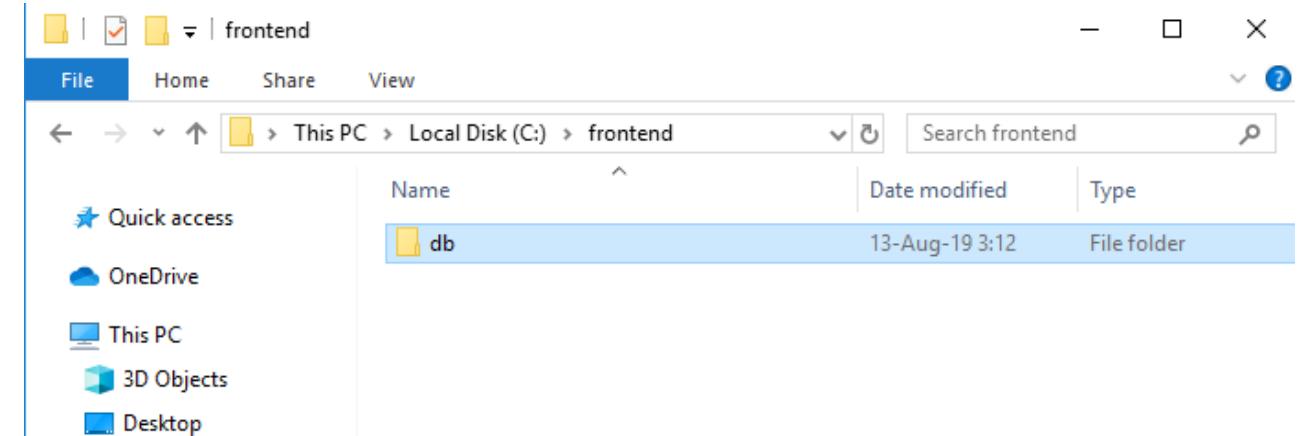
C:\frontend>npm install --global ganache-cli
C:\Users\Suppakorn\AppData\Roaming\npm\ganache-cli -> C:\Users\Suppakorn\AppData\Roaming\npm\node_modules\ganache-cli\cli.js
+ ganache-cli@6.4.4
added 54 packages from 46 contributors in 5.877s

C:\web3lab>
```



Start Ganache-cli

1. make folder **C:\frontend\db**



2. Run command

C:\frontend>ganache-cli -m "month apology submit inside oyster into evidence example

among future slot guide" --db "C:\frontend\db"

```
(0) 0xAcC4B24f45191F99f547df190081BeE16A0bA45c (100 ETH)
(1) 0xB0a3657602343ee44Ff08047e3F20E756cC848E5 (100 ETH)
(2) 0x1A1D881891E509b711428f23FeB31e42D27bCA29 (100 ETH)
(3) 0xc0139Dd870C61ec9ce6fA6F9c7867a615B5f1D3D (100 ETH)
(4) 0x59E8FcD228D65106dd4Ef730563999F27C4fdB30 (100 ETH)
(5) 0x1e49a7953859E72CCF3B8345138Cdb57f241E5a1 (100 ETH)
(6) 0x2204a92a3b6527C1E506f28681F83a24f337d7d0 (100 ETH)
(7) 0x7C37a440f2899922e22a9Bb427A36DED3d21DbFD (100 ETH)
(8) 0x7709f03A8a7AB87727c61C850c180d7C2f5192A1 (100 ETH)
(9) 0xf978D06d16293889D212F9a19cE41E5985A2C340 (100 ETH)
```

Private Keys

```
=====
```

```
(0) 0x92bdb1add582770c9a16d5ffc1be99ef655e338d9ca02035c1d30e7b032b6846
(1) 0x009bed89b474d01521562066c94efb196c56ecf602f1961bf1c1c2c0a5db6915
(2) 0x961529dc879566968c3f0ef88206346e77195caf9263df404696c048290917a
(3) 0x75d188701cc0c738a2559319da1af9ab06312787a15eceec7e8289dfc60fe464
(4) 0x4b6b88ea03fde80bd9ac0be24e676d9afe85a3337f4ebb22f8f4341d3d01f7e1
(5) 0x90990303cdb69f3f0691c37bae80bbc355d7fb45d24efd6c8a0abc2ab25d0a05
(6) 0xf3b5c2668565e479a51332519cd52d5abf22acb63bdae078daed13b6c9d8596
(7) 0x8c56986978ca76184c384467b05de637b566b13a79a162b57744469454468f36
(8) 0x44f0e2742518b6d1542a4c5c2db75d9a91183410c20703986fc25520972d8b50
(9) 0x914b8755879f32c62ab43813206d994f9d8f39d706def98ba425c3f64b82522d
```

HD Wallet

```
=====
```

```
Mnemonic: month apology submit inside oyster into evidence example among future slot guide
Base HD Path: m/44'/60'/0'/0/{account_index}
```

Gas Price

```
=====
```

```
20000000000
```

Gas Limit

```
=====
```

```
6721975
```

```
Listening on 127.0.0.1:8545
```

```
eth_blockNumber
```

```
eth_blockNumber
```

```
eth_getBlockByNumber
```

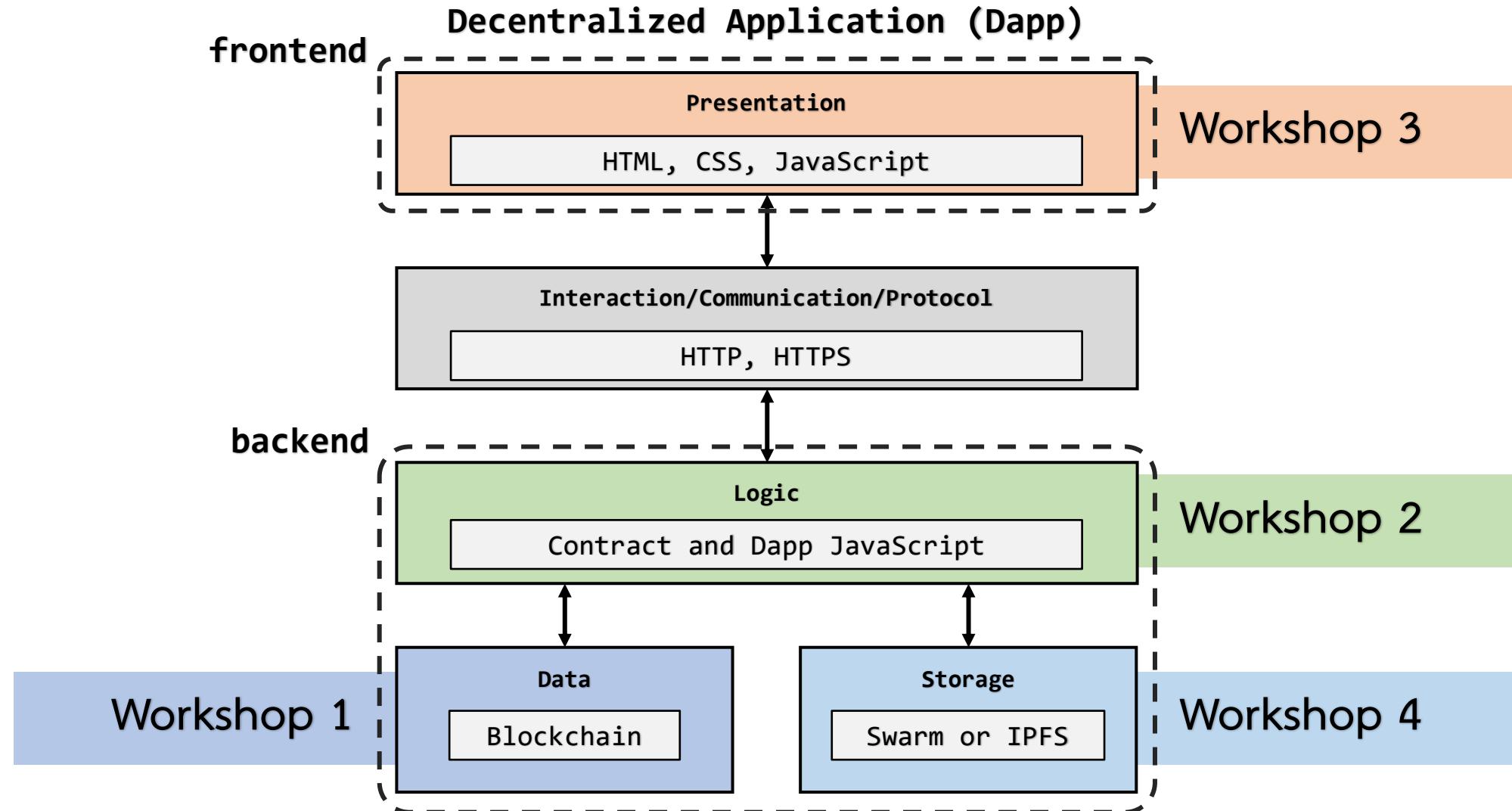
File Edit Selection View Go Debug Terminal Help ganache-key.js - frontend - Visual Studio Code

EXPLORER OPEN EDITORS FRONTEND db node_modules ganache-key.js package-lock.json package.json

OUTLINE No symbols found in document 'ganache-key.js'

1 C:\gethlab>ganache-cli -m "month apology submit inside oyster into evidence example among future slot guide" --db "C:\frontend\db"
2 Ganache CLI v6.5.1 (ganache-core: 2.6.1-beta.0)
3 >
4 Available Accounts
5 =====
6 (0) 0xAcC4B24f45191F99f547df190081BeE16A0bA45c (100 ETH)
7 (1) 0xBdA3657602343ee44Ff08047e3F20E756cC848E5 (100 ETH)
8 (2) 0x1A1D881891E509b711428f23FeB31e42D27bCA29 (100 ETH)
9 (3) 0xc0139Dd870C61ec9ce6fA6F9c7867a615B5f1D3D (100 ETH)
10 (4) 0x59E8FcD228D65106dd4Ef730563999F27C4fdB30 (100 ETH)
11 (5) 0x1e49a7953859E72CCF3B8345138Cdb57f241E5a1 (100 ETH)
12 (6) 0x2204a92a3b6527C1E506f28681F83a24f337d7d0 (100 ETH)
13 (7) 0x7C37a440f2899922e22a9Bb427A36DED3d21DbFD (100 ETH)
14 (8) 0x7709f03A8a7AB87727c61C850c180d7C2f5192A1 (100 ETH)
15 (9) 0xf978D06d16293889D212F9a19cE41E5985A2C340 (100 ETH)
16 |
17 Private Keys
18 =====
19 (0) 0x92bdb1add582770c9a16d5ffc1be99ef655e338d9ca02035c1d30e7b032b6846
20 (1) 0x009bed89b474d01521562066c94efb196c56ecf602f1961bf1c1c2c0a5db6915
21 (2) 0x961529dc879566968c3f0ef88206346e77195caf9263df404696c048290917a
22 (3) 0x75d188701cc0c738a2559319da1af9ab06312787a15eceec7e8289dfc60fe464
23 (4) 0x4b6b88ea03fde80bd9ac0be24e676d9afe85a3337f4ebb22f8f4341d3d01f7e1
24 (5) 0x90990303cdb69f3f0691c37bae80bbc355d7fb45d24efd6c8a0abc2ab25d0a05
25 (6) 0xf3b5c2668565e479a51332519cd52d5abf22acb63bdae078daed13b6c9d8596
26 (7) 0x8c56986978ca76184c384467b05de637b566b13a79a162b57744469454468f36
27 (8) 0x44f0e2742518b6d1542a4c5c2db75d9a91183410c20703986fc25520972d8b50
28 (9) 0x914b8755879f32c62ab43813206d994f9d8f39d706def98ba425c3f64b82522d
29
30 HD Wallet
31 =====
32 Mnemonic: month apology submit inside oyster into evidence example among future slot guide
33 Base HD Path: m/44'/60'/0/{account_index}
34
35 Gas Price
36 =====
37 20000000000
38
39 Gas Limit
40 =====
41 6721975

13 ▲ 0 Ln 16, Col 1 Spaces: 4 UTF-8 CRLF JavaScript





Workshop 2 Solidity

workshop นี้ทำอะไร

1. เราจะเริ่มจากเรียน smart contract structure กันก่อน
2. แล้วจากนั้นเราจะเรียน solidity value type, function and constructor, special function and variable ผ่านตัวอย่าง lottery contract
3. เราจะเก็บตก trick & tip ต่างๆในการเขียน solidity



Smart Contract Structure

```
pragma solidity ^0.5.11;

contract Storage {

    uint256 public data;

    function setData(uint256 _data) public{
        data = _data;
    }
}
```



```
pragma solidity ^0.5.11;

contract Storage {

    uint256 public data;

    function setData(uint256 _data) public{
        data = _data;
    }
}
```



```
pragma solidity ^0.5.11;

contract Storage {

    uint256 public data;

    function setData(uint256 _data) public{
        data = _data;
    }
}
```

Contract



```
pragma solidity ^0.5.11;

contract Storage {

    uint256 public data; State Variable

    function setData(uint256 _data) public{
        data = _data;
    }
}
```



```
pragma solidity ^0.5.11;

contract Storage {

    uint256 public data;

    function setData(uint256 _data) public{
        data = _data;
    }
}
```

Function



Smart Contract Structure

```
pragma solidity ^0.5.11;
```

Pragma

```
contract Storage {
```

```
    uint256 public data;
```

State Variable

```
    function setData(uint256 _data) public{
        data = _data;
    }
```

Function

```
}
```

Contract



Editor: Remix-Ethereum

The screenshot shows the Remix-Ethereum IDE interface. At the top, there's a browser header with the URL `remix.ethereum.org/#optimize=false&evmVersion=null`. On the left, a sidebar titled "FILE EXPLORERS" shows a "browser" section with files "ballot_test.sol" and "ballot.sol". The main area has a blue header bar with the text "The new layout has arrived" and two buttons: "Learn more" and "Use previous version". Below this, there are sections for "Environments" (with "Solidity" highlighted by a red circle) and "Featured Plugins" (listing "Pipeline" and "Debugger" with a "See all Plugins" button). The "File" section includes "New File", "Open Files", "Connect to Localhost", and "Import From:". The "Resources" section lists "Documentation", "Gitter channel", and "Medium Posts". At the bottom, there's a footer with a search bar and a note about using exports/register/remove/clear.



Editor: Remix-Ethereum

The screenshot shows the Remix-Ethereum IDE interface. On the left, there's a sidebar with icons for remix, browser, and other tools. A red circle labeled '1 click here' points to the 'remix' icon. In the center, a 'FILE EXPLORERS' section shows a folder named 'browser' containing files 'ballot_test.sol' and 'ballot.sol'. A red circle labeled '2 New File' points to the '+' icon next to the folder. A modal window titled 'Create new file' is open in the center, with a red circle labeled '3' pointing to the 'OK' button. The modal has a 'File Name' input field containing 'LotterySeller.sol'. Below the modal, the main interface shows sections for 'Environments' (Solidity, Vyper), 'Featured Plugins' (Pipeline, Debugger, See all Plugins), 'File' (New File, Open Files, Connect to Localhost, Import From), and 'Resources' (Documentation, Gitter channel, Medium Posts). At the bottom, there's a search bar and a note about using exports/register/unregister.

1 click here

2 New File

3

File Name

LotterySeller.sol

OK Cancel

Environments

Solidity Vyper

Featured Plugins

Pipeline Debugger See all Plugins

File

New File Open Files Connect to Localhost Import From:

Resources

Documentation Gitter channel Medium Posts

Search with transaction hash or address

• Use exports/.register(key, obj)/.remove(key)/.clear() to register and reuse object across script executions.



Editor: Remix-Ethereum

The screenshot shows the Remix Ethereum IDE interface. The title bar reads "Remix - Ethereum IDE". The address bar shows the URL "remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js". The left sidebar has a "FILE EXPLORERS" section with a "browser" folder containing "LotterySeller.sol", "ballot_test.sol", and "ballot.sol". The main editor area displays the text "Ready to Code!". At the bottom, there is a search bar with the placeholder "Search with transaction hash or address" and some help text: "Run a JavaScript script." followed by a bullet point: "• Use exports/.register(key, obj)/.remove(key)/.clear() to register and reuse object across script executions."





Smart Contract Outline

State Variable

- chairman
- price
- copy
- lottery
- owner
- sellCount

Function

- buy lottery
- how many left?
- check lottery owner
- check state
- contract owner
- price
- copy



Smart Contract Outline

State Variable

- chairman address
 - price uint
 - copy uint
 - lottery
 - owner
 - sellCount
- }
- Struct

Function

- buy lottery
- how many left?
- check lottery owner
- check state
- contract owner
- price
- copy



```
pragma solidity ^0.5.11;
contract LotteryShop{
    struct Lottery{
        address[] lotteryOwner;
        uint sellCount ;
    }
    Lottery[1000000] public lottery;
    uint public copy;
    uint public price;
    address public contractOwner;
}
```



Type

Visibility

Variable Name

uint256 public price;

Type

Visibility

Variable Name

address[1000000] public lottery;



Variable Type

Type	Example
bool	bool isTrue = true;
int_size	int number = -100;
uint_size	uint number = 100;
string	string message = "Hello, World!";
address	address wallet = 0xca35b7d915458ef540ade6068dfe2f44e8fa733c;
bytes_size	bytes code = 0x4beef;



Type	Example
enum	 <code>enum state{Created, Locked, Inactive};</code>
<code>ArrayType[_size]</code>	<code>uint[] set_of_Numbers;</code>
<code>struct _name{ _type _name; _type _name; }</code>	<code>struct Student{ string name; uint age; }</code>
<code>mapping(_key=>_value)</code>	 <code>mapping(address=>uint) wallet;</code>



Smart Contract Outline

State Variable

✓ chairman	address
✓ price	uint
✓ copy	uint
✓ lottery	
✓ owner	
✓ sellCount	

}

Struct

Function

- buy lottery
- how many left?
- check lottery owner
- ✓ check state
- ✓ contract owner
- ✓ price
- ✓ copy



```
pragma solidity ^0.5.11;
contract LotteryShop {
//state variables done!

constructor(uint _copy,uint _price) public{
    copy = _copy;
    price = _price;
    contractOwner = msg.sender;
}
}
```



```
constructor (<arguments>) public {}
```

A **constructor** is a special method that is used to **initialize** a newly created object.

After the **constructor** has executed, the final code of the contract is deployed to the blockchain.



```
pragma solidity ^0.5.11;
```

```
contract LotteryShop {  
    //state variables done!  
    //constructor done!
```

```
function getOwner(uint _number)public view returns(address[] memory){  
    return lottery[_number].lotteryOwner;  
}  
}
```



```
function _name(<input_arguments>
    {visibility}
    {state mutability}
    {modifier}★★
    returns(<return_parameter>){}
```

public - all can access

private - can be accessed only from this contract

internal - can be accessed only from this contract and its subclasses

external - can be accessed only from other contracts



Mutability	Read	Modify	note	Use Case
-none-	yes	yes	-	setter
payable	yes	yes	can received ETH!	deposit, purchase
view	yes	no	Free!	getter
pure	no	no	Free!	helper



Smart Contract Outline

State Variable

✓ chairman	address
✓ price	uint
✓ copy	uint
✓ lottery	
✓ owner	
✓ sellCount	

}

Struct

Function

- buy lottery payable
- how many left? view
- ✓ check lottery owner view
- ✓ check state
- ✓ contract owner
- ✓ price
- ✓ copy



```
pragma solidity ^0.5.11;
contract LotteryShop {
    //...

    function howManyLeft(uint _number)public view returns(uint){
        require(_number<1000000,"we do not have the number");
        return copy-lottery[_number].sellCount;
    }
}
```



require(bool condition, string memory message)

-if **condition** is false, abort the execution, revert state changes and return all the remaining gas to the sender. Also provides an error message

msg.sender - sender of the message (current call)

msg.value - number of wei sent with the message



Smart Contract Outline

State Variable

✓ chairman	address
✓ price	uint
✓ copy	uint
✓ lottery	
✓ owner	
✓ sellCount	

}

Struct

Function

- buy lottery payable
- ✓ how many left? view
- ✓ check lottery owner view
- ✓ check state
- ✓ contract owner
- ✓ price
- ✓ copy



```
pragma solidity ^0.5.11;
contract LotteryShop {
//...

function buyLottery(uint _number)public payable{
    require(msg.sender != address(0),"address(0) is not allowed");
    require(msg.value == price,"payment must be equal to price");
    require(lottery[_number].sellCount < copy,"soldout");
    require(_number<1000000,"we do not have the number");
    lottery[_number].lotteryOwner.push(msg.sender);
    lottery[_number].sellCount++;
}

}
```



Smart Contract Outline

State Variable

✓ chairman	address
✓ price	uint
✓ copy	uint
✓ lottery	
✓ owner	
✓ sellCount	

} Struct

Function

✓ buy lottery	payable
✓ how many left?	view
✓ check lottery owner	view
✓ check state	
✓ contract owner	
✓ price	
✓ copy	



1.click here

The screenshot shows the Remix Ethereum IDE interface. On the left, there's a sidebar with various icons and settings like 'Auto compile', 'Enable Optimization', and 'Hide warnings'. The main area has a title 'SOLIDITY COMPILER' and a sub-section 'Compiler' set to '0.5.11+commit.c082d...'. Below that is another 'Compiler' section with 'EVM Version' set to 'compiler default'. A prominent red circle highlights the 'Compile' button, which is part of a larger button labeled 'Compile LotterySeller.sol'. To the right of the compiler controls is the code editor window titled 'LotterySeller.sol' containing the following Solidity code:

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[100000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<100000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
}
```

At the bottom of the interface, there are buttons for 'Publish on Swarm', 'Compilation Details', 'ABI', and 'Bytecode'. The bottom pane shows a transaction history with several entries, including calls to the 'getOwner' function of the 'LotteryShop' contract.



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

SOLIDITY COMPILER

Compiler: 0.5.11+commit.c082d0b4

Language: Solidity

EVM Version: compiler default

2

Compile LotterySeller.sol

Compiler Configuration

Auto compile

Enable Optimization

Hide warnings

Contract: LotteryShop (LotteryS)

Publish on Swarm

Compilation Details

ABI Bytecode

Search with transaction hash or address

call to LotteryShop.getOwner

call to LotteryShop.getOwner

CALL [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>

Debug



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

SOLIDITY COMPILER

Compiler: 0.5.11+commit.c082d0b4

Language: Solidity

EVM Version: compiler default

Contract: LotteryShop (LotteryS)

Compile LotterySeller.sol

Compiler Configuration

Auto compile

Enable Optimization

Hide warnings

LotterySeller.sol

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[100000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<100000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27 }
```

3. done!

Publish on Swarm

Compilation Details

ABI Bytecode

Search with transaction hash or address

call to LotteryShop.getOwner

call to LotteryShop.getOwner

CALL [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>

Debug



Remix - Ethereum IDE

remix

SOLIDITY COMPILER

Compiler: 0.5.11+commit.c082d0b4.js

Language: Solidity

EVM Version: compiler default

Compile LotterySeller.sol

No Contract Compiled Yet

browser/LotterySeller.sol:35:6:
ParserError: Function, variable,
struct or modifier declaration
expected.
.:

LotterySeller.sol

```
address public contractOwner;
constructor(uint _copy,uint _price) public{
    copy = _copy;
    price = _price;
    contractOwner = msg.sender;
}

function getOwner(uint _number)public view returns(address[] memory){
    return lottery[_number].lotteryOwner;
}

function howManyLeft(uint _number)public view returns(uint){
    require(_number<1000000,"we do not have the number");
    return copy-lottery[_number].sellCount;
}

function buyLottery(uint _number)public payable{
    require(msg.sender != address(0),"address(0) is not allowed");
    require(msg.value == price,"payment must be equal to price");
    require(lottery[_number].sellCount < copy,"Lottery is soldout");
    require(_number<1000000,"we do not have the number");
    lottery[_number].lotteryOwner.push(msg.sender);
    lottery[_number].sellCount++;
}
```

ContractDefinition LotteryShop 0 reference(s)

0 listen on network Search with transaction hash or address

call to LotteryShop.getOwner

call to LotteryShop.getOwner

CALL [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>

If something is wrong, errors will show up



1.click here

DEPLOY & RUN TRANSACTIONS

Environment: JavaScript VM

Account: 0xca3...a733c

Gas limit: 3000000

Value: 0 wei

LotteryShop

Deploy: uint256 _copy, uint256 _price

or

At Address: Load contract from Address

Transactions recorded: 3

Deployed Contracts

call to LotteryShop.getOwner

call to LotteryShop.getOwner

CALL [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[1000000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<1000000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27
```



2. Select Injected Web3

The screenshot shows the Remix Ethereum IDE interface. On the left, there's a sidebar with various icons for file operations, deployment, and settings. The main area is titled "DEPLOY & RUN TRANSACTIONS". In the "Environment" section, a dropdown menu is open, showing "JavaScript VM" (which is highlighted with a red circle) and "Injected Web3". Below the dropdown, it says "Execution environment has been provided by Metamask or similar provider.". The "Account" section shows a single account listed. The "Gas limit" is set to 3000000. The "Value" field is set to 0 wei. The "Contract" dropdown is set to "LotteryShop". The "Deploy" button is orange, and below it, there's an "At Address" option and a "Load contract from Address" button. The "Transactions recorded:" section shows 3 recorded transactions. The bottom part of the interface shows a list of recent calls made to the "LotteryShop.getOwner" function, with the most recent one being a call from address 0xca35b7d915458ef540ade6068dfe2f44e8fa733c to the contract. There's also a search bar and a "Debug" button.

```
pragma solidity ^0.5.11;
contract LotteryShop{
    struct Lottery{
        address[] lotteryOwner;
        uint sellCount ;
    }
    Lottery[100000] public lottery;
    uint public price;
    address public contractOwner;
    constructor(uint _copy,uint _price) public{
        copy = _copy;
        price = _price;
        contractOwner = msg.sender;
    }
    function getOwner(uint _number)public view returns(address[] memory){
        return lottery[_number].lotteryOwner;
    }
    function howManyLeft(uint _number)public view returns(uint){
        require(_number<100000,"we do not have the number");
        return copy-lottery[_number].sellCount;
    }
}
```



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

DEPLOY & RUN TRANSACTIONS

Environment: Injected Web3 (Kovan (42) network)

Account: 3000000 wei

Value: 0 wei

Contract: LotteryShop

Deploy: 88,70

Transactions recorded: 0

Deployed Contracts: Currently you have no contract instances to interact with.

LotterySeller.sol

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[100000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<100000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27 }
```

call to LotteryShop.getOwner

call to LotteryShop.getOwner

CALL [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

DEPLOY & RUN TRANSACTIONS

Environment: Injected Web3 (Kovan (42) network)

Account: 3000000 wei

Gas limit: 0 wei

Value: 0 wei

Contract: LotteryShop

Deploy: 88.0

4. click deploy!

Transactions recorded: 0

Deployed Contracts

Currently you have no contract instances to interact with.

LotterySeller.sol

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[1000000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<1000000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27
```

call to LotteryShop.getOwner

call to LotteryShop.getOwner

call [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>



RENDERING

remix

DEPLOY & RUN TRANSACTIONS

Environment Injected Web3

Kovan (42) network

Account 0x00aa39d30f

Gas limit 3000000

Value 0 wei

LotteryShop

Deploy 88.70

At Address Load contract from Address

Transactions recorded: 1

Deployed Contracts

Currently you have no contract instances to interact with.

Home LotterySeller.sol

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[100000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<100000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27
```

call to LotteryShop.getOwner

creation of LotteryShop pending...

[vm] from:0xca3...a733c to:LotteryShop.(constructor) value:0 wei data:0x608...00046 logs:0 hash:0x449...04767

creation of LotteryShop pending...

>

Kovan Test Network

node1 author → New Contract

CONTRACT DEPLOYMENT

0

DETAILS DATA

GAS FEE 0.000597 No Conversion Rate Available

AMOUNT + GAS FEE

TOTAL 0.000597 No Conversion Rate Available

Reject Confirm

Debug

5. Metamask popup will show up
click Confirm to continue



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

DEPLOY & RUN TRANSACTIONS

Environment: Injected Web3 (Kovan (42) network)

Account: 0x00a...597c2

Gas limit: 3000000

Value: 0 wei

Contract: LotteryShop

Deploy: 88,70

At Address: Load contract from Address

Transactions recorded: 1

Deployed Contracts: LotteryShop at 0xa78...d61b7 (block 0)

LotterySeller.sol

```
pragma solidity ^0.5.11;
contract LotteryShop{
    struct Lottery{
        address[] lotteryOwner;
        uint sellCount ;
    }
    Lottery[1000000] public lottery;
    uint public copy;
    uint public price;
    address public contractOwner;
    constructor(uint _copy,uint _price) public{
        copy = _copy;
        price = _price;
        contractOwner = msg.sender;
    }
    function getOwner(uint _number)public view returns(address[] memory){
        return lottery[_number].lotteryOwner;
    }
    function howManyLeft(uint _number)public view returns(uint){
        require(_number<1000000,"we do not have the number");
        return copy-lottery[_number].sellCount;
    }
}
```

call to LotteryShop.getOwner

creation of LotteryShop pending...

[vm] from:0xca3...a733c to:LotteryShop.(constructor) value:0 wei data:0x608...00046 logs:0 hash:0x449...04767

creation of LotteryShop pending...

<https://kovan.etherscan.io/tx/0x537cb471bc4a0a191e9233c9a68fe295347dd1e7b0d725db77a004b721ecb3e4>

6. Our smart contract will show up here



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

DEPLOY & RUN TRANSACTIONS

LotteryShop

Deploy 88,70

At Address Load contract from Address

Transactions recorded: 1

Deployed Contracts

LotteryShop at 0xa...d61b7 (b)

buyLottery uint256 _number

contractOw... (button circled in red)

copy

getOwner uint256 _number

howManyLeft uint256 _number

lottery howManyLeft - call uint256

price

Home LotterySeller.sol

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[100000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<100000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27
```

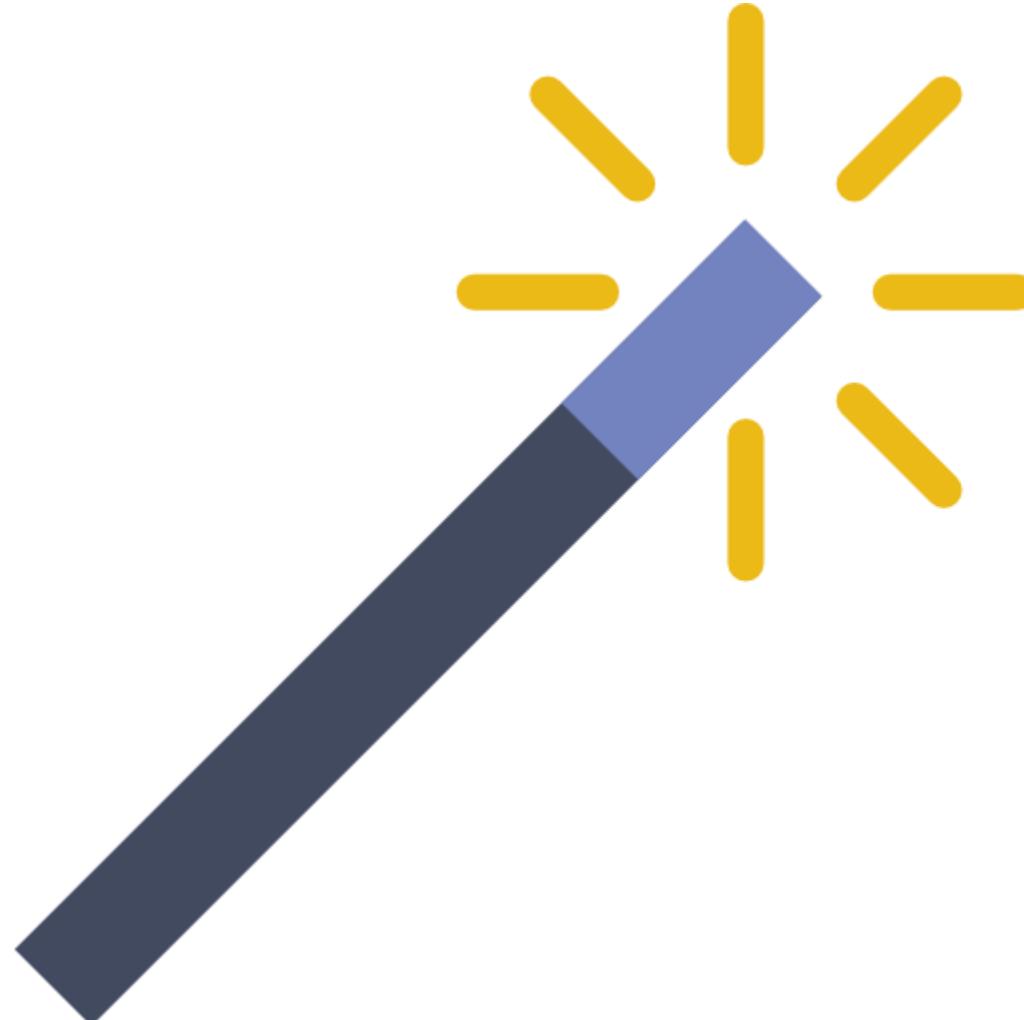
0 listen on network Search with transaction hash or address

[vm] from:0xca3...a733c to:LotteryShop.(constructor) value:0 wei data:0x608...00046 logs:0 hash:0x449...04767 Debug

creation of LotteryShop pending...

<https://kovan.etherscan.io/tx/0x537cb475bc4a0a191e9233c9a68fe295347dd1e7b0d725db77a004b721ecb3e4>

[block:12970684 txIndex:0] from:0x0a...597c2 to:LotteryShop.(constructor) value:0 wei data:0x608...00046 logs:0 hash:0x537...cb3e4 Debug





value Type key Variable Name
address[5] wallet;

Variable Name key value
wallet[4]= 0xaA;

key value Type Variable Name
mapping(uint => address) wallet;



```
mapping(string => string) dictionary;
```

```
dictionary["ant"] = "a small insect";
```



key value Type Variable Name
mapping(string => string) dictionary;

Variable Name key value
dictionary["ant"]="a small insect";



key value Type Variable Name
mapping(address => uint) score;

Variable Name key value
score[0x045ad] = 99;



```
pragma solidity ^0.5.11;
contract test{
    mapping(address => uint) score;
```

```
function getGrade(address student)public view returns (uint){
    uint a = score[student];
    if(a > 80) return 0;← A
    else if(a > 70) return 1;← B
    else if(a > 60) return 2;← C
    else if(a > 50) return 3;← D
    else return 4;← F
}
```



```
pragma solidity ^0.5.11;
contract test{
    mapping(address => uint) score;
    enum grade {A,B,C,D,F}

    function getGrade(address student)public view returns (grade){
        uint a = score[student];
        if(a > 80) return grade.A;
        else if(a > 70) return grade.B;
        else if(a > 60) return grade.C;
        else if(a > 50) return grade.D;
        else return grade.F;
    }
}
```



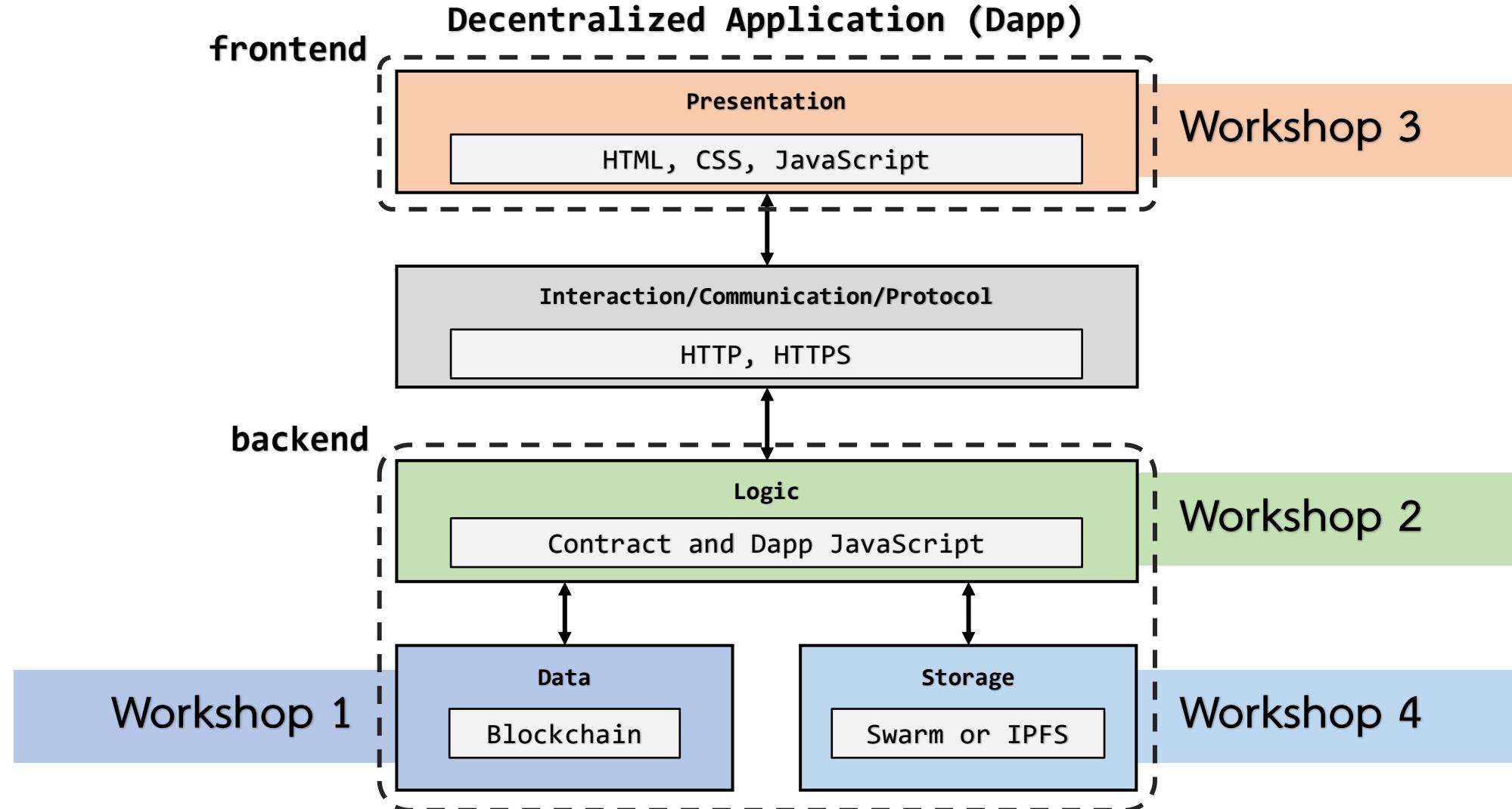
```
contract test{
    address owner;
    constructor()public{
        owner = msg.sender;
    }
    function superPower1()public{
        require(msg.sender == owner);
        //...
    }
    function superPower2()public{
        require(msg.sender == owner);
        //...
    }
}
```

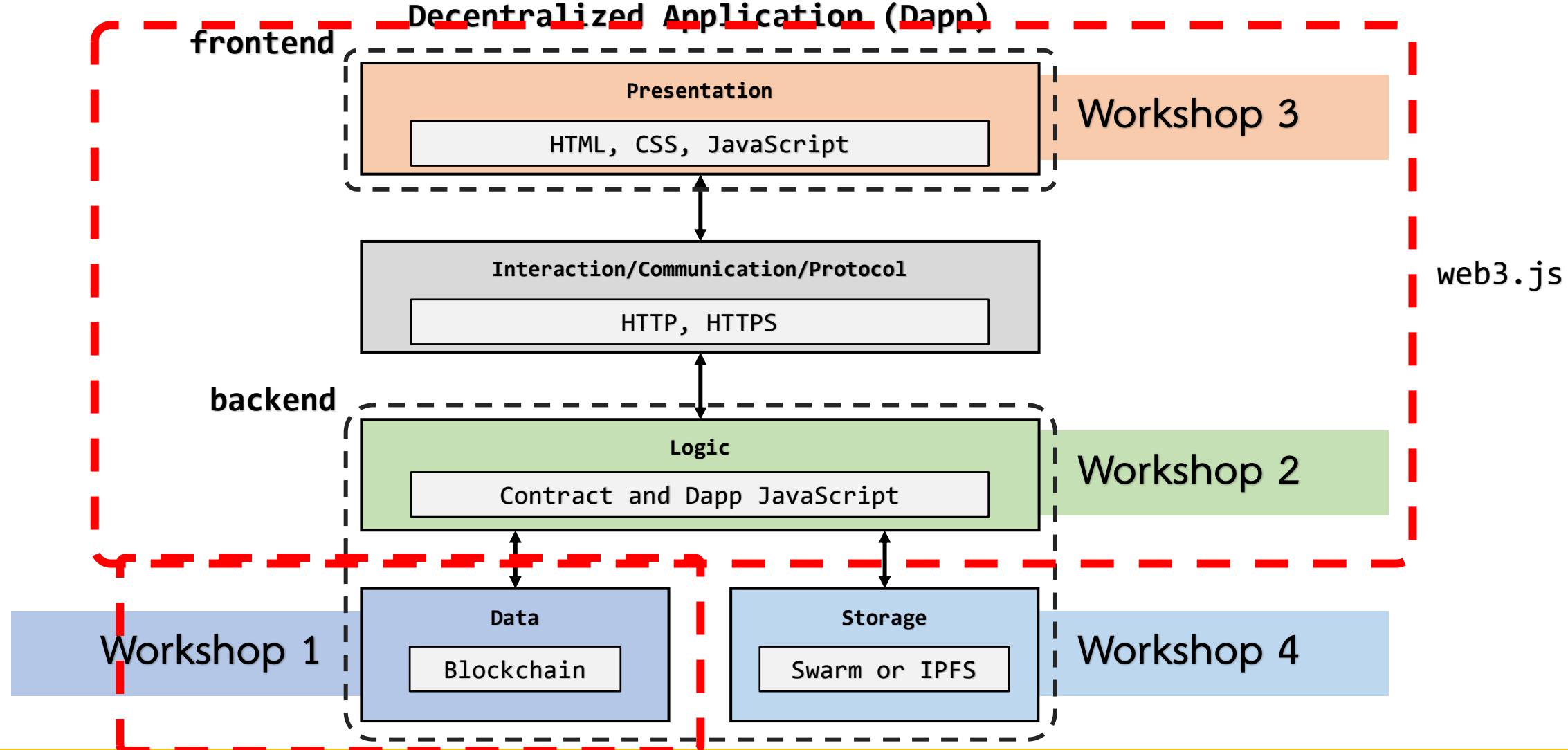


```
contract test{
    address owner;
    constructor()public{
        owner = msg.sender;
    }
    modifier onlyOwner{
        require(msg.sender==owner);
        _;
    }
    function superPower1()public onlyOwner{
        //...
    }
    function superPower2()public onlyOwner{
        //...
    }
}
```



```
< + browser/erc20.sol >
1 pragma solidity ^0.5.0;
2 > library SafeMath { }
30 // -----
31 // ERC Token Standard #20 Interface
32 // https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md
33 // -----
34 > contract ERC20Interface {
35     function totalSupply() public view returns (uint);
36     function balanceOf(address tokenOwner) public view returns (uint balance);
37     function allowance(address tokenOwner, address spender) public view returns (uint remaining);
38     function transfer(address to, uint tokens) public returns (bool success);
39     function approve(address spender, uint tokens) public returns (bool success);
40     function transferFrom(address from, address to, uint tokens) public returns (bool success);
41
42     event Transfer(address indexed from, address indexed to, uint tokens);
43     event Approval(address indexed tokenOwner, address indexed spender, uint tokens);
44 }
45 > contract ApproveAndCallFallBack { }
46 > contract Owned { }
63 // -----
64 // ERC20 Token, with the addition of symbol, name and decimals and a
65 // fixed supply
66 // -----
67 > contract FixedSupplyToken is ERC20Interface, Owned {
68     using SafeMath for uint;
69
70     string public symbol;
71     string public name;
72     uint8 public decimals;
73     uint _totalSupply;
```







Workshop 3 Front End

workshop นี้ทำอะไร

1. เราจะสร้าง Web Application กัน ด้วยนั่นสิ่งที่ต้องรู้ก็คือภาษาที่ใช้เขียนเว็บได้แก่

1.1 HTML (div, input, button)

1.2 JavaScript (document.innerHTML, document.value)

1.3 JavaScript [, callback], Promise, Async/Await

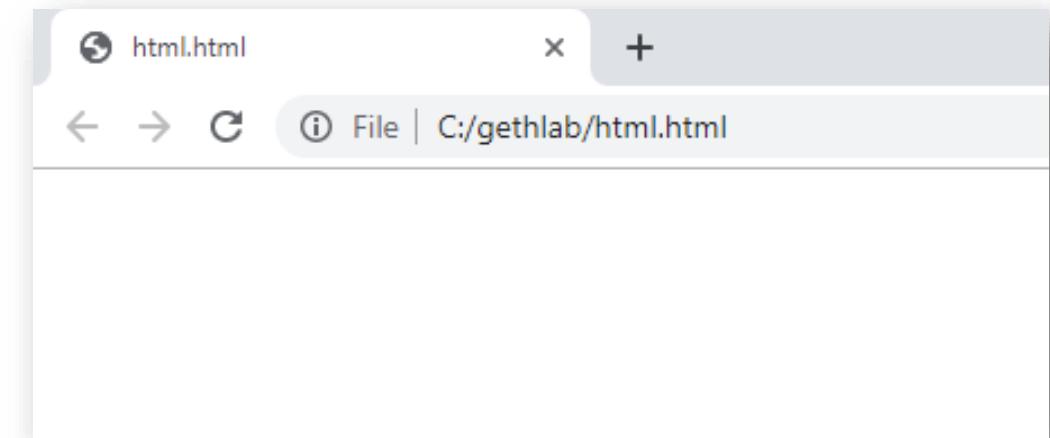
2. เราจะใช้ web3.js module เป็น open source ที่ใช้ในการติดต่อกับ Ethereum blockchain

2.1 web3.js script



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>my first HTML page!</title>
</head>
<body>

</body>
</html>
```



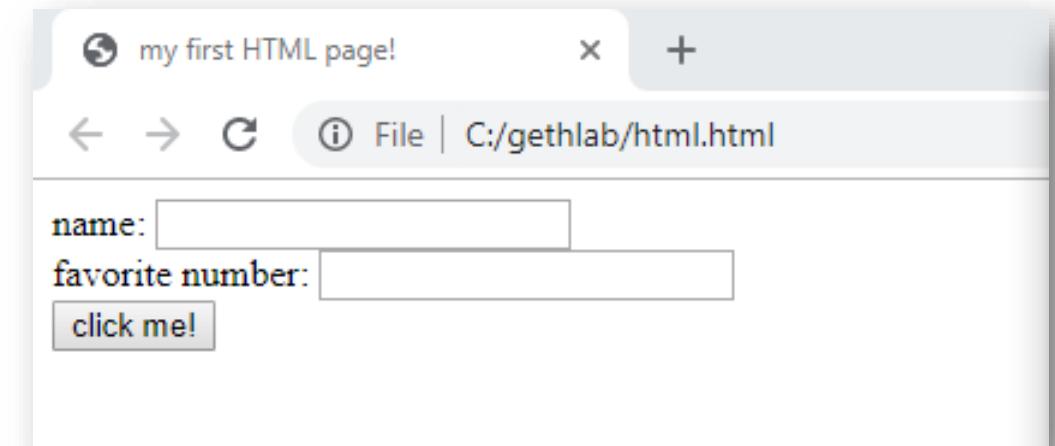


```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
</head>
<body>
    name: <input type="text" id="name"><br>
    favorite number: <input type="number" id="fav_num"><br>
</body>
</html>
```

The screenshot shows a web browser window titled "my first HTML page!". The address bar indicates the file is located at "C:/gethlab/html.html". The browser interface includes standard navigation buttons (back, forward, refresh) and a "File" menu. Below the header, there are two input fields. The first field is labeled "name:" and contains a text input box. The second field is labeled "favorite number:" and contains a numeric input box.



```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
</head>
<body>
    <!--input-HTML done!-->
    <button onclick="myFunction()">click me!</button>
</body>
</html>
```





```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
</head>
<body>
    <!--input-HTML done!-->
    <!--button done!-->
    <div id="placeholder"></div>
</body>
</html>
```

my first HTML page!

File | C:/gethlab/html.html

name:

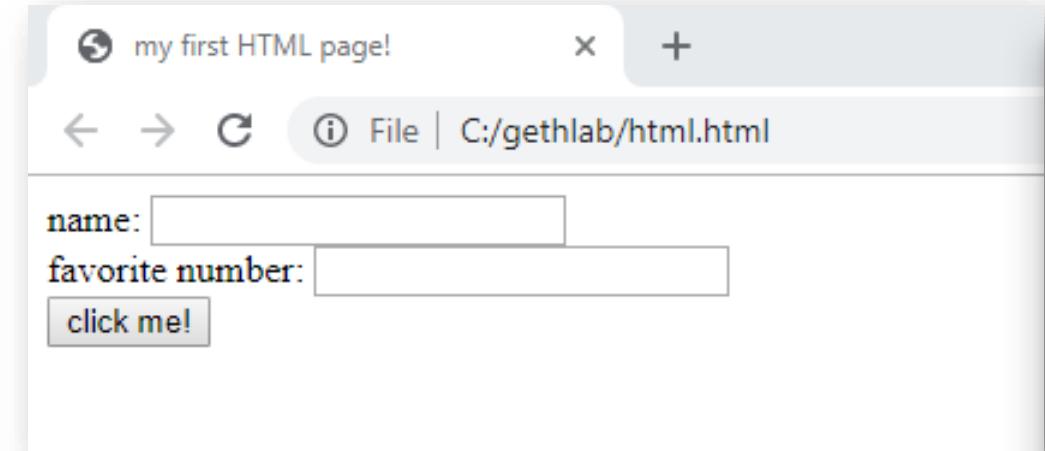
favorite number:

click me!

Output-HTML

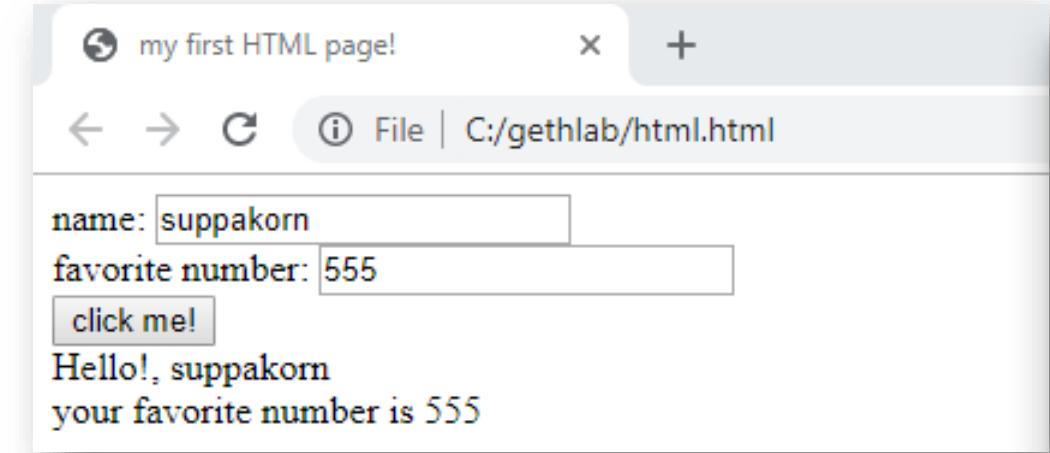


```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
</head>
<body>
    <!--input-HTML done!-->
    <!--button done!-->
    <!--output-HTML done!-->
    <script>
        function myFunction(){
            let name = document.getElementById("name").value
            let fav_num = document.getElementById("fav_num").value
        }
    </script>
</body>
</html>
```



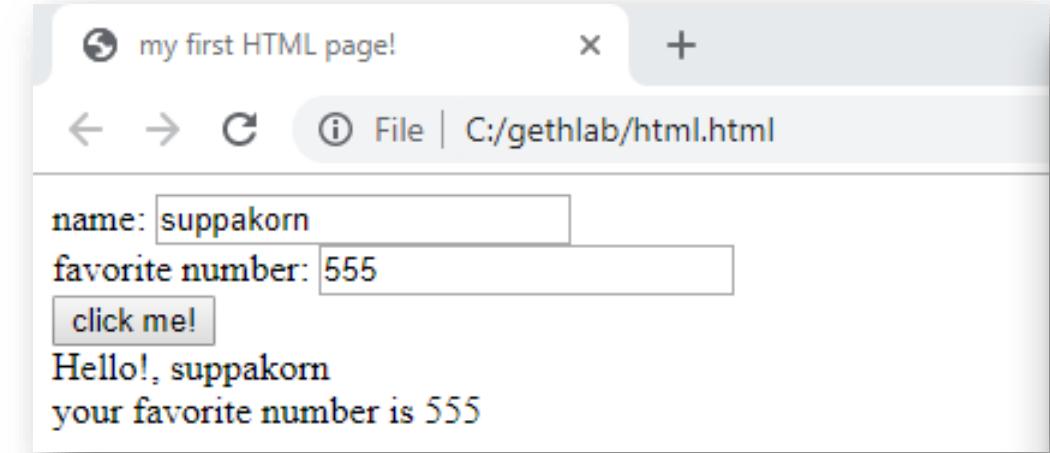


```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
</head>
<body>
    <!--input-HTML done!-->
    <!--button done!-->
    <!--output-HTML done!-->
    <script>
        function myFunction(){
            //input-JS done!
            document.getElementById("placeholder").innerHTML = "Hello!, "+name+"<br>
            your favorite number is "+fav_num
        }
    </script>
</body>
</html>
```



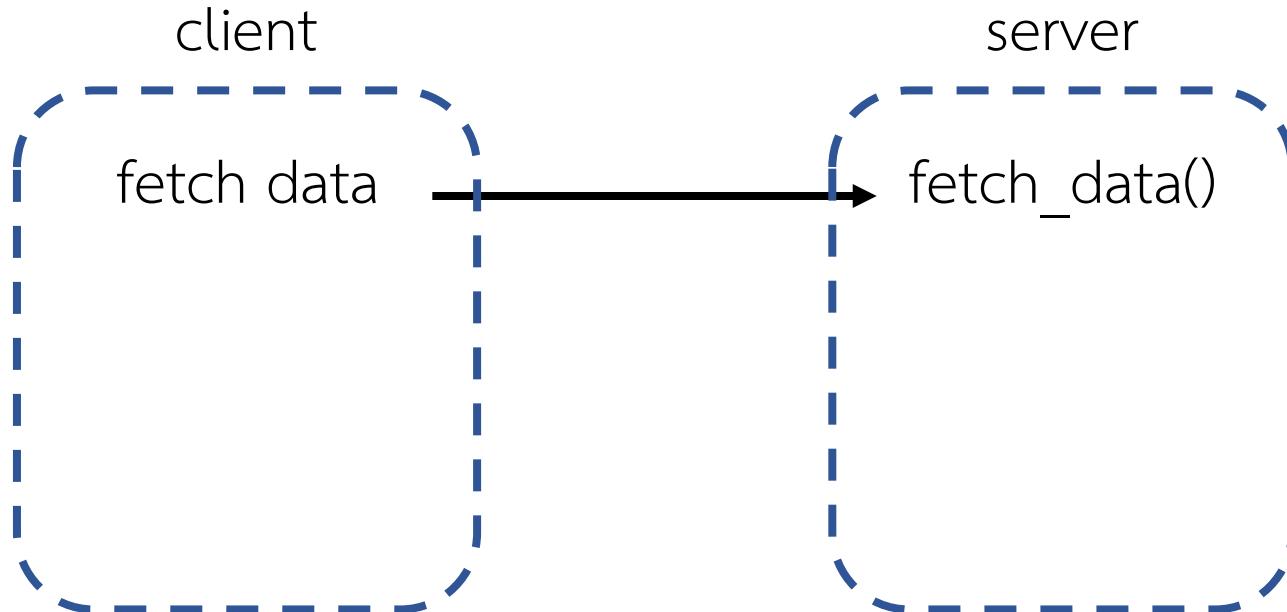


```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>my first HTML page!</title>
</head>
<body>
    name: <input type="text" id="name"></input><br>
    favorite number: <input type="number" id="fav_num"></input><br>
    <script>
        let name = document.getElementById("name").value;
        let fav_num = document.getElementById("fav_num").value;
    </script>
    <button onclick="myFunction()">click me!</button>
    <div id="placeholder"></div>
    <script>
        function myFunction(){
            let name = document.getElementById("name").value;
            let fav_num = document.getElementById("fav_num").value;
            document.getElementById("placeholder").innerHTML = "Hello!, "+name+"<br> your favorite number is "+fav_num;
        }
    </script>
</body>
</html>
```



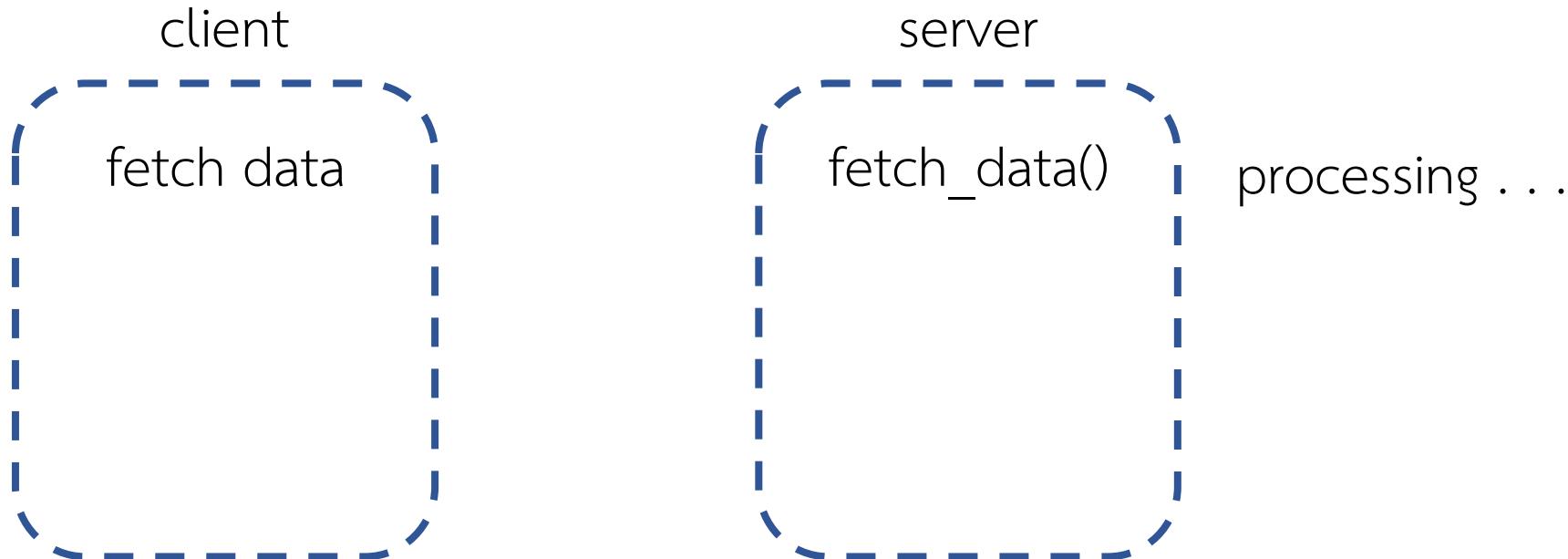


```
<script>
    function myFunction(){
        let name = fetch_data_from_blockchain()
        document.getElementById("placeholder").innerHTML = name
    }
</script>
```



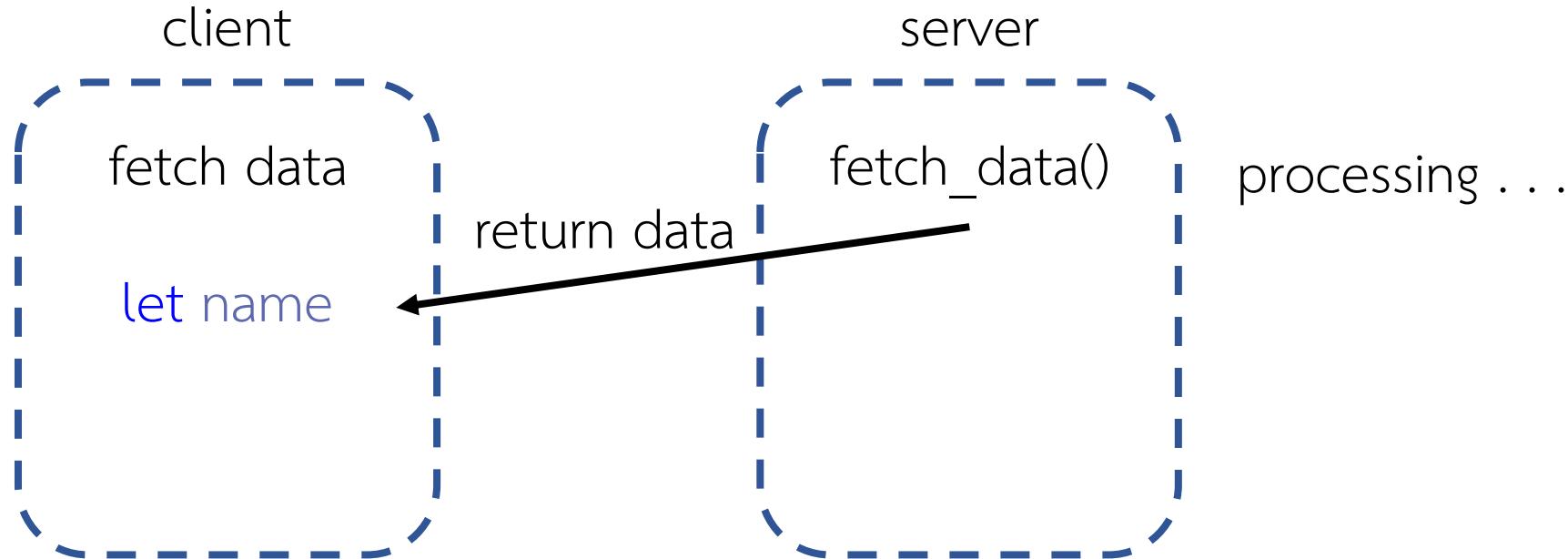


```
<script>
    function myFunction(){
        let name = fetch_data_from_blockchain()
        document.getElementById("placeholder").innerHTML = name
    }
</script>
```





```
<script>
    function myFunction(){
        let name = fetch_data_from_blockchain()
        document.getElementById("placeholder").innerHTML = name
    }
</script>
```



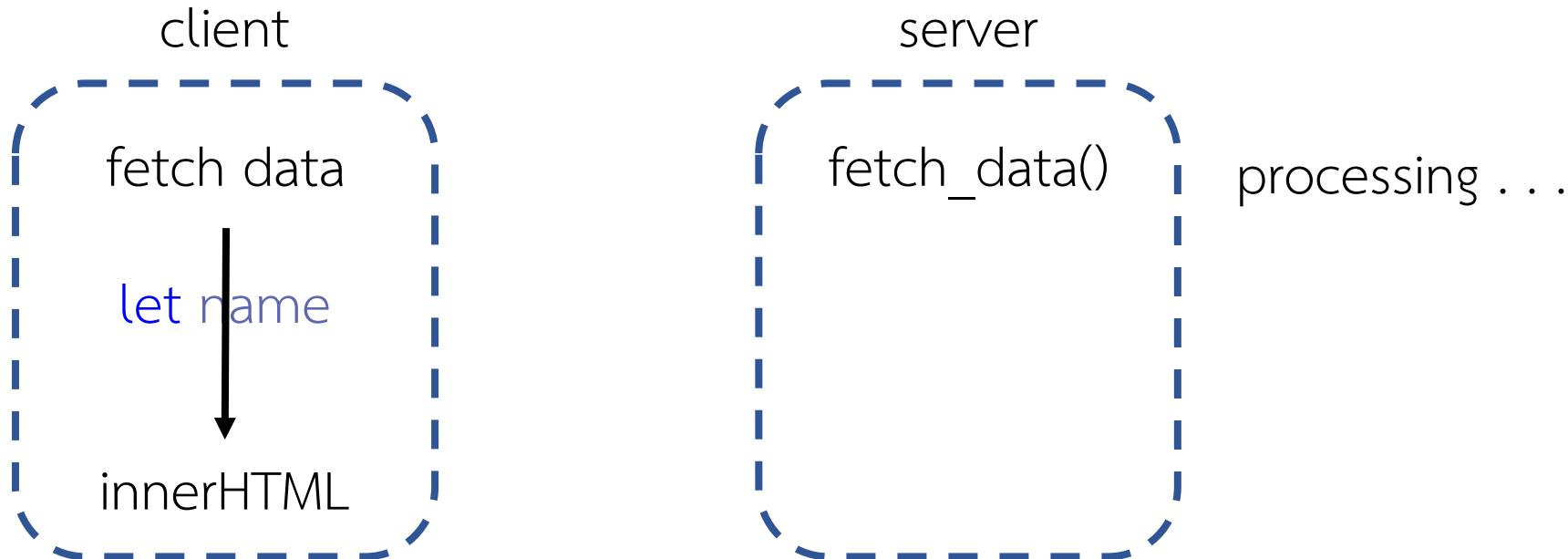


```
<script>
    function myFunction(){
        let name = fetch_data_from_blockchain()
        document.getElementById("placeholder").innerHTML = name
    }
</script>
```





```
<script>
    function myFunction(){
        let name = fetch_data_from_blockchain()
        document.getElementById("placeholder").innerHTML = name
    }
</script>
```





```
function myFunction(){
    let name = fetchDataFromBlockchain(
        ,function(error,data){
            if(error){
                console.log(error)
            }else{
                name = data;
                document.getElementById("placeholder").innerHTML = name
            }
        });
}
```



```
dataBase.verifyUser(username, password, (error, userInfo) => {
    if (error) {
        console.log(error)
    }else{
        dataBase.getRoles(username, (error, roles) => {
            if (error){
                console.log(error)
            }else {
                dataBase.logAccess(username, (error) => {
                    if (error){
                        console.log(error)
                    }else{
                        User.Update(userInfo, roles)
                    }
                })
            }
        })
    });
});
```



```
database.verifyUser(username, password)
  .then(userInfo => DataBase.getRoles(userInfo))
  .then(rolesInfo => DataBase.logAccess(rolesInfo))
  .then(finalResult => {
    User.Update(userInfo, roles)
  })
  .catch((err) => {
    console.log(err)
  })
}
```



```
async function(username, password){  
    try {  
        const userInfo = await DataBase.verifyUser(username, password);  
        const rolesInfo = await DataBase.getRoles(userInfo);  
        const logStatus = await DataBase.logAccess(userInfo);  
        User.Update(userInfo,rolesInfo);  
    }catch (e){  
        console.log(e);  
    }  
};
```



- <input>
- <button>
- <div>
- document.getElementById("").innerHTML
- document.getElementById("").value
- Async/Await



1. Run command

C:\frontend>**http-server**

A screenshot of a terminal window titled '1: node'. The window shows the command 'C:\frontend>http-server' being run, followed by output indicating the server is starting up and serving files from the current directory. It provides two available addresses: 'http://10.42.183.10:8080' and 'http://127.0.0.1:8080'. A note at the bottom says 'Hit CTRL-C to stop the server'.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: node
C:\frontend>http-server
Starting up http-server, serving .
Available on:
  http://10.42.183.10:8080
  http://127.0.0.1:8080
Hit CTRL-C to stop the server
```



สร้างไฟล์ index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>
  hello

</body>
</html>
```



The screenshot shows a list of functions for a blockchain contract:

- buyLottery** (orange button) - uint256 _number
- contractOw...** (blue button)
- copy** (blue button)
- getOwner** (blue button) - uint256 _number
- howManyLeft** (blue button) - uint256 _number
- lottery** (blue button) - uint256
- price** (blue button)

localhost:8080/444.html

ContractOwner : 0xAcC4B24f45191F99f547df190081BeE16A0bA45c

Copy : 5

Price : 0

หมายเลขสลากกินแบ่ง:

[getOwner](#) [howManyLeft](#) [buyLottery](#)

Elements Console Sources Network Performance »

top

▶ ["0xacc4b24f45191f99f547df190081bee16a0ba45c"]

>

Console What's New

Highlights from the Chrome 76 update

Autocomplete with CSS keyword values

Typing a keyword value like "bold" in the Styles pane now
autocomplete to "font-weight: bold".

A new UI for network settings

The "Use large request rows", "Group by frame", "Show overview",
and "Capture screenshots" options have moved to the new
Network Settings pane.





The screenshot shows a list of functions for a lottery contract. The functions are:

- buyLottery** (orange button) - uint256 _number
- contractOw...** (blue button)
- copy** (blue button)
- getOwner** (blue button) - uint256 _number
- howManyLeft** (blue button) - uint256 _number
- lottery** (blue button) - uint256
- price** (blue button)

The buttons for **contractOw...** and **price** are highlighted with red boxes.



The screenshot displays a user interface for interacting with a blockchain contract. It features several buttons and input fields:

- A top button labeled "buyLottery" with a "uint256 _number" input field.
- A "copy" button.
- Two buttons highlighted with red boxes:
 - "getOwner" with a "uint256 _number" input field.
 - "howManyLeft" with a "uint256 _number" input field.
- A bottom button labeled "lottery" with a "uint256" input field.
- A bottom-most button labeled "price".



```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
</head>
<body>
    <div id="showContractOwner"></div>
    <div id="showCopy"></div>
    <div id="showPrice"></div>
</body>
</html>
```



HTML

The screenshot shows a list of functions for a lottery contract. Each function is represented by a blue button with its name and a parameter type. To the left of each button is a green circular icon with a white checkmark. The 'copy' button under 'contractOw...' and the 'price' button under 'price' are both highlighted with a red rectangular border.

Function	Parameter Type
buyLottery	uint256 _number
contractOw...	
copy	
getOwner	uint256 _number
howManyLeft	uint256 _number
lottery	uint256
price	

JavaScript



```
<body>
```

```
  <!--dashboard done!-->
```

```
  หมายเลขสลากกินแบ่ง: <input type="number" id="lottery"><br>
    <button onclick="getOwner()">getOwner</button>
    <button onclick="howManyLeft()">howManyLeft</button>
    <button onclick="buyLottery()">buyLottery</button>
    <div id="showGetOwner"></div>
    <div id="showHowManyLeft"></div>
    <div id="showBuyLottery"></div>
```

```
</body>
```



HTML

The screenshot shows a list of functions for a lottery contract. Each function is accompanied by a green checkmark icon. The functions are:

- buyLottery (orange button, highlighted with a red box)
- contractOwner (blue button)
- copy (blue button)
- getOwner (blue button)
- howManyLeft (blue button, highlighted with a red box)
- lottery (blue button)
- price (blue button)

Each function has a corresponding input field next to it, labeled "uint256 _number". A dropdown arrow icon is located to the right of each input field.

JavaScript



1. Run command

```
C:\frontend>browserify -r web3 > bundle.js
```

```
js bundle.js  x
js bundle.js
1  require=(function(){function r(e,n,t){function o(i,f){if(!n[i]){if(!e[i]){var c="function"==typeof require&&require;if(!f&&c)return c(i,!0);i
2  var asn1 = exports;
3
4  asn1.bignum = require('bn.js');
5
6  asn1.define = require('./asn1/api').define;
7  asn1.base = require('./asn1/base');
8  asn1.constants = require('./asn1/constants');
9  asn1.decoders = require('./asn1/decoders');
10  asn1.encoders = require('./asn1/encoders');
11
12  },{"./asn1/api":2,"./asn1/base":4,"./asn1/constants":8,"./asn1/decoders":10,"./asn1/encoders":13,"bn.js":16}],2:[function(require,module,exports)
13  var asn1 = require('../asn1');
14  var inherits = require('inherits');
15
16  var api = exports;
17
18  api.define = function define(name, body) {
      
```



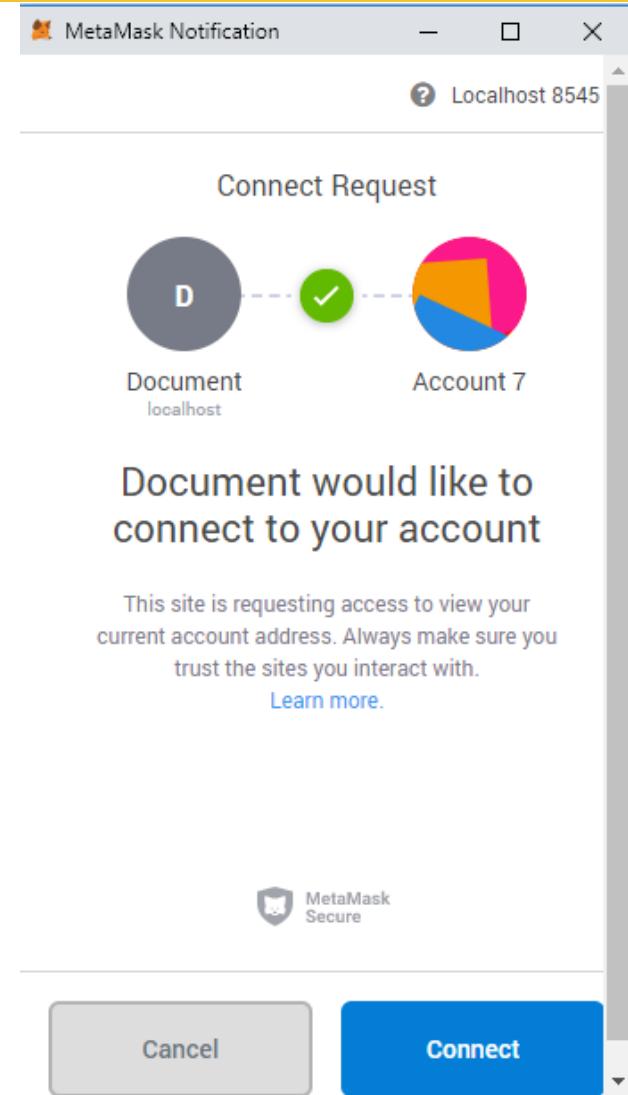
1. Run command

C:\frontend>browserify -r web3 -r ipfs -r buffer > bundle.js

```
JS bundle.js x
JS bundle.js
1 require=(function(){function r(e,n,t){function o(i,f){if(!n[i]){if(!e[i]){var c="function"==typeof require&&require;if(!f&&c) return c(i,!0);
2 var asn1 = exports;
3
4 asn1.bignum = require('bn.js');
5
6 asn1.define = require('./asn1/api').define;
7 asn1.base = require('./asn1/base');
8 asn1.constants = require('./asn1/constants');
9 asn1.decoders = require('./asn1/decoders');
10 asn1.encoders = require('./asn1/encoders');
11
12 },{"../asn1/api":2,"./asn1/base":4,"./asn1/constants":8,"./asn1/decoders":10,"./asn1/encoders":13,"bn.js":16}],2:[function(require,module,exports){
13 var asn1 = require('../asn1');
14 var inherits = require('inherits');
15
16 var api = exports;
17
18 api.define = function define(name, body) {
```



```
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- ... -->
    <script src="./bundle.js"></script>
    <script>
        ethereum.enable()
        const Web3 = require('web3')
        const web3 = new Web3(Web3.givenProvider)
    </script>
</head>
<body>
    <!--UI done!-->
</body>
</html>
https://web3js.readthedocs.io/en/v1.2.1/getting-started.html
```





The **web3.eth** is for the ethereum blockchain and smart contracts

The **web3.shh** is for the whisper protocol to communicate p2p and broadcast

The **web3.bzz** is for the swarm protocol, the decentralized file storage

The **web3.utils** contains useful helper functions for Dapp developers.



The **web3.eth** is for the ethereum blockchain and smart contracts

The **web3.shh** is for the whisper protocol to communicate p2p and broadcast

The **web3.bzz** is for the swarm protocol, the decentralized file storage

The **web3.utils** contains useful helper functions for Dapp developers.



```
<head>
  <!-- ... -->
<script>
  //web3 done!
  const abi =
  const contractAddress = ""
  const lotteryContract = new web3.eth.Contract(abi,contractAddress)
</script>
</head>
```



contract instance

contract instance

```
const lotteryContract = new web3.eth.Contract(abi, contractAddress)
```

Application Binary Interface

a web3 methods

location of our contract



Application Binary Interface

1.click here

The screenshot shows the Remix Ethereum IDE interface. On the left, the Solidity Compiler sidebar is open, with the EVM Version dropdown circled in red. The main area displays a Solidity contract named 'LotterySeller.sol' with the following code:

```
pragma solidity ^0.5.11;
contract LotteryShop{
    struct Lottery{
        address[] lotteryOwner;
        uint sellCount ;
    }
    Lottery[100000] public lottery;
    uint public copy;
    uint public price;
    address public contractOwner;
}
constructor(uint _copy,uint _price) public{
    copy = _copy;
    price = _price;
    contractOwner = msg.sender;
}
function getOwner(uint _number)public view returns(address[] memory){
    return lottery[_number].lotteryOwner;
}
function howManyLeft(uint _number)public view returns(uint){
    require(_number<100000,"we do not have the number");
    return copy-lottery[_number].sellCount;
}
```

Below the code, there are buttons for 'Publish on Swarm' and 'Compilation Details'. The 'Compilation Details' button is highlighted in grey. At the bottom, there are tabs for 'ABI' and 'Bytecode', with 'ABI' selected. The bottom right corner shows a 'Debug' button.



Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

SOLIDITY COMPILER

Compiler: 0.5.11+commit.c082d0b4.js

Language: Solidity

EVM Version: compiler default

Compile LotterySeller.sol

Compiler Configuration

Auto compile

Enable Optimization

Hide warnings

Contract: LotteryShop (LotteryS)

LotterySeller.sol

```
1 pragma solidity ^0.5.11;
2 contract LotteryShop{
3     struct Lottery{
4         address[] lotteryOwner;
5         uint sellCount ;
6     }
7
8     Lottery[1000000] public lottery;
9     uint public copy;
10    uint public price;
11    address public contractOwner;
12
13    constructor(uint _copy,uint _price) public{
14        copy = _copy;
15        price = _price;
16        contractOwner = msg.sender;
17    }
18
19    function getOwner(uint _number)public view returns(address[] memory){
20        return lottery[_number].lotteryOwner;
21    }
22
23    function howManyLeft(uint _number)public view returns(uint){
24        require(_number<1000000,"we do not have the number");
25        return copy-lottery[_number].sellCount;
26    }
27 }
```

Publish on Swarm

ABI Bytecode

call to LotteryShop.getOwner

call to LotteryShop.getOwner

call [call] from:0xca35b7d915458ef540ade6068dfe2f44e8fa733c to:LotteryShop.getOwner(uint256) data:0xc41...00000

call to LotteryShop.getOwner

>

2. click here to copy abi

The ABI button is circled in red.



Application Binary Interface

```
[ { "constant": true, "inputs": [], "name": "copy", "outputs": [ { "internalType": "uint256", "name": "", "type": "uint256" } ], "payable": false, "stateMutability": "view", "type": "function" }, { "constant": true, "inputs": [ { "internalType": "uint256", "name": "_number", "type": "uint256" } ], "name": "howManyLeft", "outputs": [ { "internalType": "uint256", "name": "", "type": "uint256" } ], "payable": false, "stateMutability": "view", "type": "function" }, { "constant": true, "inputs": [], "name": "price", "outputs": [ { "internalType": "uint256", "name": "", "type": "uint256" } ], "payable": false, "stateMutability": "view", "type": "function" }, { "constant": true, "inputs": [ { "internalType": "uint256", "name": "", "type": "uint256" } ], "name": "lottery", "outputs": [ { "internalType": "uint256", "name": "sellCount", "type": "uint256" } ], "payable": false, "stateMutability": "view", "type": "function" }, { "constant": true, "inputs": [ { "internalType": "uint256", "name": "_number", "type": "uint256" } ], "name": "getOwner", "outputs": [ { "internalType": "address[]", "name": "", "type": "address[]" } ], "payable": false, "stateMutability": "view", "type": "function" }, { "constant": true, "inputs": [], "name": "contractOwner", "outputs": [ { "internalType": "address", "name": "", "type": "address" } ], "payable": false, "stateMutability": "view", "type": "function" }, { "constant": false, "inputs": [ { "internalType": "uint256", "name": "_number", "type": "uint256" } ], "name": "setNumber", "outputs": [ { "internalType": "uint256", "name": "newNumber", "type": "uint256" } ], "payable": true, "stateMutability": "nonpayable", "type": "function" } ]
```



Application Binary Interface

```
<head>
  <!-- ... -->
<script>
  //web3 done!
  const abi = <paste your abi here>
  const contractAddress = ""
  const lotteryContract = new web3.eth.Contract(abi,contractAddress)
</script>
</head>
```



contract address

1.click here

The screenshot shows the Remix Ethereum IDE interface. On the left, there's a sidebar with various icons and buttons. The main area has tabs for 'Home' and 'LotterySeller.sol'. The code editor contains the following Solidity code:

```
pragma solidity ^0.5.11;
contract LotteryShop{
    struct Lottery{
        address[] lotteryOwner;
        uint sellCount ;
    }
    Lottery[1000000] public lottery;
    uint public copy;
    uint public price;
    address public contractOwner;
    constructor(uint _copy,uint _price) public{
        copy = _copy;
        price = _price;
        contractOwner = msg.sender;
    }
    function getOwner(uint _number)public view returns(address[] memory){
        return lottery[_number].lotteryOwner;
    }
    function howManyLeft(uint _number)public view returns(uint){
        require(_number<1000000,"we do not have the number");
        return copy-lottery[_number].sellCount;
    }
}
```

The deployment section shows the following details:

- Environment: Injected Web3 (Kovan (42) network)
- Account: 0x00a...597c2 (with a red circle around the arrow icon)
- Gas limit: 3000000
- Value: 0 wei
- Contract: LotteryShop
- Gas limit: 88,70
- Deploy button highlighted in orange
- At Address: Load contract from Address

Below the deployment area, the 'Transactions recorded' section shows one transaction:

- call to LotteryShop.getOwner
- creation of LotteryShop pending...
- [vm] from:0xca3...a733c to:LotteryShop.(constructor) value:0 wei data:0x608...00046 logs:0 hash:0x449...04767 creation of LotteryShop pending...

A link to Etherscan is provided: <https://kovan.etherscan.io/tx/0x537cb475bc4a0a191e9233c9a68fe295347dd1e7b0d725db77a004b721ecb3e4>



contract address

Remix - Ethereum IDE

remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

DEPLOY & RUN TRANSACTIONS

Environment: Injected Web3 (Kovan (42) network)

Account: 0x00a...597c2 (Gas limit: 3000000, Value: 0 wei)

Contract: LotteryShop (Deploy: 88,70 or At Address)

LotterySeller.sol

```
pragma solidity ^0.5.11;
contract LotteryShop{
    struct Lottery{
        address[] lotteryOwner;
        uint sellCount ;
    }
    Lottery[1000000] public lottery;
    uint public copy;
    uint public price;
    address public contractOwner;
    constructor(uint _copy,uint _price) public{
        copy = _copy;
        price = _price;
        contractOwner = msg.sender;
    }
    function getOwner(uint _number)public view returns(address[] memory){
        return lottery[_number].lotteryOwner;
    }
    function howManyLeft(uint _number)public view returns(uint){
        require(_number<1000000,"we do not have the number");
        return copy-lottery[_number].sellCount;
    }
}
```

Transactions recorded: 1

Deployed Contracts: LotteryShop at 0xa78...d61b7 (block 0)

call to LotteryShop.getOwner

creation of LotteryShop pending...

[vm] from:0xca3...a733c to:LotteryShop.(constructor) value:0 wei data:0x608...00046 logs:0 hash:0x449...04767

creation of LotteryShop pending...

<https://kovan.etherscan.io/tx/0x537cb475bc4a0a191e9233c9a68fe295347dd1e7b0d725db77a004b721ecb3e4>

2. click here to copy contract address



```
<head>
  <!-- ... -->
<script>
  //web3 done!
  const abi = <paste your abi here>
  const contractAddress = "0x5d4e134aef2f88a2dedd35757c5fb9e72dfbfe97"
  const lotteryContract = new web3.eth.Contract(abi,contractAddress)
</script>
</head>
```



DOMContentLoaded

```
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- ... -->
    <script>
        //web3 and contract done!
        window.addEventListener('DOMContentLoaded', (event) => {
            //DOM fully loaded and parsed
        })
    </script>
</head>
<body>
    <!--UI done!-->
</body>
</html>
https://web3js.readthedocs.io/en/v1.2.1/getting-started.html
```



```
window.addEventListener('DOMContentLoaded', async function(event) {  
    let contractOwner = await lotteryContract.methods.contractOwner().call()  
    document.getElementById("showContractOwner").innerHTML =  
        "ContractOwner : "+ contractOwner  
})
```



```
window.addEventListener('DOMContentLoaded', async function(event) {  
    let contractOwner = await lotteryContract.methods.contractOwner().call()  
    document.getElementById("showContractOwner").innerHTML =  
        "ContractOwner : "+ contractOwner  
  
    let copy = await lotteryContract.methods.copy().call()  
    document.getElementById("showCopy").innerHTML = "Copy : "+copy  
})
```



```
window.addEventListener('DOMContentLoaded', async function(event) {  
    let contractOwner = await lotteryContract.methods.contractOwner().call()  
    document.getElementById("showContractOwner").innerHTML =  
        "ContractOwner : "+ contractOwner  
  
    let copy = await lotteryContract.methods.copy().call()  
    document.getElementById("showCopy").innerHTML = "Copy : "+copy  
  
    let price = await lotteryContract.methods.price().call()  
    document.getElementById("showPrice").innerHTML = "Price : "+price  
})
```



HTML

The screenshot shows a list of functions for a lottery contract. Each function is represented by a blue button with its name and a parameter type. To the left of each button is a green circular icon with a white checkmark. The functions listed are:

- buyLottery (uint256 _number)
- contractOw... (highlighted with a red box)
- copy (highlighted with a red box)
- getOwner (uint256 _number)
- howManyLeft (uint256 _number)
- lottery (uint256)
- price (highlighted with a red box)

JavaScript



HTML

The screenshot shows a list of functions for a lottery contract. Each function is accompanied by a green checkmark icon. The first two functions, 'buyLottery' and 'getOwner', are highlighted with a red border.

✓	buyLottery	uint256 _number
✓	contractOw...	
✓	copy	
✓	getOwner	uint256 _number
✓	howManyLeft	uint256 _number
	lottery	uint256
✓	price	

JavaScript



```
<head>
<script>
//web3 and contract
window.addEventListener('DOMContentLoaded', async function(event){
    //done
})
async function getOwner(){}
async function howManyLeft(){}
async function buyLottery(){}
</script>
</head>
```



```
async function getOwner(){  
    let _number = document.getElementById("lottery").value  
    let owner = await lotteryContract.methods.getOwner(_number).call()  
    document.getElementById("showGetOwner").innerHTML = "Owner : "+owner  
}
```



HTML

The image shows a user interface for interacting with a smart contract. On the left, there are green checkmarks next to each method name. The methods listed are: buyLottery, uint256 _number; contractOw..., uint256; copy, uint256; getOwner, uint256 _number; howManyLeft, uint256 _number; lottery, uint256; and price, uint256. Two methods are highlighted with red boxes: 'buyLottery' and 'howManyLeft'. Each highlighted method has a dropdown arrow icon to its right.

Method	Type
buyLottery	uint256 _number
contractOw...	uint256
copy	uint256
getOwner	uint256 _number
howManyLeft	uint256 _number
lottery	uint256
price	uint256

JavaScript



```
async function howManyLeft(){
    let _number = document.getElementById("lottery").value
    let left = await lotteryContract.methods.howManyLeft(_number).call()
    document.getElementById("showHowManyLeft").innerHTML = "left : "+left
}
```



HTML

The image shows a user interface for interacting with a smart contract. On the left, there is a vertical list of green circular icons with white checkmarks. To the right of each icon is a method name and its return type. The methods listed from top to bottom are: **buyLottery** (return type: uint256 _number), **contractOw...**, **copy**, **getOwner** (return type: uint256 _number), **howManyLeft** (return type: uint256 _number), **lottery** (return type: uint256), and **price**. Two specific methods are highlighted with red boxes: **buyLottery** and **getOwner**.

Method	Return Type
buyLottery	uint256 _number
contractOw...	
copy	
getOwner	uint256 _number
howManyLeft	uint256 _number
lottery	uint256
price	

JavaScript



```
async function buyLottery(){
    let _number = document.getElementById("lottery").value
    let price = await lotteryContract.methods.price().call()
    let wallet = await web3.eth.getAccounts()
    lotteryContract.methods.buyLottery(_number).send({from: wallet[0], value: price})
    .on('receipt', function(receipt){
        console.log('receipt')
        document.getElementById("showBuyLottery").innerHTML = "คุณได้ซื้อสลากหมายเลข" + _number +
            " เรียบร้อยแล้ว"
    })
    .on('error', function(error){
        console.log(error)
        document.getElementById("showBuyLottery").innerHTML = error
    })
}
```



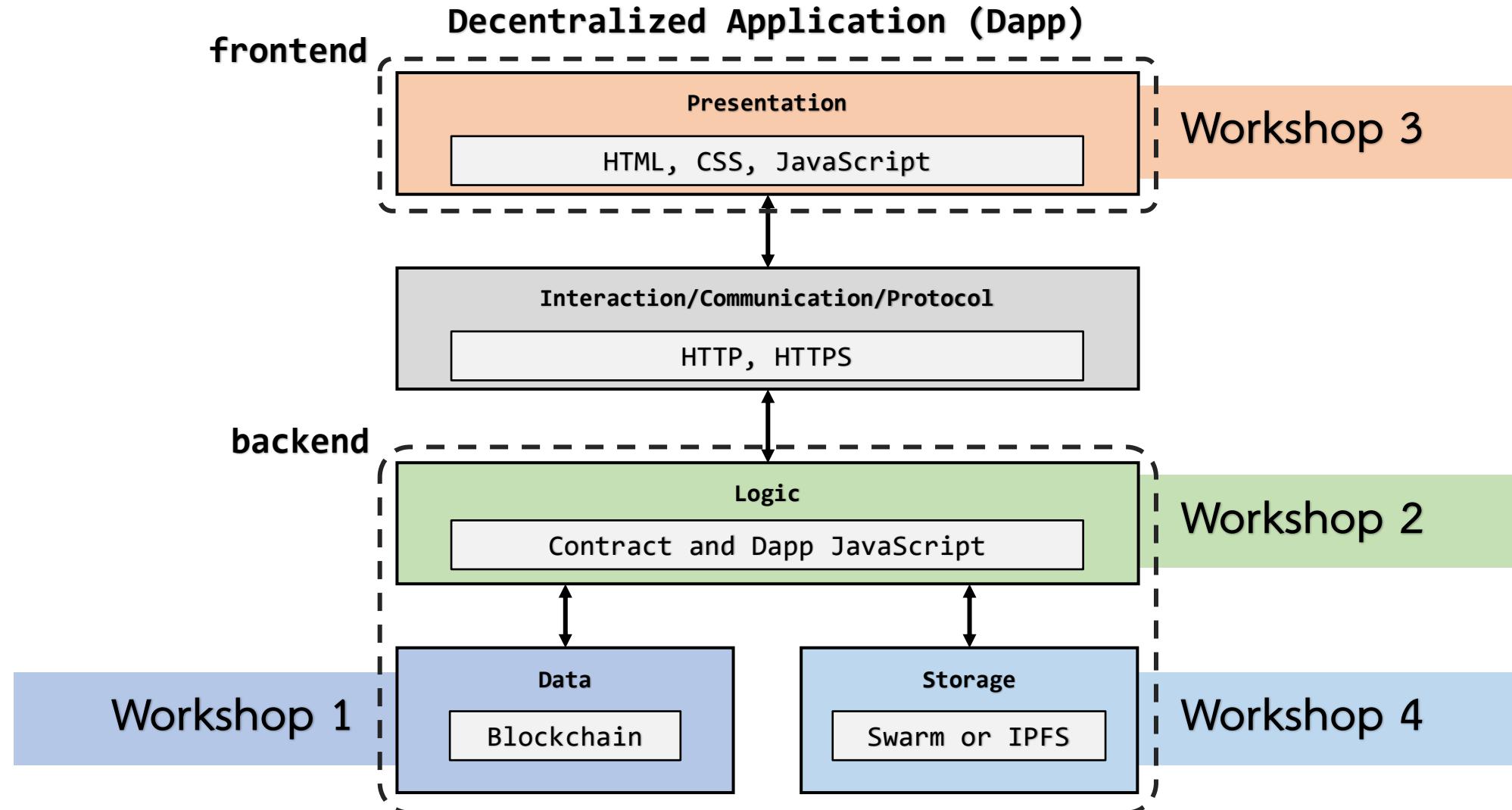
HTML

The screenshot shows a list of functions for a lottery contract. Each function has a green checkmark icon to its left. The functions are:

- buyLottery (orange button, highlighted with a red box)
- contractOwner
- copy
- getOwner (highlighted with a red box)
- howManyLeft (highlighted with a red box)
- lottery
- price

Each function is followed by a text input field labeled "uint256 _number". A dropdown arrow icon is located to the right of each input field.

JavaScript





Workshop 4 Storage

workshop นี้ทำอะไร

1. จากที่เราตะลุยมา 3 Workshops เราพบว่าจุดอ่อนของ smart contract คือมันเก็บได้แต่ตัวแปร ดังนั้นใน workshop นี้ เราจะมาเก็บไฟล์กัน โดยมีจะไขวธีได้แก่
 - 1.1 IPFS ที่ใช้เก็บไฟล์ประเทคโนโลยีที่เป็นกระแสอยู่พักนึงนั่นเอง
 - 1.2 แปลงไฟล์ให้เป็น data (byte) และเก็บในตัวแปร bytes ของ smart contract



A peer-to-peer hypermedia protocol
to make the web faster, safer, and more open.



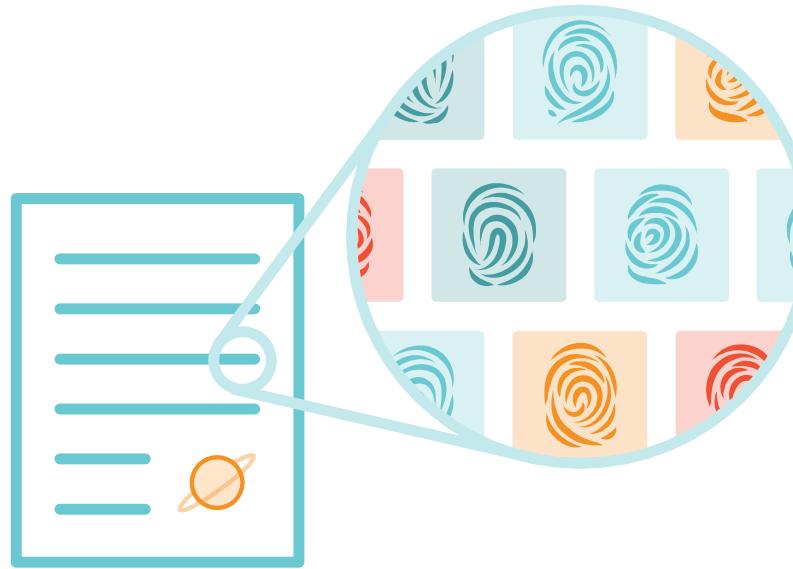
What is IPFS?



IPFS is a distributed system for
storing and accessing files, websites, applications, and data.



How IPFS works?



Each file and all of the **blocks within it** are given a **unique fingerprint** called a **cryptographic hash**.



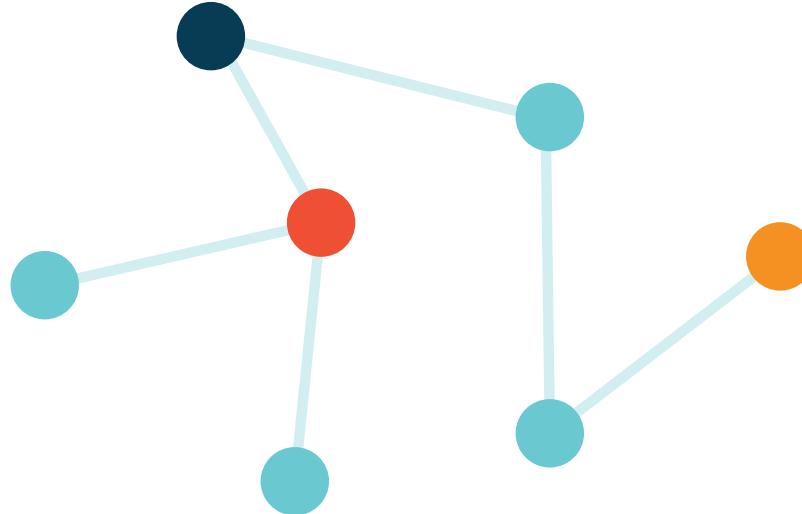
How IPFS works?



IPFS removes duplications across the network.



How IPFS works?



Each **network node** stores only content it is interested in, and some indexing information that helps figure out who is storing what.



How IPFS works?



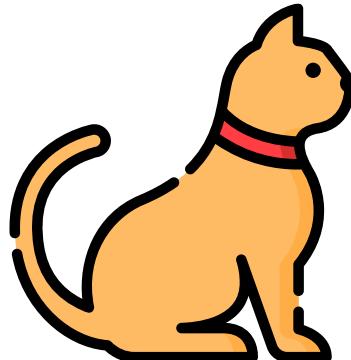
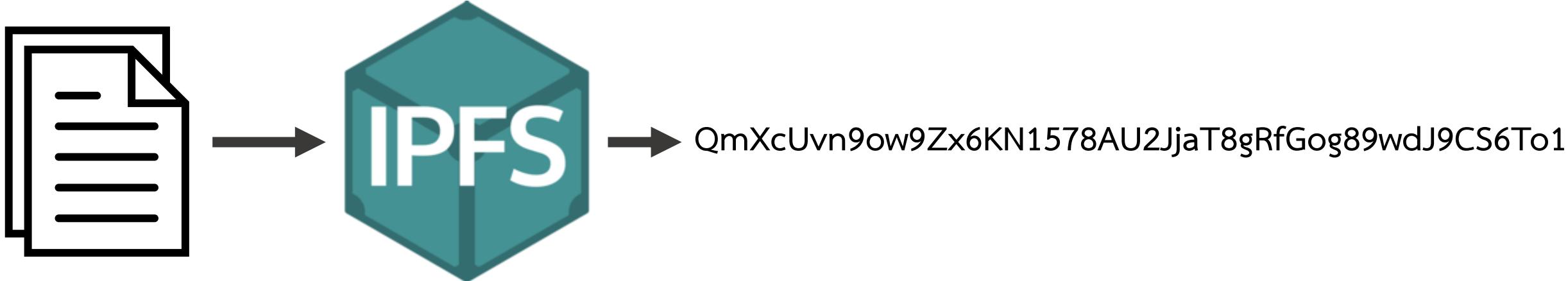
When **looking up files**, you're asking the network to find nodes storing the content behind a unique hash.



How IPFS works?



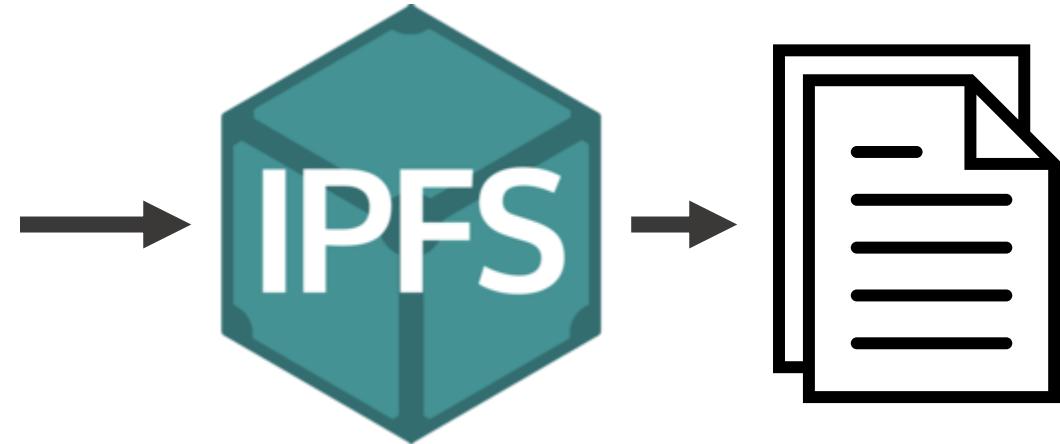
Every file can be found by **human-readable names** using a decentralized naming system called **IPNS**.



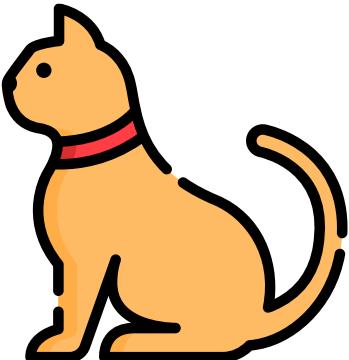
สมชาย



QmXcUvn9ow9Zx6KN1578AU2JjaT8gRfGog89wdJ9CS6To1



สมหมาย





สร้างไฟล์ ipfs.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>
  hello

</body>
</html>
```



Document x +

localhost:8080/ipfs.html

JS IPFS - Add data to IPFS from the browser

No file chosen

found in ipfs:
[ipfs hash]
[ipfs url]



```
<!DOCTYPE html>
<html lang="en">
<head>
  <!--meta and title done!-->
</head>
<body>
  <h1>JS IPFS - Add data to IPFS from the browser</h1>
  <input type="file" id="source"></input>
  <button onclick="store()">add to ipfs</button>
  <div>found in ipfs:</div>
  <input type="text" id="hash" value="[ipfs hash]">
  <button onclick="show()">show</button><br>
  <a id="url">[ipfs url]</a>
  <iframe id="content" width="100%" height="400">[ipfs content]</iframe>
</body>
</html>
```



Document X +

localhost:8080/ipfs.html

JS IPFS - Add data to IPFS from the browser

Choose File pic.png add to ipfs

found in ipfs:
QmXcUvn9ow9Zx6KN1578A show
<https://ipfs.io/ipfs/QmXcUvn9ow9Zx6KN1578AU2JjaT8gRfGog89wdJ9CS6To1>





2. require('ipfs')

1. Run command

C:\frontend>**browserify -r web3 -r ipfs -r buffer > bundle.js**

```
JS bundle.js  x
JS bundle.js
1  require=(function(){function r(e,n,t){function o(i,f){if(!n[i]){if(!e[i]){var c="function"==typeof require&&require;if(!f&&c)return c(i,!0);i
2  var asn1 = exports;
3
4  asn1.bignum = require('bn.js');
5
6  asn1.define = require('./asn1/api').define;
7  asn1.base = require('./asn1/base');
8  asn1.constants = require('./asn1/constants');
9  asn1.decoders = require('./asn1/decoders');
10  asn1.encoders = require('./asn1/encoders');
11
12  },{"/./asn1/api":2,"./asn1/base":4,"./asn1/constants":8,"./asn1/decoders":10,"./asn1/encoders":13,"bn.js":16}],2:[function(require,module,exports)
13  var asn1 = require('../asn1');
14  var inherits = require('inherits');
15
16  var api = exports;
17
18  api.define = function define(name, body) {

```



2. require('ipfs')

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!--meta and title done!-->
  <script src="bundle.js"></script>
  <script>
    const IPFS = require('ipfs')
    const Buffer = require('buffer') </script>
</head>
<body>
  <!--UI done!-->
</body>
</html>
```

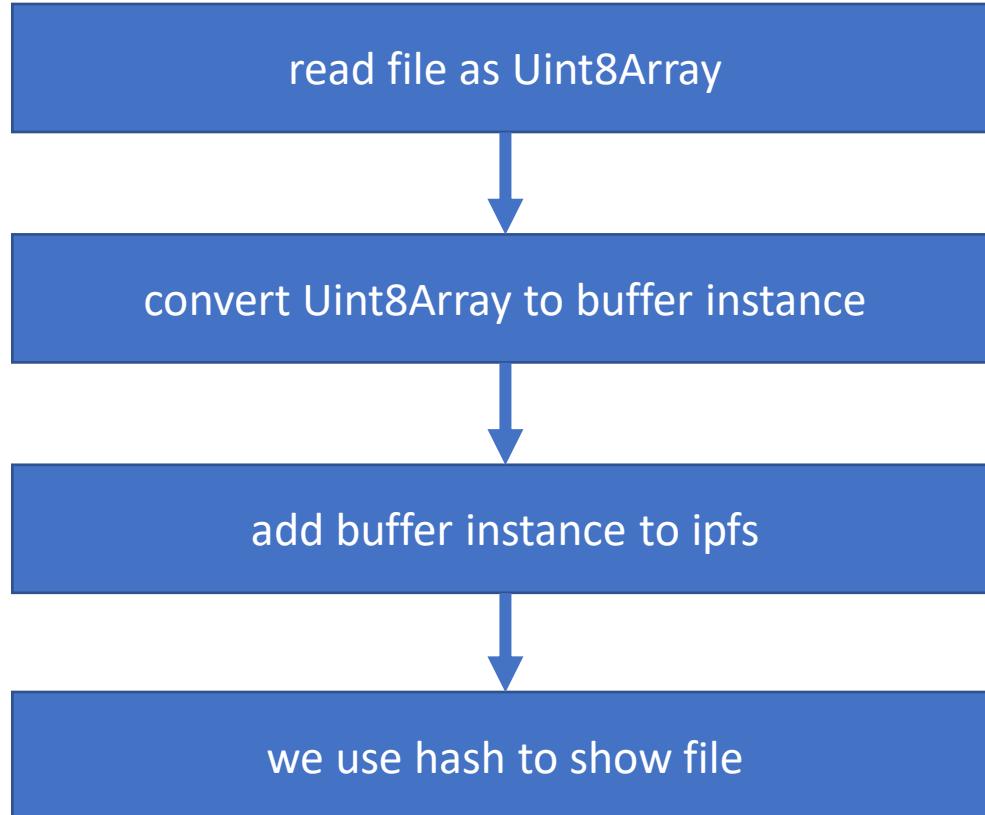


3. node instance

```
<!DOCTYPE html>
<html lang="en">
<head>
    <!--meta and title done!-->
    <!--require done!-->
    <script>
        let node
        IPFS.create().then((res)=>{
            node =res
        })
        console.log('IPFS node is ready')
    </script>
</head>
<body>
    <!--UI done!-->
</body>
</html>
```

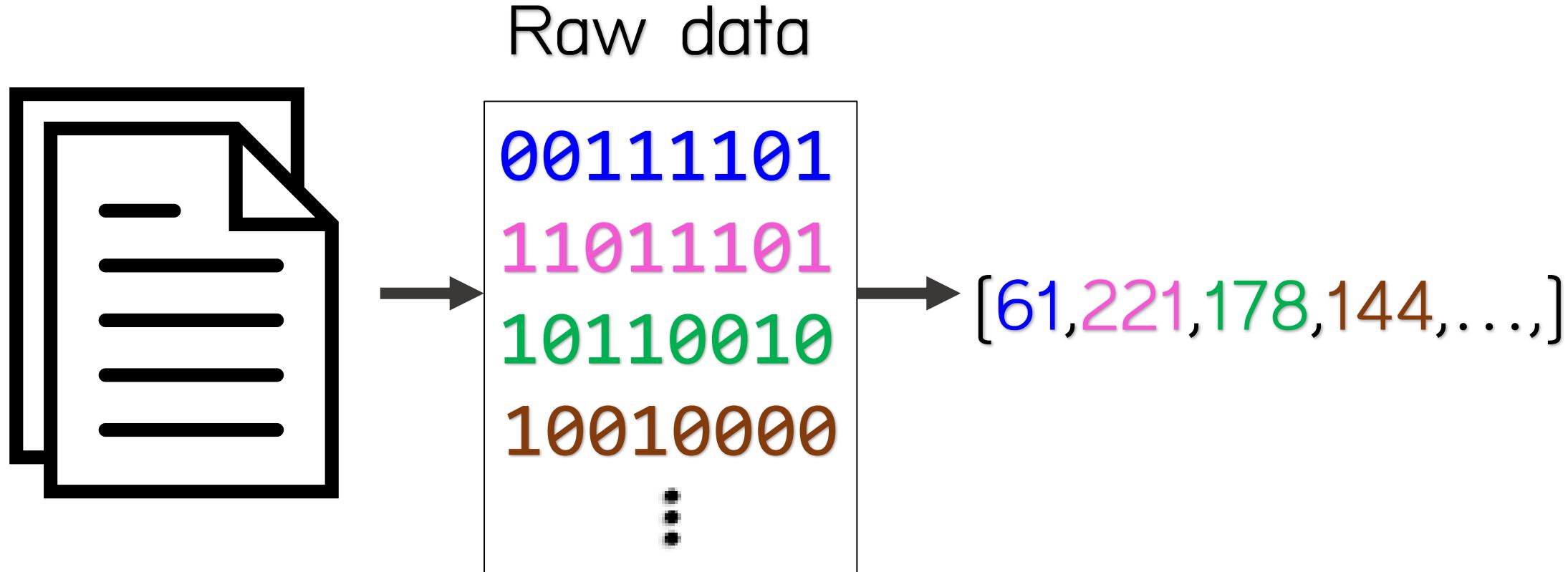


4. store()





4.1 file to Uint8Array





4.1 file to Uint8Array

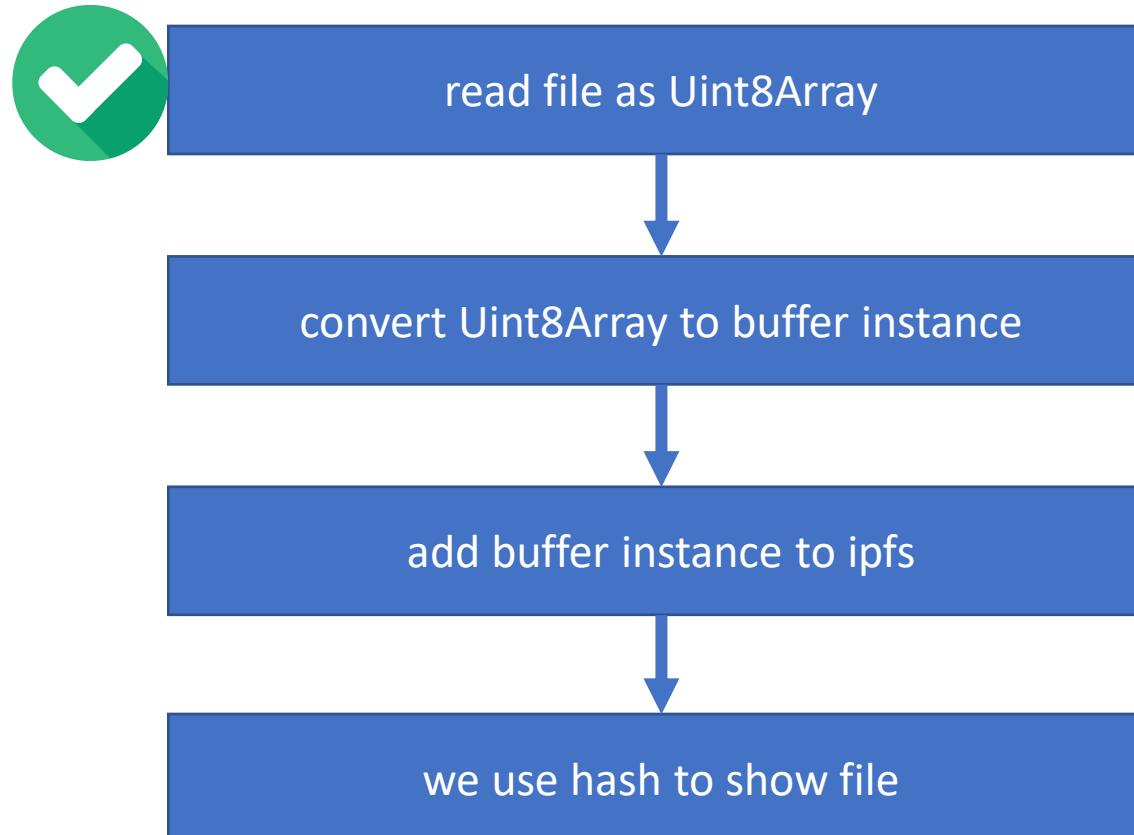
```
async function store(){
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()

        fileReader.onloadend = async function(fileLoadedEvent) {
            let uint8Array = fileLoadedEvent.target.result
            //uint8Array is stored in variable uint8Array
        };

        await fileReader.readAsArrayBuffer(fileToLoad)//read data as uint8Array
    }
}
```



4.1 file to Uint8Array



```
read file= ipfs.html:28
▼ArrayBuffer(266289) {}
  ► [[Int8Array]]: Int8Array(266289) [37, 80, 68, 70, 45, 49, 4...
  ► [[Uint8Array]]: Uint8Array(266289) [37, 80, 68, 70, 45, 49, ...
  byteLength: ...
  ► __proto__: ArrayBuffer
```



4.2 buffer instance

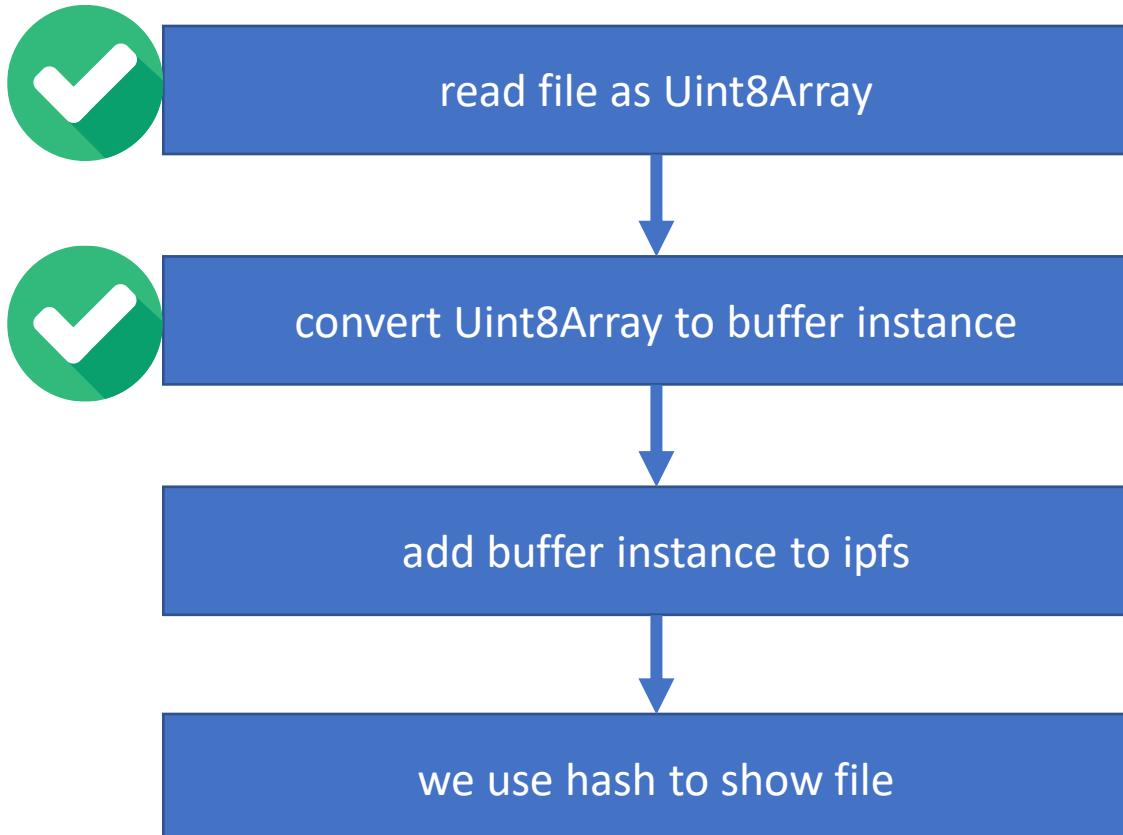
```
async function store(){
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()

        fileReader.onloadend = async function(fileLoadedEvent) {
            let uint8Array = fileLoadedEvent.target.result;
            let buff = Buffer.Buffer(fileLoadedEvent.target.result)
            //convert uint8Array buffer to a buffer instance
        };

        await fileReader.readAsArrayBuffer(fileToLoad)//read data as uint8Array
    }
}
```



4.2 buffer instance



```
buff= ipfs.html:30
Uint8Array(266289) [37, 80, 68, 70, 45, 49, 46, 55, 13, 10, 37
, 181, 181, 181, 181, 13, 10, 49, 32, 48, 32, 111, 98, 106, 13
, 10, 60, 60, 47, 84, 121, 112, 101, 47, 67, 97, 116, 97, 108,
111, 103, 47, 80, 97, 103, 101, 115, 32, 50, 32, 48, 32, 82, 4
7, 76, 97, 110, 103, 40, 101, 110, 45, 85, 83, 41, 32, 47, 83,
116, 114, 117, 99, 116, 84, 114, 101, 101, 82, 111, 111, 116,
32, 51, 48, 32, 48, 32, 82, 47, 77, 97, 114, 107, 73, 110, 102
, 111, 60, 60, 47, ...]
```



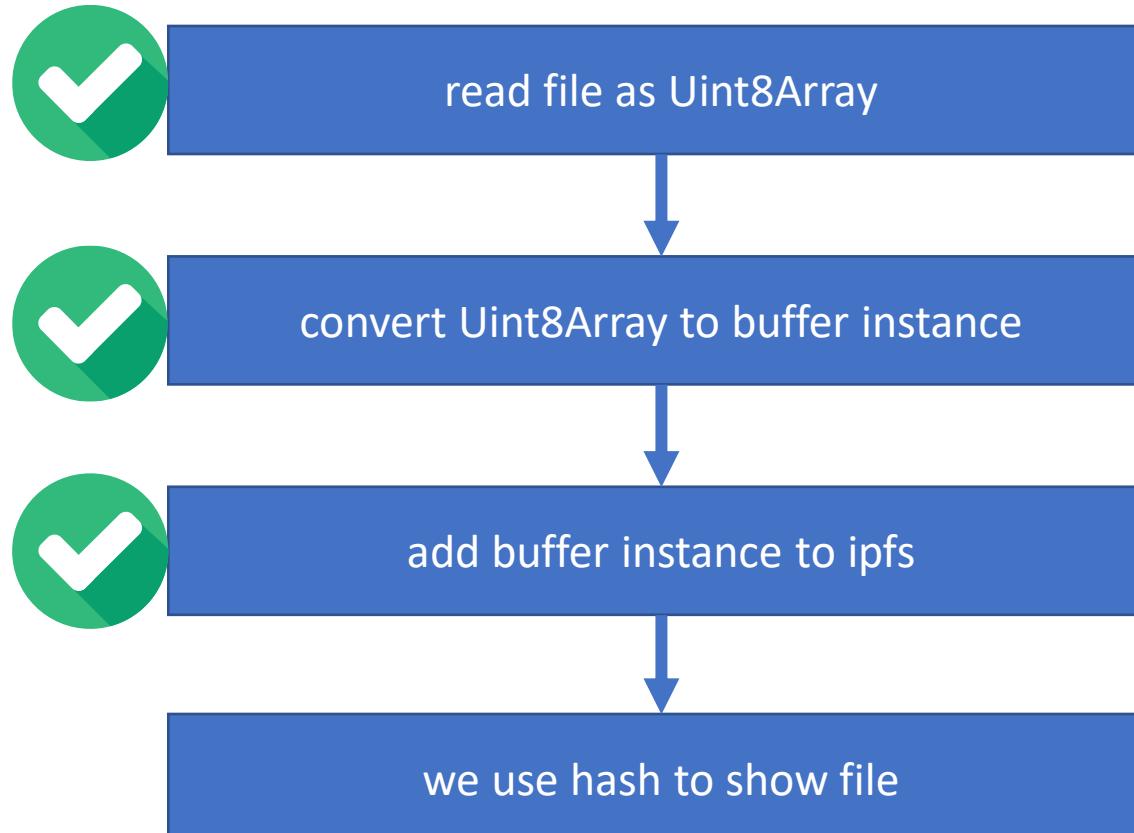
4.3 add to ipfs

```
async function store(){
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()

        fileReader.onloadend = async function(fileLoadedEvent) {
            let uint8Array = fileLoadedEvent.target.result
            let buff = Buffer.Buffer(fileLoadedEvent.target.result)
            const res = await node.add(buff)//add buffer to ipfs
        };
        await fileReader.readAsArrayBuffer(fileToLoad)//read data as uint8Array
    }
}
```



4.3 add to ipfs



```
res= ipfs.html:32
▼ [...]
  ▼ 0:
    hash: "QmYgXPUvjwls...ZGc"
    path: "QmYgXPUvjwls...ZGc"
    size: 266416
    ► __proto__: Object
    length: 1
    ► __proto__: Array(0)
```

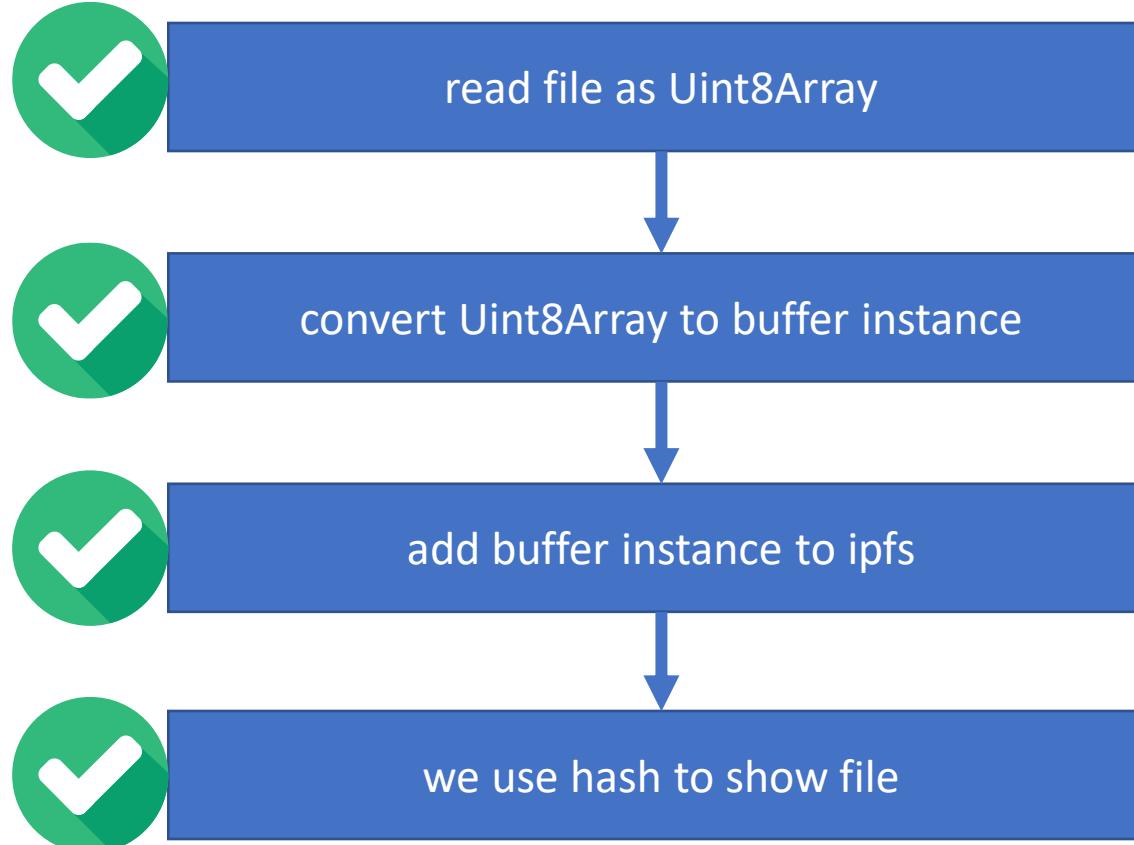


4.3 add to ipfs

```
let selectedFile = document.getElementById("source").files  
if (selectedFile.length > 0) {  
    let fileToLoad = selectedFile[0]  
    let fileReader = new FileReader()  
  
    fileReader.onloadend = async function(fileLoadedEvent) {  
        let uint8Array = fileLoadedEvent.target.result  
        let buff = Buffer.Buffer(fileLoadedEvent.target.result)  
        const res = await node.add(buff)  
        hash = res[0].hash  
        let url = 'https://ipfs.io/ipfs/' + hash  
        console.log('url =', url)  
        document.getElementById('hash').value = hash  
        document.getElementById('url').innerHTML = url  
        document.getElementById('url').href = url  
        document.getElementById('content').src = url  
  
        let data = await node.cat(hash)  
        console.log("data = ", data)  
    }  
    await fileReader.readAsArrayBuffer(fileToLoad); //read data as uint8Array  
}
```



4. store()





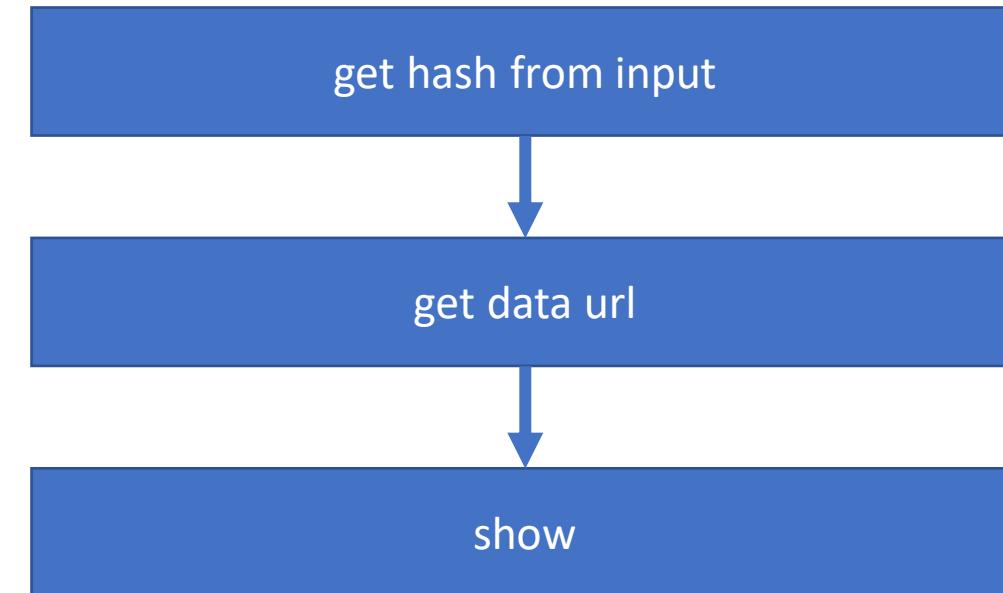
4. store()

```
async function store(){
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()
        fileReader.onloadend = async function(fileLoadedEvent) {
            let uint8Array = fileLoadedEvent.target.result
            let buff = Buffer.Buffer(fileLoadedEvent.target.result)
            const res = await node.add(buff)
            hash = res[0].hash
            let url = 'https://ipfs.io/ipfs/' + hash
            console.log('url = ', url)
            document.getElementById('hash').value = hash
            document.getElementById('url').innerHTML = url
            document.getElementById('url').href = url
            document.getElementById('content').src = url

            let data = await node.cat(hash)
            console.log("data = ", data)
        }
        await fileReader.readAsArrayBuffer(fileToLoad); //read data as uint8Array
    }
}
```



5. show()





5. show()

```
async function show(){
    let hash = document.getElementById('hash').value
    let data = await node.cat(hash)
    let url = 'https://ipfs.io/ipfs/' + hash
    document.getElementById('url').innerHTML = url
    document.getElementById('url').href = url
    document.getElementById('content').src = url
    data = await node.cat(hash)
    console.log("data = ", data)
}
```



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title><!DOCTYPE html>
    <script src="bundle.js"></script>
<script>
    const IPFS = require('ipfs')
    const Buffer = require('buffer')
    let node
    IPFS.create({ repo: String(Math.random() + Date.now()) }).then((res)=>{
        node = res;
    })
    console.log('IPFS node is ready')

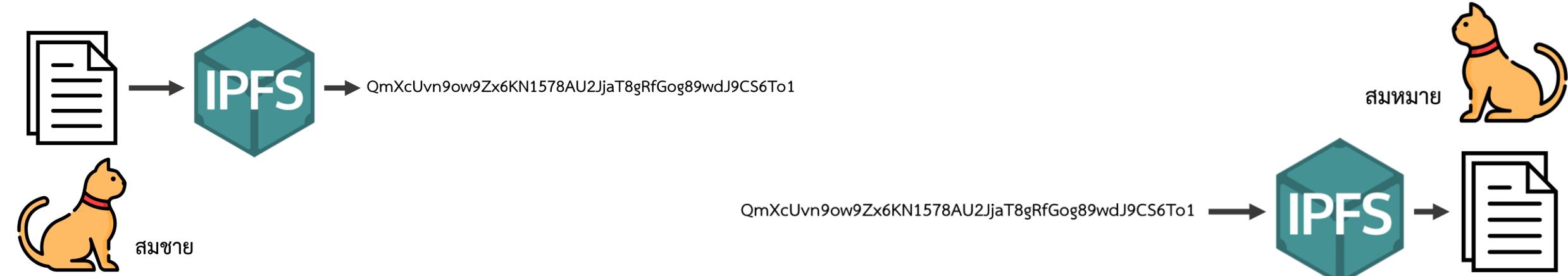
    async function store(){
        let selectedFile = document.getElementById("source").files;
        if (selectedFile.length > 0) {
            let fileToLoad = selectedFile[0];
            let fileReader = new FileReader();
            fileReader.onloadend = async function(fileLoadedEvent) {
                buff = Buffer.Buffer(fileLoadedEvent.target.result)//convert array buffer to buffer
                const res = await node.add(buff)//add buffer to ipfs
                hash = res[0].hash
                let url = 'https://ipfs.io/ipfs/' + hash
                console.log('url = ',url);
                document.getElementById('hash').value = hash
                document.getElementById('url').innerHTML = url
                document.getElementById('url').href = url
                document.getElementById('content').src = url

                data = await node.cat(hash)
                console.log("data = ",data)
            };
            await fileReader.readAsArrayBuffer(fileToLoad);//read data as array buffer
        }
    }

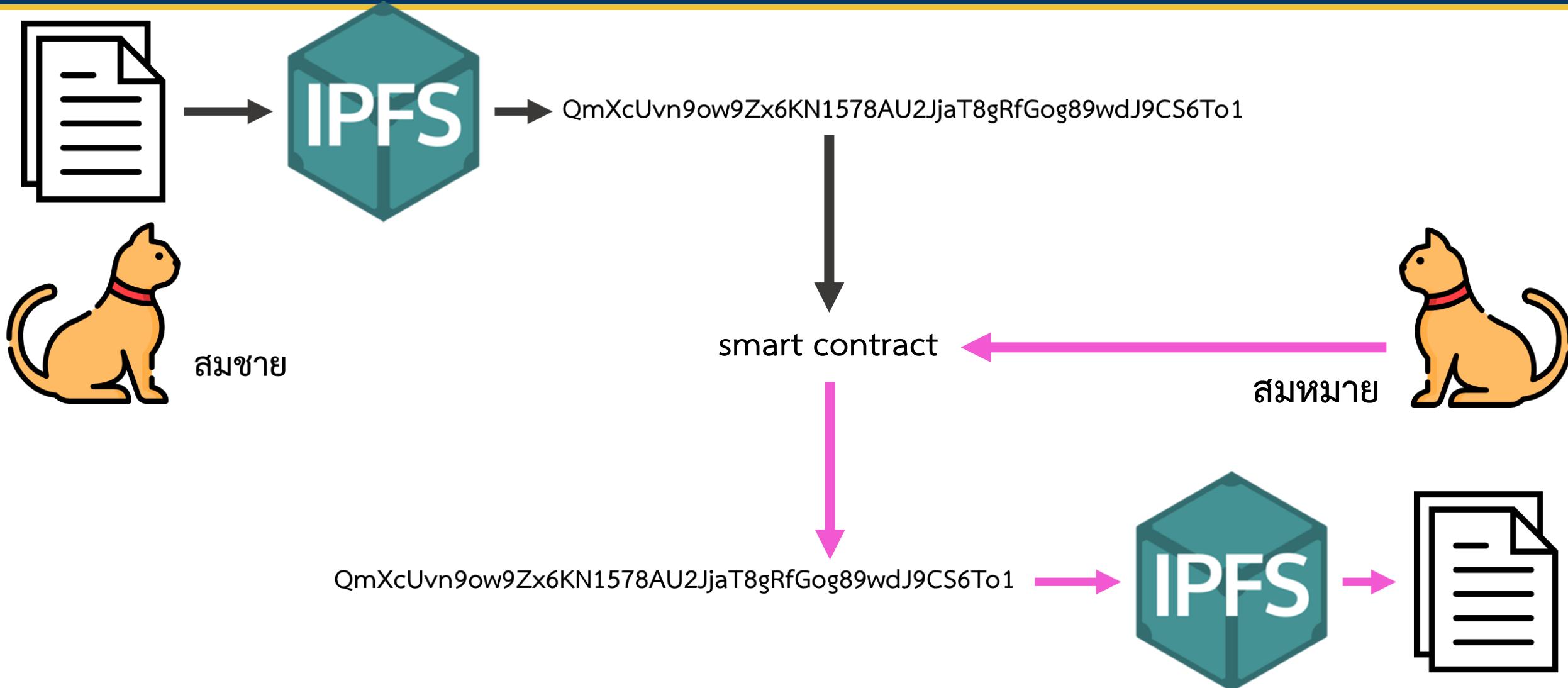
    async function show(){
        let hash = document.getElementById('hash').value
        let data = await node.cat(hash)
        let url = 'https://ipfs.io/ipfs/' + hash
        document.getElementById('url').innerHTML = url
        document.getElementById('url').href = url
        document.getElementById('content').src = url
        data = await node.cat(hash)
        console.log("data = ",data)
    }
</script>
</head>
<body>
    <h1>JS IPFS - Add data to IPFS from the browser</h1>
    <input type="file" id="source"></input>
    <button onclick="store()">Add to ipfs</button>
    <div>Found in ipfs:</div>
    <input type="text" id="hash" value="ipfs hash">
    <button onclick="show()">Show</button><br>
    <a id="url">[ipfs url]</a>
    <iframe id="content" width="100%" height="400">[ipfs content]</iframe>
</body>
</html>
```



ipfs + blockchain!??



Where is blockchain include!??





set contract()

```
pragma solidity ^0.5.11;

contract hashStorage {
    string public ipfsHash;
    address public owner;

    constructor()public{
        owner = msg.sender;
    }

    function addHash(string memory _ipfsHash)public {
        require(msg.sender == owner);
        ipfsHash=_ipfsHash;
    }
}
```



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title><!DOCTYPE html>
<script src="bundle.js"></script>
<script>
  const IPFS = require('ipfs')
  const Buffer = require('buffer')
  const Web3 = require('web3')
  ethereum.enable().then(x=>{console.log(x)})
  const web3 = new Web3(Web3.givenProvider || new Web3.providers.HttpProvider('http://localhost:8545'))

  const contractABI=[...]

  const contractAddress="0x5e050bde777039933d646ddb6ce1de104f6565dc"
  const ipfsContract = new web3.eth.Contract(contractABI,contractAddress)
```

```
let node
IPFS.create({ repo: String(Math.random() + Date.now()) }).then((res)=>{
node =res;
})
console.log('TPFS node is ready')
```



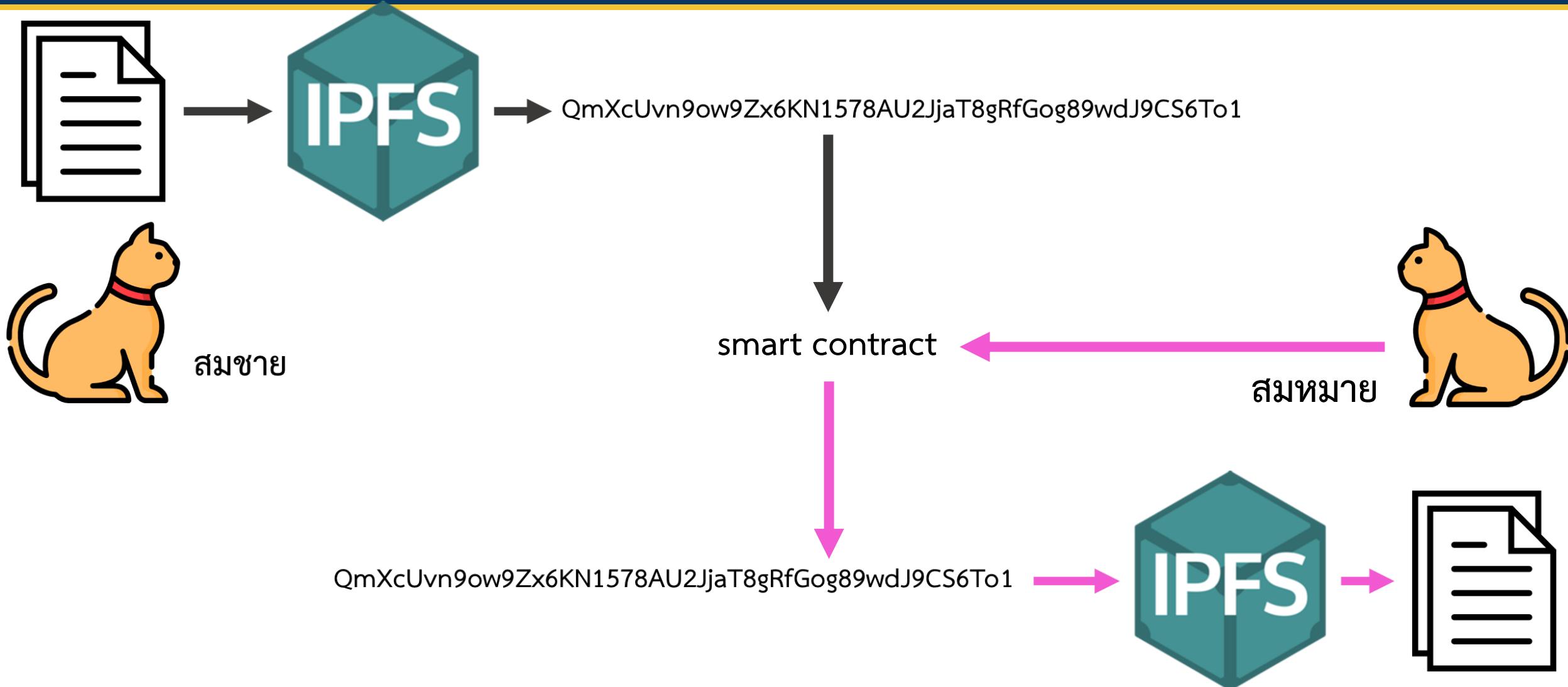
```
async function store(){
let selectedFile = document.getElementById("source").files;
if (selectedFile.length > 0) {
    let fileToLoad = selectedFile[0];
    let fileReader = new FileReader();
    fileReader.onloadend = async function(fileLoadedEvent) {
        buff = Buffer.Buffer(fileLoadedEvent.target.result)//convert array buffer to buffer
        const res = await node.add(buff)//add buffer to ipfs
        hash = res[0].hash
        let url = 'https://ipfs.io/ipfs/'+hash
        console.log('url =',url);
        document.getElementById('hash').value = hash
        document.getElementById('url').innerHTML = url
        document.getElementById('url').href = url
        document.getElementById('content').src= url

        data = await node.cat(hash)
        console.log("data = ",data)

        let wallet = await web3.eth.getAccounts()
        let result = await ipfsContract.methods.addHash(hash).send({from: wallet[0]}) 
        console.log(result)
    };
    await fileReader.readAsArrayBuffer(fileToLoad);//read data as array buffer
}
}
```



```
async function show_eth(){
    let hash = await ipfsContract.methods.ipfsHash().call();
    document.getElementById('hash').value = hash
    let data = await node.cat(hash)
    let url = 'https://ipfs.io/ipfs/' + hash
    document.getElementById('url').innerHTML = url
    document.getElementById('url').href = url
    document.getElementById('content').src= url
}
```



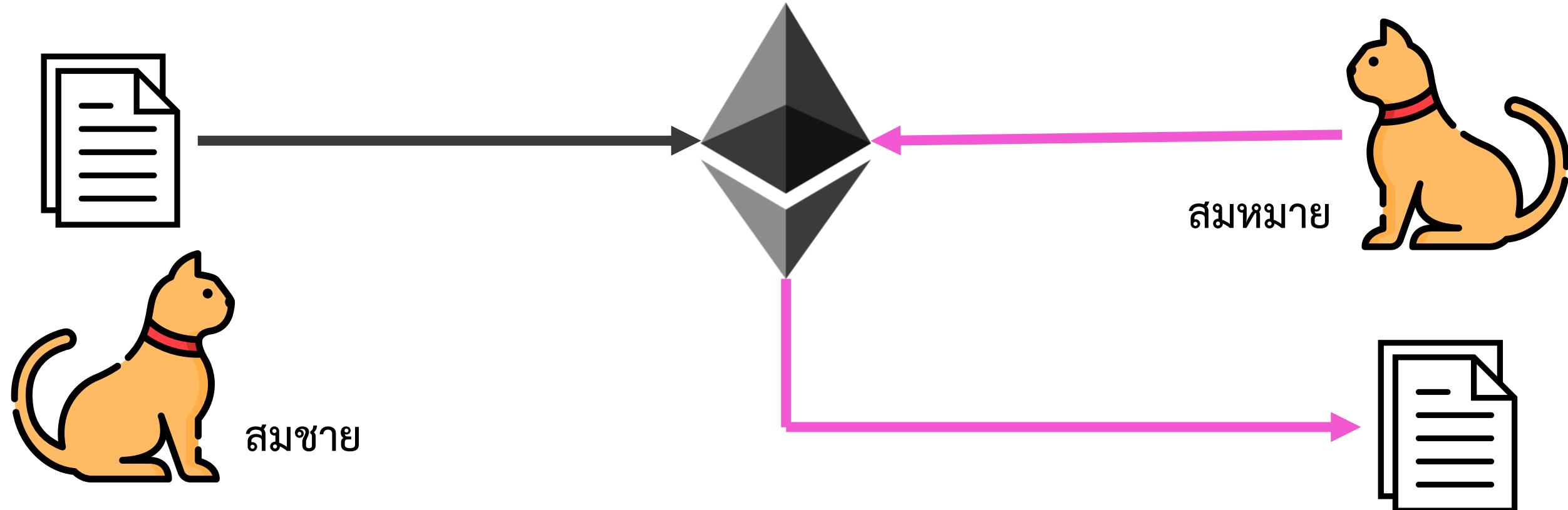


IT Management
Faculty of Engineering



ประเทคโนโลยี

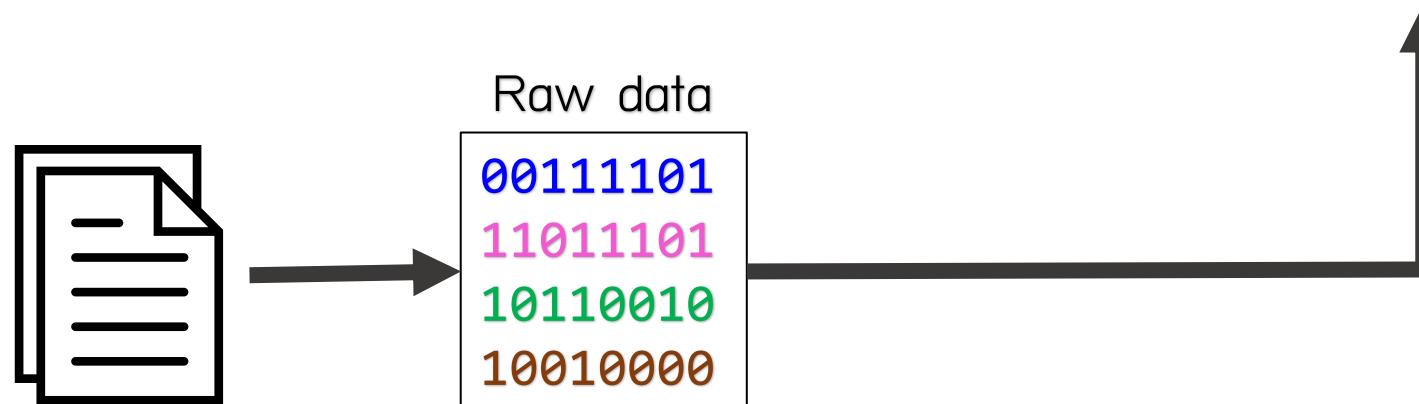
QmXFf5QLQi3DhbwyrgR7qH8tHAZcGyWtWtsBy6ijj4B7aL





`data:[<mediatype>][;base64],<data>`

`data:application/pdf;base64,3ddb290...`





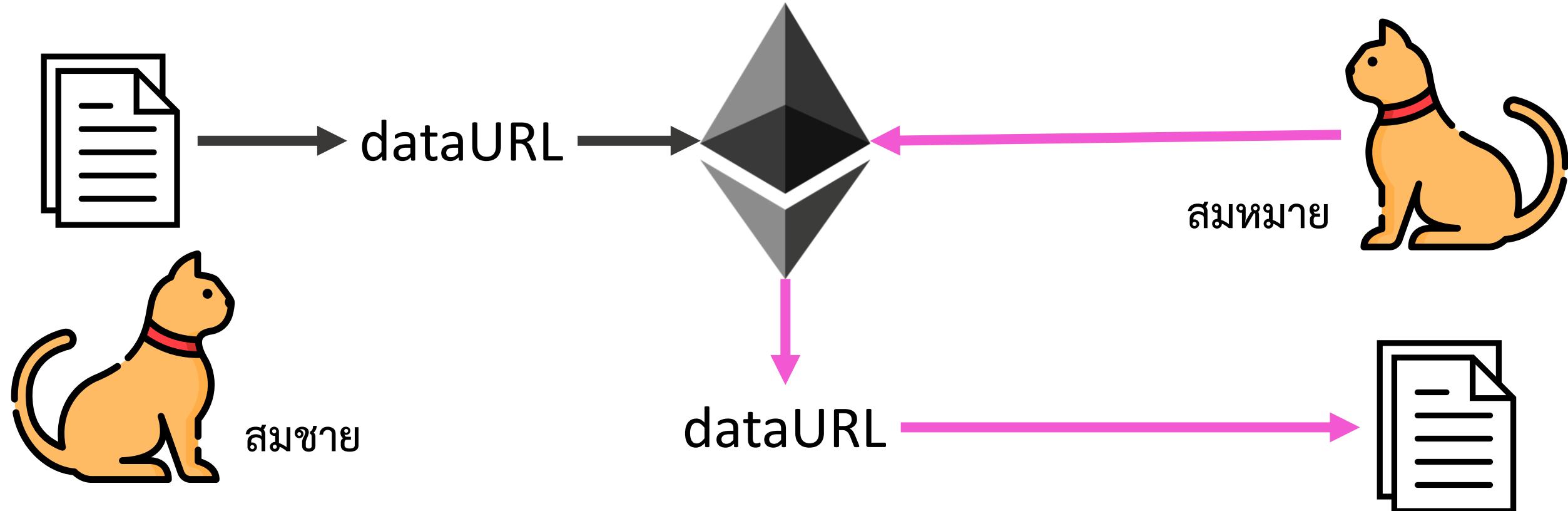
data:text/plain;base64,<data>

data:image/png;base64,<data>

data:video/mp4;base64,<data>

More Info ...

https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types





```
pragma solidity ^0.5.11;
contract Storage{
    string public dataURL;

    function set(string memory _dataURL)public{
        dataURL = _dataURL;
    }
}
```



c615932526c5a6d646f6157707a6448563264336835656f4f45685961486949
6d4b6b704f556c5a61586d4a6d616f714f6b7061616e714b6d7173724f30746
26133754c6d367773504578636248794d6e4b3074505531646258324e6e6134
654c6a354f586d352b6a7036764879382f5431397666342b66722f784141664
151414441514542415145424151454241414141414141414151494442415547
4277674a4367762f78414331455141434151494542414d45427755454241414
2416e6341415149444551514649544547456b46524232467845794979675167
55517047687363454a497a4e53384256696374454b466951303453587846786
75a4769596e4b436b714e5459334f446b3651305246526b6449535570545646
56575631685a576d4cc15a375a0e014/bc/105555231010c643465587143673
45340066f6549695971536b3553566c70 [Show more \(1.4 MB\)](#) [Copy](#)

{code: -32603, message: "Error: Error: [ethjs-rpc] rpc error with payload {...}": "eth_sendRawTransaction"} Error: oversized data"} 1

code: -32603

message: "Error: Error: [ethjs-rpc] rpc error with payload {

► __proto__: Object



```
pragma solidity ^0.5.11;
contract Storage{
    bytes public dataURL;
    bytes public a;

    function set(string memory _dataURL)public{
        a = bytes(_dataURL);
        for(uint i=0;i<a.length;i++){
            dataURL.push(a[i]);
        }
    }

    function clear()public{
        dataURL = "";
    }
}
```



Document x +

localhost:8080/eth.html

ETH - Add data to Ethereum from the browser

No file chosen

index
 show
[data url]

[Large empty rectangular input field]



```
<!DOCTYPE html>
<html lang="en">
<head>
  <!--meta and title done!-->
</head>
<body>
  <h1>ETH - Add data to Ethereum from the browser</h1>
  <input type="file" id="source"></input>
  <button onclick="store()">add to ETH</button>
  <div>index</div>
  <input type="number" id="index" value="">
  <button onclick="show()">show</button><br>
  <a id="url">[data url]</a>
  <iframe id="content" width="100%" height="400">[data content]</iframe>
</body>
</html>
```



2. connect to contract

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!--meta and title done!-->
  <script src="bundle.js"></script>
  <script>
    const Web3 = require('web3')
    //ethereum.enable().then(x=>{console.log(x)})
    const web3 = new Web3(Web3.givenProvider)

    const contractABI = [ ... ]
    const contractAddress="0xbd2627f94b55c42525da8374814287f13b63ce56"
    const byteContract = new web3.eth.Contract(contractABI,contractAddress)
  </script>
</head>
<body>
  <!--UI done!-->
</body>
</html>
```



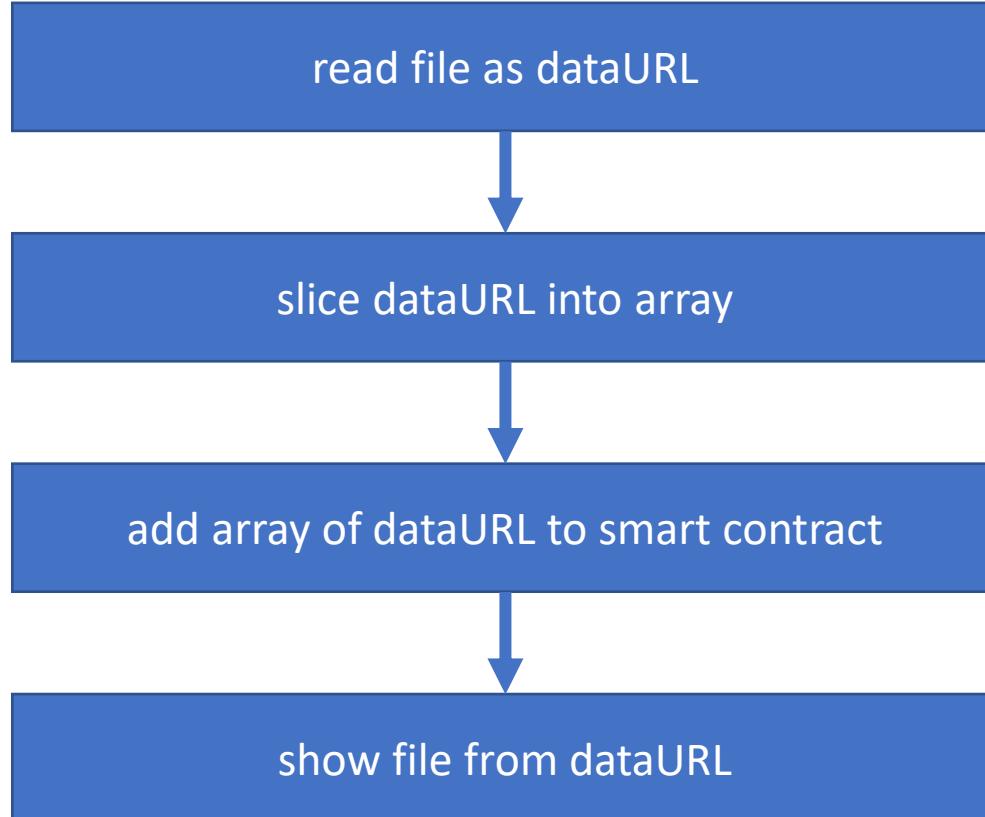
2. connect to contract

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!--meta and title done!-->
  <script src="bundle.js"></script>
  <script>
    const Web3 = require('web3')
    //ethereum.enable().then(x=>{console.log(x)})
    const web3 = new Web3(Web3.givenProvider)

    const contractABI = [ ... ]
    const contractAddress="0xbd2627f94b55c42525da8374814287f13b63ce56"
    const byteContract = new web3.eth.Contract(contractABI,contractAddress)
  </script>
</head>
<body>
  <!--UI done!-->
</body>
</html>
```



3. store()





3.1 file to dataURL

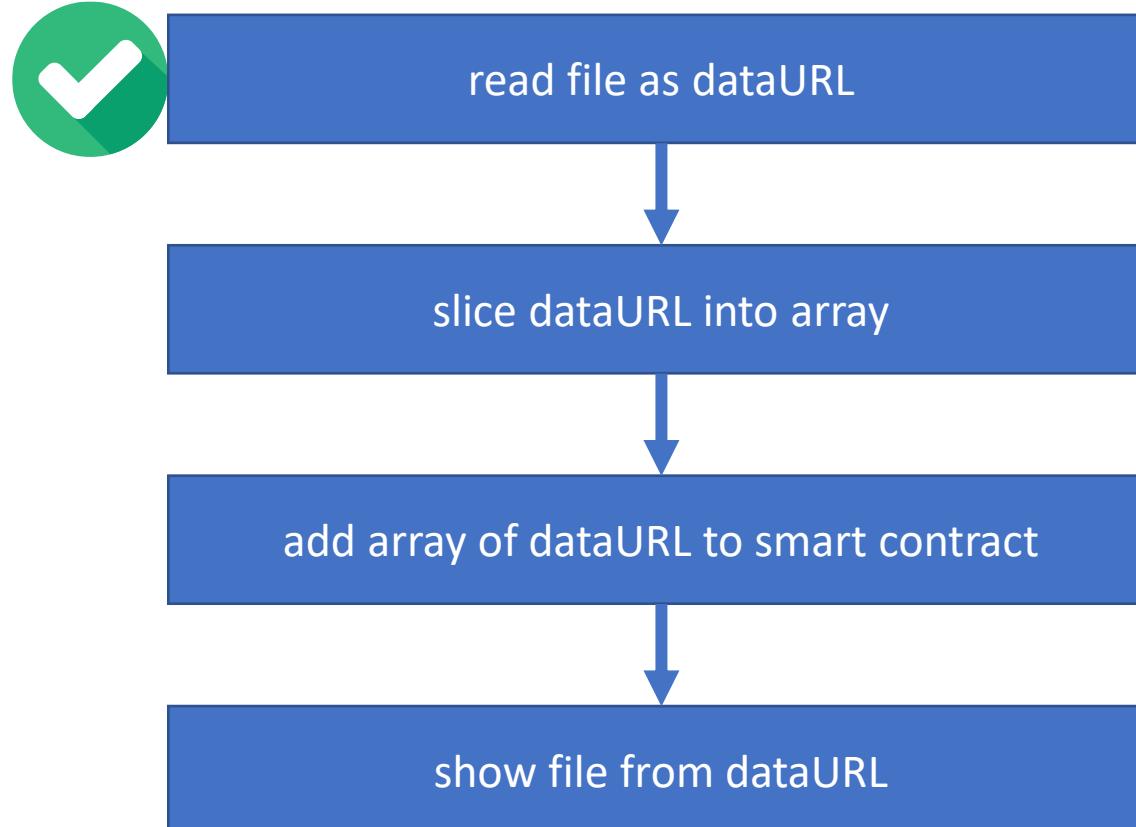
```
async function store(){
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()

        fileReader.onloadend = async function(fileLoadedEvent) {
            let dataURL = fileLoadedEvent.target.result
            // dataURL is stored in variable dataURL
        };

        fileReader.readAsDataURL(fileToLoad)//read data as dataURL
    }
}
```



3.1 file to dataURL



dataURL= data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAXwAAADPCAIAAB5rP3yAAAAAXNSR...VEByg10kAp0QFKiQ5QSsAUqID1BIdoJToAKVEByj0+/f/AI0GLsLHE0+wAAAAAE1FTkSuQmCC



3.2 slice into array

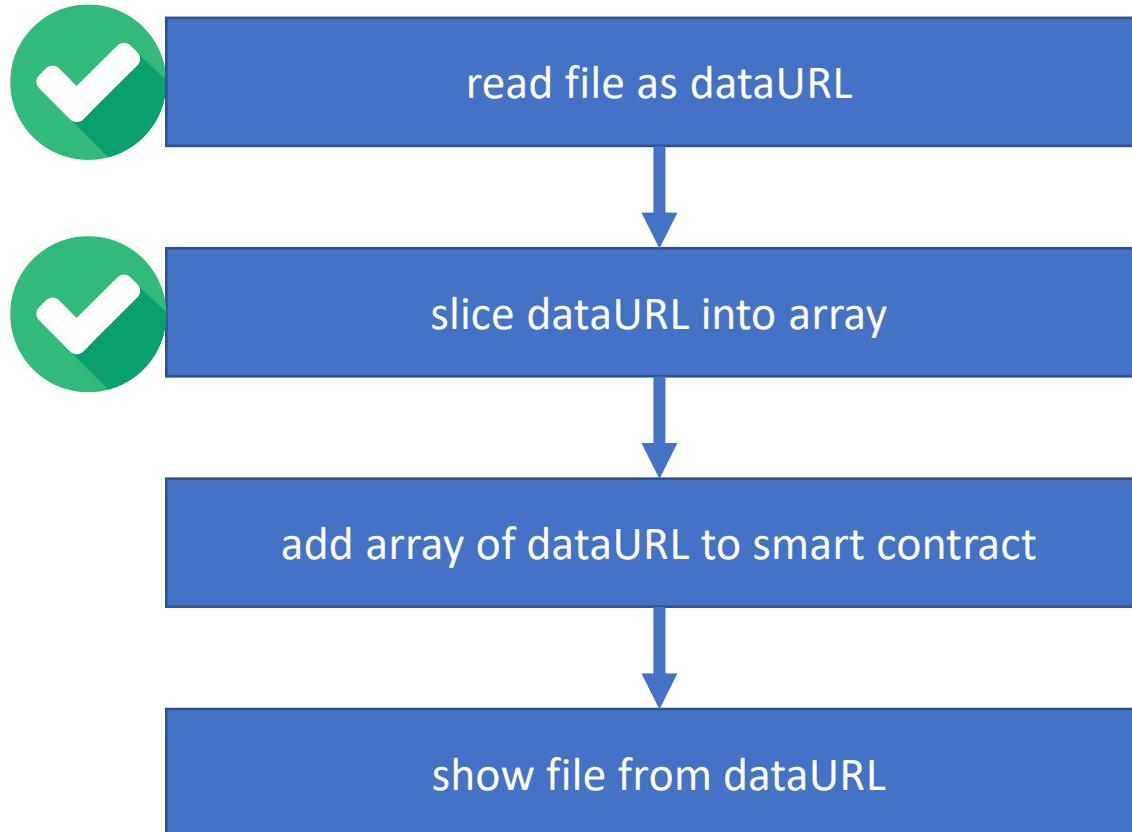
```
async function store(){
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()

        fileReader.onloadend = async function(fileLoadedEvent) {
            let dataURL = fileLoadedEvent.target.result
            let dataURLSlice = dataURL.match(/.{1,500}/g)
            // each elements contain 500 characters
            for(let i=0;i<dataURLSlice.length;i++){
                // TODO
            }
        };

        fileReader.readAsDataURL(fileToLoad)
    }
}
```



3.2 file to dataURL



```
dataURLSlice=
  ▼ 0: "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAXwAAADPCAIAAAAB5rP3yAAA...
  1: "CnRKed/IbbfMs3NzeFP39+8+ZNDuvv4u+cnZ1dX1/nI+E+rTfm79+/j1fq1evctUBGN9...
  2: "q67fLksEqic+c8muJQoEV0wp2TLPNtgbEZ5Kqtik0rvvJ4iqHPLk8OaPfRWT2kCopDjs7...
  3: "h20Z1XkaI+uarQe0qQy0vIEWxjDA6saKhddJY6oTOMpw65vIQcwcZjuL6+dmBFQ72iE4c...
  4: "zyNOYh5PKa8sGiQ1ddXpr0Ik9jHkIurykfLDp01eW16SzyNOYh5PKa8sGiQ1ddXpr0Ik9...
  5: "kJuWoD15eXR0dH+eX+L+Z8pzs+sZ01e1JpVVQv/xIHafnoOLa615iTkMsbuzc9W9nxib5...
  6: "SAUqID18IdoJToAKVEByg10kAp0QFKiQ5QSnsAUqID1BIdoJToAKVEByg10kAp0QFKiQ5...
length: 7
__proto__: Array(0)
```



3.3 add to contract

1. `tx` - `Object` : The transaction object as follows:

- `nonce` - `String` : (optional) The nonce to use when signing this transaction. Default will use `web3.eth.getTransactionCount()`.
- `chainId` - `String` : (optional) The chain id to use when signing this transaction. Default will use `web3.eth.net.getId()`.
- `to` - `String` : (optional) The receiver of the transaction, can be empty when deploying a contract.
- `data` - `String` : (optional) The call data of the transaction, can be empty for simple value transfers.
- `value` - `String` : (optional) The value of the transaction in wei.
- `gasPrice` - `String` : (optional) The gas price set by this transaction, if empty, it will use `web3.eth.gasPrice()`
- `gas` - `String` : The gas provided by the transaction.



3.3 add to contract

```
for(let i=0;i<dataURLSlice.length;i++){
    let tx = {
        nonce: ,
        gasPrice: ,
        gas: ,
        to: ,
        value: '',
        data:
    }
}
```



```
for(let i=0;i<dataURLSlice.length;i++){
    let wallet = await web3.eth.getAccounts()
    let latestNonce = await web3.eth.getTransactionCount(wallet[0])
    let tx = {
        nonce: latestNonce,
        gasPrice: ,
        gas: ,
        to: ,
        value: "",
        data:
    }
}
```



3.3.2 gasPrice and gasUsed

```
for(let i=0;i<dataURLSlice.length;i++){
    let wallet = await web3.eth.getAccounts()
    let latestNonce = await web3.eth.getTransactionCount(wallet[0]);
    let estimateGas = await byteContract.methods.set(dataURLSlice[i]).estimateGas({from: wallet[0]});
    let tx = {
        nonce: latestNonce,
        gasPrice: web3.utils.toHex(web3.utils.toWei('1', 'gwei')),
        gas: web3.utils.toHex((estimateGas + 1000)),
        to: ,
        value: "",
        data:
    }
}
```



```
for(let i=0;i<dataURLSlice.length;i++){
  let wallet = await web3.eth.getAccounts()
  let latestNonce = await web3.eth.getTransactionCount(wallet[0]);
  let estimateGas = await byteContract.methods.set(dataURLSlice[i]).estimateGas({from: wallet[0]})
  let tx = {
    nonce: latestNonce,
    gasPrice: web3.utils.toHex(web3.utils.toWei('1', 'gwei')),
    gas: web3.utils.toHex((estimateGas + 1000)),
    to: contractAddress,
    value: "",
    data:
  }
}
```



```
for(let i=0;i<dataURLSlice.length;i++){
    let wallet = await web3.eth.getAccounts()
    let latestNonce = await web3.eth.getTransactionCount(wallet[0]);
    let estimateGas = await byteContract.methods.set(dataURLSlice[i]).estimateGas({from: wallet[0]}) 
    let tx = {
        nonce: latestNonce,
        gasPrice: web3.utils.toHex(web3.utils.toWei('1', 'gwei')),
        gas: web3.utils.toHex((estimateGas + 1000)),
        to: contractAddress,
        value: '',
        data: byteContract.methods.set(dataURLSlice[i]).encodeABI()
    }
}
```



3.3.5 signTransaction

```
for(let i=0;i<dataURLSlice.length;i++){
    let wallet = await web3.eth.getAccounts()
    let latestNonce = await web3.eth.getTransactionCount(wallet[0]);
    let estimateGas = await byteContract.methods.set(dataURLSlice[i]).estimateGas({from: wallet[0]})
    let tx = {
        nonce: latestNonce,
        gasPrice: web3.utils.toHex(web3.utils.toWei('1', 'gwei')),
        gas: web3.utils.toHex((estimateGas + 1000)),
        to: contractAddress,
        value: '',
        data: byteContract.methods.set(dataURLSlice[i]).encodeABI()
    }
    const privateKey = "0x262CA48C689471E77309BFD778628170F03B12C3C745670DEFF9D53B5E40066A"
    let signedTx = await web3.eth.accounts.signTransaction(tx,privateKey)
}
```

eth.html:100

```
{messageHash: "0x33fff2b0eb01f307a7c6f6784db97cee6bb306ee0de4f69bc806556d
2e66b0d1", v: "0x77", r: "0x89e0989ff9409dbb536f43e574fc028d27c036743387e
▶ 59644466f33a0630322", s: "0x6ca0a9b02c141788830a79e71b7fb379953099669424
d7de22d7dd0867d6bb9", rawTransaction: "0xf902ab1b8502540be4008365fa9394aa
e9eedf707340af01..8830a79e71b7fb379953099669424d7de22d7dd0867d6bb9"}
```



3.3.6 sendTransaction

```
for(let i=0;i<dataURLSlice.length;i++){
    let wallet = await web3.eth.getAccounts()
    let latestNonce = await web3.eth.getTransactionCount(wallet[0]);
    let estimateGas = await byteContract.methods.set(dataURLSlice[i]).estimateGas({from: wallet[0]})
    let tx = {
        nonce: latestNonce,
        gasPrice: web3.utils.toHex(web3.utils.toWei('1', 'gwei')),
        gas: web3.utils.toHex((estimateGas + 1000)),
        to: contractAddress,
        value: "",
        data: byteContract.methods.set(dataURLSlice[i]).encodeABI()
    }
    const privateKey = "0x262CA48C689471E77309BFD778628170F03B12C3C745670DEFF9D53B5E40066A";
    let signedTx = await web3.eth.accounts.signTransaction(tx,privateKey)
    let res = await web3.eth.sendSignedTransaction(signedTx.rawTransaction)
    console.log(res)
}

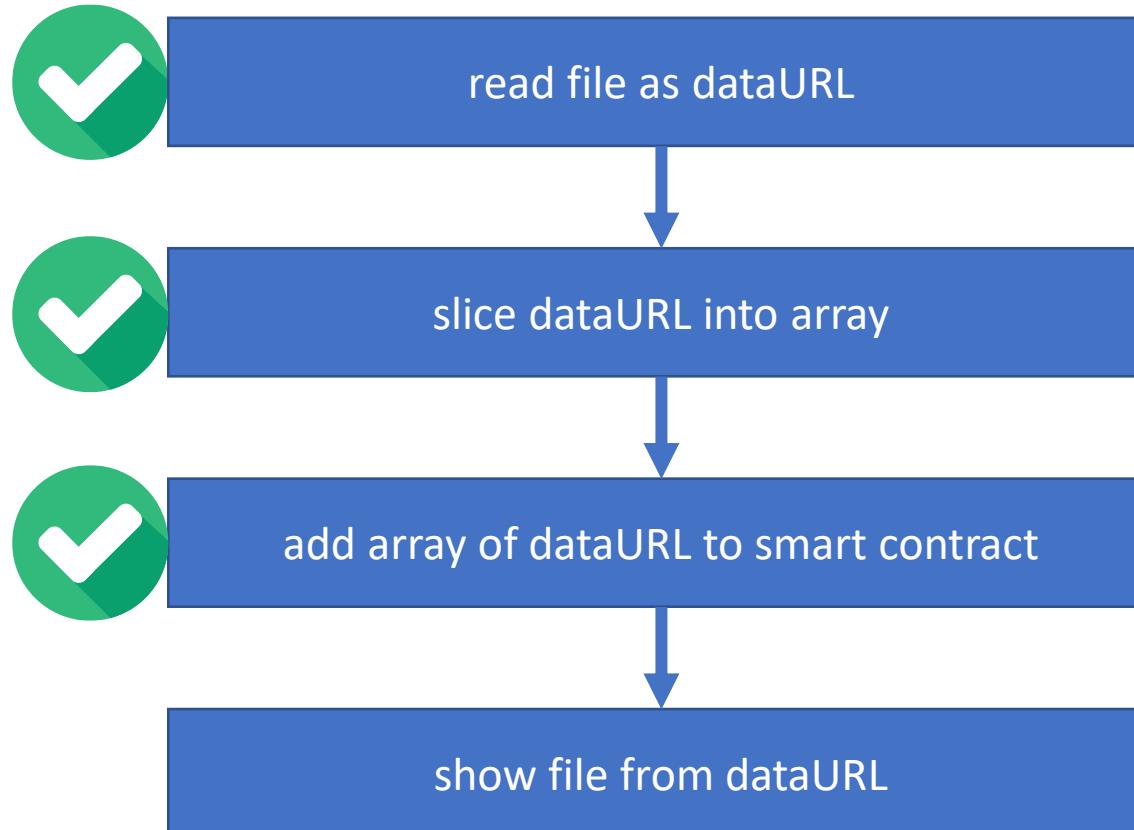
{blockHash: "0x3c2a8f249ec7e135d537642f1ee645648e6ebef2f5f193bc41be124f44
▶ 3f0b0c", blockNumber: 12867071, contractAddress: null, cumulativeGasUsed:
6424605, from: "0x00aa39d30f0d20ff03a22ccfc30b7efbfca597c2", ...}
```

[eth.html:102](#)

{blockHash: "0x3c2a8f249ec7e135d537642f1ee645648e6ebef2f5f193bc41be124f44
▶ 3f0b0c", blockNumber: 12867071, contractAddress: null, cumulativeGasUsed:
6424605, from: "0x00aa39d30f0d20ff03a22ccfc30b7efbfca597c2", ...}



3.3 add to contract



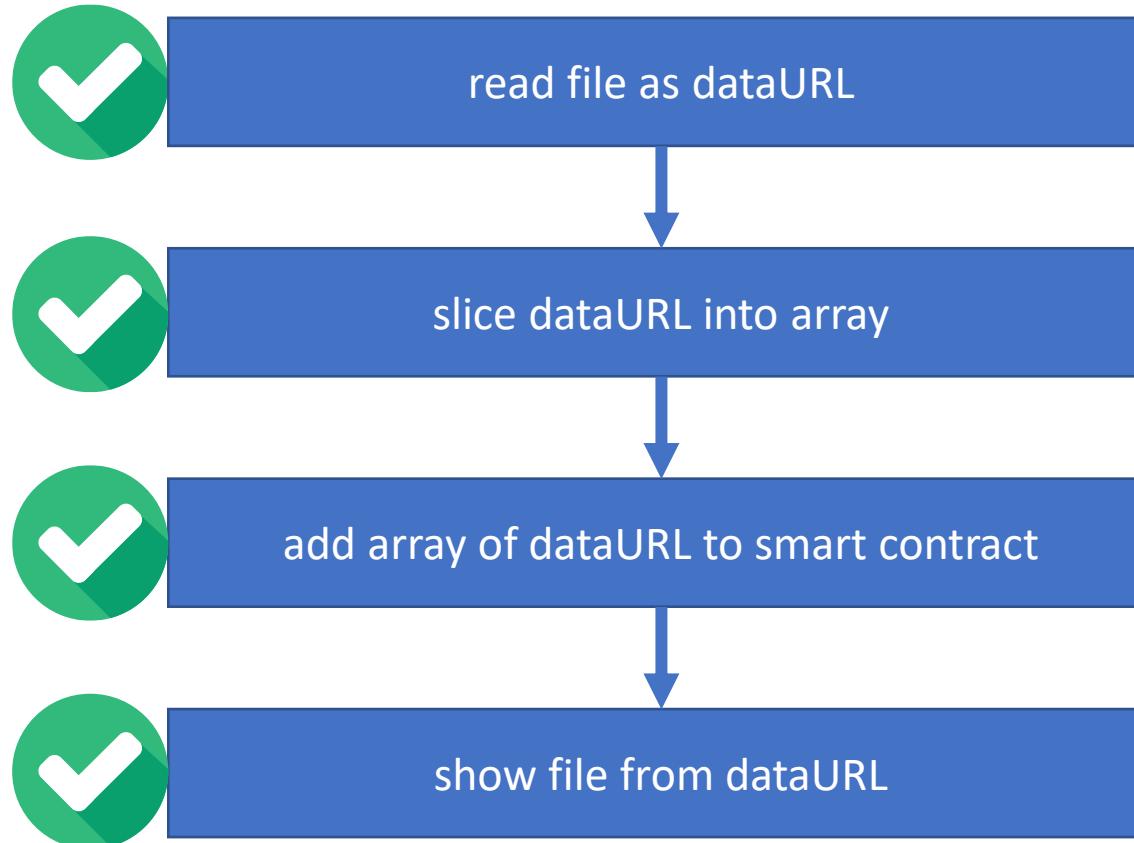


3.4 show data

```
async function store() {
    let selectedFile = document.getElementById("source").files
    if (selectedFile.length > 0) {
        let fileToLoad = selectedFile[0]
        let fileReader = new FileReader()
        fileReader.onload = async function(fileLoadedEvent) {
            //dataURLSlice done!
            for(let i=0;i<dataURLSlice.length;i++){
                //sendRawTransaction done!
            }
            let dataURLFromBlockchain = await byteContract.methods.dataURL().call()
            let hexURL = dataURLFromBlockchain.slice(2,dataURLFromBlockchain.length)// remove 0x
            let url = ""
            for (let i = 0; i < hexURL.length; i += 2) {
                url += String.fromCharCode(parseInt(hexURL.substr(i, 2), 16));
            }
            console.log('url =',url)
            document.getElementById('url').innerHTML = url
            document.getElementById('url').href = url
            document.getElementById('content').src= url
        }
        fileReader.readAsDataURL(fileToLoad)
    }
}
```



3.4 show data



```
dataURLSlice=
  ▼ 0: "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAXwAAADPCAIAAAAB5rP3yAAA...
  1: "CnRKed/IbbfMs3NzeFP39+8+ZNDuvv4u+cnZ1dX1/nI+E+rTfm79+/j1fq1evctUBGN9...
  2: "q67fLksEqic+c8muJQoEV0wp2TLPNtgbEZ5Kqtik0rvvJ4iqHPLk8OaPfRWT2kCopDjs7...
  3: "h20Z1XkaI+uarQe0qQy0vIEWxjDA6saKhddJY6oTOMpw65vIQcwcZjuL6+dmBFQ72iE4c...
  4: "zyNOYh5PKa8sGiQ1ddXpr0Ik9jHkIurykfLDp01eW16SzyNOYh5PKa8sGiQ1ddXpr0Ik9...
  5: "kJuWoD15eXR0dH+eX+L+Z8pzs+sZ01e1JpVVQv/xIHafnoOLa615iTkmbsuzc9W9nxib5...
  6: "SAUqID18IdoJToAKVEBygl0kAp0QFKiQ5QSnsAUqID1BIdoJToAKVEBygl0kAp0QFKiQ5...
length: 7
__proto__: Array(0)
```



4. show data

```
async function show(){
    let dataURLFromBlockchain = await byteContract.methods.dataURL().call()
    let hexURL = dataURLFromBlockchain.slice(2,dataURLFromBlockchain.length)// remove 0x
    let url = ""
    for (let i = 0; i < hexURL.length; i += 2) {
        url += String.fromCharCode(parseInt(hexURL.substr(i, 2), 16));
    }
    console.log('url =',url)
    document.getElementById('url').innerHTML = url
    document.getElementById('url').href = url
    document.getElementById('content').src= url
}
```



4. show data

Document x + localhost:8080/eth.html - X ☆ 🌐 🌱 ⚙ :

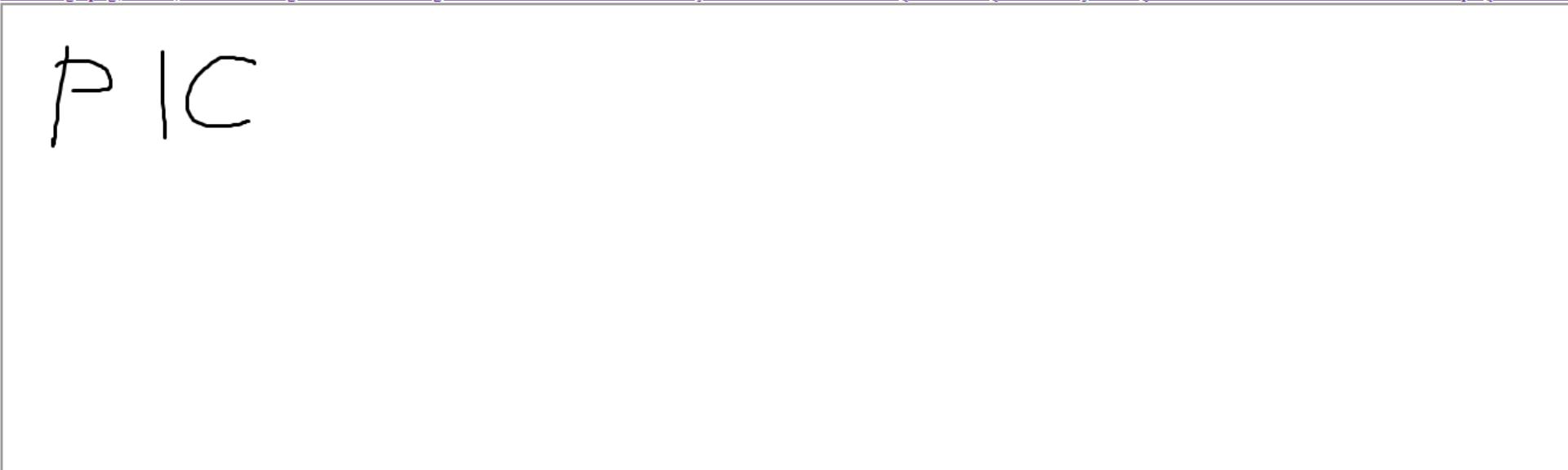
ETH - Add data to Ethereum from the browser

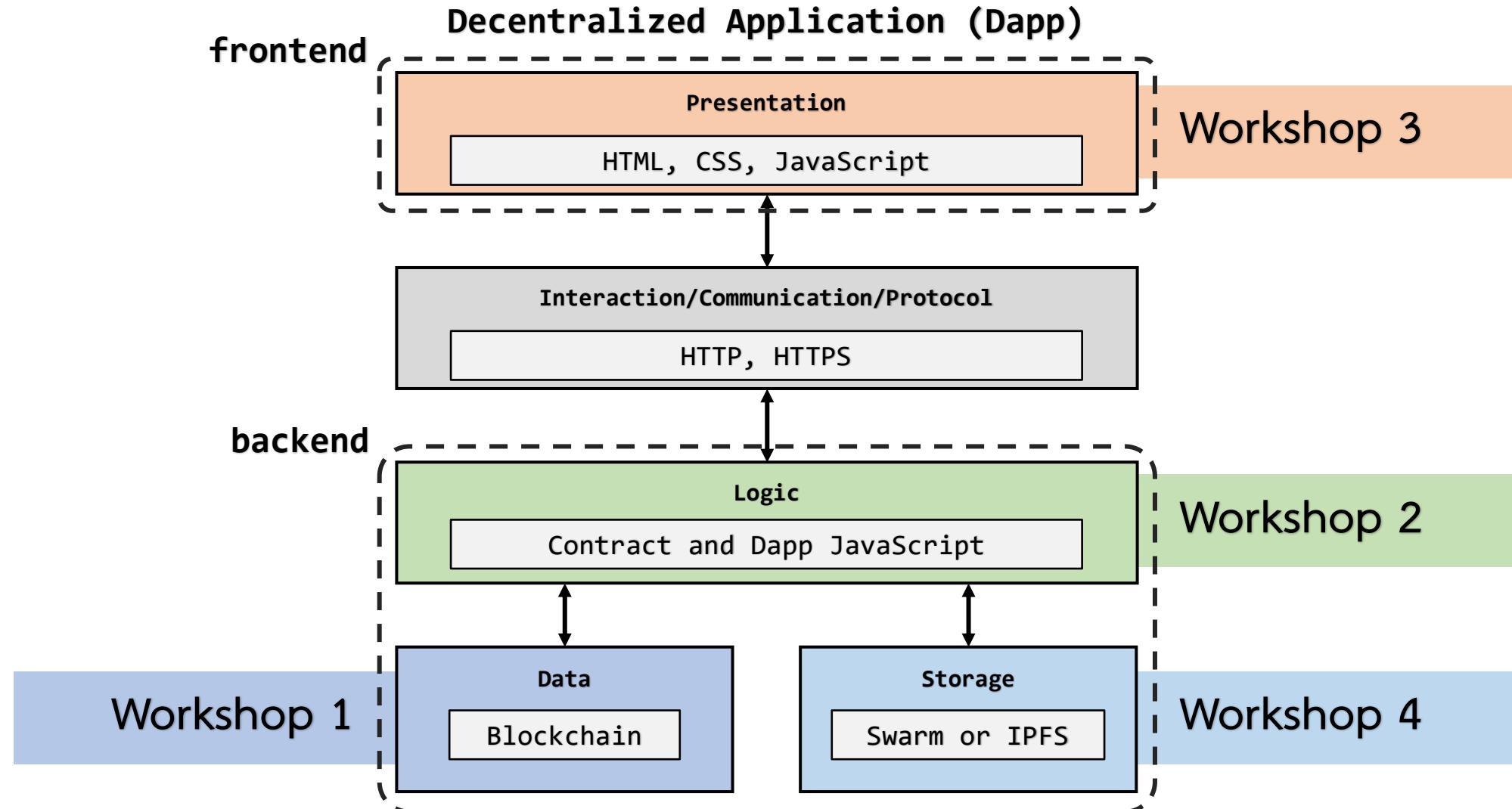
Choose File a.png add to ETH

index

show

[data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAXwAADPCAIAAAB5rP3yAAAAAXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAAJcEhZcwAADsMAAA7DAcdvqGQAAAk0SURB](#)







Blockchain Voting

1. เกิดอคติต่อกัน
2. เกิดการซื้อสิทธิ์ขายเสียมากขึ้น

เพราะคนซื้อตรวจสอบได้ว่าโหวตตามที่สัญญาไว้หรือเปล่า
และคนขายก็ไม่กล้าเบี้ยว เพราะตรวจสอบได้



<https://media.consensys.net/40-ethereum-apps-you-can-use-right-now-d64333769f7>



THANK YOU!



THANK YOU! :3