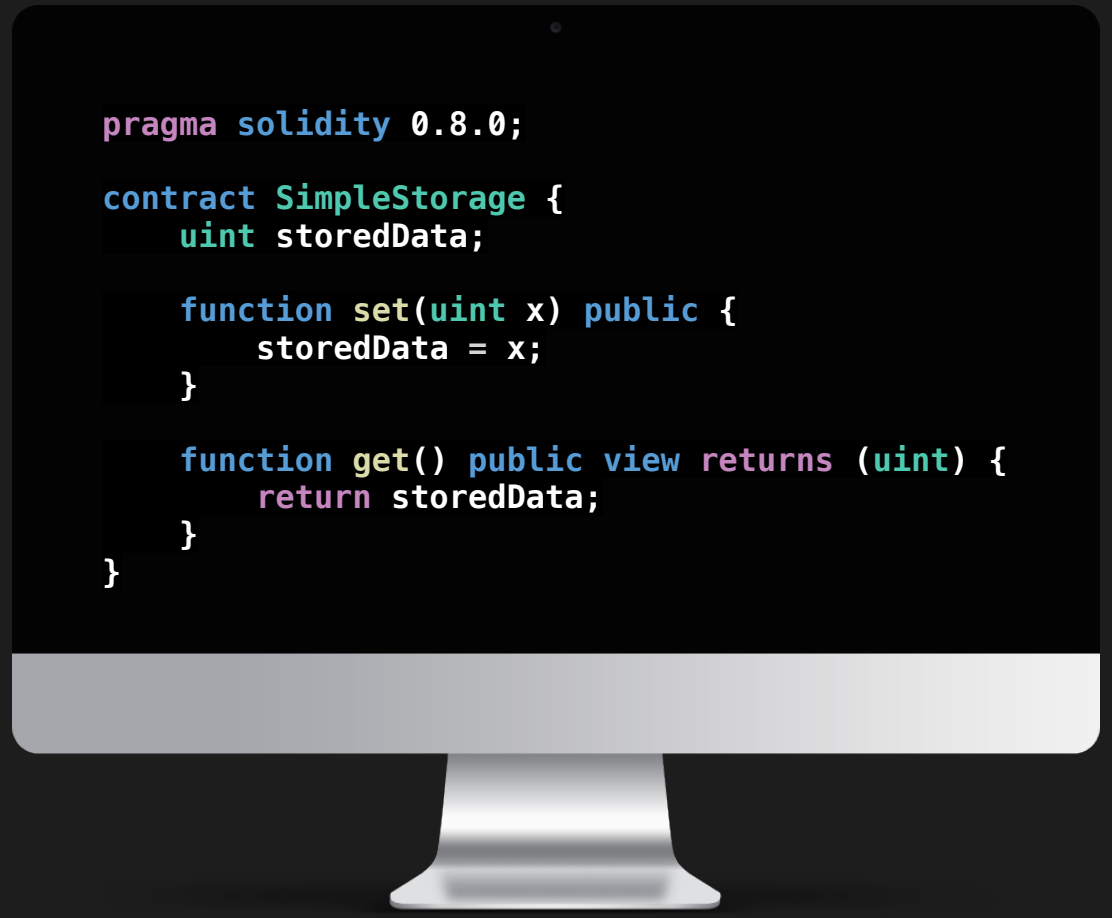
Smart Contract

- BChen, Bill Hsu -

目錄



Gas的設計

- 每種運算都有其相對應的成本
- Gas Price
  - 每個單位 Gas 的價格
  - 1 Gwei = 0.000000001 BNB (ETH)
- Gas Limit
  - 單筆交易中所願意支付 Gas 單位的最大數量
- Tx Fee
  - 最多為 Gas Limit * Gas Price

合約的部署

1. 寫好合約
2. 編譯合約
   - Bytecode
   - ABI
3. 透過線上 IDE 部署 or 其他

呼叫合約

1. Function 的識別碼
2. 放上需要的參數

Function : setName(string)

↓

Kecaak-256

↓

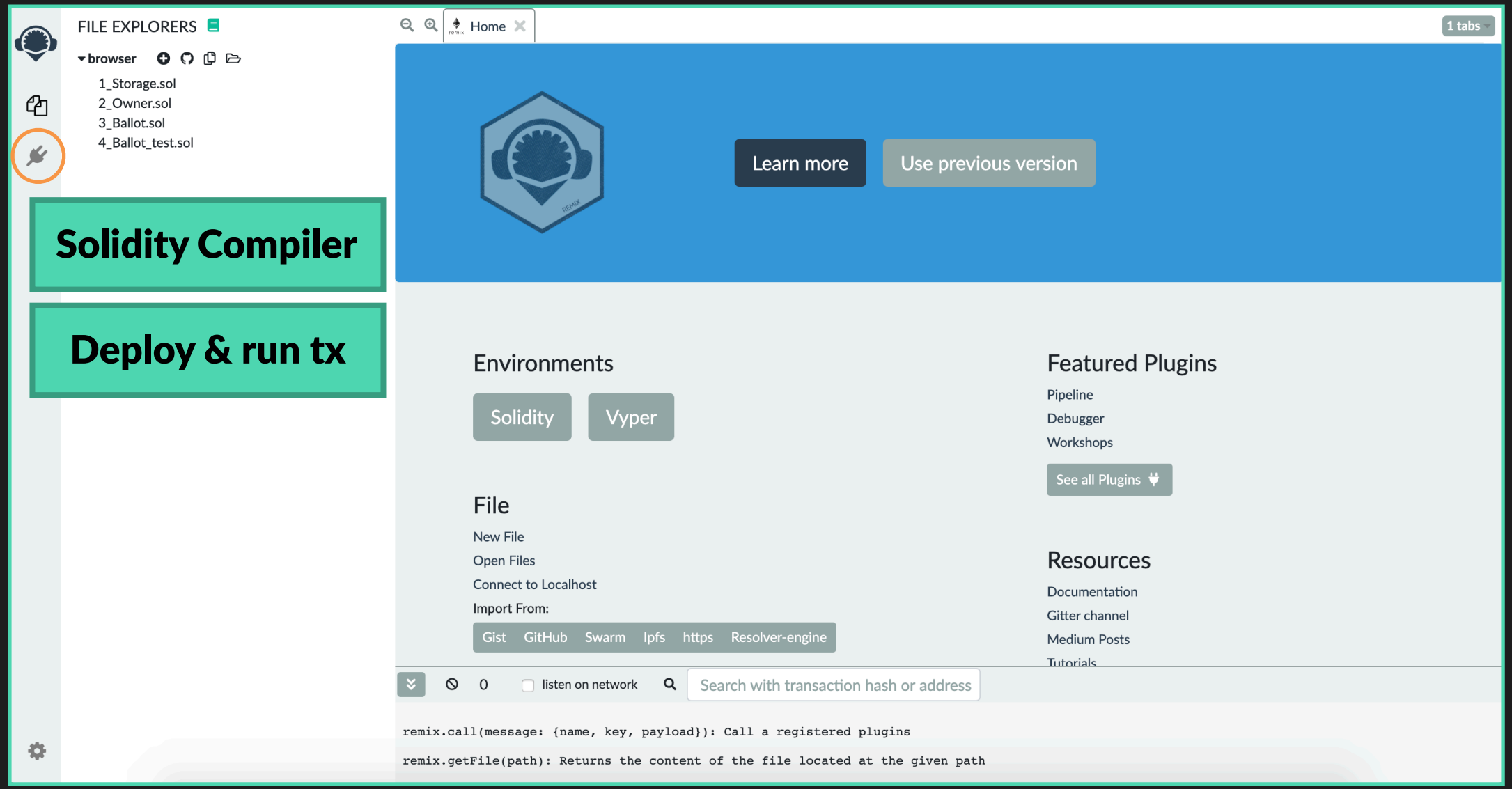
c47f0027.....

- 開發環境介紹 -

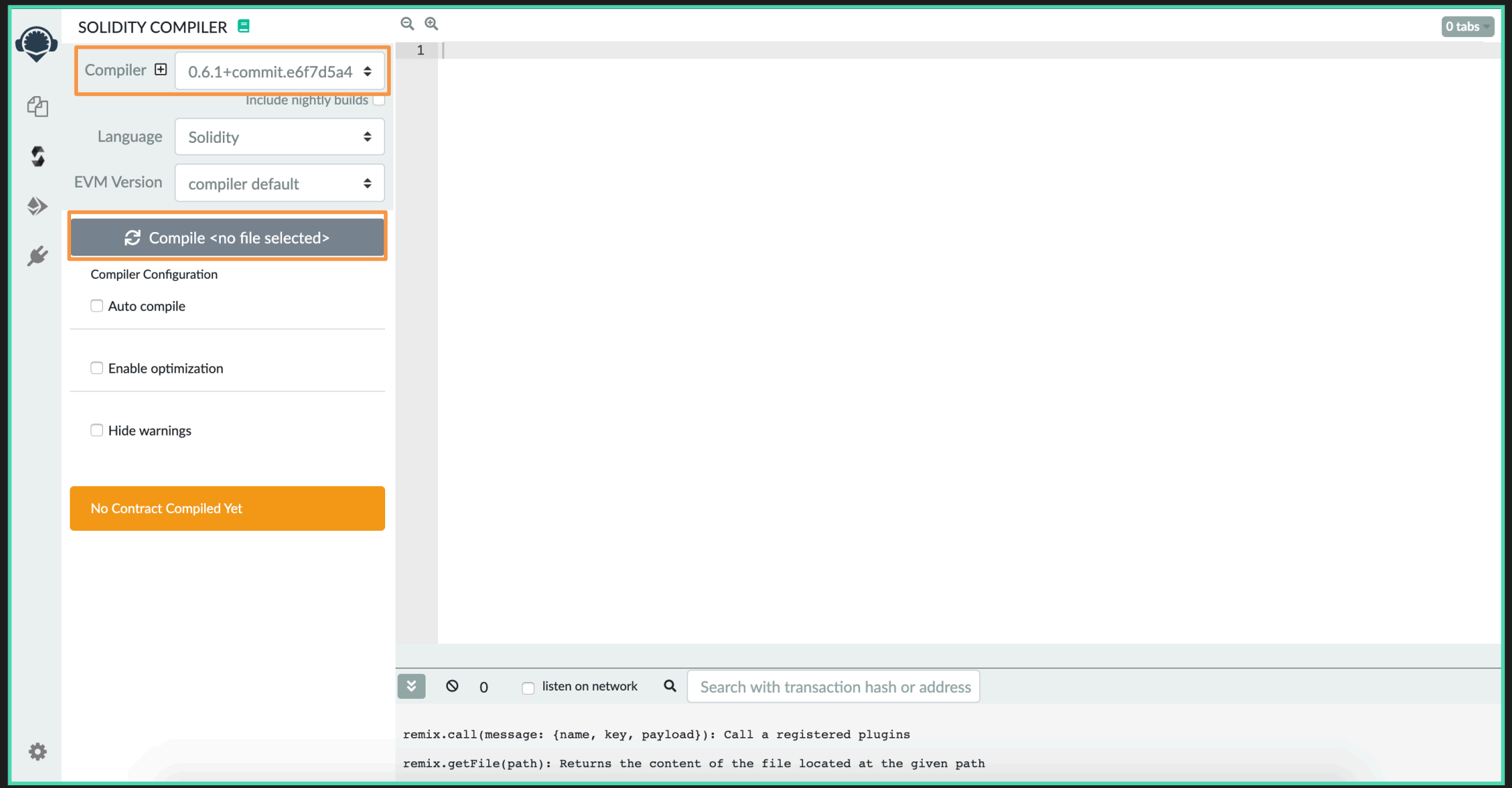
編譯合約

- Remix - 線上 IDE
  - http://remix.ethereum.org
- 安裝本機版 Remix IDE
  - `npm install remix-ide -g`
- Solc (Solidity Compiler) - localhost
  - https://www.npmjs.com/package/solc

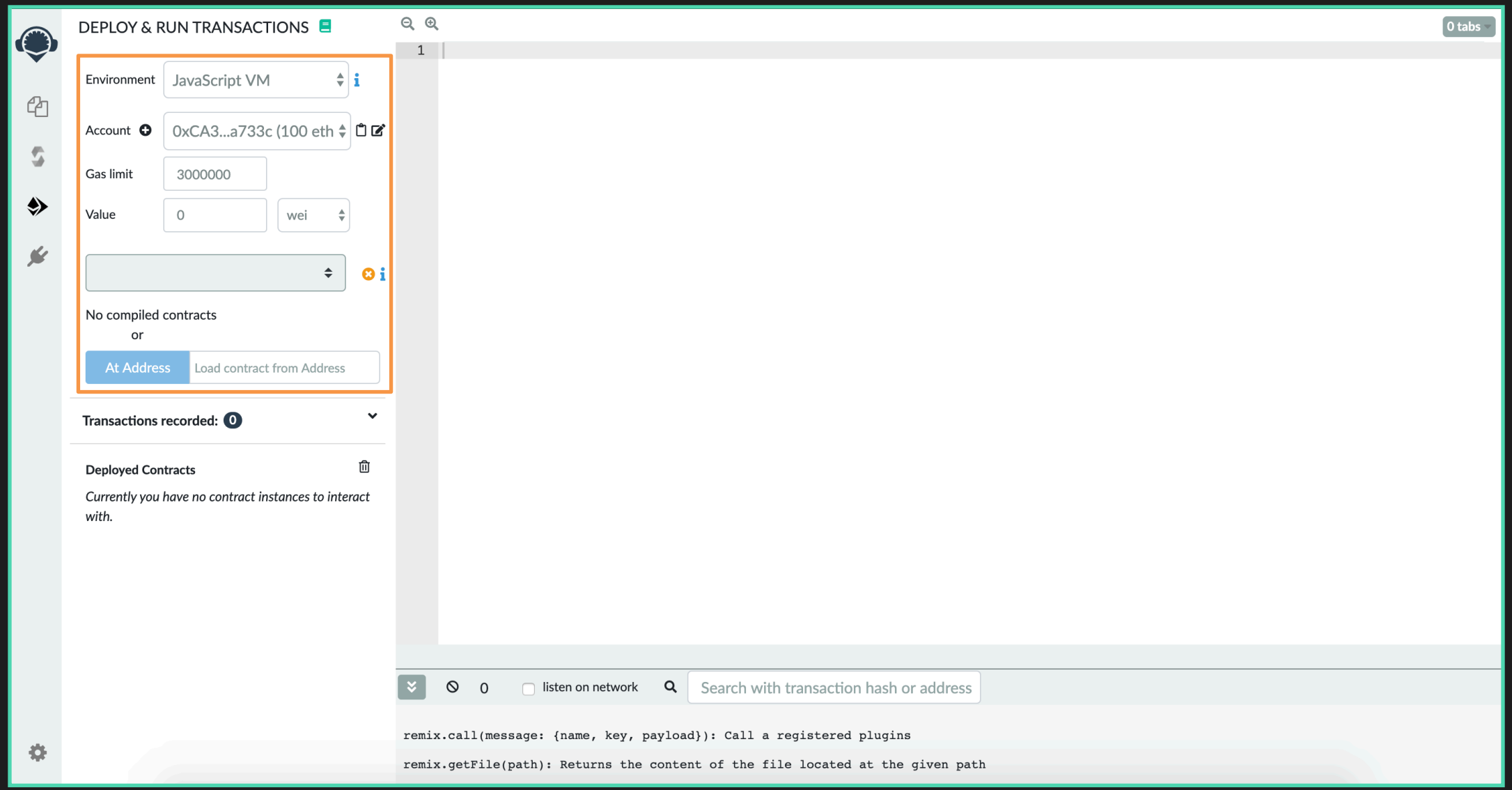
Remix - 線上 IDE


FILE EXPLORERS
browser
1_Storage.sol
2_Owner.sol
3_Ballot.sol
4_Ballot_test.sol
Solidity Compiler
Deploy & run tx
Home
1 tabs
Learn more
Use previous version
Environments
Solidity
Vyper
File
New File
Open Files
Connect to Localhost
Import From:
Gist
GitHub
Swarm
Ipfs
https
Resolver-engine
Featured Plugins
Pipeline
Debugger
Workshops
See all Plugins
Resources
Documentation
Gitter channel
Medium Posts
Tutorials
0
listen on network
Search with transaction hash or address
remix.call(message: {name, key, payload}): Call a registered plugins
remix.getFile(path): Returns the content of the file located at the given path

Remix - 線上 IDE


SOLIDITY COMPILER
Compiler
0.6.1+commit.e6f7d5a4
Include nightly builds
Language
Solidity
EVM Version
compiler default
Compile <no file selected>
Compiler Configuration
Auto compile
Enable optimization
Hide warnings
No Contract Compiled Yet
0 tabs
1
0
listen on network
Search with transaction hash or address
remix.call(message: {name, key, payload}): Call a registered plugins
remix.getFile(path): Returns the content of the file located at the given path

Remix - 線上 IDE


DEPLOY & RUN TRANSACTIONS
Environment
JavaScript VM
Account
0xCA3...a733c (100 eth
Gas limit
3000000
Value
0
wei
No compiled contracts
or
At Address
Load contract from Address
Transactions recorded: 0
Deployed Contracts
Currently you have no contract instances to interact with.
0 tabs
1
0
listen on network
Search with transaction hash or address
remix.call(message: {name, key, payload}): Call a registered plugins
remix.getFile(path): Returns the content of the file located at the given path

- 智能合約架構簡介 -

基礎架構

- 官方文檔：https://docs.soliditylang.org/en/v0.8.0/

```
pragma solidity 0.8.0;

contract SimpleStorage {
    uint256 storedData;

    function set(uint256 x) public {
        storedData = x;
    }

    function get() public view returns (uint256) {
        return storedData;
    }
}
```

變數宣告

很多函數

變數宣告

- 變數型態
  - bool
  - int / uint
  - bytes
  - address
  - string
  - array
  - mapping
- \+
- 能見度
  - public
  - private
  - internal
  - external
- \+
- 變數名稱

```
int8 public age;
bool private isOwner;
string name;
```

變數宣告

- address

```
address payable public bank;
```

- mapping

```
mapping(address => uint256) public balances;
balances[address] = 10;
uint256 balance = balances[address];
```

- array
  - push
  - pop
  - length

```
uint256[4] fixArr;
uint256[] dynamicArr;
```

函數宣告

- 函數名稱(參數) + 能見度 + 回傳值
  - public
  - private
  - internal
  - external

```
function funName() private {…}
function funName2(uint num) external returns(uint8) {…}
function deposit() public payable {…}
```

函數宣告

- View function
- Pure function
  - 不改變合約狀態
  - 函數執行不消耗 gas
  - 不需經過礦工驗證

```
function viewFun(uint256 a, uint256 b) public view returns (uint256) {
    return a * (b + 42) + now;
}

function pureFun(uint256 a, uint256 b) public pure returns (uint256) {
    return a * (b + 42);
}
```

特殊函數

- Constructor
  - 合約建構子
  - 只會執行一次
  - 非必須
- Selfdestruct
  - 合約自殺
  - 唯一參數為地址
  - 把合約剩餘的錢給該地址

```
contract shop {
    address payable owner;

    constructor() {
        owner = msg.sender;
    }

    function close() public {
        require(owner == msg.sender);
        selfdestruct(owner);
    }

}
```

特殊函數

- Fallback / Receive [payable]
  - 沒有 function 宣告
  - 沒有參數與回傳值
  - 必須是 external
  - 預設只有 2300 gas
  - 非必要
  - 觸發條件：
    1. 單純的轉帳
    2. 呼叫合約沒有的函數

```
contract StandardFallback {

    receive() external payable {}

    fallback() external {}
}
```

Event

- 合約內部函數觸發
- 將觸發參數存進 log 中
- Contract 無法直接取 log 的資料
- 額外的儲存空間，很便宜
- 方便 DAPP 監聽事件
- 搭配 **emit** 使用

```
event 事件名稱(參數型態1 參數名稱1, 參數型態2 參數名稱2, ... );
```

- 撰寫第一個智能合約 -

- NFT 基礎介紹 -

ERC 20 vs ERC 721

- Fungible Token
  - 可替代
  - 可分割
  - 幣幣等值
- Non Fungible Token
  - 不可替換性
  - 不可分割
  - 每個 Token 是獨一無二的 (tokenId)
  - 加密收藏品、虛擬寶物、
    房地產、股票債券所有權
  - metadata

NFT 價值

- 作品的證明與背後故事 > 作品本身
- 背後的發行商，降低認證門檻與成本
- 商品的價格資訊、交易紀錄可朔源
- 數位資產，擴大交易市場

NFT 基礎合約架構


```
function balanceOf(address owner) external view returns (uint256 balance);
function ownerOf(uint256 tokenId) external view returns (address owner);
function safeTransferFrom(address from, address to, uint256 tokenId) external;
function transferFrom(address from, address to, uint256 tokenId) external;
function approve(address to, uint256 tokenId) external;
function getApproved(uint256 tokenId) external view returns (address operator);
function setApprovalForAll(address operator, bool _approved) external;
function isApprovedForAll(address owner, address operator) external view returns(bool);
function safeTransferFrom(address from, address to, uint256 tokenId, bytes calldata) external;
event Transfer(address indexed from, address indexed to, uint256 indexed tokenId);
event Approval(address indexed owner, address indexed approved, uint256 indexed tokenId);
event ApprovalForAll(address indexed owner, address indexed operator, bool approved);
```

NFT Metadata


```
function name() external view returns (string _name);

function symbol() external view returns (string _symbol);

function tokenURI(uint256 _tokenId) external view returns (string);
```

```
{
    "name": "NAME",
    "description": "DESCRIPTION",
    "image": "URL",
}
```

-Deploy NFT on BSC -

- END -