スマート・コントラクト作成手順

スマートコントラクト(Solidity/Ethereum)開発環境構築と
コントラクトのサンプル実装について

一般社団法人フィンテック研究振興協会

2018 年 5 月 20 日

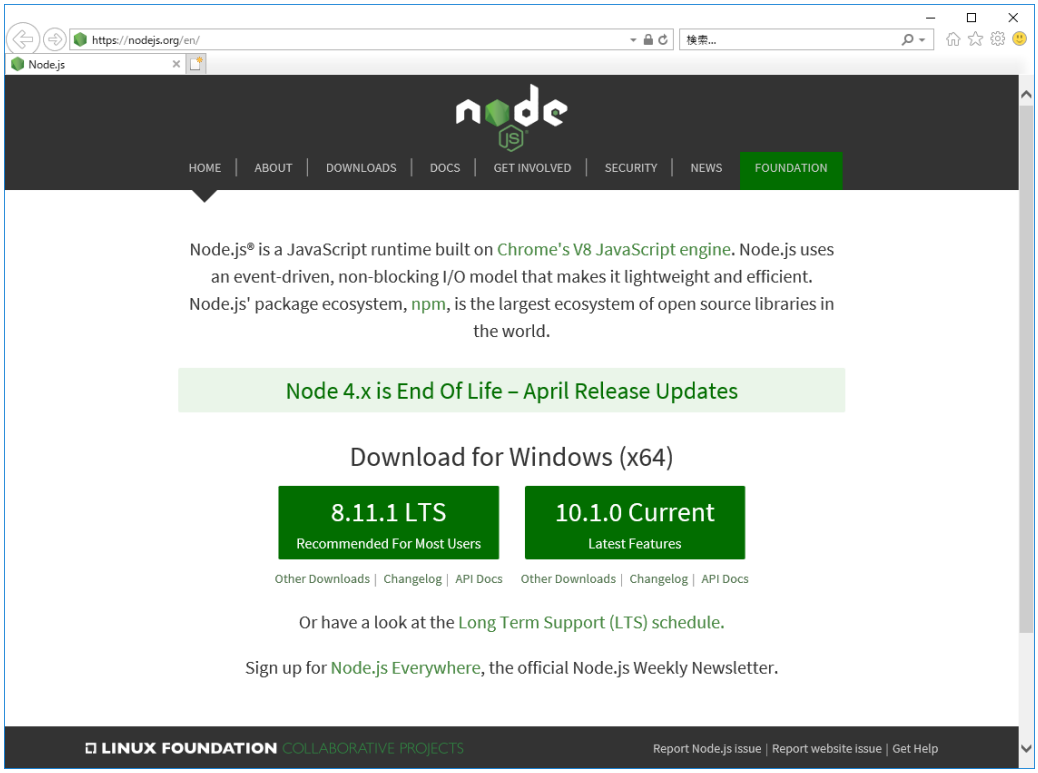
スマート・コントラクト作成手順

◆ 開発環境を構築する

■ Node.js をダウンロードする

以下のサイトより LTS 版をダウンロードし、インストールします。

https://nodejs.org/en/



コマンドプロンプトを起動し、「node -v」コマンドにてバージョンが表示されるか確認します。

```
C:¥Users¥xxx¥>node -v
v8.11.1
```

Node.js インストール時に npm も自動的にインストールされるのでこちらのバージョンも確認します。

```
C:¥Users¥xxx¥>npm -v
5.6.0
```

■ Remix(Browser-Solidity)をダウンロードする

コマンドプロンプトを起動し、以下のコマンドを実行します。

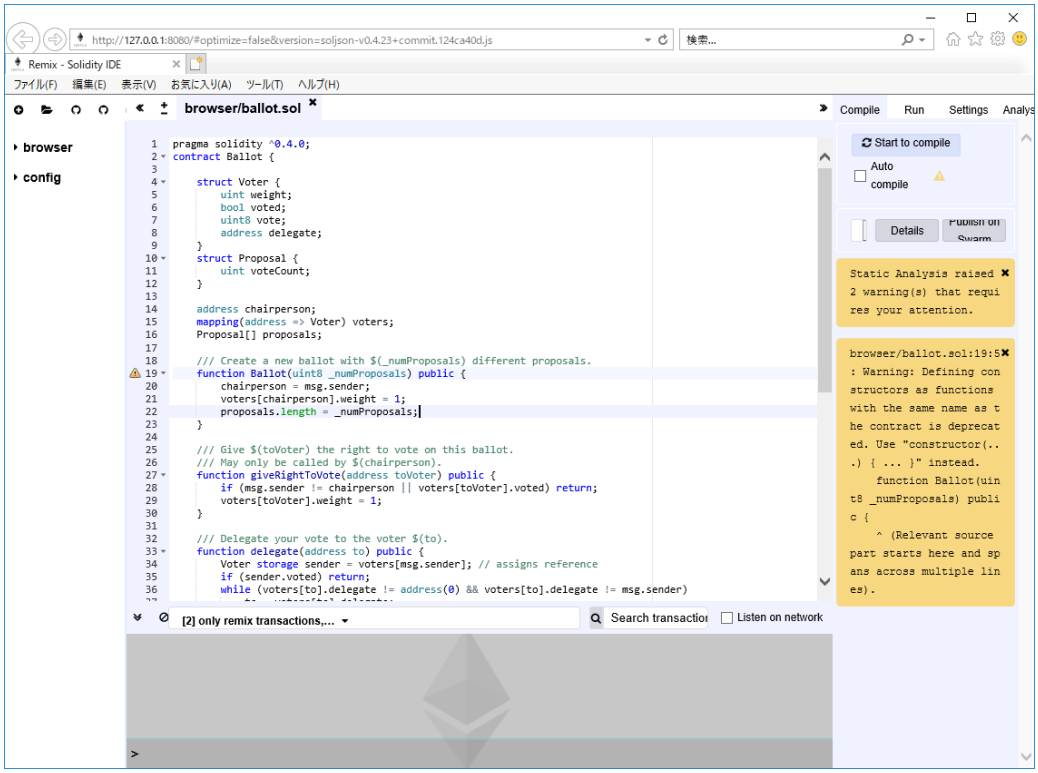
```
C:¥Users¥xxx¥>npm install remix-ide -g
```

■ Remix(Browser-Solidity)を起動する

コマンドプロンプトを起動し、「remix-ide」コマンドにて Remix を起動します。

```
C:¥Users¥xxx¥>remix-ide
Setup notifications for C:¥user¥xxx
Shared folder : C:¥user¥xxx
Starting Remix IDE at http://localhost:8080 and sharing C:¥user¥xxx
Mon May 14 2018 10:31:50 GMT+0900 (東京 (標準時)) Remixd is listening on 127.0.0.1:65520
```

ブラウザを起動し、「http://localhost:8080」へアクセスします。

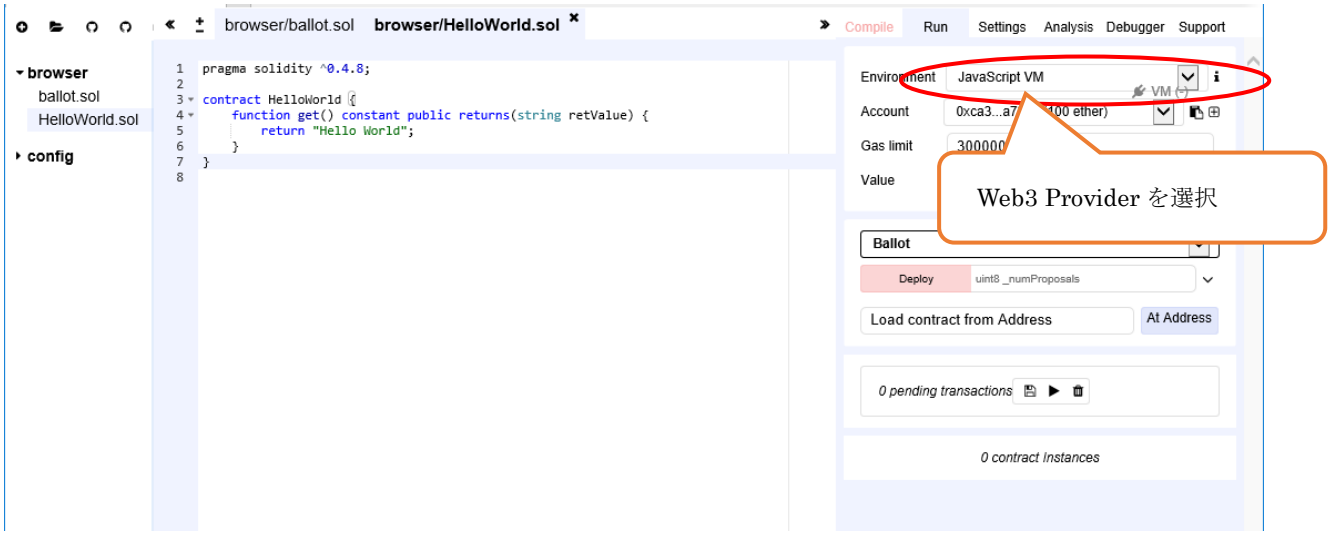


■ Remix と geth を接続する

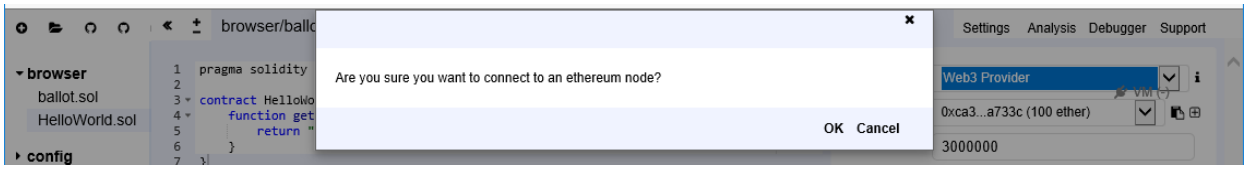
以下のコマンドで geth を起動します。

```
C:¥Users¥xxx¥data>geth --networkid "10" --nodiscover --rpc --rpcaddr "localhost" --rpcport "8545"
--rpcapi "web3,eth,neget,personal" --rpccorsdomain "*" --maxpeers 1 --datadir ./ console 2>> ./gethr.log
```

「Run」タブをクリックし、Environment を"Web3 Provider"に切り替えます。



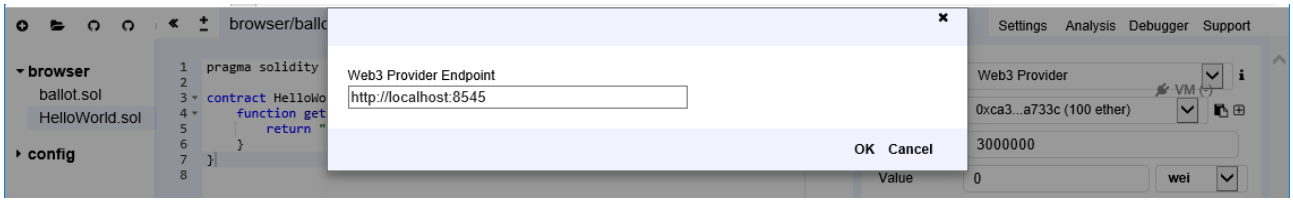
メッセージが表示されるので、「OK」をクリックします。



geth 起動時にオプションで指定した情報を入力します。

・「localhost」：--rpcaddr で指定した値

・「8545」：--rpcport で指定した値

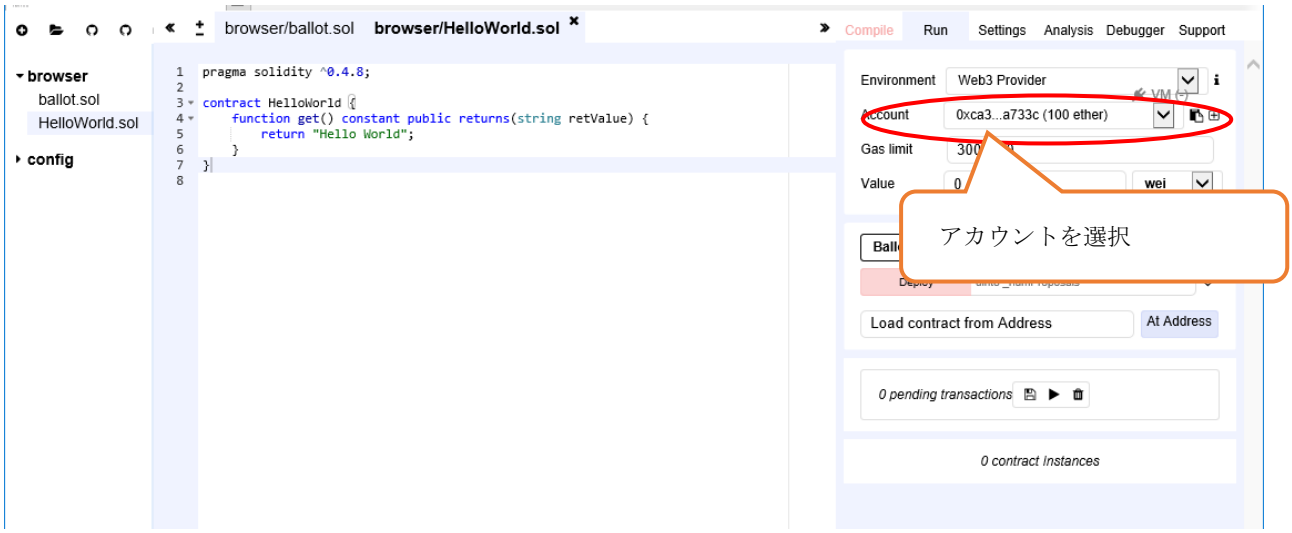


Geth コンソール上でマイニングを開始します。

```
>miner,start(1)
null
```

■ コントラクトをデプロイする

Account でデプロイするアカウントを指定します。

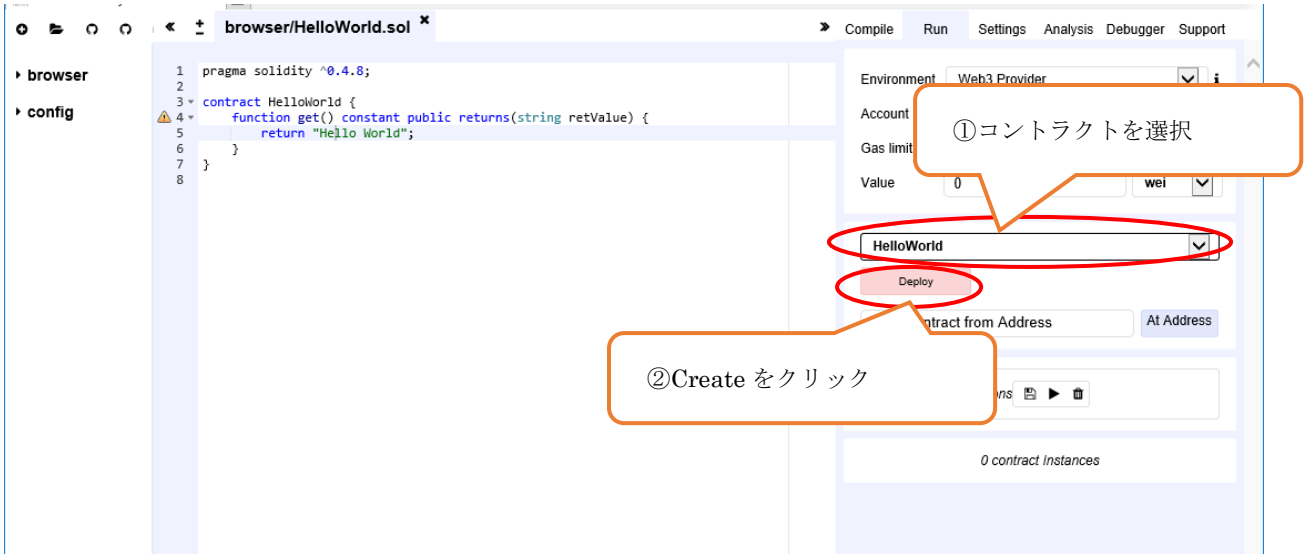


コントラクトを実行するにはアカウントのロックを解除する必要があります。

geth でデプロイするアカウントのロックを解除します。

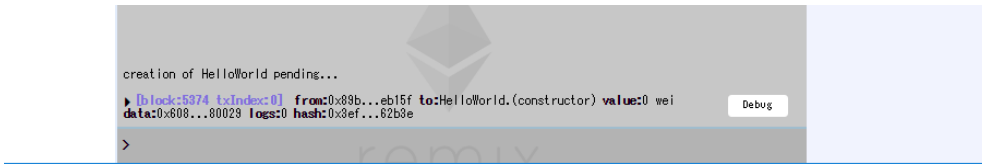
```
>personal.unlockAccount(eth.account[0])
Unlock account 0x89bdfcf98c7f240cf31b2f577485d9234a1eb15f
Passphrase:
true
```

デプロイするコントラクトを選択し「Deploy」をクリックします。



「Creation of (コントラクト名) pending...」と表示されます。

マイニングによりコントラクトがブロックチェーン上に取り込まれると「block:xxx txIndex:0 ...」と表示されます。



◆ コントラクトを作成する

■ Hello World

簡単なコントラクトを記述します。

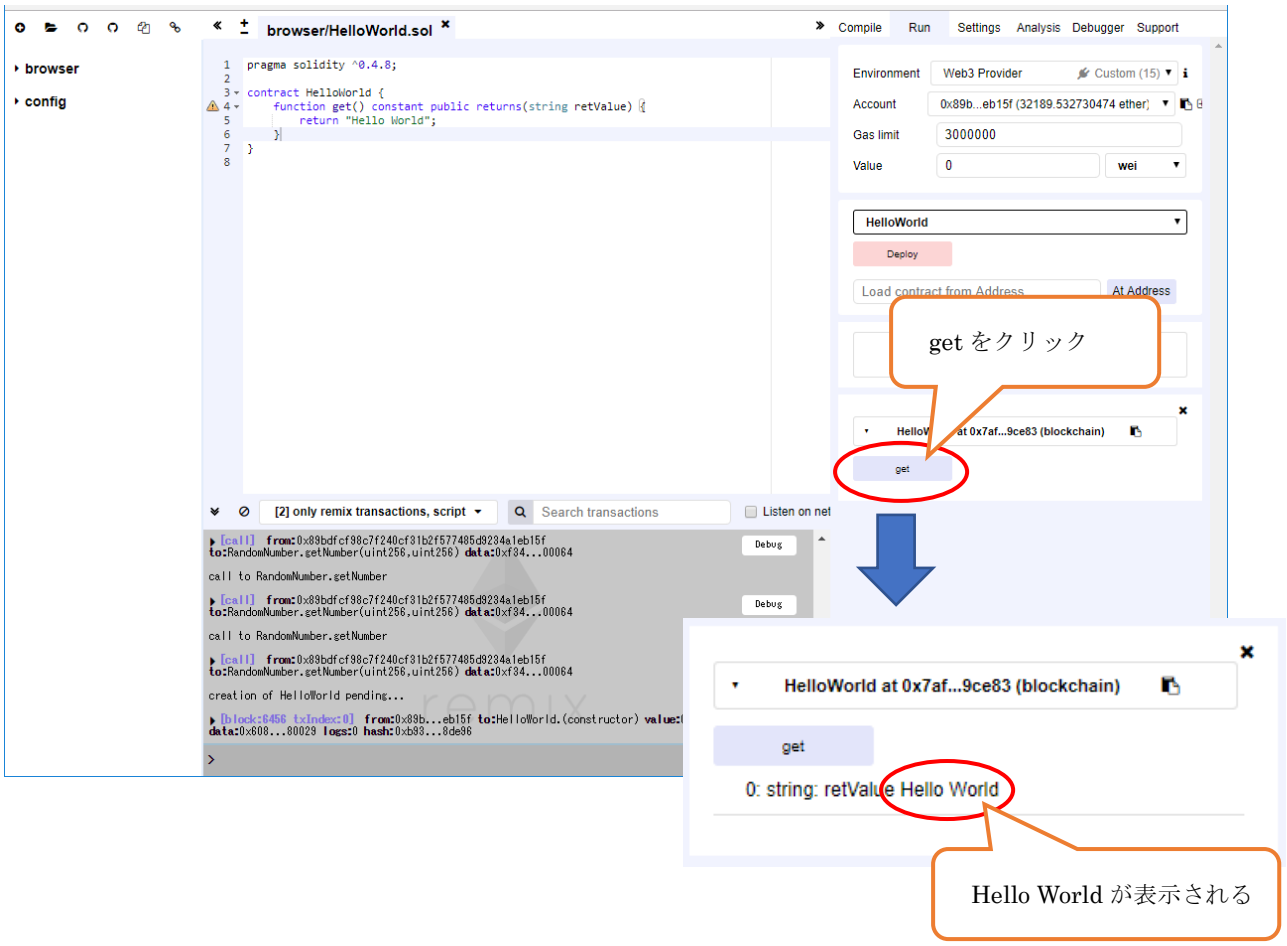
get()関数で"Hello World"を返すだけのコントラクトです。

```
pragma solidity ^0.4.8;


contract HelloWorld {
    function get() constant public returns(string retValue) {
        return "Hello World";
    }
}
```

Remix でデプロイし、実行します。

「get」をクリックすると、Hello World が表示されます。



■ 乱数を生成する

乱数を生成するコントラクトを記述します。

Solidity には乱数を生成するメソッドが存在しない為、最新ブロック番号と seed 値を基に生成します。

関数の引数は seed 値と乱数の最大値です。

```
pragma solidity ^0.4.8;

contract RandomNumber {
    function getNumber(uint seed, uint max) constant returns (uint randomNumber) {
        uint num        = uint(sha3(block.blockhash(block.number - 1), seed));
        num             %= max;
        return num;
    }
}
```

Remix でデプロイし、実行します。

「set」の右の入力に 100 と入力し「set」クリック後、「get」をクリックすると、乱数が表示されます。