

Desarrollo Blockchain Ethereum con Solidity

Módulo 5 - Web3

Web3

JSON RPC API

- **JSON:** Javascript Object Notation
- **RPC:** Remote Procedure Call

JSON RPC es un protocolo de comunicación que se utiliza para comunicarse con los nodos.

Mediante este protocolo podemos pedirle información a los nodos de Ethereum, por ejemplo, el último número de bloque.

```
// Request
curl -X POST --data '{"jsonrpc":"2.0","method":"eth_blockNumber","params":[],"id":83}'

// Result
{
  "id":83,
  "jsonrpc": "2.0",
  "result": "0x4b7" // 1207
}
```

Web3

- Son una serie de librerías que nos permiten interactuar con nodos locales o remotos de Ethereum.
- Encapsula la comunicación del JSON RPC.
- Sus implementaciones más populares son las de Javascript y Python.
- En este curso utilizaremos web3js para conectar los contratos con un sitio web.



Utilización de Web3js

- Su utilización es similar a cualquier librería de javascript.
- En general se utilizan funciones dentro del módulo **web3.eth**
- Por ejemplo el número de bloque actual se consulta con:
web3.eth.getBlockNumber()

Fuente: Google

getBlockNumber

```
web3.eth.getBlockNumber([callback])
```

Returns the current block number.

Returns

Promise returns **Number** - The number of the most recent block.

Example

```
web3.eth.getBlockNumber()  
  .then(console.log);  
> 2744
```

Instancias de Web3js

- Luego de instalar web3js se debe crear una instancia web3 y establecer un proveedor.
- Los navegadores compatibles con Ethereum tendrán un proveedor (ethereumProvider o web3.currentProvider) disponible.
- Si esta propiedad es nula, debe conectarse a un nodo remoto / local.

```
// in node.js use: var Web3 = require('web3');  
  
var web3 = new Web3(Web3.givenProvider || "ws://localhost:8546");
```

Web3 desde Truffle

- La consola de truffle nos permite ejecutar código Javascript.
- Además, la librería de web3js ya está funcionando en la consola.
- Podemos ejecutar sentencias de web3 en la consola de Truffle.

```
Truffle Develop started at http://127.0.0.1:8545/

Accounts:
(0) 0x42f85495ca1afed22d3152c35479e6ce4b00d12a
(1) 0x03ccd5cf358d60dc1116964a24da6849f04a12a1
(2) 0xa6f313092b3a00cd6a87d0cbe5acb8a9823de256
(3) 0xd48b9a5e8ab9dd3a21235f669fc810767bf70ba7
(4) 0x4beb8ea1c9b23d8b8bafb3fc75e752a0b55bc96b
(5) 0x43ed04a858bab9d5e4ce0788eb50d423fc1bb4dd
(6) 0xb2d854d9e35f12d3419d63ce137476ec826bef12
(7) 0xe78bdd579667a21c7d432450d006deff34585949
(8) 0xd61ac6d50423365bf5f3cb6a2d19644f69612ba4
(9) 0x1c90d3d4b58c4da98766cb75354a37901f26d756

Private Keys:
(0) 91d87459794248d039bfb355ec7ee2b594c40a4c921148b327a904b2a82588a2
(1) 9e751f4e4bc6c55ab41c78788e5f15968b7b2f01badf6d44711280ab8e38318d
(2) 7fea35b81cc80b6ebe040443aae4be58d89ed5c68e38fcd5a08a074e80b56276
(3) ba5dfe96c29fd8c7f12babacd41593809f32ca596a1d9b4dd247e299e9d49a82
(4) 24c7ca430277251acf404f2f1d92b7531154e0d267a7f854028925385d7923d4
(5) 82e5d3d4b2191f9d1267c04996fbb935b29b268fec39547d8c13f799297a0b90
(6) fccf28fa122aba66843593b53354a98da46fafce40d43ecbd644640d07ba561e
(7) c4c2e9a4248c4b512c469ad42170d0dd627bf0627425907110609a7be83ad183
(8) e1cfc1352a4f13cbf76faf14726388c4737329c19dd90b6899b4d03613f7542a
(9) 20f8b550b9a6b0b3cb3a953eeb01575113ca78845f41b1470123c8c54a9381a4

Mnemonic: bonus fiscal above priority file trip scatter minimum joy rate build sand

⚠ Important ⚠ : This mnemonic was created for you by Truffle. It is not secure.
Ensure you do not use it on production blockchains, or else you risk losing funds.

truffle(develop)> web3.eth.getBlockNumber()
0
truffle(develop)> █
```

¡Muchas gracias!

¡Sigamos trabajando!