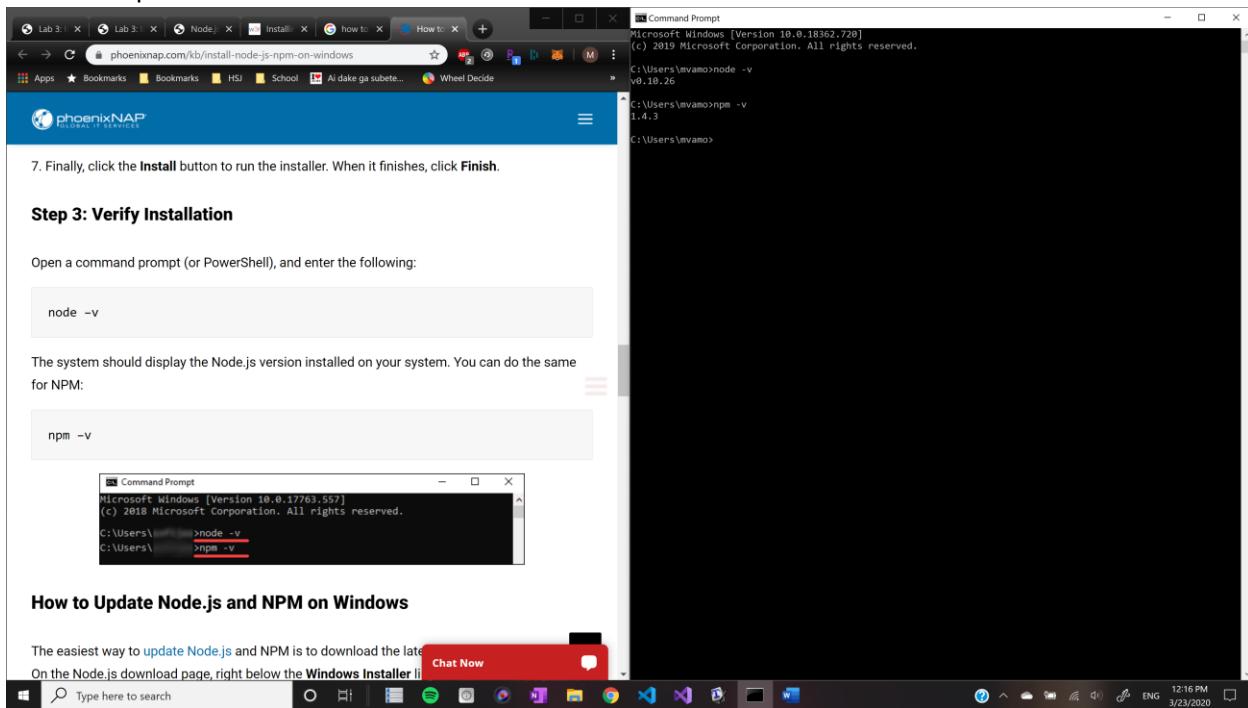


First attempt:



is the final

```
api "web3,eth,net,debug" --rpccorsdomain "*" --ipcdisable
```

t localhost:8545.

am or batch file.

d to [ganache-cli](#).
use the command
ave the latest geth
is in my testing and

```
env
```

pc has been
issue again, make
aming npm
ironment. This is

```
"8545" --rpcapi "web3,eth,net,debug" --rpccorsdomain "*" --ipcdisable
```

through all of the environment options.
Once finished, close and reload your console and re-run the commands above. They should now provide you with version number.

3. Use NPM to install the [Ethereumjs-testrpc](#):

```
> npm install -g ethereumjs-testrpc
```

Note:
 • There are several issues for the above command running on my Windows environment. This is the final command that works for me

```
d:\Geth> geth --dev --rpc --mine --rpccaddr "0.0.0.0" --rpcport "8545" --rpcapi "web3,eth,net,debug" --rpccorsdomain "*" --ipcdisable
```

To make this command work, you need to [Installing Geth](#).

4. Once finished, run the following command to start it:

```
> testrpc
```

Note:
 • This provides you with 10 different accounts and [private keys](#), along with a [local server](#) at [localhost:8545](#).
 • Error
 ▪ Symptom
 ○ 'testrpc' is not recognized as an internal or external command, operable program or batch file.
 ▪ Investigation
 ○ cannot start 'testrpc'
 ○ command Testrpc is not working
 ○ From the rpm docs the testrpc package has been deprecated and moved to [ganache-cli](#).
 My suggestion is to use that instead. But if you want to use test rpc then use the command [geth --rpc](#) to set it going. Of course it would be better to check if you have the latest [geth](#) version as well in case it does not work but so far I have been using this in my testing and it worked flawlessly.
 ○ geth --rpc will use a lot of memory by storing files in [c:\Users\Henry\Chang\AppData\Roaming](#).
 ○ [This package ethereumjs-testrpc has been deprecated](#). ethereumjs-testrpc has been renamed to [ganache-cli](#), would be better to use new one. If you face the issue again, make sure that Environment Variables contain path to [C:\Users\appData\Roaming\npm](#)
 • [Setting up Geth Mist TestRPC - Part 1 Getting Start with Smart Contracts](#) **
 • Solution

```
C:\Users\mvamo> geth --dev --rpc --mine --rpccaddr "0.0.0.0" --rpcport "8545" --rpcapi "web3,eth,net,debug" --rpccorsdomain "*" --ipcdisable
INFO [03-23|12:21:24.108] Maximum peer count                                     INFO [03-23|12:21:26.004] Using developer account
INFO [03-23|12:21:26.014] Starting peer-to-peer node                         INFO [03-23|12:21:26.014] Starting peer-to-peer node
ws://md46/g01.13.8                                                       ws://md46/g01.13.8
INFO [03-23|12:21:26.017] Allocated trie memory caches                         INFO [03-23|12:21:26.017] Allocated trie memory caches
INFO [03-23|12:21:26.023] Writing custom genesis block                      INFO [03-23|12:21:26.023] Writing custom genesis block
INFO [03-23|12:21:26.028] Persisted trie from memory database                  INFO [03-23|12:21:26.028] Persisted trie from memory database
INFO [03-23|12:21:26.037] Initialising chain configuration                   config["ChainID": 1337, "Homestead": 0, "DAO": 0, "EIP150": 0, "EIP155": 0, "EIP158": 0, "Byzantium": 0, "Constantinople": 0, "Petersburg": 0, "Istanbul": 0, "MuirGlacier": null, "Engine": "clique"]
INFO [03-23|12:21:26.104] Initialising Ethereum protocol                      INFO [03-23|12:21:26.104] Initialising Ethereum protocol
  <n/a>
INFO [03-23|12:21:26.122] Upgrade blockchain database version                  from=<n/a> to=7
INFO [03-23|12:21:26.152] Loaded most recent local header                    number=0 hash=7e4b5c_371b36 to=1 age=50y11m
  <n/a>
INFO [03-23|12:21:26.183] Loaded most recent local full block                 number=0 hash=7e4b5c_371b36 to=1 age=50y11m
  <n/a>
INFO [03-23|12:21:26.207] Loaded most recent local fast block                number=0 hash=7e4b5c_371b36 to=1 age=50y11m
  <n/a>
INFO [03-23|12:21:26.406] Allocated fast sync bloom                          size=512.00MB
INFO [03-23|12:21:26.409] Initialized fast sync bloom                        item=1 generate=0.000 elapsed=0s
INFO [03-23|12:21:26.413] Stored checkpoint snapshot to disk                  number=0 hash=7e4b5c_371b36 to=1 age=50y11m
INFO [03-23|12:21:26.417] started whisper v.6.0                           seq=1 tx=c8892380006119c ip=127.0.0.1 udp=0 tcp=63913
INFO [03-23|12:21:26.426] HTTP endpoint opened                                url=http://0.0.0.0:8545 cors=* vhosts=local
host
INFO [03-23|12:21:26.449] Started P2P networking                            self="enode://76a65f96202881958dd67547490e1da64d33fb0f05e5df7e0a5470a872bd446b1240af4446a9da08ab542f54c0e88508370264ccc93d3d3004ccfc2@127.0.0.1:63913"
INFO [03-23|12:21:26.502] Transaction pool price threshold updated          price=<10000000000
INFO [03-23|12:21:26.559] Transaction pool price threshold updated          price=<10000000000
INFO [03-23|12:21:26.559] Etherbase automatically configured               address=<0x3c576a888a39bd242f48880c6045AD02D08425ac
INFO [03-23|12:21:26.578] Sealing paused, waiting for transactions          self="enode://76a65f96202881958dd67547490e1da64d33fb0f05e5df7e0a5470a872bd446b1240af4446a9da08ab542f54c0e88508370264ccc93d3d3004ccfc2@127.0.0.1:63913"
INFO [03-23|12:21:26.578] Commit new mining work                           number=1 sealhash=722a69..ba659d uncles=<0 tx=<0x0> proto=tcp export=63913 import=63913
INFO [03-23|12:21:26.652] Mapped network port                            proto=<0x0> ip=127.0.0.1 port=8545
  <n/a>="UPnP IGv1-IP1"
```

Stuck on the same line for 9 hours

```
C:\Users\mvamo>eth --dev --rpc --mine --praddr "0.0.0.0" --rpcport 8545 --rpccorsdomain "*" --ipcdisable
Microsoft Windows [Version 10.0.18362.720]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\mvamo>eth --dev --rpc --mine --praddr "0.0.0.0" --rpcport 8545 --rpccorsdomain "*" --ipcdisable
INFO [03-23 14:30:45.657] Maximum peer count
INFO [03-23 14:30:45.657] Maximum peer count
INFO [03-23 14:30:47.603] Starting peer-to-peer node
INFO [03-23 14:30:47.604] Allocated trie memory caches
INFO [03-23 14:30:47.607] Writing custom genesis block
INFO [03-23 14:30:47.610] Persisted chain from memory database
INFO [03-23 14:30:47.612] Persisted chain configuration
INFO [03-23 14:30:47.612] Persisted chain configuration
INFO [03-23 14:30:47.615] Initializing Ethereum protocol
INFO [03-23 14:30:47.617] Upgrade blockchain database version
INFO [03-23 14:30:47.620] Loaded most recent local header
INFO [03-23 14:30:47.622] Loaded most recent local full block
INFO [03-23 14:30:47.623] Loaded most recent local fast block
INFO [03-23 14:30:47.624] Loaded most recent local header
INFO [03-23 14:30:47.754] Initialized fast sync bloom
INFO [03-23 14:30:47.754] Stored checkpoint snapshot to disk
INFO [03-23 14:30:47.758] Started whisper v.6.0
INFO [03-23 14:30:47.759] New local node record
INFO [03-23 14:30:47.761] HTTP endpoint opened
INFO [03-23 14:30:47.762] Started P2P networking
INFO [03-23 14:30:50.075] Mapped network port
7223discport=0
INFO [03-23 14:30:47.765] Transaction pool price threshold updated price=1000000000
INFO [03-23 14:30:47.767] Transaction pool price threshold updated price=1
INFO [03-23 14:30:47.771] Etherbase automatically configured address=0x039A8a41eeefebADD07f42Ddbf447ca8267E4b41A
INFO [03-23 14:30:47.773] Commit new mining work number=1 sealhash=55d377_51c783 uncles=0 txs=0 gas=0 fees=0 elapsed=0s
INFO [03-23 14:30:47.773] Sealing paused, waiting for transactions proto=tcp extport=53272 intport=53272 interface="UPNP IGv1-IP1"
INFO [03-23 14:30:50.075] Mapped network port
```

Type here to search

Hulu Student In Exercises Lab 3: Inter lab3_Basic... +

Apps Bookmarks Bookmarks HSJ School Bay Area counties a... Random Dinner Ge...

9:26 PM 3/23/2020

Command Prompt

Microsoft Windows [Version 10.0.18362.720]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\mvamo>mkdir coursetro-eth

C:\Users\mvamo>cd coursetro-eth

C:\Users\mvamo\coursetro-eth>npm init

This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sane defaults.

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

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

Press `Ctrl+C` at any time to quit.

name: (coursetro-eth)
version: (0.0.0)
description:
entry point: (Index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to C:\Users\mvamo\coursetro-eth\package.json:

```
{
  "name": "coursetro-eth",
  "version": "0.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}
```

Is this ok? (yes)

C:\Users\mvamo\coursetro-eth>

Type here to search

11:55 AM 3/30/2020

1. Switch over to the [Remix IDE](#), click on the **Run** tab, and then change the **Environment dropdown** from **Javascript VM to Web3 Provider**.

- Error
 - Symptom
 - Got this error when trying to switch from "Javascript VM" to "Web3 provider" in Remix.

"Not possible to connect to the Web3 provider. Make sure the provider is running and a connection is open (via IPC or RPC)"
 - Investigation
 - [Testrpc running on 8545, yet Remix can't connect #588](#)
 - Because <https://ethereum.github.io/browser-solidity> is working on https, using http

Note:

- Hit enter through all of the prompts.

4. Run the following command to install `web3.js`:

```
> npm install ethereum/web3.js
```

Note:

- There are several issues for the above command running on my Windows environment. This is the final command that works for me

```
npm install web3@0.20.6 --save
```

`web3.js` is the Ethereum compatible JavaScript API which implements the Generic [JSON RPC](#) spec. It's available on `npm` as a node module, for Bower and component as embeddable scripts, and as a `meteor.js` package.

Step 3: Changing the Environment in Remix

- Switch over to the [Remix IDE](#), click on the **Run** tab, and then change the **Environment dropdown** from **Javascript VM** to **Web3 Provider**.
 - Error
 - Symptom
 - Got this error when trying to switch from "Javascript VM" to "Web3 provider" in Remix

"Not possible to connect to the Web3 provider. Make sure the provider is running and a connection is open (via IPC or RPC)"
 - Investigation
 - [Testrpc running on 8545, yet Remix can't connect #588](#)
 - Because https://ethereum.github.io/browser-solidity is working on https, using http protocol with http://remix.ethereum.org/ in my case problem was solved
 - [Connecting to remote Web3 provider #975](#) •
 - Solution

```
d:\Geth> geth --dev --rpc --mine --rpccaller "0.0.0.0" --rpcport "8545" --rpccaller "web3,eth,n
```

- Hit "OK" and then specify the `testrpc` localhost address (by default, it's <http://localhost:8545>)
 - This means instead of deploying and testing in the **Javascript VM**, we're now using the **TestRPC client**

Step 3: Changing the Environment in Remix

- Switch over to the [Remix IDE](#), click on the **Run** tab, and then change the **Environment dropdown** from **Javascript VM** to **Web3 Provider**.
 - Error
 - Got this error when trying to switch from "Javascript VM" to "Web3 provider" in Remix

"Not possible to connect to the Web3 provider. Make sure the provider is running and a connection is open (via IPC or RPC)"
 - Investigation
 - [Testrpc running on 8545, yet Remix can't connect #588](#)
 - Because https://ethereum.github.io/browser-solidity is working on https, using http protocol with http://remix.ethereum.org/ in my case problem was solved
 - [Connecting to remote Web3 provider #975](#) •
 - Solution

```
d:\Geth> geth --dev --rpc --mine --rpccaller "0.0.0.0" --rpcport "8545" --rpccaller "web3,eth,n
```

- Hit "OK" and then specify the `testrpc` localhost address (by default, it's <http://localhost:8545>)
 - This means instead of deploying and testing in the **Javascript VM**, we're now using the **TestRPC client** on your computer.

If you haven't been following along since the previous lesson, paste in this contract in a new solidity file called "`Coursetro.sol`:

```
pragma solidity ^0.4.18;

contract Coursetro {
    string fName;
    uint age;
    function setInstructor(string _fName, uint _age) public {
        fName = _fName;
        age = _age;
    }
    function getInstructor() public constant returns (string, uint) {
        return (fName, age);
    }
}
```

index.html (The final version will be created later.)

```
<!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>
    <link rel="stylesheet" type="text/css" href="main.css">
    <script src="../node_modules/web3/dist/web3.min.js"></script>
</head>
<body>
    <div class="container">
        <h1>CoursePro Instructor</h1>
        <h2 id="instructor"></h2>
        <label for="name" class="col-lg-2 control-label">Instructor Name</label>
        <input id="name" type="text">
        <label for="age" class="col-lg-2 control-label">Instructor Age</label>
        <input id="age" type="text">
        <button id="button">Update Instructor</button>
    </div>
    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>
    <script>
        // Our future code here...
    </script>
</body>
</html>
```

main.css

```
body {
    background-color:#F0F0F0;
    padding: 2em;
    font-family: 'Raleway', 'Source Sans Pro', 'Arial';
}

.container {
    width: 50%;
    margin: 0 auto;
}

label {
    display: block;
    margin-bottom: 10px;
}

input {
    padding: 10px;
    width: 50%;
    margin-bottom: 1em;
}

button {
    margin: 2em 0;
    padding: 1em 4em;
    display: block;
}

#instructor {
    padding: 1em;
    background-color: #ffff;
    margin: 1em 0;
}
```

Step 5: Using Web3.js to Connect & Interact with the Smart Contract

- Going back to the `index.html`, at the bottom of the file we have an empty `<script>` tag. This is where we will write the necessary code to work with our smart contract.
- In the `head tags`, we're already importing the `Web3.js` library, so now, let's use it to connect to our `testrpc client`:

```

<script>
if (typeof web3 != 'undefined') {
  web3 = new Web3(web3.currentProvider);
} else {
  // set the provider you want from Web3.providers
  web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:8545"));
}
</script>

```

Note:

- Normally the book's author would never use `jQuery` (He is a big Angular fan), but this keeps things more simple.
- This code comes directly from the [Web3.js GitHub](#) page.
- It's saying that if `web3` is not undefined, then we'll use that as our `provider`. If it's undefined (else), we can manually specify the `provider` ourselves.
- You may be wondering, how would `web3` be defined? Well, if you're using the Chrome extension `MetaMask` (which we will use later in this course) or an Ethereum browser like `Mist`, the `provider` is automatically injected.

- Specify a `default ethereum account` to use through the `web3.eth.defaultAccount` method:

```

<script>
// Previous if/else statement removed for brevity
web3.eth.defaultAccount = web3.eth.accounts[0];
</script>

```

Note:

- Remember when we ran the `testrpc` console command? It provided us with `10 accounts`. We're simply choosing the `first account` here to use.

- If you switch back to the `Remix IDE`, click on the `Compile` tab and click `Details`. Scroll down until you see the `Interface - ABI` section and click the `copy` icon as shown below:

The screenshot shows the Remix IDE interface. In the center, there is a code editor with the following code:

```

<script>
if (typeof web3 != 'undefined') {
  web3 = new Web3(web3.currentProvider);
} else {
  // set the provider you want from Web3.providers
  web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:8545"));
}
</script>

```

Below the code editor, the interface tab is selected. In the `ABI` section, there is a copy icon (a blue square with a white arrow) next to the ABI JSON string. The ABI JSON is as follows:

```

{
  "name": "Course",
  "inputs": [
    {
      "name": "name",
      "type": "string"
    },
    {
      "name": "age",
      "type": "uint8"
    }
  ],
  "stateMutability": "nonpayable",
  "type": "contract"
}

```

The screenshot also shows the Windows taskbar at the bottom with various application icons.

Note:

- Remember when we ran the `testrpc` console command? It provided us with [10 accounts](#). We're simply choosing the [first account](#) here to use.

4. If you switch back to the [Remix IDE](#), click on the [Compile](#) tab and click [Details](#). Scroll down until you see the [Interface - ABI](#) section and click the [copy icon](#) as shown below:

`pragma solidity ^0.4.18;`
`contract Coursetro {`
 `string fName;`
 `uint age;`
`}`
`//Function Course{() public {`
`constructor() public {`
`fName = "John";`
`age = 34;`
`}`
`}`
`// setInstructor accepts 2 parameters, _fName and _age.`
`// Once called, we set our string fName to the returned`
`value. This is where we can change the fName variable. This is where`
`we return the fName and age variable once it's called.`
`function setInstructor(string _fName, uint _age) public {`
`fName = _fName;`
`age = _age;`
`}`
`}`
`// The getInstructor() function is defined as being`
`constant. This means that it will always return the same value. This is where`
`we return the fName and age variable once it's called.`
`function getInstructor() public constant returns (string, uint) {`
`return (fName, age);`
`}`

lab3.sol

```
pragma solidity ^0.4.18;
contract Coursetro {
    string fName;
    uint age;
}
function Coursetro() public {
    constructor();
}
```

SOLIDITY COMPILER

Compiler: CourseTro
Language: Solidity
EVM Version: 0.4.25+commit.5dbfbf1

ABI: [Copy value to clipboard](#)

Contract: Coursetro

Public: [Copy value to clipboard](#)

Code: [Copy value to clipboard](#)

METADATAHASH: [Copy value to clipboard](#)

VS Code

File | file:///D:/nodejs/coursetro-eth/index.html

Coursetro Instructor

Instructor Name:

Instructor Age:

Update Instructor

index.html

```
<script>
// Previous if/else statement removed for brevity
web3.eth.defaultAccount = web3.eth.accounts[0];
var CoursetroContract = web3.eth.contract(YOUR ABI);
var Coursetro = CoursetroContract.at('PASTE CONTRACT ADDRESS HERE');
console.log(Coursetro);
</script>
```

Error List: 0 Errors, 0 Warnings, 0 Messages

6. Go back to **Remix** and click the **Run** tab, and click on the **copy** icon next to the **contract** that we created earlier on the right column.

Step 1
Step 2
Step 3

7. Back in **index.html** add the following line:

```
<script>
// Previous if/else statement removed for brevity
web3.eth.defaultAccount = web3.eth.accounts[0];
var Coursetro = web3.eth.contract(YOUR ABI);
var Coursetro = CoursetroContract.at('PASTE CONTRACT ADDRESS HERE');
console.log(Coursetro);
</script>
```

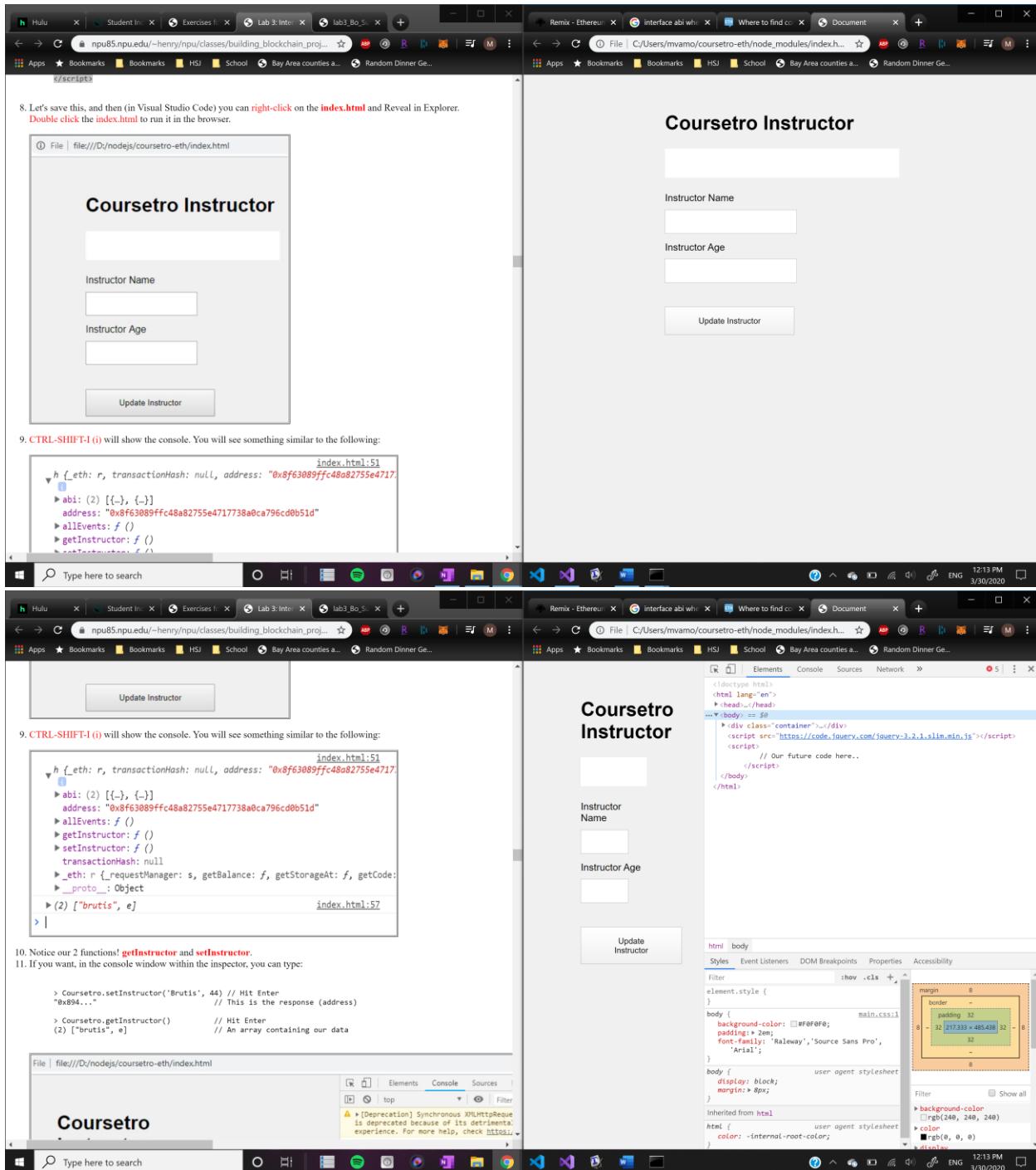
Step 1
Step 2
Step 3

7. Back in **index.html** add the following line:

```
<script>
// Previous if/else statement removed for brevity
web3.eth.defaultAccount = web3.eth.accounts[0];
var CoursetroContract = web3.eth.contract(YOUR ABI);
var Coursetro = CoursetroContract.at("PASTE CONTRACT ADDRESS HERE");
console.log(Coursetro);
</script>
```

8. Let's save this, and then (in Visual Studio Code) you can right-click on the **index.html** and Reveal in Explorer. Double click the **index.html** to run it in the browser.

Step 1
Step 2



10. Notice our 2 functions! `getInstructor` and `setInstructor`.

11. If you want, in the console window within the inspector, you can type:

```
> Coursetro.setInstructor('Brutis', 44) // Hit Enter
"0x894..." // This is the response (address)

> Coursetro.getInstructor() // Hit Enter
(2) ["brutis", e] // An array containing our data
```

File | file:///D:/nodejs/coursetro-eth/index.html

Coursetro Instructor

Bruce (40 years old)

Instructor Name

Instructor Age

Update Instructor

Console Output:

```
[Deprecation] Synchronous XMLHttpRequest is deprecated because of its detrimental experience. For more help, check https://xhr.spec.whatwg.org/
connected
Creating Coursetro
Creating
> h [eth] _ transactionHash: null, add: "78f5325431c", abi: Array(3), setInstructor
Inside Coursetro.setInstructor
Inside error
> (2) ["Bruce", X]
> Coursetro.getInstructor()
> (2) ["Bruce", X]
> Coursetro.setInstructor('Brutis', 44)
> Uncaught Error: invalid address
at u (web3.min.js:1)
at InputTransactionFormatter (web3.js:1)
at add: (web3.min.js:1)
at Array.map (<anonymous>)
at l.formatInput (web3.min.js:1)
at t._onTxSend (web3.js:1)
at t._sendTransaction (web3.js:1)
at c.sendTransaction (web3.min.js:1)
at c.execute (web3.min.js:1)
at <anonymous>:1:11
```

Type here to search

12:14 PM 3/30/2020

10. Notice our 2 functions! `getInstructor` and `setInstructor`.

11. If you want, in the console window within the inspector, you can type:

```
> Coursetro.setInstructor('Brutis', 44) // Hit Enter
"0x894..." // This is the response (address)

> Coursetro.getInstructor() // Hit Enter
(2) ["brutis", e] // An array containing our data
```

File | file:///D:/nodejs/coursetro-eth/index.html

Coursetro Instructor

Bruce (40 years old)

Instructor Name

Instructor Age

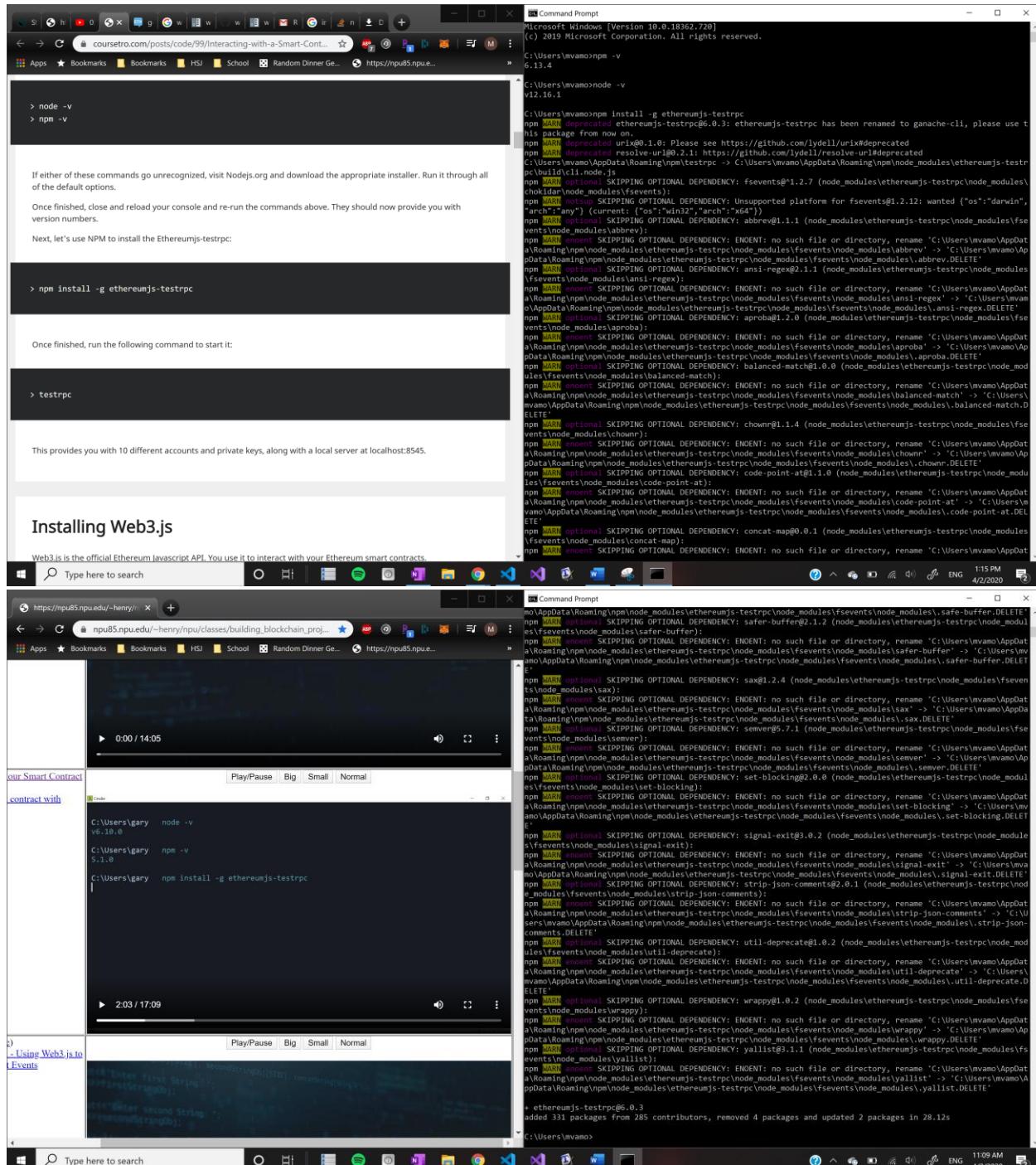
Update Instructor

Console Output:

```
[Deprecation] Synchronous XMLHttpRequest is deprecated because of its detrimental experience. For more help, check https://xhr.spec.whatwg.org/
connected
Creating Coursetro
Creating
> h [eth] _ transactionHash: null, add: "78f5325431c", abi: Array(3), setInstructor
Inside Coursetro.setInstructor
Inside error
> (2) ["Bruce", X]
> Coursetro.getInstructor()
> (2) ["Bruce", X]
> Coursetro.setInstructor('Brutis', 44)
> Uncaught Error: invalid address
at u (web3.min.js:1)
at InputTransactionFormatter (web3.js:1)
at add: (web3.min.js:1)
at Array.map (<anonymous>)
at l.formatInput (web3.min.js:1)
at t._onTxSend (web3.js:1)
at t._sendTransaction (web3.js:1)
at c.sendTransaction (web3.min.js:1)
at c.execute (web3.min.js:1)
at <anonymous>:1:11
```

Type here to search

12:14 PM 3/30/2020



Second attempt:

Administrator: Command Prompt

```
C:\Users\mvamo\coursetro-eth>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.

Press ^C at any time to quit.
package name: (coursetro-eth)
version: (0.0.0)
description:
git repository:
repository:
author:
license: (ISC)
About to write to C:\Users\mvamo\coursetro-eth\package.json:

{
  "name": "coursetro-eth",
  "version": "0.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "web3": "^1.2.6"
  },
  "devDependencies": {},
  "description": ""
}

Is this OK? (yes)
```

Just do
npm install web3

ateebahmed commented on Aug 30, 2018

Just do
npm install web3

@leonprou tried it but was unsuccessful.

nivida added the support label on Nov 28, 2018

Huanzhang89 commented on Nov 30, 2018

Hi @nil3sh99 , like @leonprou suggested npm install web3 should work (Please refer to the readme at <https://github.com/ethereum/web3.js>)

@ateebahmed Could you elaborate on the issue? What operating system are you running on, also what versions of node and npm ?

Some error logs would also be helpful to pin-point the exact issue you are experiencing.

Type here to search

https://npn85.npu.edu/~henry/ https://github.com/ethereum/web3.js/issues/1894

03. Web3.js Tutorial - Att... https://npn85.npu.edu...

Open up your preferred code editor (I use Visual Studio Code) with the project folder we created. Here, you'll notice a node_modules folder, which includes web3 that we installed via npm earlier.

Let's create an index.html in the project folder:

We're not going to create anything too fancy in terms of a UI, but we'll have some limited CSS, and a UI that consists of a place that retrieves the Instructor's name and age from the `getInstructor()` function, and a form with 2 input fields for a name and age, which will be set via jQuery from 2 input textfields.

To get started, paste the following contents into the empty index.html file:

```
<!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>

  link rel="stylesheet" type="text/css" href="main.css">

  script src="./node_modules/web3/dist/web3.min.js"></script>
```

```
</head>
body>
  div class="container">

    h1>Coursetro Instructor</h1>

    h2 id="instructor"></h2>

    label for="name" class="col-lg-2 control-label">Instructor Name</label>
    input id="name" type="text">

    label for="name" class="col-lg-2 control-label">Instructor Age</label>
    input id="age" type="text">

    button id="button">Update Instructor</button>
```

Welcome index.html

File Edit Selection View Go Run Terminal Help index.html - coursetro-eth - Visual Studio Code

EXPLORER OPEN EDITOR... 1 UNSAVED

index.html

node_modules

npm-debug.log

package-lock.json

package.json

LN 39 Col 8 Spaces 4 UTF-8 CRLF HTML R

11:18 AM 4/2/2020

File Edit Selection View Go Run Terminal Help index.html - coursetro-eth - Visual Studio Code

EXPLORER OPEN EDITOR... 1 UNSAVED

index.html

node_modules

npm-debug.log

package-lock.json

package.json

LN 39 Col 8 Spaces 4 UTF-8 CRLF HTML R

11:26 AM 4/2/2020

As you can see, we're referencing a `main.css` file, so create that file and paste in the following rulesets real quickly:

```

body {
    background-color:#F0F0F0;
    padding: 2em;
    font-family: 'Raleway','Source Sans Pro', 'Arial';
}

.container {
    width: 50%;
    margin: 0 auto;
}

label {
    display:block;
    margin-bottom:10px;
}

input {
    padding:10px;
    width: 50%;
    margin-bottom: 1em;
}

button {
    margin: 2em 0;
    padding: 1em 4em;
    display:block;
}

.instructor {
    padding:1em;
    background-color:#fff;
    margin: 1em 0;
}

```

Save it.

Using Web3.js to Connect & Interact with the Smart Contract

Going back to the `index.html`, at the bottom of the file we have an empty `<script>` tag. This is where we will write the necessary code to work with our smart contract.

Normally I would never use jQuery (I'm a big Angular fan), but this keeps things more simple.

In the head tags, we're already importing the Web3.js library, so now, let's use it to connect to our testrpc client:

```

<script>

    if (typeof web3 !== 'undefined') {
        web3 = new Web3(web3.currentProvider);
    } else {
        // set the provider you want from Web3.providers
        web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:8545"));
    }

</script>

```

This code comes directly from the [Web3.js Github page](#).

It's saying that if `web3` is not undefined, then we'll use that as our provider. If it's undefined (else), we can manually specify the provider ourselves.

You may be wondering, how would `web3` be defined? Well, if you're using the Chrome extension [MetaMask](#) (which we will use later in this course) or an Ethereum browser like [Mist](#), the provider is automatically injected.

```

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>

<script>
    if (typeof web3 !== 'undefined') {
        web3 = new Web3(web3.currentProvider);
    } else {
        // set the provider you want from Web3.providers
        web3 = new Web3(new Web3.providers.HttpProvider(
            "http://localhost:8545"));
    }
</script>

```

This code comes directly from the [Web3.js Github page](#).

It's saying that if web3 is not defined, then we'll use that as our provider. If it's undefined (else), we can manually specify the provider ourselves.

You may be wondering, how would web3 be defined? Well, if you're using the Chrome extension [MetaMask](#) (which we will use later in this course) or an Ethereum browser like [Mist](#), the provider is automatically injected.

Next, we have to specify a default ethereum account to use through the `web3.eth.defaultAccount` method:

```

<script>
  // Previous if/else statement removed for brevity
  web3.eth.defaultAccount = web3.eth.accounts[0];
</script>

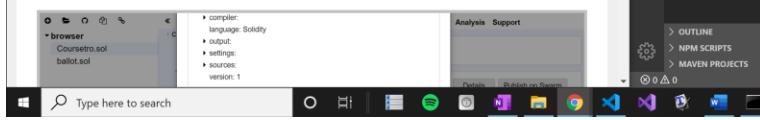
```

Remember when we ran the `testrpc` console command? It provided us with 10 accounts. We're simply choosing the first account here to use.

Next, need to use the `web3.eth.contract()` method to initialize (or create) the contract on an address. It accepts one parameter, which is referred to as the ABI (Application Binary Interface).

This ABI allows you to call functions and receive data from your smart contract.

If you switch back to the Remix IDE, click on the [Compile tab](#) and click [Details](#). Scroll down until you see the [Interface - ABI](#) section and click the copy icon as shown below:



Going back to `index.html` paste the following code:

```

<script>
  // Previous if/else statement removed for brevity
  web3.eth.defaultAccount = web3.eth.accounts[0];
  var CoursetroContract = web3.eth.contract(PASTE ABI HERE!);
</script>

```

Great. Now that we have the interface for interacting with our contract through the `CoursetroContract` variable, the last thing to do is to define the actual contract address.

VS Code screenshot showing the completed `index.html` file with the ABI pasted in:

```

<script>
  // Previous if/else statement removed for brevity
  web3.eth.defaultAccount = web3.eth.accounts[0];
  var CoursetroContract = web3.eth.contract([
    {
      "constant": false,
      "inputs": [
        {
          "name": "_fName",
          "type": "string"
        },
        {
          "name": "_age",
          "type": "uint256"
        }
      ],
      "name": "setInstructor",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [],
      "name": "getInstructor",
      "outputs": [
        {
          "name": "",
          "type": "string"
        },
        {
          "name": "",
          "type": "uint256"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    }
  ]);
</script>

```

Great. Now that we have the interface for interacting with our contract through the `CoursetroContract` variable, the last thing to do is to define the actual contract address.

We used Remix to create the contract earlier, and it has an associated address.

Go back to Remix and click the `Run` tab, and click on the copy icon next to the contract that we created earlier on the right column.

Back in `index.html` add the following line:

```
<script>
    // Previous if/else statement removed for brevity
    web3.eth.defaultAccount = web3.eth.accounts[0];
    var CoursetroContract = web3.eth.contract(YOUR ABI);
    var Coursetro = CoursetroContract.at('PASTE CONTRACT ADDRESS HERE');
    console.log(Coursetro);
</script>
```

Great. Let's save this, and then (in Visual Studio Code) you can right-click on the `index.html` and Reveal in Explorer. Double click the `index.html` to run it in the browser.

CTRL-SHIFT-I (i) will show the console. You will see something similar to the following:

The browser console output is as follows:

```
h {_eth: r, transactionHash: null, address: "0x8f63089ffc48a82755e4717738a0ca796cd0b51d"
  abi: (2) [{...}, {...}]
  address: "0x8f63089ffc48a82755e4717738a0ca796cd0b51d"
  allEvents: f ()
  getInstructor: f ()
```

CTRL-SHIFT-I (i) will show the console. You will see something similar to the following:

The browser console output is as follows:

```
Uncaught TypeError: web3.eth.contract is not a function
at index.html:44
```

This video is the 3rd lesson from my 100% free course:

03. Web3.js Tutorial - Attach a GUI to your Ethereum Smart Contract

86,374 views • Oct 24, 2017

DesignCourse 549K subscribers

Free Course Page (In development): <https://goo.gl/EiKpPP>
Written tutorial of this video: <https://goo.gl/RdXuz9>
<https://coursetro.com>

Type here to search

The screenshot shows a dual-monitor setup. The left monitor displays a Stack Exchange question page for a Web3.js tutorial. The right monitor displays Visual Studio Code (VS Code) with an open file named 'index.html'.

Stack Exchange Question (Left Monitor):

https://ethereum.stackexchange.com/questions/22983/web3-eth-contract-deploy-is-not-a-function

03. Web3.js Tutorial - Interacting with Ethereum Contracts

I'm using this version itself and yet getting error: contract.deploy is not a function. Any ideas? - Ani Nov 16 '18 at 12:12

share improve this answer follow answered Apr 19 '18 at 11:10 Ftila Abadi 111 2

Requires `c` capital of contract(Here: `web3.eth.Contract`). So it should be

```
0 var contractAbi = new web3.eth.Contract(abi)
```

look that is `c` and not `C` in: `web3.eth.Contract`

share improved Dec 3 '19 at 16:26 shane 5,406 0 15 37 follow answered Dec 3 '19 at 16:07 sngq 43 4

add a comment

Visual Studio Code (Right Monitor):

File Edit Selection View Go Run Terminal Help index.html - coursetro-eth - Visual Studio Code

index.html # main.css

```
</div>
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>
<script>
  if (typeof web3 === 'undefined') {
    web3 = new Web3(web3.currentProvider);
  } else {
    // set the provider you want from Web3.providers
    web3 = new Web3(new Web3.providers.HttpProvider(
      "http://localhost:8545"
    ));
  }
  web3.eth.defaultAccount = web3.eth.accounts[0];
  var CoursetroContract = new web3.eth.Contract([
    {
      "constant": false,
      "inputs": [
        {
          "name": "_fName",
          "type": "string"
        },
        {
          "name": "_age",
          "type": "uint256"
        }
      ],
      "name": "setInstructor",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [],
      "name": "getInstructor",
      "outputs": [
        {
          "name": "",
          "type": "string"
        }
      ]
    }
  ],
```

Line numbers 30 to 70 are visible on the left side of the code editor. The status bar at the bottom of VS Code shows: In 44, Col 54 Spaces: 4 UTF-8 CRLF HTML 11:46 AM 4/2/2020

Third attempt (Successful):

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** New Tab - Document
- Address Bar:** File | C:/Users/mvamo/AppData/Local/Packages/CanonicalGroupLimited.Ubuntu18.04onWindows_79rhkp1fdgsc/LocalState/rootfs/home/mva456/coursetro-eth/index.html
- Content Area:** A form titled "Coursetro Instructor" with fields:
 - Instructor Name: 123
 - Instructor Age: 34A button labeled "Update Instructor" is visible below the form.
- Console Tab:** Shows developer tools with the following log output:

```
GET file:///C:/Users/mvamo/AppData/Local/Packages/CanonicalGroupLimited.Ubuntu18.04onWindows_79rhkp1fdgsc/LocalState/rootfs/home/mva456/coursetro-eth/main_1ab3.. index.html:8
CSS net::ERR_FILE_NOT_FOUND
⚠ [Deprecation] Synchronous XMLHttpRequest on the main thread is deprecated because of its detrimental effects to the end user's experience. For more help, check https://xhr.spec.whatwg.org/.
connecter
Creating Coursetro
▶ h {eth: _, transactionHash: null, address: "0xb36d7ae5e0ffcf7869631521d8d5dd3ab9a30b0", abi: Array(3), setInstructor: f, ...}
Inside Coursetro.getInstructor
Inside terror
▶ (2) ["123", X]
Inside click
>
```
- Taskbar:** Shows the Windows Start button, a search bar, and a taskbar with various pinned icons.

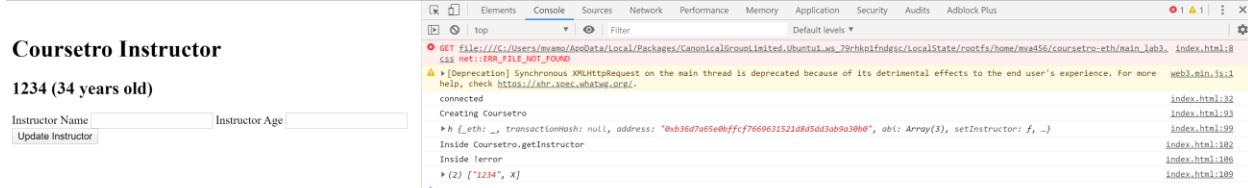
The screenshot shows the Ethereum IDE interface in Remix. The top navigation bar includes tabs for 'Compile', 'Run', 'Analysis', 'Testing', 'Debugger', 'Settings', and 'Support'. The main area displays the Solidity code for 'Coursetro.sol':

```

pragma solidity ^0.4.18;
contract Coursetro {
    string _name;
    uint _age;
    function () public {
        _name = "Gary";
        _age = 32;
    }
    // setInstructor accepts 2 parameters, _name and _age.
    // Once called, we set our string _name to the returned
    // value and set _age with _age.
    function setInstructor(string _name, uint _age) public {
        _name = _name;
        _age = _age;
    }
    // The getInstructor() function is defined as being
    // constant, and it returns a string and a uint. This is where
    // we return the _name and _age variable once it's called.
    function getInstructor() public constant returns (string, uint) {
        return (_name, _age);
    }
}

```

The right panel shows the environment settings: Web3 Provider (Custom), Account (0x2c4...c97ca), Gas limit (3000000), and Value (0 wei). Below this is a 'Transactions recorded' section and a 'Deployed Contracts' section listing 'Coursetro' at address 0xb36...a30b0.





Reason for error:

Your commands have mixed Web3 1.0.x beta commands (your first line) with Web3 0.x.x commands (second line).

See below for two versions on how you can get this to work, assuming your change string function is "changeString":

Web3 1.0.x beta | [web3.eth.Contract](#)

```
var contractInstance = new web3.eth.Contract(abi, _contractAddress);

// use .send() to send the transaction to the blockchain and change state
contractInstance.methods.changeString('new string').send({ from: _yourAccount })
  .then(receipt => { /** some action **/ });
```

Web3 0.x.x released | [web3.eth.contract](#)

```
var MyContract = web3.eth.contract(abi);
var contractInstance = MyContract.at(_address);
contractInstance.changeString('new string',
  { from: _myAccount, gas: _gasLimit },
  (err, res) => { /** callback **/ }
);
```

Node, Npm, Web3 version (Command Prompt on Windows 10):

```
C:\Users\mvamo\coursetro-eth>node -v
v13.12.0
```

```
C:\Users\mvamo\coursetro-eth>npm -v
6.14.4
```

```
C:\Users\mvamo\coursetro-eth>npm ls web3
coursetro-eth@0.0.0 C:\Users\mvamo\coursetro-eth
`-- web3@1.2.6
```

Solution:

1. Install Ubuntu from Microsoft Store
2. Look for Turn Windows features on or off from the Start Menu or Control Panel
3. Add a checkmark on Windows Subsystem for Linux and click OK
4. Search Xming on your browser and install
5. Open XLaunch and click Next all the way to the end
6. If your antivirus is Comodo, disable Auto-Containment permanently and put on silent
7. Open Ubuntu and create username and password
8. On Ubuntu, type

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install vim-gtk
export DISPLAY=:0
```
9. Follow lab 3 video instructions
10. When creating/editing index.html and main.css, type geany on Ubuntu
 - a. Editing features are similar to Visual Studio Code
11. When opening index.html on browser, open File Explorer and go to
`C:\Users\YourUsername\AppData\Local\Packages\CanonicalGroupLimited.Ubuntu18.04onWindows_79rhkp1fndgsc\LocalState\rootfs\home\UbuntuUsername\coursetro-eth`