

①

66
Movies → The imitation game 99
A genius

1/1

22 15 Oct 2022

Agenda -

What?
Why?
How?

- Ethereum intro
- Beginning of Ethereum
- Wallet
- Introduction to dapp
- Drawback of bitcoin

Alan Turing

Why moving bitcoin to Ethereum

Turing

Any language learn Turing is called
Turing completeness (C++, Java, JavaScript)

1) Draw back of Bitcoin

- Lack of Turing completeness
- Value Blind
- Lack of State
- Blockchain-blind

Bitcoin → ~~loop~~ why Turing completeness
building application.

Turing completeness - is Good or Bad?

- Value Blind - not read values
- Blockchain Blind -

UTXO's - Unspent transaction output.

②

1/1

Solution of All is "Ethereum"

Ethereum

Component

- World Computer
- open Source
- Smart Contract
- ~~ethereum~~ Ether

↑
Currency

"Vitalik Buterin"

Dr. Gavin Wood

Solidity

Polka dot

World Computer - multiple computer

Smart Contract -

- Bit coin is Blockchain protocol and Cryptocurrency
- Ether is platform

★ Web2 pay - Amazon Server

★ Web3 " , " Miners.

Turing completeness Why Bad?

operation consume some pay fee - Rs 1

loops

5 time loop
7 operation
+ 5 ⇒ 12 operation

Rs - 7
Rs - 12

infinite loop ∞ Rs

GAS?

③

G Wei -

Ether - 10^{18}

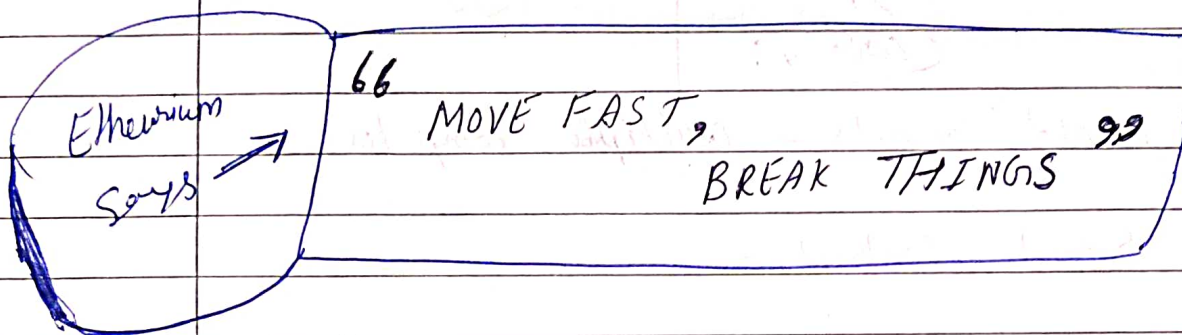
1/1

#Dapp — Decentralized application
or D-Eth application (latin)

example

Dapp is used instagram installed on ethereum platform.

- Smart Contract — Code
- Web frontend UI



Ethereum Account

— Fields

- EOA [
- Nonce
 - Ether balance
 - Contract code
 - Storage

Two types of Account

- External owned account — (managed private key)
- Contract Accounts.

Meta Mask (Wallet) (extension

— Web Wallet

ethereum mainnet

④

NodeJS

⇒ How to create own web server?

Agenda →

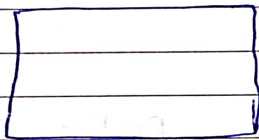
Web server

1) Module - http → url

http → Hyper text transfer Protocol

Ethereum account

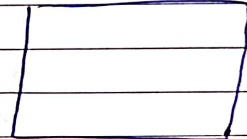
EoA -



client

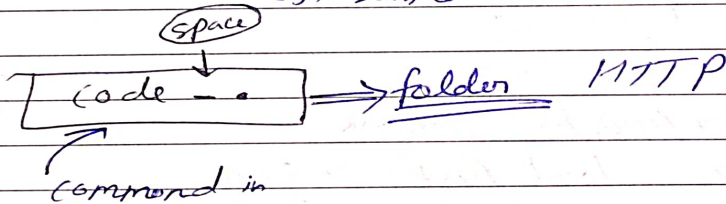
India

→ USA

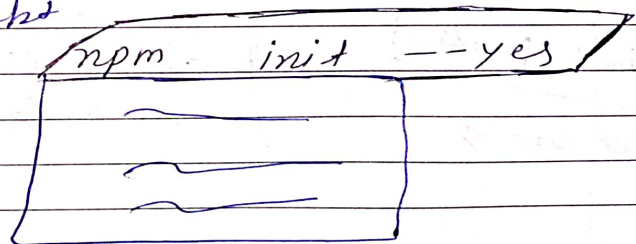


request

established → client server communication



⇒ Command prompt



package.json - file

5

1/1

Server.js

```
const http = require('http');
```

~~// create server~~

// createServer - method - create http server

// www.google.com/inewton

```
const server = http.createServer((req, res) =>
```

```
{ if (req.url === '/') {
```

```
  res.write("hello, this is http  
  server response\n");
```

```
  res.end();
```

```
}
```

Port - Communication

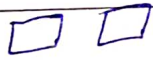
↳ Sending & receiving data

A-B

B

listening forever

Client - Server



file transfer

port → Computer Network

DOM → local host → 5000
3000

ip address → domain

127.0.0.1

local web hosting

loop back ip address

web app running

```
server.listen(3000);
```

```
console.log('listening on port 3000');
```

```
if (req.url === '/inewton/courses') {
```

```
  res.write(JSON.stringify(['Blockchain', 'data  
  science', 'web devops']));
```

```
  res.end();
```

```
};
```

⑥

__/__/__

Right click console side Network
* check speed

get — url

post — body of html

API (Application Programming Interface)

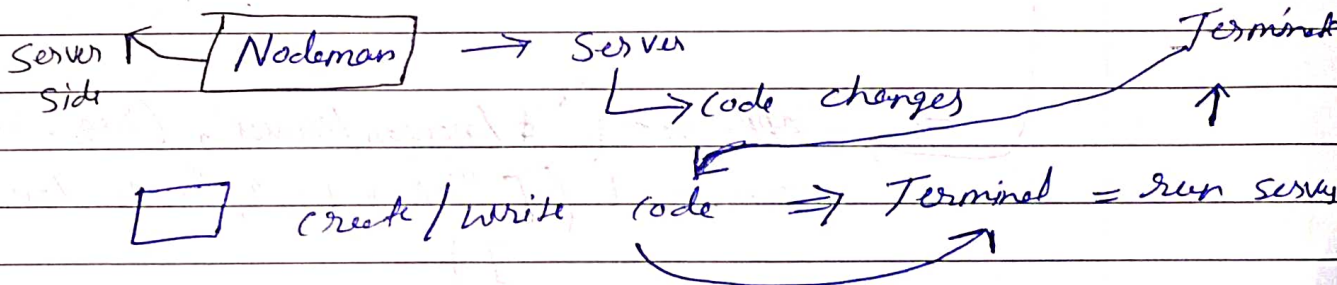
Interface front between frontend & backend.

ExpressJS

expressJS is nodeJS web application framework.

① / terminal | npm i -g express --save ⇒ command

② npm install -g -nodemon

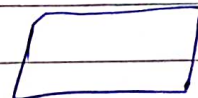


live Server → it only work HTML

Azure

HTML
CSS

Javascript



⑦

`api.js` `const express = require('express');`

`const app = express();`

// `app` is an object used to interact with express functional.

`app.get('/', (req, res) => {`

`res.send("hello");`

`}) ;`

`const port = 6000;`

`app.listen(6000, () => console.log('listening on port ' + port));`

`app.get('/neuron/courses', (req, res) => {
res.send(["Blockchain", "web dev", "devops"]);
})`