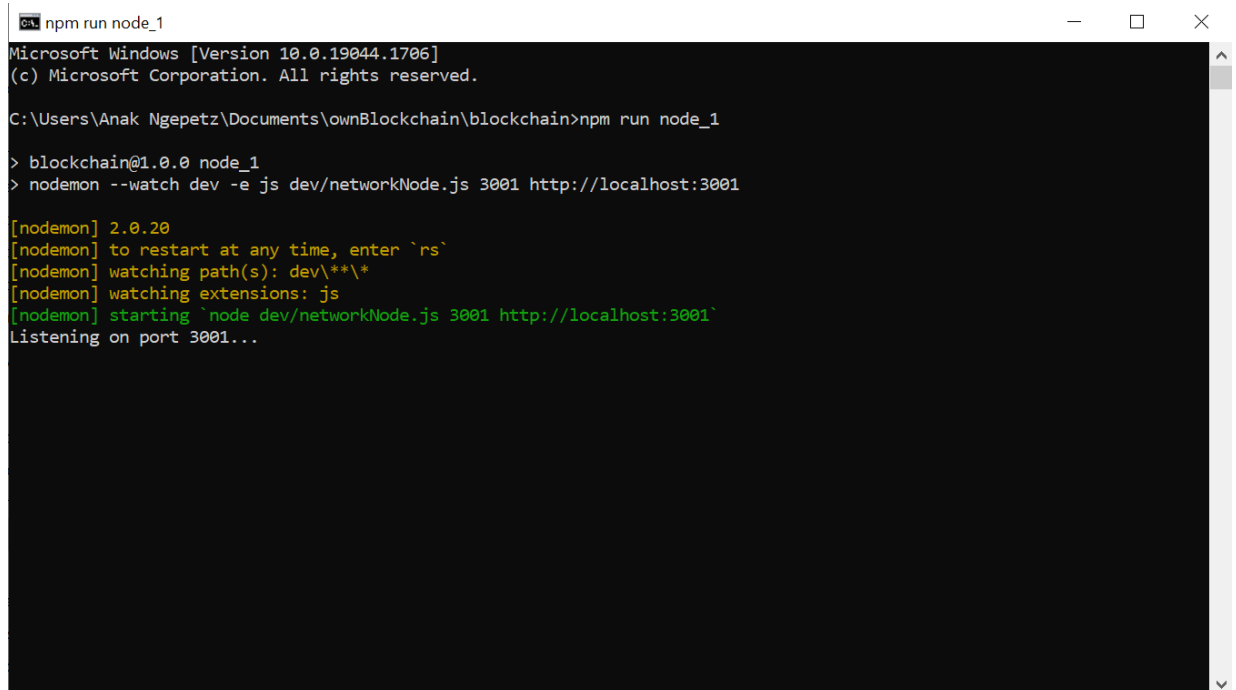


CARA PAKAI

Saya asumsikan yang *clone* sudah tahu *basic* dari konsep Blockchain. Install juga Postman (aplikasi tes API) dan Visual Studio Code (aplikasi edit Javascript) sebagai pendukung *project* ini. Jangan lupa untuk melakukan npm install dan *dependencies* yang diperlukan.

SERVER CONNECTION

1. Di dunia nyata Blockchain berkonsep *decentralized* yang terdiri dari berbagai perangkat keras. Di *project* ini karena hanya menggunakan 1 perangkat keras, jadi saya bagi *server* nya jadi 5 dengan asumsi kelima *server* tersebut adalah 5 perangkat keras di dunia nyata.
2. Koneksikan *server* ke *localhost* lewat Windows Command Prompt dalam *folder* Blockchain sendiri menggunakan fungsi npm run node_1. Kemudian buka lagi Windows Command Prompt **yang baru** , lakukan hingga node_5. Maka akan seperti:



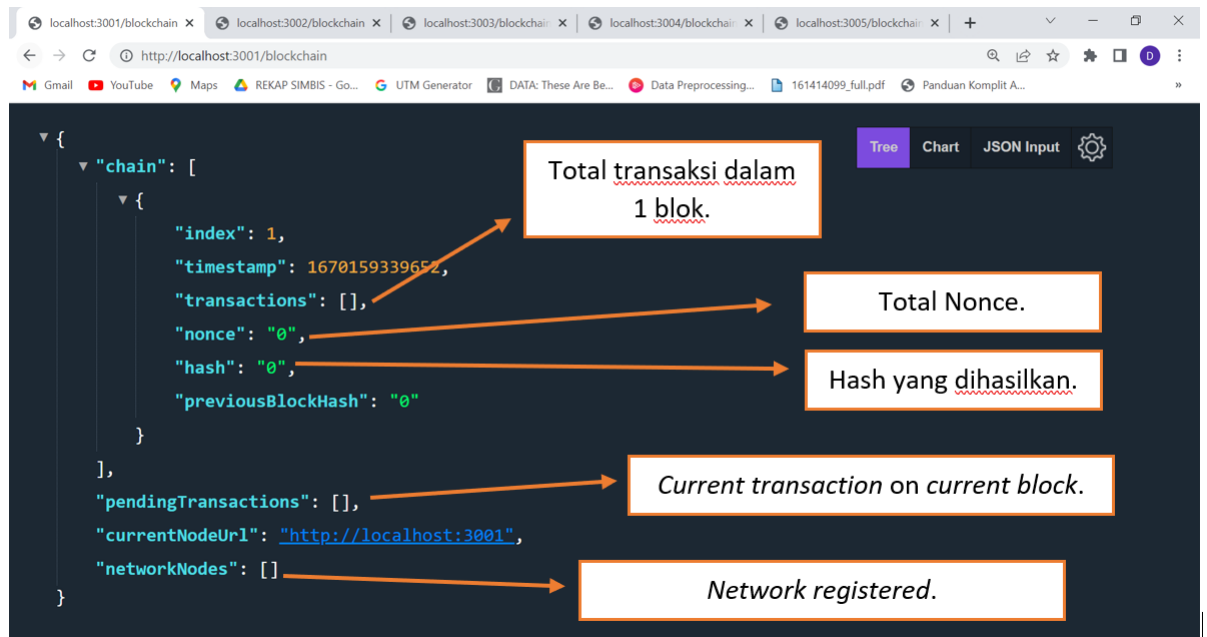
```
npm run node_1
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Anak Ngepetz\Documents\ownBlockchain\blockchain>npm run node_1

> blockchain@1.0.0 node_1
> nodemon --watch dev -e js dev/networkNode.js 3001 http://localhost:3001

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): dev/**/*
[nodemon] watching extensions: js
[nodemon] starting `node dev/networkNode.js 3001 http://localhost:3001`
Listening on port 3001...
```

3. Buka di *browser* dengan link <http://localhost:3001/blockchain> . Buka di *browser* yang sama tetapi *tab* baru dengan link <http://localhost:3002/blockchain> dan lakukan hingga <http://localhost:3005/blockchain> . Maka akan seperti:

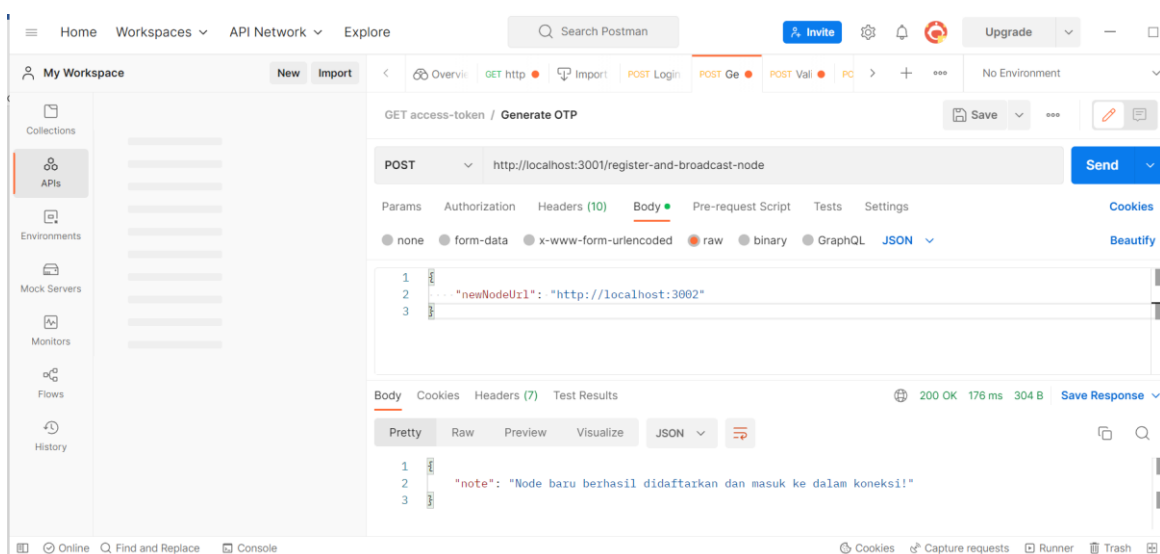


REGISTER THE NETWORK BETWEEN LOCALHOST:3001 UNTIL LOCALHOST:3002

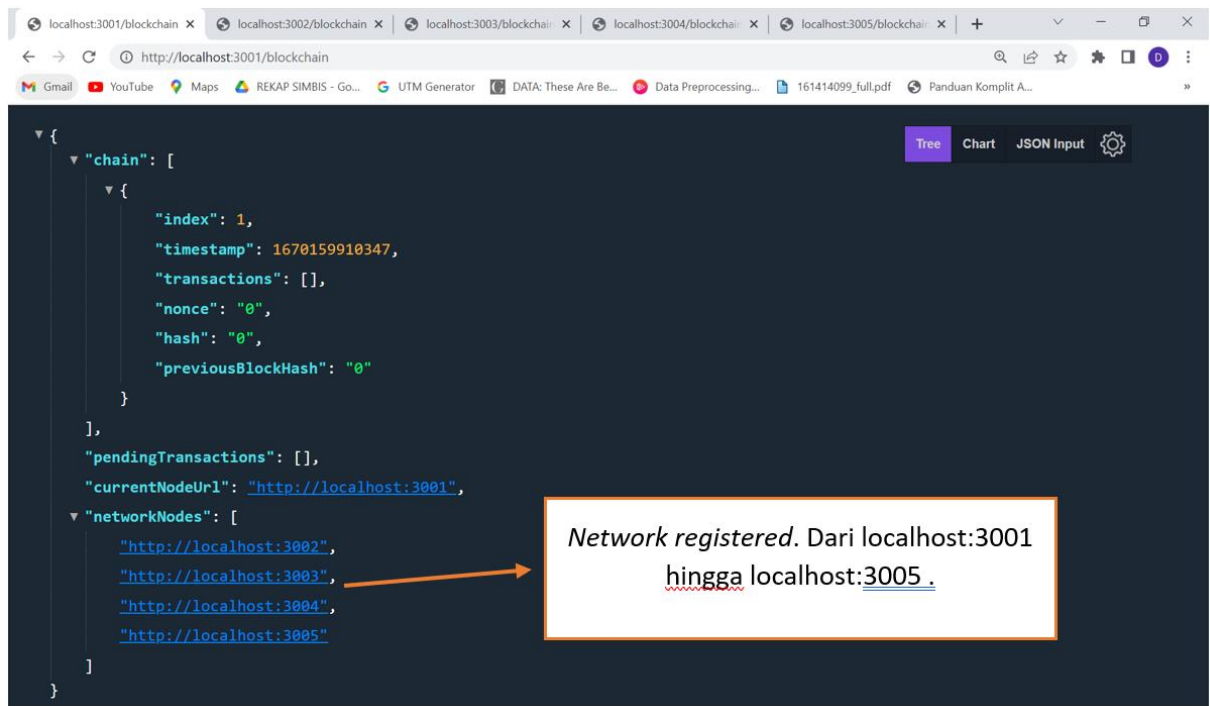
Sekarang buka post man API, ketik link <http://localhost:3001/register-and-broadcast-node> dengan metode POST. Lalu buka *tab body* ketik *script* dalam format JSON seperti dibawah ini

```
{
  ... "newNodeUrl": "http://localhost:3002"
}
```

Lakukan juga yang localhost:3003 hingga localhost:3005. Maka akan seperti ini:

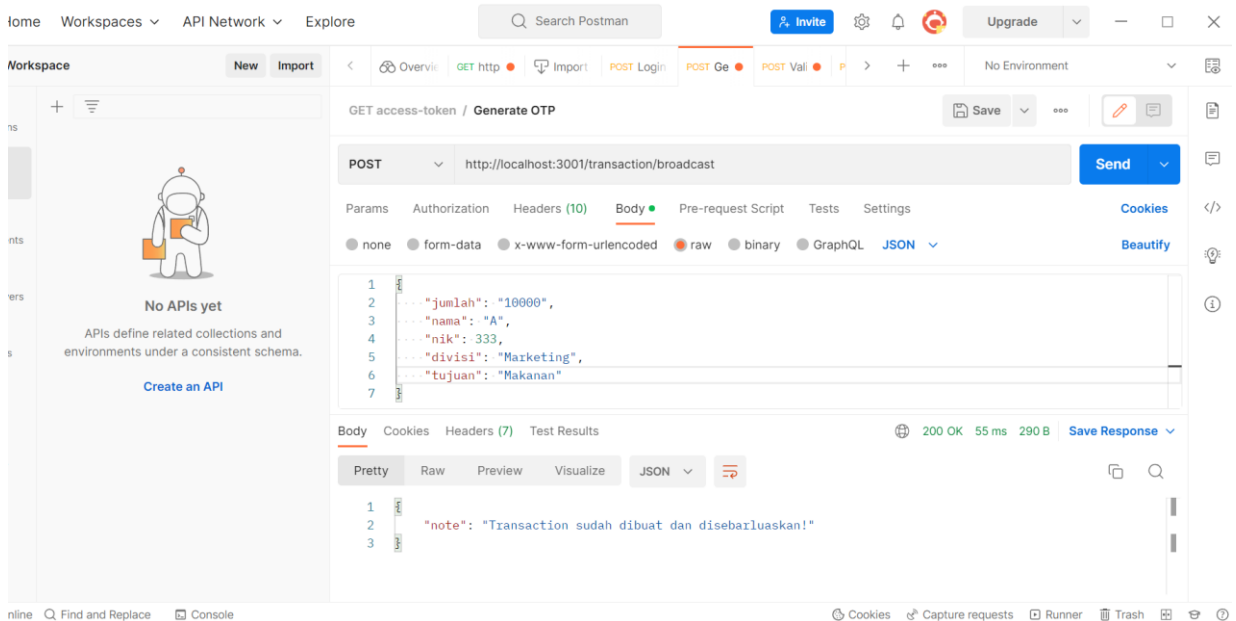


Ketika membuka link <http://localhost:3001/blockchain> lagi dalam *browser*, maka akan seperti ini:

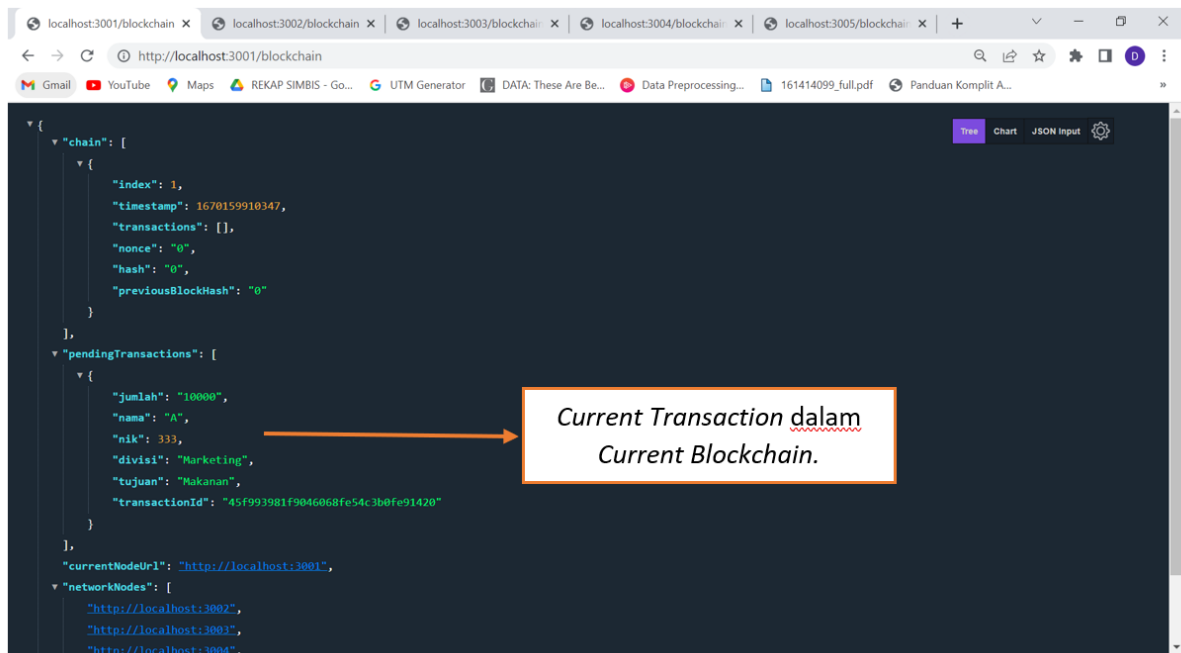


TRY TO ADD TRANSACTION

Langkah ini saya asumsikan penerapannya dalam sebuah pencatatan transaksi perusahaan. Anggaplah seorang karyawan bernama “A” ingin melakukan reimburse pembelian makanan dan diminta untuk *upload bill* serta jumlahnya ke *ledger* dalam *blockchain*. Maka yang dilakukan ketik link <http://localhost:3001/transaction/broadcast> (dengan metode POST) dan masukan informasi yang dibutuhkan yaitu jumlah, nama, nik, divisi, dan tujuan. Maka akan seperti ini:

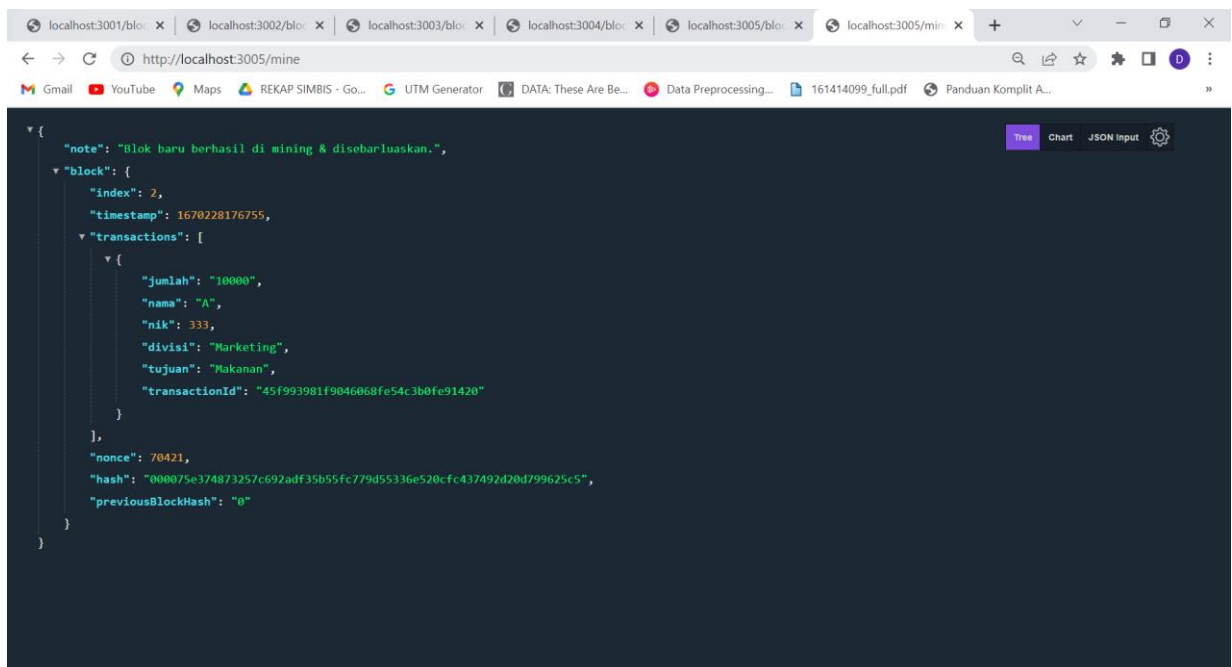


Ketika membuka link <http://localhost:3001/blockchain> lagi dalam *browser*, maka akan seperti ini:



HOW TO MINE

Proses *mining* sendiri dalam aplikasi dunia nyata dilakukan secara berkala bila studi kasusnya merupakan pencatatan transaksi. Untuk melakukan proses *mining* sendiri, silahkan bukan browser dan ketikkan link <http://localhost:3001/mine> . Proses ini akan *mine* blok baru dan akan memakan waktu yang cukup lama. Maka akan seperti ini:



Ketika membuka link <http://localhost:3001/blockchain> lagi dalam *browser*, maka akan seperti ini:

localhost:3001/bloc x localhost:3002/bloc x localhost:3003/bloc x localhost:3004/bloc x localhost:3005/bloc x localhost:3005/min x +

http://localhost:3001/blockchain

Gmail YouTube Maps REKAP SIMBIS - Go... UTM Generator DATA: These Are Be... Data Preprocessing... 161414099_full.pdf Panduan Komplit A...

```
{
  "chain": [
    {
      "index": 1,
      "timestamp": 1670159910347,
      "transactions": [],
      "nonce": "0",
      "hash": "0",
      "previousBlockHash": "0"
    },
    {
      "index": 2,
      "timestamp": 1670228176755,
      "transactions": [
        {
          "jumlah": "10000",
          "nama": "A",
          "nik": 333,
          "divisi": "Marketing",
          "tujuan": "Makanan",
          "transactionId": "45f993981f9046068fe54c3b0fe91420"
        }
      ],
      "nonce": 70421,
      "hash": "000075e374873257c692adf35b55fc779d55336e520cfc437492d20d799625c5",
      "previousBlockHash": "0"
    }
  ]
}
```