1 – Şimdi burada ne yaptık? “http” modülünü import ediyoruz. Daha sonra host ve port u oluşturuyoruz. Host, bilgisayarda kullanacağımız adresin ne olduğunu yazıyoruz. Local server ı belirliyoruz. Localden bağlanacağımız için 127.0.0.1 ‘i yazdık. Port, server a bağlanacağımız port numarasını belirliyoruz. Daha sonra bu http modulü ile bir server oluşturuyoruz. Bu server’ın içerisinde, res olarak; status kodunu, header’ını, end değerini girip-gönderiyoruz.

Şimdi http-host-port işlemlerini tamamladıktan sonra, sırada bu server’ı dinlememiz lazım. Bu yüzden en aşağıda “listen” ile çağırıp, port-host ‘u parametre olarak gönderiyoruz. Son olarak da console.log mesajı yayınlıyoruz.

const http = require("http");

const host = "127.0.0.1";

const port = 3002;

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

*res*.statusCode = 200;

*res*.setHeader("Content-type", "text/html");

*res*.end("this is node server");

});

server.listen(port, host, () => {

  console.log("server started ", port);

});

2- Burada ise, http modulü ile yaptığımız işlerin aynısını express framework’ü ile daha kolay bir şekilde yapabiliyoruz. Express modulünü import ediyoruz. Sonra express() i çağırıp, server.get ile (/..) url’ye gelen req-res lere göre işlemlerimiz yapıyoruz.

const express = require("express");

const data = require("./data");

const port = 3002;

const server = express();

server.get("/", (*\_req*, *res*) => {

*res*.send("express added and server is ready");

});

server.get("/data", (*\_req*, *res*) => {

*res*.status(200).json(data);

});

server.get("/data/:id", (*req*, *res*) => {

  const { id } = *req*.params;

  const trump = data.find((*item*) => *item*.id === parseInt(id));

  if (trump) {

*res*.status(200).json(trump);

  } else {

*res*.status(404).send("could not find it");

  }

});

server.listen(port, () => {

  console.log(port, " server is listening");

});

3- server.get(/users) ile users sayfasına istek geldiği zaman nasıl bir yönlendirme yapacağımızı belirliyoruz. bu şekilde routing yapabiliyoruz. /users adresine req geldiği (istek yapıldığı) zaman buna res olarak çeşitli işlemler yapıyoruz. Örneğin;

*res*.status(200).json(data);

şeklinde status=200 olan ve json şeklinde data dosyasını gönderiyoruz.

  const { id } = *req*.params;

  const trump = data.find((*item*) => *item*.id === parseInt(id));

  if (trump) {

*res*.status(200).json(trump);

şeklinde params olarak gelen id’yi alıp bizim data listemizdeki id ile eşleşirse o id’nin ait olduğu datayı-kullanıcıyı res olarak dönebiliyoruz.

* url kısmına;

localhost:3002/users/1?name=www, şeklinde gelen url’nin içerisinden query olarak “?” key=value, şeklinde

req.query, objesini alabiliriz. Yani url ye yapılan query sorgusunu alıp-kullanabiliriz.

* url nin body kısmından da gelen datayı alabiliriz,

postman ile body kısmından, raw formunda JSON formatını seçip, JSON formatında mesajımızı yazıyoruz. Tabi bunu okumak için yani gelen res’i okumak için JSON formatını parse etmemiz lazım. Bunun için, şeklinde yapmamız lazım.

server.use(express.json());

3- Sırada middleware kısmı var;