 node’s built-in http module.

const http = require('http')//get moudle to http

const server = http.createServer(function (request, response) { //make a server , passing the request and response to the callback funtion

response.writeHead(200, { 'Content-Type': 'text/plain' }) / write a head perfix to the response socket

response.write('Hello, world') // data to the response

response.end(); //finish the request-response cycle

})

const port = 3000

server.listen(port, function () {

console.log(`Node server created at port ${port}`)// function make the server running ad listing to the port

})

const express = require("express");

const app = express();

app.get("/", function (req, res) {

  console.log("the server had been called from a clinet");

  res.send("here is the response");

});

const port = 3000;

app.listen(port, function () {

  console.log("server is listing now at port of ", port);

});

Basic req resp using express npm moudle

A listen function open the server to start listing

Get is the function active when someone make a get request to the server only

app.get("/", function (request, response) {

  console.log("Someone has come into the server. Brace yourselves.");

  response.send("Ending the cycle, thanks for visiting");

});

app.get("/maps", function (request, response) {

  response.send("Here's some stuff related to maps");

});

app.get("/shoobi", function (request, response) {

  response.send("This here is the shoobi \*route\*");

});

(“/”) is the root of Routes and it activate once you make localhost:portNumber

Others is just the / after the localhost and port

app.get("/landing/:username", function (req, res) {

  console.log("landing");

  res.send(`the user input ${req.params.username}`);

});

Make a parameters in the request

In url (localhost:portnumber/landing/input)

//////////

Optional query

We can make optional query by just (/?) and get it from (req.query)

SERVE A FILES (HTML CSS JAVASCRIPT)

1. Const path=require(“path’)
2. App.use(express.static(path.join(\_\_dirname,’dist’)));

Use is for middleware (like don’t do anything until you do thsis)

It make us serve a static files

const express = require('express')

const path = require('path')

const app = express()

app.use(express.static(path.join(\_\_dirname, 'dist')))

const port = 3000

app.listen(port, function () {

    console.log(`Running server on port ${port}`)

})

How to serve files to webpage

CRUD

Creating , Retreiving , updating , deleting

Post , get , update,delete;