**Express**

Express generator

Express

Start the app - set PORT=5000 && npm start

bikram-choudhury Github

repo - githubAPI

Import express module to use express

const express = require('express');

explore express methods

* Use
* Route

Routes

In ejs

viewengine -  in app.js, views

Template Engines - Used to render data from the server side

res.render(index,{title:'Express'}) - To render data

-> Will go to app.js , checks the views path checks the file in that path

-> Then checks the viewengine and sets the extension to that

->

In EJS file - For evaluating - <% %>

  For printing - <%= %>

  assigning values - <%= %>

Callback functions - middleware

Routes

Top to bottom appr

Middlewares- Top-down Goes to the next middleware in the scope when we give next()

Middleware - Control flow functions

Next - Callback argument

* Next() should always be the last statement in a middleware.
* Any statements within the middleware after next() will not be executed.
* The next() statement will call the next middleware in the scope until the request is resolved. If the request is not resolved in the scope, it’ll go to the parent middleware and so on.

router.route('/users')

.get((req, res,next) => {

req.users = "Wecome to users API";

next();

},(req, res) => {

res.send(req.users);

})

Fetching data – (From JSONPlaceholder )

Axios – Module used for access one server from another.

Promise object

Array.isArray(data) - To check if the data is array

Object.keys(data) - To check if the data is object

Controllers

Moving functions - from one module to another

moduleName.function

bind() - lazy execution

apply() - Need to pass array eager to execute

call() - eager to execute

**params**

Passing parameters to url in express

/:paramName

/:param

**Query**

**Promise –** takes a callback function as an argument which takes two objects resolve and reject as arguments.

* Promise is eager to execute

.**then**

.**catch**

**Circular json –** One response object cannot be added to another response object.

**Explore**

Promise hell

Promise series

Async package parallel

Promise.all – takes two arguments – 1. 2. Array of callbacks

Search the following in Bikram’s Repo

Waterfall

series