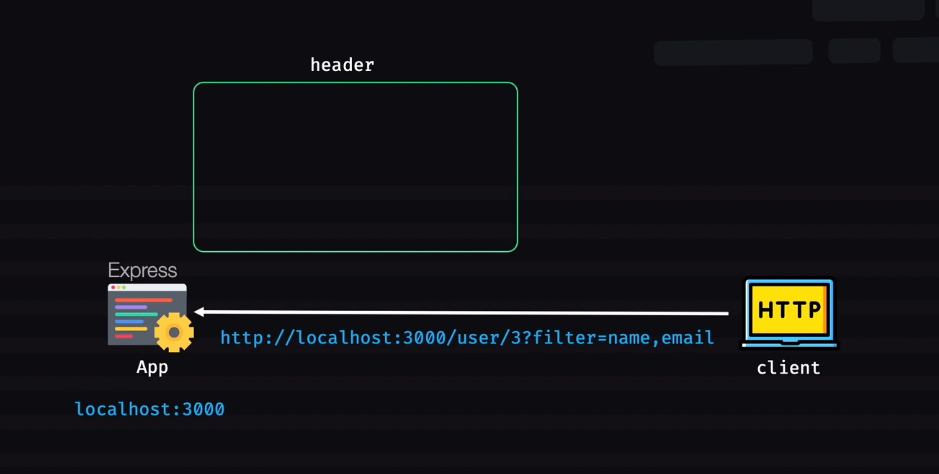
Ajke amra express js er ekta important aspect request object niye discuss korbo. Express ekta micro-framework jeta actually request and response niyei concerned.

Kono client jokhon server ke request kore tokhon etar mane holo kono ekta url e she hit kore ba ekta htttp protocol or https protocol er maddhome she server e request kore. Express e etake ‘req’ object diye define kora hoy. Ajke ei object er properties ebong methods niye kaj korbo.



Dhore nilam dan side er holud icon ta holo amader http client (browser or mobile app or postman). Ar left side e amader express app cholche 3000 port e. amra jeta run korchi amader local machine e. ekhon ei client ta amader express app e ekta request korlo. Tar mane kono ekta url (http://localhost:3000/user/3....,email) e hit korlo. Client ei request korar shomoy request er shathe besh kichu jinish diye dicche. Kichu jinish automatically chole ashce. Kichu jinish manually diye dicche. Automatically chole asha jinisher moddhe header propertiesgula (host, content-type etc) ache. Echara url er moddheo bivinno type er jinish thake. Jemon pic er url er 3 hocche parameter. Filter = name, email --- eta hocche query. Erokom bivinno amader request er shathe automatically chole ashche.

Core nodejs er req object ter ja ja property and method ache shegula express js eo kaj korbe. express core nodejs er req object ta ke just extend kore extra kichu property and method vore dise.

Req.baseurl

Index.js …

*const* express = *require*("express");

*const* app = *express*();

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.baseUrl);

  res.*send*("Hello World...");

});

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Nodemon index kore postman er url e localhost:3000/user/4 likhe get mehod e send korle console e req.baseUrl blank dekhacche. Er karon holo ami jokhoni kono express app create kori tokhon by default amar ekta root route create hoy. App object er upor base kore ekta route create hoise. Amar pura app ta holo ei ‘app’ ta. Kintu ei app charao amra subapp banate pari. Subapp er khetre base ta change hoye jabe. Ei ‘app’ er base ta ekdom root e . ekarone eta blank dicche. baseUrl check korar jonno ekta subapp adminRoute banai.

Index.js …

*const* express = *require*("express");

*const* app = *express*();

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*(req.baseUrl);

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.baseUrl);

  res.*send*("Hello World...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

adminRoute app er dashboard e get method e gele ekta CB call hoy jekhane request object er baseUrl ta console e dekha jay. Ekhon ami amar main ‘app’ keo bole dicchi tumi ‘/admin’ path e adminRoute app ta use koro.

Orthat eta hocche amar main ba root app. “localhost:3000/user/:id” dile amar ei route ta kaj korbe. onno dike jokhoni express ‘localhost:3000/admin/…’ pabe tokhoni bujhbe amar router change hoye gese. ‘/admin’ route e jete hobe. Orthat jodi ami ‘/admin/dashboard’ dei tahole amar adminRoute.get() er route ta kaj korbe. ekhon nodemon index kore postman e localhost:3000/admin/dashboard e get method e hit korle adminRoute.get() er baseUrl dekhacche “/admin”. So kono ekta route je subapp e mount hobe shei subapp er base path ta hocche baseUrl.

Req.originalUrl

Ager code baseUrl er jaygay originalUrl likhe dilam. Nodemon index kore get method e “localhost:3000/user/3?filter=name” url e hit korle originalUrl ashtese “/user/3?filter=name”. onnodike “localhost:3000/admin/dashboard” url e hit korle originalUrl ashtese “/admin/dashboard”. Ekhon adminRoute.get() ebong app.get() dui jayagay te req.url o diye dilam.

Index.js …

*const* express = *require*("express");

*const* app = *express*();

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*(req.originalUrl); *// /admin/dashboard*

  console.*log*(req.url); *// /dashboard*

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.originalUrl); *// /user/3?filter=name*

  console.*log*(req.url); *// /user/3?filter=name*

  res.*send*("Hello World...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Ekahne dekhte pacchi main app er khetre req.originalUrl ebong req.url same. But subapp er khetre req.url modified hoye geche. so req.url er upor depend kora jay na. so client actually kon url e hit korsilo sheta jante hole originalUrl use korbo.

Req.path

AdminRoute.get() er khetre “localhost:3000/admin/dashboard” url e hit korlam. Req.path ashlo “/dashboard”.

App.get() er khetre “localhost:3000/user/3?filter=name” url e hit korlam. Req.path ashlo “/user/3”

Index.js …

*const* express = *require*("express");

*const* app = *express*();

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*(req.path);

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.path);

  res.*send*("Hello World...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

So app.get() er khetre dekhlam query string gula thake na. sub app er khetre root path er porer part tuku pacchi.

Req.hostname and req.ip

Req.path shoriye req.hostname boshalam. App.get() ebong adminRoute.get() both case ei hostname pelam ‘localhost’. Ebong req.ip er belay dui jaygatei ‘::1’ pacchi.

Req.method and req.protocol

Both case ei ‘GET’ and ‘http’ show kortese.

Req.params

App.get() er amar request parameter holo ‘id’. Express amake req.params er moddhe ekta object dey. Shei object er moddhe id namok ekta property thakbe. tar value hobe amar set kora value.

Index.js …

*const* express = *require*("express");

*const* app = *express*();

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.params);

  console.*log*("-----------");

  res.*send*("Hello World...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Ekhon postman e get method e “localhost:3000/user/3?filter=name” url e hit korlam. Console e req.params hishebe pailam ekta object {id : ‘3’}. Ekhane ei 3 kintu ekta string.

Req.query

Index.js…

*const* express = *require*("express");

*const* app = *express*();

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.query);

  console.*log*("-----------");

  res.*send*("Hello World...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Ekhon postman e get method e “localhost:3000/user/3?filter=name” url e hit korlam. Console e req.query hishebe pailam ekta object {filter : ‘name’}.

Req.body

Index.js …

*const* express = *require*("express");

*const* app = *express*();

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.query);

  console.*log*("-----------");

  res.*send*("Hello World Get...");

});

app.*post*("/user", (*req*, *res*) *=>* {

  console.*log*(req.body);

  console.*log*("-----------");

  res.*send*("Hello World Post...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Postman e giye content-type “application/json” choose kore body te ‘raw’ select kore json data {“country” : “USA”} set localhost:3000/user e hit korlam. Console e req.body undefined ashlo. Amra jani raw nodejs e jokhon kono request er response ashe tokhon sheta buffer akare pai. Ei buffer ke avoid kore kivabe data ta ekbare pete pari tar ekta system express kore dise. Bivinno parsing function like express.json(), express.raw(), express.urlencode() etc. jehetu ami postman theke json pathaisi ami express er onnopashe ekta json parser rakhle tahole she data ta amake dite parto. Ekarone code e ekhon app.use(express.json()) likhe ‘app’ ke bole dibo ei json parser ta use koro. Ekhon postman e localhost:3000/user hit korle console req.body hishebe ekta js object {country : ‘USA’} pacchi. Postmane setting thik rekhe kono blank data pathale ekta blank object dibe.

Req.cookies

Ekhane postman amar server er client. Cookie always client e thake. Postman e manually cookie set kore dekhabo amra server e cookie pacchi kina. Postman e cookies option e giye localhost domain er jonno ekta cookie set korlam. Then index.js app.get() er console e req.cookies dekhar system korlam. Nodemon index kore postman e “localhost:3000/user/5” hit korle req.cookies undefined dekhay. Cz amra ekhono cookie parser install kore amader application e niye ashi nai. ‘npm install cookie-parser’ kore install korlam. Require(‘cookie-parser’) kore cookie parser niye ashlam. App.use() kore bole dibo cookie parser takeo use korte. Ekhon abar nodemon index kore localhost:3000/user/8 kore dekhlam console e object akare cookie chole ashce {Turza : ‘great’}

Index.js …

*const* express = *require*("express");

*const* cookieParser = *require*("cookie-parser");

*const* app = *express*();

app.*use*(express.*json*());

app.*use*(*cookieParser*());

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.cookies);

  console.*log*("-----------");

  res.*send*("Hello World Get...");

});

app.*post*("/user", (*req*, *res*) *=>* {

  console.*log*(req.body);

  console.*log*("-----------");

  res.*send*("Hello World Post...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");});

cookie signed ebong unsigned hote pare. Etokkhon unsigned cookie dekhlam. Ami jodi secure cookie use kori jeta ke tamper kora jay na, cryptograhically secure tahole sheta thakbe signed cookie er moddhe.

Req.secure

Jodi amader protocol http hoy tahole req.secure false ashbe. Protocol https hole true ashbe.

Req.app

Etokkhon porjonto amra kintu moduler system e kaj kori nai. Amra ekta file index.js ei shob kaj korlam. But normal application e amader multiple file niye kaj korte hobe. App.get() er CB ta ekta handler function hishebe bairer kono file e thakte pare. Ekta file create korlam handler.js name. tate app.get() er CB ta niye sheta abar export kore index.js e import kore anlam. Then app.get() e boshiye dilam 2nd parameter e. ekhon handler function er moddhe amar ‘app’ object er kono property proyojon. ‘app’ object ta handler function er moddhe kivabe pabo? Express amader kaj easy korar jonno req object er moddhe app object ta property akare diye dise.

Index.js

*const* express = *require*("express");

*const* cookieParser = *require*("cookie-parser");

*const* handler = *require*("./handler");

*const* app = *express*();

app.*use*(express.*json*());

app.*use*(*cookieParser*());

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  console.*log*("-----------");

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", handler);

app.*post*("/user", (*req*, *res*) *=>* {

  console.*log*(req.body);

  console.*log*("-----------");

  res.*send*("Hello World Post...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Handler.js…

*const* *handler* = (*req*, *res*) *=>* {

  console.*log*(req.app);

  res.*send*("Hello World...");};

module.exports = handler;

Req.route

Ami je route e achi like /user --- ei /user get ba post jekono method e hit hote pare. Ei /user --- ei route tar shob info jodi pete chai tahole req.route use korte hobe.

Index.js …

*const* express = *require*("express");

*const* app = *express*();

app.*use*(express.*json*());

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user", (*req*, *res*) *=>* {

  console.*log*(req.route);

  res.*send*("Hello World Get...");

});

app.*post*("/user", (*req*, *res*) *=>* {

  console.*log*(req.route);

  res.*send*("Hello World Post...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

So get ba post je methodei ‘localhost:3000/user’ url e hit kori na keno output hishebe route object pacchi.

Ekhon request object er kichu methods er kaj dekhbo.

req.accepts()

ektu age amra post method e application/json content-type er data postman (client) theke pathaisilam. Postman er header e ‘Accept’ bole ekta jinish ache jar value by default ‘\*/\*’ deya ache. Etar ortho client bole dicche ami shob type er request accept korte pari. Abar ami ‘Accept’ er value modify kore bole ‘application/json’ kore bole dite pari client only ‘application/json’i accept korbe. tahole ekhetre server end e jante hobe client ki accept kore. Req.accept() diye amra sheta verify korte pari.

Postman er header e Accept er value ‘application/json’ boshaya dilam.

Index.js …

*const* express = *require*("express");

*const* app = *express*();

app.*use*(express.*json*());

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

console.*log*(req.*accepts*("html"));

  res.*send*("Hello World Get...");

});

app.*post*("/user", (*req*, *res*) *=>* {

  res.*send*("Hello World Post...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Now postman e get method e ‘localhost:3000/user/4’ hit korlam. Etar maddhome ami jiggesh korlam tumi ki html data accept koro? Answer ashlo ‘false’. Cz client agei header e ‘application/json’ bole dise.



Html er jaygay json boshaya ekhon o/p dekhlam. o/p ashlo ‘json’ orthat true return korlo. Orthat accept kore.

Req.get()

client request er shathe besh kichu header pathay ja amra postman er headers er moddhe dekhte pacchi. Req.get() er maddhome amra ei header gular value gula dekhte pari.

Index.js…

*const* express = *require*("express");

*const* app = *express*();

app.*use*(express.*json*());

*const* adminRoute = express.*Router*();

adminRoute.*get*("/dashboard", (*req*, *res*) *=>* {

  res.*send*("We are in admin dashboard.");

});

app.*get*("/user/:id", (*req*, *res*) *=>* {

  console.*log*(req.*get*("connection"));

  res.*send*("Hello World Get...");

});

app.*post*("/user", (*req*, *res*) *=>* {

  res.*send*("Hello World Post...");

});

app.*use*("/admin", adminRoute);

app.*listen*(3000, () *=>* {

  console.*log*("Listening on port 3000");

});

Nodemon index kore get method e ‘localhost:3000/user/4’ url e hit korlam. Req.get(‘connection’) value ashlo ‘keep-alive’. ‘connection' shoraya ‘content-type’ ba ‘accept’ dile ashe ‘application/json’.