Weaver – Documentation

for version 0.0.1
by Modula.dev

Table of Contents

Getting Started

Function Documentation

Weaver – Documentation

for version 0.0.1
by Modula.dev

weaver.add_route(regex, function)

A route is an object containing a regular expression and a function which will be passed in a request object if the route is called. Routes that are added earlier are given higher priority over routes added later. The request object will contain useful pieces of data like the request.url, the request.query, request.uid, and request.method.

weaver.add_file(filename)

This function takes a filename and creates a route who’s regex is a simple name-match. **Make sure to include the leading “/” in the filename**. If provided a filename like

```
weaver.add_file(“/static/mainpage.md”)
```

It will do basically the same thing as

```
weaver.add_route(/^\/static/mainpage.md/, (request)=>{
return {status:200, mime:’text/plain’,
content: fs.readFileSync(“/static/mainpage.md”)} })
```

weaver.router(*request*)

The router is a function that takes a request-object and returns an object containing a status code, a MIME-type, and some content-body. The default router's behavior is to find the first route which matches the *request.url*, and then pass the *request* into it and return.

For a very simple example, let's say you added a route like:

```
weaver.add_route(/\/home\/.*/, (request)=>{
      return {status:200,
      mime:’text/raw’,
      content: ‘hello, world’}
})
```

When you request a resource with the URL, the default router will test whether the URL matches the regex `/\/home\/.*/`. If you decide to replace the default router, remember that it is always expected to return either a valid page, or undefined

weaver.pre_router(*request*)

The pre-router is a function that takes the request-object and modifies it before it is handed off to the router. The default pre-router's behavior is to promise the request object will have properties like request.uid, request.method, request.url, and to decode html symbols like spaces from `%20` in the URL

weaver.pre_uid(*request*)

The pre-uid is a function that takes the request object and hashes it into a number. The algorithm used is very basic and intentionally information-lossy; it does not contain any information that is *personally identifiable*, while still allowing applications to remember which user they're responding to

weaver.server(*optional: port number*)

At the end of your file, once you've created all of your routes, you'll call this function to start hosting the server over either the specified or default port. For shared-hosting applications, leave this value empty.

Example Application

Let's say you wanted an application with one catch-all route that just tells the user what their UID was, the method they requested the page with, and the resource they requested.
That file might look something like this:

```
const weaver = require(“./weaver.js”)
weaver.add_route(/\/.*/, main)
function main(request) {
return {
status: 200,
mime: “text/html”,
content: `UID ${request.uid} URL ${request.url} VIA ${request.method}`
}}
weaver.listen()
```