

ExpressJS - Debugging



- Express uses the Debug module to internally log information about route matching, middleware functions, application mode, etc.

To see all internal logs used in Express, set the DEBUG environment variable to **Express:*** when starting the app -

```
DEBUG = express:* node index.js
```

The following output will be displayed.

```
ayushgp@swaggy:~/hello-world$ DEBUG=express:* node index.js
express:application set "x-powered-by" to true +0ms
express:application set "etag" to 'weak' +3ms
express:application set "etag fn" to [Function: wetag] +2ms
express:application set "env" to 'development' +0ms
express:application set "query parser" to 'extended' +0ms
express:application set "query parser fn" to [Function: parseExtendedQueryString] +0ms
express:application set "subdomain offset" to 2 +0ms
express:application set "trust proxy" to false +1ms
express:application set "trust proxy fn" to [Function: trustNone] +0ms
express:application booting in development mode +0ms
express:application set "view" to [Function: View] +0ms
express:application set "views" to '/home/ayushgp/hello-world/views' +0ms
express:application set "jsonp callback name" to 'callback' +0ms
express:router use / query +8ms
express:router:layer new / +1ms
express:router use / expressInit +1ms
express:router:layer new / +0ms
express:router use / jsonParser +0ms
express:router:layer new / +0ms
express:router use / urlencodedParser +2ms
express:router:layer new / +0ms
express:router use / multerMiddleware +1ms
express:router:layer new / +0ms
express:router:route new / +0ms
express:router:layer new / +0ms
express:router:route get / +1ms
express:router:layer new / +0ms
express:router:route new /:id([0-9]{3,}) +0ms
express:router:layer new /:id([0-9]{3,}) +0ms
express:router:route get /:id([0-9]{3,}) +0ms
```

These logs are very helpful when a component of your app is not functioning right. This verbose output might be a little overwhelming. You can also restrict the DEBUG variable to specific area to be logged. For example, if you wish to restrict the logger to application and router, you can use the following code.

```
DEBUG = express:application,express:router node index.js
```

Debug is turned off by default and is automatically turned on in production environment. Debug can also be extended to meet your needs, you can read more about it at [its npm page](#).