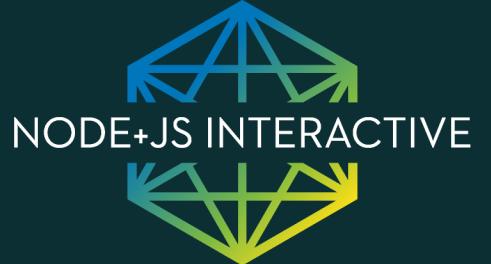


December 11–12, 2019
Montreal, Canada



December 11-12, 2019
Montreal, Canada

Why you Should Maintain Type Definitions for your JavaScript Project

Ethan Arrowood, Software Engineer, *Microsoft*

Pronouns: he, his



Fast and low overhead web framework, for Node.js

★ Star

12,702

fork

 **fastify / fastify**

Used by 4.7k Unwatch 257 Unstar 12.7k Fork 824

[Code](#) [Issues 48](#) [Pull requests 5](#) [Actions](#) [Projects 1](#) [Wiki](#) [Security](#) [Insights](#)

Fast and low overhead web framework, for Node.js <https://www.fastify.io>

[webframework](#) [performance](#) [speed](#) [nodejs](#)

 2,019 commits	 29 branches	 0 packages	 128 releases	 195 contributors	View license
---	---	---	--	--	------------------------------

 **JavaScript 97.3%**  **TypeScript 2.7%**



```
// Import the framework and instantiate it
const fastify = require('fastify')()

// Declare a route
fastify.get('/', async (request, reply) => {
  return { hello: 'world' }
})

// Run the server!
(async () => {
  try {
    await fastify.listen(3000)
    fastify.log.info(`server listening on ${fastify.server.address().port}`)
  } catch (err) {
    fastify.log.error(err)
    process.exit(1)
  }
})()
```



TypeScript

fastify.d.ts



DefinitelyTyped

The repository for high quality TypeScript type definitions



```
npm install @types/<library>
```

Branch: next ▾

fastify / fastify.d.ts

Find file

Copy path



Ethan-Arrowood Update route generics to use object instead (#1951)

433801e 13 days ago

45 contributors



79 lines (74 sloc) | 4 KB

Raw

Blame

History



```
1 import * as http from 'http'
2 import * as http2 from 'http2'
3 import * as https from 'https'
4
5 import { FastifyRequest } from './types/request'
6 import { RawServerBase, RawServerDefault, RawRequestDefaultExpression, RawReplyDefaultExpression } from './types/utils'
7 import { FastifyLoggerOptions } from './types/logger'
8 import { FastifyInstance } from './types/instance'
9 import { FastifyServerFactory } from './types/serverFactory'
10
11 /**
12  * Fastify factor function for the standard fastify http, https, or http2 server instance.
13  *
14  * The default function utilizes http
15  *
16  * @param opts Fastify server options
17  * @returns Fastify server instance
18  */
19 export default function fastify<
20     RawServer extends RawServerBase = RawServerDefault,
21     RawRequest extends RawRequestDefaultExpression<RawServer> = RawRequestDefaultExpression<RawServer>,
22     RawReply extends RawReplyDefaultExpression<RawServer> = RawReplyDefaultExpression<RawServer>,
23     Logger = FastifyLoggerOptions<RawServer>
24 >(opts?: FastifyServerOptions<RawServer, Logger>): FastifyInstance<RawServer, RawRequest, RawReply, Logger>;
```



```
export default function fastify<
  RawServer extends RawServerBase = RawServerDefault,
  RawRequest extends RawRequestDefaultExpression<RawServer> = RawRequestDefaultExpression<RawServer>,
  RawReply extends RawReplyDefaultExpression<RawServer> = RawReplyDefaultExpression<RawServer>,
  Logger = FastifyLoggerOptions<RawServer>
>(opts?: FastifyServerOptions<RawServer, Logger>): FastifyInstance<RawServer, RawRequest, RawReply, Logger>;
```



Generics



```
export default function fastify<
  RawServer extends RawServerBase = RawServerDefault,
  RawRequest extends RawRequestDefaultExpression<RawServer> = RawRequestDefaultExpression<RawServer>,
  RawReply extends RawReplyDefaultExpression<RawServer> = RawReplyDefaultExpression<RawServer>,
  Logger = FastifyLoggerOptions<RawServer>
>(opts?: FastifyServerOptions<RawServer, Logger>): FastifyInstance<RawServer, RawRequest, RawReply, Logger>;
```



RawServer *extends* RawServerBase = RawServerDefault



```
export type RawServerBase = http.Server | https.Server | http2.Http2Server | http2.Http2SecureServer
```

```
export type RawServerDefault = http.Server
```



```
export default function fastify<
  RawServer extends RawServerBase = RawServerDefault,
  RawRequest extends RawRequestDefaultExpression<RawServer> = RawRequestDefaultExpression<RawServer>,
  RawReply extends RawReplyDefaultExpression<RawServer> = RawReplyDefaultExpression<RawServer>,
  Logger = FastifyLoggerOptions<RawServer>
>(opts?: FastifyServerOptions<RawServer, Logger>): FastifyInstance<RawServer, RawRequest, RawReply, Logger>;
```



```
RawRequest extends RawRequestDefaultExpression<RawServer> = RawRequestDefaultExpression<RawServer>
```



```
export type RawRequestDefaultExpression<RawServer> = RawServerDefault> =  
( RawServer extends http.Server | https.Server )  
? http.IncomingMessage  
: http2.Http2ServerRequest
```



```
export default function fastify<
  RawServer extends RawServerBase = RawServerDefault,
  RawRequest extends RawRequestDefaultExpression<RawServer> = RawRequestDefaultExpression<RawServer>,
  RawReply extends RawReplyDefaultExpression<RawServer> = RawReplyDefaultExpression<RawServer>,
  Logger = FastifyLoggerOptions<RawServer>
>(opts?: FastifyServerOptions<RawServer, Logger>): FastifyInstance<RawServer, RawRequest, RawReply, Logger>;
```



```
import fastify from 'fastify'
```

```
const server = fastify() // returns a http server
```



```
import fastify from 'fastify'  
import * as http2 from 'http2' // @types/node  
  
const server = fastify<http2.Http2Server>({ http2: true })
```



```
export type FastifyServerOptions<
  RawServer extends RawServerBase = RawServerDefault,
  Logger = FastifyLoggerOptions<RawServer>
> = {
  http2?: RawServer extends http2.Http2Server ? true : false
}
```



```
import fastify, {
  RouteShorthandOptions,
  RequestBodyDefault,
  RequestParamsDefault,
  RequestHeadersDefault } from 'fastify'

const server = fastify()

type QuerystringType = {
  username: string,
  password: number
}

server.get<
  RequestBodyDefault,
  QuerystringType,
  RequestParamsDefault,
  RequestHeadersDefault
>('/auth', async (request, reply) => {
  const { username, password } = request.query
  return `logged in!`
})
```

Named Generic Parameters

```
import fastify from 'fastify'

const server = fastify()

type QuerystringType = {
  username: string,
  password: number
}

server.get<{
  Querystring: QuerystringType
}>('/auth', async (request, reply) => {
  const { username, password } = request.query
  return `logged in!`
})
```

Testing



tsd build passing

Check TypeScript type definitions

Install

```
$ npm install tsd
```

<https://github.com/SamVerschueren/tsd>

```
import fastify, { FastifyInstance } from '..../fastify'
import * as http from 'http'
import * as http2 from 'http2'
import { expectType, expectError } from 'tsd'

expectType<
  FastifyInstance<
    http.Server,
    http.IncomingMessage,
    http.ServerResponse
  >
>(fastify())

expectType<
  FastifyInstance<
    http2.Http2Server,
    http2.Http2ServerRequest,
    http2.Http2ServerResponse
  >
>(fastify<http2.Http2Server>())

expectError(fastify<http2.Http2Server>({ http2: false }))
```

Testing is Important
even for Type Definitions

Missing Feature: Coverage



Documentation



“The best way to learn the Fastify type system is by example!”

- Me

Good Documentation Includes:

- Examples/Tutorials
- Detailed API Guide
- References to source code
- Frequently asked questions (with answers)
- Links for contributing

Code

Issues 49

Pull requests 5

Actions

Projects 1

Wiki

Security

Insights

[WIP] Typescript docs #1966

Edit

Open

Ethan-Arrowood wants to merge 7 commits into `fastify:next` from `Ethan-Arrowood:typescript-docs`

Conversation 7

Commits 7

Checks 18

Files changed 1

+408 -236



Ethan-Arrowood commented 4 days ago

Member

+ 😊

...

Totally revamping TypeScript documentation for the `next` types.

Still work in progress. I'm going to be busy for the next couple weeks so I wanted to get this up for some early feedback. Please try out the examples and let me know if there are any issues. You'll need to use `npm link` to get access to the fastify v3 `next` typings -- if anyone wants steps on this let me know and I'll add them to this PR.



Ethan-Arrowood added 7 commits 21 days ago

Init commit of TypeScript v3 docs

7df9fd9

Reviewers



mcollina



Assignees

No one—assign yourself



Labels

documentation

invalid

typescript

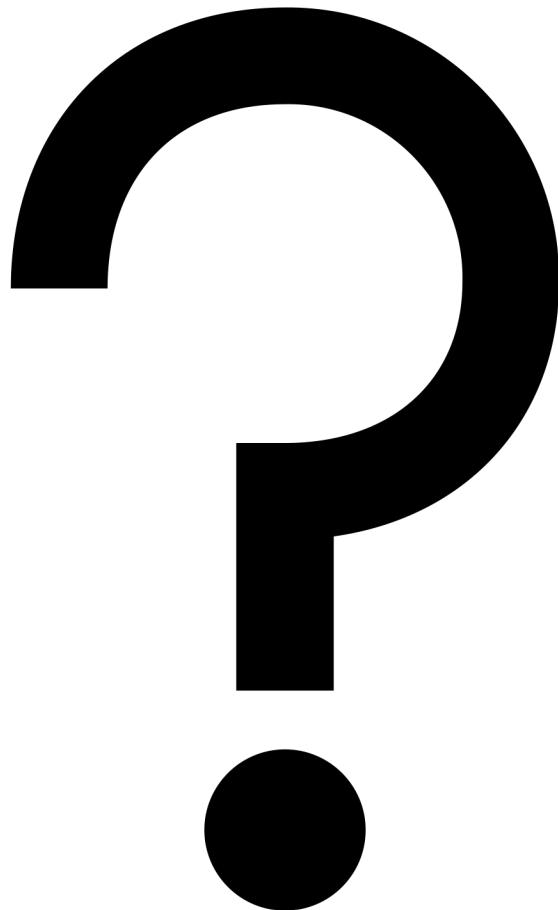
v3.x



Projects

<https://github.com/fastify/fastify/pull/1966/>

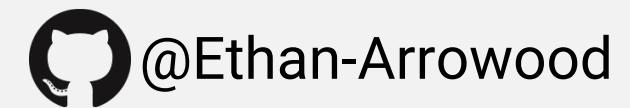
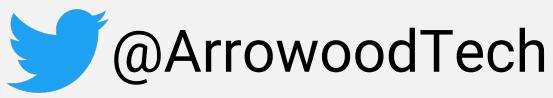
Why all this effort?



JS



TS



NODE+JS INTERACTIVE

December 11-12, 2019
Montreal, Canada

Ethan Arrowood
Software Engineer, Microsoft

You're welcome to chat with me at anytime here or online!