



Curso de Backend con Node.js

Artículo

Manejo de errores asíncronos y síncronos en Express



glrodasz ⌚ 26 de Julio de 2019

El manejo de errores en Express es el proceso de capturar un error de manera asíncrona como síncrona . Por defecto Express viene con un manejador de errores por defecto, así que no es necesario escribir uno para empezar a usarlo.

Los errores que ocurren de manera síncrona dentro un manejador de rutas o un *middleware* no requieren trabajo extra. Si un código síncrono lanza un error Express automáticamente capturará el error. Por ejemplo:

```
app.get("/", function(req, res) {  
  throw new Error("BROKEN"); // Express capturara este error por sí solo.  
});
```

Para errores que se retornan desde funciones asíncronas invocadas desde un manejador de ruta o un *middleware*, es necesario pasar el error como argumento a la función `next()` de esta



Capa de validación de datos ...

```
app.get("/", function(req, res, next) {  
  fs.readFile("/file-does-not-exist", function(err, data) {  
    if (err) {  
      next(err); // Se debe pasar el error a Express.  
    } else {  
      res.send(data);  
    }  
  });  
});
```

Es responsabilidad de nosotros capturar errores que puedan ocurrir en código asíncrono invocado desde un manejador de ruta o *middleware* para que Express lo procese. Por ejemplo:

```
app.get("/", function(req, res, next) {  
  setTimeout(function() {  
    try {  
      throw new Error("BROKEN");  
    } catch (err) {  
      next(err);  
    }  
  }, 100);  
});
```

El ejemplo de arriba usa un bloque `try...catch` para capturar los errores en el código asíncrono y pasarlo a Express. Si el bloque `try...catch` fuese omitido, Express no podría capturar el error debido a que no es parte de un manejador síncrono de código.



Capa de validación de datos ...

capturará el error. Por ejemplo:

```
app.get("/", function(req, res, next) {  
  Promise.resolve()  
    .then(function() {  
      throw new Error("BROKEN");  
    })  
    .catch(next); // Errores serán pasados a Express.  
});
```



Escribe aquí tu pregunta

+ 2



Capa de validación de datos ...