

Clase 14: Express - estático

Sitio: [Centro de E-Learning - UTN.BA](#)
Curso: Curso de Backend Developer - Turno
Noche
Libro: Clase 14: Express - estático

Imprimido
por: Nelson Brian Avila Solano
Día: Tuesday, 30 de December de 2025,
19:58

Descripción

Objetivos

- Abordar conceptos básicos de Express en Node.js
- Servir archivos estáticos como HTML y CSS desde el servidor.

Tabla de contenidos

1. Introducción
2. Express - estático
3. ¡A practicar!

1. Introducción

Express es un framework para Node.js que facilita la creación de servidores web y la gestión de rutas de manera sencilla y eficiente. Permite servir archivos estáticos como páginas HTML, hojas de estilo e imágenes, lo que resulta útil para desarrollar sitios web y aplicaciones sin necesidad de configuraciones complejas.

Además, Express ayuda a organizar mejor el código separando las rutas en distintos archivos, lo que hace que las aplicaciones sean más estructuradas y fáciles de mantener. Gracias a su flexibilidad y simplicidad, es una herramienta muy utilizada en el desarrollo web con Node.js.

2. Express - estático

Express - estático

Mostrar contenido estático

La función `express.static()` es una función de middleware incorporada en Express para entregar archivos estáticos.

```
app.use(express.static("public"));
```

Definir las rutas correspondientes

```
app.get('/', (req, res) => {  
  res.sendFile(__dirname + "/public/index.html")  
});  
  
app.get('/generic', (req, res) => {  
  res.sendFile(__dirname + '/public/generic.html')  
});  
  
app.get('/elements', (req, res) => {  
  res.sendFile(__dirname + '/public/elements.html')  
}) app.get('*', (req, res) => {  
  res.sendFile(__dirname + '/public/404.html')  
})
```

Organizar el código

Integrar las rutas desde un archivo separado

```
const express = require('express'); const router =
express.Router(); router.get('/', (req, res) => {
res.sendFile(__dirname + '/../public/index.html');
});
router.get('/sobre-nosotros', (req, res) => {
res.sendFile(__dirname + '/../public/sobre-nosotros.html');
});
router.get('/contacto', (req, res) => {
res.sendFile(__dirname + '/../public/contacto.html'); });

module.exports = router;
```

En app.js

Indicar desde donde se toman las rutas

```
const routes = require('./routes/routes');
app.use('/', routes);
```

3. ¡A practicar!

¡A practicar!

1. Crear una carpeta public.
2. Crear 4 páginas html, una de ellas index.html con su correspondiente contenido css e imágenes.
3. Incluir página 404.
4. Definir las rutas correspondientes en archivo routes.

Recordá utilizar.

```
app.use(express.static("public"));
```

¡Anexo!

1. Instalar GIT <https://git-scm.com/>
2. Abrir una cuenta en github: <https://github.com/>