

■ Ficha 2 – Rutas y respuestas en Express

■ 1. ¿Qué es una ruta?

Una **ruta** es una dirección dentro del servidor a la que puede acceder un cliente (por ejemplo, un navegador o una aplicación). Cada ruta está asociada a un **método HTTP** y define qué respuesta enviará el servidor cuando se acceda a esa dirección.

Por ejemplo:

```
app.get("/", (req, res) => { res.send("Bienvenido"); });
```

Significa: “Cuando el cliente hace una petición **GET** a la dirección `/`, el servidor responde con el texto ‘Bienvenido’.”

■ 2. Estructura de una ruta en Express

```
app.METODO('/ruta', (req, res) => { ... });
```

Donde:

- **app**: la aplicación de Express que creamos con **const app = express()**.
- **METODO**: el tipo de petición HTTP (por ahora usamos **GET**).
- **'/ruta'**: la dirección o endpoint al que el cliente accederá.
- **(req, res)**: los parámetros que representan la petición y la respuesta.

■ 3. Tipos de respuesta más comunes

- **res.send()**: envía texto o HTML.
- **res.json()**: envía datos en formato JSON.
- **res.sendFile()**: envía un archivo (como un HTML o una imagen).

■ 4. Métodos HTTP (por ahora solo GET)

HTTP define diferentes **métodos** para indicar qué acción quiere realizar el cliente. Por ahora, usamos **GET** porque solo pedimos información al servidor. Más adelante conoceremos otros como **POST**, **PUT** y **DELETE**.

Ejemplo:

```
app.get("/info", (req, res) => { res.send("Información del servidor"); });
```

■ 5. Esquema visual

Cliente (navegador) → Petición GET a /info → Servidor Express → Respuesta con texto o datos.

Podemos imaginar que cada ruta del servidor es una **ventanilla diferente** que ofrece una información distinta al cliente.

■ **En resumen:** las rutas permiten que nuestro servidor Express responda de distintas maneras según la dirección o recurso que el cliente solicite.