

Rutas en Express

“

A través del **sistema** de **ruteo** de Express podemos definir cómo va a responder nuestra aplicación según el **método** HTTP y la **ruta** que llegue como pedido al **servidor**.



”

Definiendo una ruta

Express nos brinda una estructura sencilla para definir cada una de las rutas que estaremos atendiendo y las respuestas que enviará nuestra aplicación.

```
{}  
  app.get('/', (req, res) => {  
    res.send('¡Hola mundo!');  
  });
```

Definiendo una ruta

El primer objeto que vemos es `app`, que guarda la ejecución de Express y contiene la mayor parte de su funcionalidad.

```
{ }  
  app.get('/', (req, res) => {  
    res.send('¡Hola mundo!');  
  });
```

Definiendo una ruta

Luego, deberemos colocar el método HTTP que queremos atender. Puede ser, por ejemplo: get, post, put, patch o delete.

```
{}  
  app.get('/', (req, res) => {  
    res.send('¡Hola mundo!');  
  });
```

Definiendo una ruta

A continuación tenemos, como primer parámetro, un string que representa la ruta a la que ha ingresado nuestro cliente. En este caso se trata del raíz o *home* del sitio.

```
{}  
  app.get('/', (req, res) => {  
    res.send('¡Hola mundo!');  
  });
```

Definiendo una ruta

Como segundo parámetro, tendremos el callback que se encargará de responder a la petición.

Este callback recibirá al menos dos parámetros:

- **req** → Representa la petición del cliente.
- **res** → Representa la respuesta del servidor.

Ambos son objetos que nos brinda Express y contendrán toda la información y funcionalidad necesaria para trabajar los pedidos y respuestas.

```
{}  
  app.get('/', (req, res) => {  
    res.send('¡Hola mundo!');  
  });
```

Definiendo una ruta

Finalmente, dentro del callback, definiremos la lógica a ejecutar para manejar la respuesta al pedido del cliente.

En este caso, utilizamos el método `send()` del objeto `res` (response) para enviar una respuesta de texto al navegador (cliente).

```
{}  
  app.get('/', (req, res) => {  
    res.send('¡Hola mundo!');  
  });
```


Definiendo una ruta

No va ser necesario que siempre pongamos el contenido directamente dentro del método.

Como todo método, `send()` puede recibir una variable como parámetro.

```
{}  
  app.get('/', (req, res) => {  
    let contenido = 'Página principal';  
    res.send(contenido);  
  });
```

DigitalHouse>
Coding School