

# RECORRIDA ECOSISTEMA EXPRESS

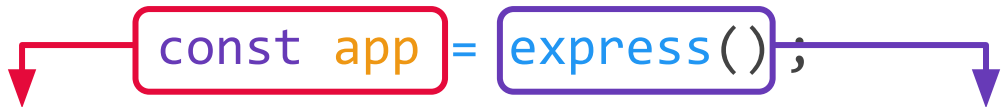
“

Dentro de este framework encontramos **funcionalidades** completamente **listas** para usar. Todas ellas **conviven** entre sí, creando juntas lo que llamamos el **ecosistema express**.



# FUNCIÓN DE ALTO NIVEL

Express cuenta con una función de alto nivel que, al invocarla, retorna un **objeto** con múltiples propiedades y métodos: la función `express()`. Al **almacenar** la ejecución de esa función en una constante, podremos tener **acceso** a través de ella a todas esas funcionalidades que nos da el objeto.



## Constante

Aquí tendremos almacenado el **objeto**. Usando la notación de punto podremos acceder a todas sus **propiedades** y **métodos**.

## Función alto nivel

La función **ejecutada** devuelve un **objeto**.

# FUNCIONALIDADES EXPRESS

El objeto que devuelve `express()` tiene métodos asociados para gestionar las peticiones que se hagan a través de get, post, delete, put y patch.

```
app.METHOD();  
app.get();  
app.post();  
app.put();  
app.patch();  
app.delete();
```

{}

Cuando trabajemos con alguno de esos métodos, tendremos acceso a dos **objetos literales** que nos da Express: **request** y **response**, que nos permitirán acceder a los **requests** del **cliente** y a los **responses** del **servidor**.



```
{ código }
```

```
request.path();  
request.method();  
response.send();
```

Este método del objeto **request** nos permite conocer la url exacta de donde vino la petición.

```
{ código }
```

```
request.path();  
request.method();  
response.send();
```

Este método del objeto **request** nos permite conocer a través de qué método HTTP viajó la petición.

```
{ código }
```

```
request.path();  
request.method();  
response.send();
```

Este método del objeto **response** nos permite enviar como respuesta un **string**, un **objeto literal** o un **array** sin la necesidad de aclarar qué tipo de dato estamos enviando.

El método, por detrás, ya trae definidas las cabeceras que necesita el navegador.