

Devolver una respuesta directamente

Cuando creas una *operación de path* normalmente puedes devolver cualquier dato: un `dict`, una `list`, un modelo Pydantic, un modelo de base de datos, etc.

Por defecto, **FastAPI** convertiría automáticamente ese valor devuelto a JSON usando el `jsonable_encoder` explicado en Codificador Compatible JSON.

Luego, tras bastidores, pondría esos datos compatibles con JSON (por ejemplo, un `dict`) dentro de una `JSONResponse` que se usaría para enviar la respuesta al cliente.

Pero puedes devolver una `JSONResponse` directamente de tu *operación de path*. Esto puede ser útil, por ejemplo, para devolver cookies o headers personalizados.

Devolver una Response

De hecho, puedes devolver cualquier `Response` o cualquier subclase de la misma.

!!! tip “Consejo” `JSONResponse` en sí misma es una subclase de `Response`.

Y cuando devuelves una `Response`, **FastAPI** la pasará directamente.

No hará ninguna conversión de datos con modelos Pydantic, no convertirá el contenido a ningún tipo, etc.

Esto te da mucha flexibilidad. Puedes devolver cualquier tipo de dato, sobrescribir cualquier declaración de datos o validación, etc.

Usando el `jsonable_encoder` en una Response

Como **FastAPI** no realiza ningún cambio en la `Response` que devuelves, debes asegurarte de que el contenido está listo.

Por ejemplo, no puedes poner un modelo Pydantic en una `JSONResponse` sin primero convertirlo a un `dict` con todos los tipos de datos (como `datetime`, `UUID`, etc) convertidos a tipos compatibles con JSON.

Para esos casos, puedes usar el `jsonable_encoder` para convertir tus datos antes de pasarlos a la respuesta:

```
Python hl_lines="4 6 20 21" {!../../../docs_src/response_directly/tutorial001.py!}
```

!!! note “Detalles Técnicos” También puedes usar `from starlette.responses import JSONResponse`.

FastAPI provee `starlette.responses` como `fastapi.responses`, simplemente como una conv

Devolviendo una Response personalizada

El ejemplo anterior muestra las partes que necesitas, pero no es muy útil todavía, dado que podrías simplemente devolver el `item` directamente, y **FastAPI** lo

pondría en una `JSONResponse` por ti, convirtiéndolo en un `dict`, etc. Todo esto por defecto.

Ahora, veamos cómo puedes usarlo para devolver una respuesta personalizada.

Digamos que quieres devolver una respuesta XML.

Podrías poner tu contenido XML en un string, ponerlo en una `Response` y devolverlo:

```
Python hl_lines="1 18" {!../../../../../docs_src/response_directly/tutorial002.py!}
```

Notas

Cuando devuelves una `Response` directamente, los datos no son validados, convertidos (serializados), ni documentados automáticamente.

Pero todavía es posible documentarlo como es descrito en Respuestas adicionales en OpenAPI.

Puedes ver en secciones posteriores como usar/declarar esas `Responses` personalizadas aún teniendo conversión automática de datos, documentación, etc.