Additional Status Codes

By default, **FastAPI** will return the responses using a `JSONResponse`, putting the content you return from your *path operation* inside of that `JSONResponse`.

It will use the default status code or the one you set in your *path operation*.

Additional status codes

If you want to return additional status codes apart from the main one, you can do that by returning a `Response` directly, like a `JSONResponse`, and set the additional status code directly.

For example, let's say that you want to have a *path operation* that allows to update items, and returns HTTP status codes of 200 "OK" when successful.

But you also want it to accept new items. And when the items didn't exist before, it creates them, and returns an HTTP status code of 201 "Created".

To achieve that, import `JSONResponse`, and return your content there directly, setting the `status_code` that you want:

Python hl_lines="4 23" {!../../../docs_src/additional_status_codes/tutorial001.py!}

!!! warning When you return a `Response` directly, like in the example above, it will be returned directly.

```
It won't be serialized with a model, etc.
```

```
Make sure it has the data you want it to have, and that the values are valid JSON (if you a
```

!!! note "Technical Details" You could also use `from starlette.responses import JSONResponse`.

```
**FastAPI** provides the same `starlette.responses` as `fastapi.responses` just as a conven:
```

OpenAPI and API docs

If you return additional status codes and responses directly, they won't be included in the OpenAPI schema (the API docs), because FastAPI doesn't have a way to know beforehand what you are going to return.

But you can document that in your code, using: Additional Responses.