

JSON 호환 가능 인코더

데이터 유형(예: Pydantic 모델)을 JSON과 호환된 형태로 반환해야 하는 경우가 있습니다. (예: `dict`, `list` 등)

예를 들면, 데이터베이스에 저장해야 하는 경우입니다.

이를 위해, **FastAPI**에서는 `jsonable_encoder()` 함수를 제공합니다.

`jsonable_encoder` 사용

JSON 호환 가능 데이터만 수신하는 `fake_db` 데이터베이스가 존재한다고 가정하겠습니다.

예를 들면, `datetime` 객체는 JSON과 호환되는 데이터가 아니므로 이 데이터는 받아들여지지 않습니다.

따라서 `datetime` 객체는 [ISO format](#) 데이터를 포함하는 `str` 로 변환되어야 합니다.

같은 방식으로 이 데이터베이스는 Pydantic 모델(속성이 있는 객체)을 받지 않고, `dict` 만을 받습니다.

이를 위해 `jsonable_encoder` 를 사용할 수 있습니다.

Pydantic 모델과 같은 객체를 받고 JSON 호환 가능한 버전으로 반환합니다:

```
{!../../../../../docs_src/encoder/tutorial001.py!}
```

이 예시는 Pydantic 모델을 `dict` 로, `datetime` 형식을 `str` 로 변환합니다.

이렇게 호출한 결과는 파이썬 표준인 [`json.dumps\(\)`](#) 로 인코딩 할 수 있습니다.

길이가 긴 문자열 형태의 JSON 형식(문자열)의 데이터가 들어있는 상황에서는 `str` 로 반환하지 않습니다. JSON과 모두 호환되는 값과 하위 값이 있는 Python 표준 데이터 구조 (예: `dict`)를 반환합니다.

!!! note "참고" 실제로 `jsonable_encoder` 는 **FastAPI** 에서 내부적으로 데이터를 변환하는 데 사용하지만, 다른 많은 곳에서도 이는 유용합니다.