

## 2.1 FastAPI의 라우팅

라우트는 HTTP 요청 메서드의 요청을 수락하고 선택적으로 인수를 받을 수 있도록 정의된다. 요청이 특정 라우트로 전달되면 애플리케이션은 라우트 처리기 `route handler`가 요청을 처리하기 전에 해당 라우트가 정의되어 있는지 확인한다. 라우트 처리기는 서버로 전송된 요청을 처리하는 함수다. 라우트 처리기의 예로는 요청을 받아 데이터베이스에서 특정 데이터를 추출하는 함수가 있다.<sup>18</sup>



### HTTP 요청 메서드

HTTP 요청 메서드는 HTTP 메서드 처리 유형을 정의하는 식별자다. 표준 메서드에는 GET, POST, PUT, PATCH, DELETE 등이 있다. 각 메서드를 자세히 알고 싶다면 웹 사이트<sup>19</sup>를 참고하자.

### 라우팅 예

〈CHAPTER 1 FastAPI 소개〉의 마지막 부분에서 단일 라우트 애플리케이션을 만들었다. 이 애플리케이션에서는 다음과 같이 `app` 변수로 초기화한 `FastAPI()` 인스턴스를 사용해서 라우팅을 처리했다.

```
from fastapi import FastAPI
app = FastAPI()

@app.get("/")

async def welcome() -> dict:
    return {
        "message": "Hello World"
    }
```

그리고 다음과 같이 `uvicorn` 도구가 `FastAPI()` 인스턴스를 가리키도록 했다.

<sup>18</sup> 옮긴이: 설명을 추가하자면, 라우팅은 요청을 HTTP 메서드 종류와 라우트에 맞게 분배하는 것을 의미한다.

<sup>19</sup> <https://developer.mozilla.org/ko/docs/Web/HTTP/Methods>