

---

```
import httpx
import pytest

from auth.jwt_handler import create_access_token
from models.events import Event
```

---

이 코드는 앞서 사용한 일반적인 라이브러리, `create_access_token(user)` 함수, 이벤트 모델 (`Event`)도 임포트한다. 몇몇 라우트는 보안이 적용되므로 접속 토큰을 생성해야 한다. 따라서 새로운 픽스처를 만들고 접속 토큰을 반환하도록 한다. 이 픽스처는 `module` 범위를 갖는다. 즉, 테스트 파일이 실행될 때 한 번만 실행되고 다른 함수가 호출될 때는 실행되지 않는다. 다음 코드를 `test_routes.py` 파일에 추가하자.

---

```
@pytest.fixture(scope="module")
async def access_token() -> str:
    return create_access_token("testuser@packt.com")
```

---

다음으로 이벤트를 데이터베이스에 추가하는 픽스처를 만든다. 이 픽스처는 CRUD 라우트 테스트에 대한 사전 테스트를 진행하는 데 사용된다. 다음과 같이 코드를 추가하자.

---

```
@pytest.fixture(scope="module")
async def mock_event() -> Event:
    new_event = Event(
        creator="testuser@packt.com",
        title="FastAPI Book Launch",
        image="https://linktomyimage.com/image.png",
        description="We will be discussing the contents of the FastAPI book in this event. Ensure to come with your own copy to win gifts!",
        tags=["python", "fastapi", "book", "launch"],
        location="Google Meet"
    )

    await Event.insert_one(new_event)

    yield new_event
```

---