

신규 이벤트 생성을 담당하는 POST 라우트를 다음과 같이 변경한다.

```
@event_router.post("/new")
async def create_event(new_event: Event,
    session=Depends(get_session)) -> dict:
    session.add(new_event)
    session.commit()
    session.refresh(new_event)

    return {
        "message": "Event created successfully."
    }
```

이 코드는 데이터베이스 처리에 필요한 세션 객체가 `get_session()` 함수에 의존하도록 설정한다. 함수 내에서는 데이터(이벤트)를 세션에 추가하고 데이터베이스에 등록(커밋)한 후 세션을 업데이트한다.

다음 명령을 사용해 라우트가 제대로 변경됐는지 확인한다.

```
(venv)$ curl -X 'POST' \
'http://127.0.0.1:8000/event/new' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "title": "FastAPI Book Launch",
  "image": "fastapi-book.jpeg",
  "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"
}'
```

윈도우에서는 다음과 같이 실행한다.

```
curl -X POST http://127.0.0.1:8000/event/new -H "accept: application/json"
-H "Content-Type: application/json" -d '{"id": 1, "title": "FastAPI Book
Launch", "image": "https://linktoimage.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", "l
aunch"], "location": "Google Meet"}'
```
