

```

location: Optional[str]

class Config:
    schema_extra = {
        "example": { "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"
        }
    }

```

---

### 3 connection.py에 데이터베이스 및 테이블 생성을 위한 설정을 작성한다.

```

from sqlmodel import SQLModel, Session, create_engine
from models.events import Event

database_file = "planner.db"
database_connection_string = f"sqlite:/// {database_file}"
connect_args = {"check_same_thread": False}
engine_url = create_engine(database_connection_string, echo=True, connect_
args=connect_args)

def conn():
    SQLModel.metadata.create_all(engine_url)

def get_session():
    with Session(engine_url) as session:
        yield session

```

---

이 코드는 먼저 필요한 의존 라이브러리와 테이블 모델을 임포트한다. 그리고 데이터베이스 파일의 위치(없는 경우 생성된다), 연결 문자열<sup>connection string</sup>, 생성된 SQL 데이터베이스의 인스턴스를 변수에 저장한다. `conn()` 함수는 `SQLModel`을 사용해서 데이터베이스와 테이블을 생성하고 `get_session()`을 사용해서 데이터베이스 세션을 애플리케이션 내에서 유지한다.