

```

        raise HTTPException(
            status_code=status.HTTP_409_CONFLICT,
            detail="User with supplied username exists"
        )
    users[data.email] = data
    return {
        "message": "User successfully registered!"
    }

```

등록 라우트에서는 애플리케이션에 내장된 데이터베이스를 사용한다(데이터베이스에 관한 내용은 <CHAPTER 6 데이터베이스 연결>에서 자세히 다룬다).

이 라우트는 사용자를 등록하기 전 데이터베이스에 비슷한 이메일이 존재하는지 확인한다.

2 로그인(/signin) 라우트를 다음과 같이 정의한다.

```

@user_router.post("/signin")
async def sign_user_in(user: UserSignIn) -> dict:
    if user.email not in users:
        raise HTTPException(
            status_code=status.HTTP_404_NOT_FOUND,
            detail="User does not exist"
        )
    if users[user.email].password != user.password:
        raise HTTPException(
            status_code=status.HTTP_403_FORBIDDEN,
            detail="Wrong credentials passed"
        )

    return {
        "message": "User signed in successfully."
    }

```

이 라우트는 로그인하려는 사용자가 데이터베이스에 존재하는지를 먼저 확인하고, 없으면 예외를 발생시킨다. 사용자가 존재하면 패스워드가 일치하는지 확인해서 성공 또는 실패 메시지를 반환한다.

여기서는 설명을 위해 패스워드를 암호화하지 않고 일반 텍스트로 저장했지만 소프트웨어 개발 관점에서는 잘못된 방식이다. <CHAPTER 6 데이터베이스 연결>에서 애플리케이션 내장