

```

payload = {
    "username": "wronguser@packt.com",
    "password": "testpassword"
}

```

```

default_client = <httpx.AsyncClient object at 0x1044a84f0>

@pytest.mark.asyncio
async def test_sign_user_in(default_client: httpx.AsyncClient) -> None:
    payload = {
        "username": "wronguser@packt.com",
        "password": "testpassword"
    }

    headers = {
        "accept": "application/json",
        "Content-Type": "application/x-www-form-urlencoded"
    }

    response = await default_client.post("/user/signin", data=payload, headers=headers)

    > assert response.status_code == 200
    E assert 404 == 200
    E + where 404 = <Response [404 Not Found]>.status_code

tests/test_login.py:41: AssertionError
===== short test summary info =====
FAILED tests/test_login.py::test_sign_user_in - assert 404 == 200
1 failed, 1 passed in 0.34s
(venv) + planner git:(main) x

```

그림 8-5 틀린 요청 페이로드로 인해 실패한 테스트

사용자 등록 및 로그인 라우트 테스트를 모두 작성했다. 이제 이벤트 플래너 API의 CRUD 라우트 테스트를 작성해보자.

CRUD 라우트 테스트

먼저 tests 폴더에 test_routes.py 파일을 만든다.

```
(venv)$ touch tests/test_routes.py
```

생성된 파일에 다음 코드를 추가한다.