

윈도우에서는 다음 명령을 사용하거나 탐색기를 사용해 신규 파일을 생성할 수 있다.

```
C:\code\fastapi\todos>rem. > todo.py
```

이제 생성된 파일 안에 코드를 작성하면 된다. 먼저 `APIRouter` 클래스를 `fastapi` 패키지에서 임포트한 후 `APIRouter()` 인스턴스를 만든다.

```
from fastapi import APIRouter

todo_router = APIRouter()
```

다음으로 앱 내부 데이터베이스를 임시로 만들고 `todos`를 생성 및 추출하는 라우트를 정의한다.

```
from fastapi import APIRouter

todo_router = APIRouter()

todo_list = []

@todo_router.post("/todo")
async def add_todo(todo: dict) -> dict:
    todo_list.append(todo)
    return {
        "message": "Todo added successfully."
    }

@todo_router.get("/todo")
async def retrieve_todos() -> dict:
    return {
        "todos": todo_list
    }
```

이 코드에서는 `todo` 처리를 위해 두 개의 라우트를 추가한다. 첫 번째 라우트는 `todo_list`에 `todo`를 추가하는 `POST` 메서드이고, 두 번째 라우트는 모든 `todo` 아이템을 `todo_list`에서 조회하는 `GET` 메서드다.

이것으로 `todo` 라우트 작성이 끝났다. 이제 이 애플리케이션을 서버에서 실행한 다음 제대로 작동하는지 테스트해보자.