

▶ 2-3 전체 프로세스 자동화 처리

- pyinstaller 설치 및 실행

[설치]

CMD

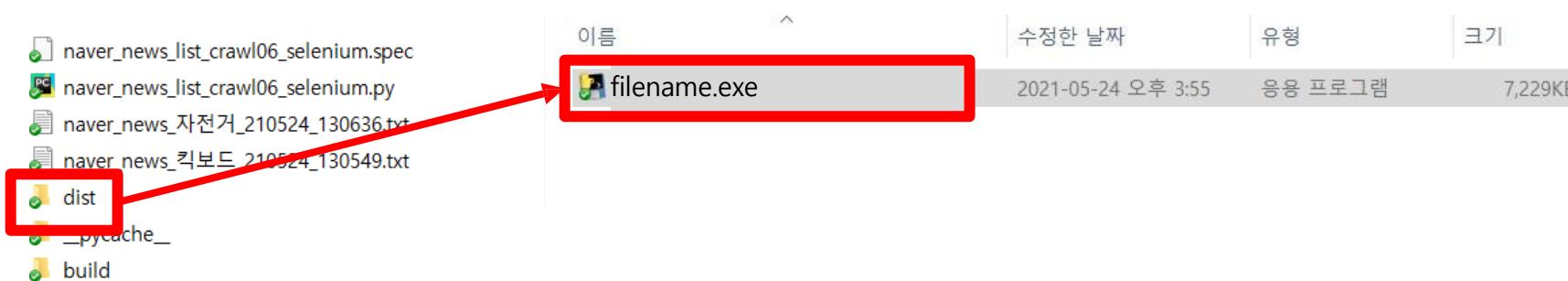
pip install pyinstaller

py가 있는 폴더로 이동 : cd 폴더경로

[실행]

pyinstaller --onefile file_name.py

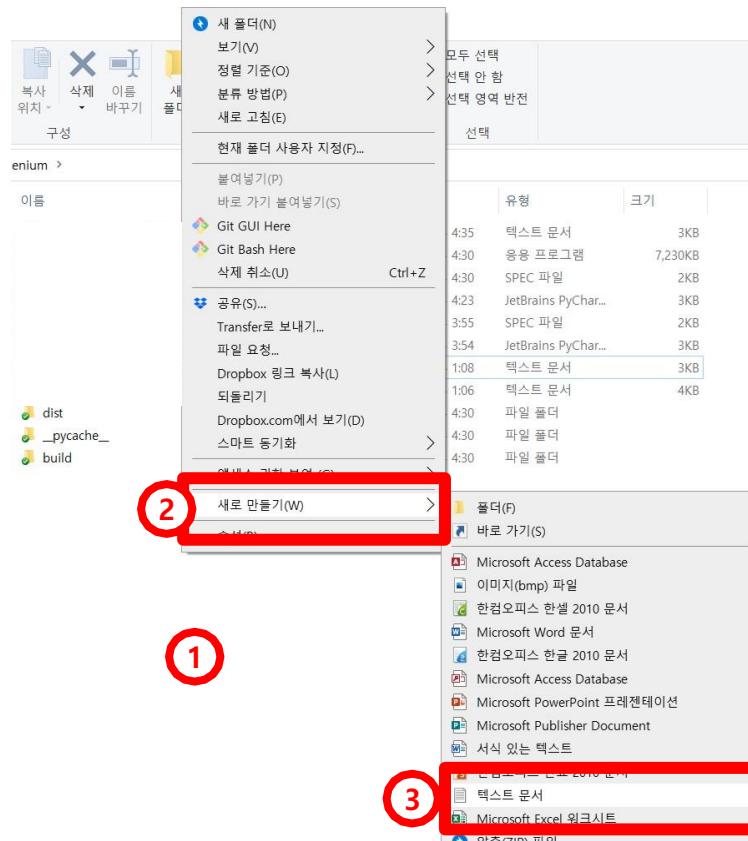
```
16065 INFO: checking EXE
16066 INFO: Building EXE because EXE-00.toc is non existent
16067 INFO: Building EXE from EXE-00.toc
16098 INFO: Appending archive to EXE C:\Users\huhju\Dropbox\강의\python\result\naver_news_selenium\dist\filename.exe
16913 INFO: Building EXE from EXE-00.toc completed successfully.
```



▶ 2-3 전체 프로세스 자동화 처리

- batch 파일 생성

매번 CMD창을 열어 실행하기 번거롭기 때문에 이 작업을 단축해주는 Batch파일을 만들어서 실행해보겠습니다.



1. 마우스 오른쪽 클릭
2. 새로만들기
3. 텍스트 문서

▶ 2-3 전체 프로세스 자동화 처리

- batch 파일 생성

1. 파일 이름을

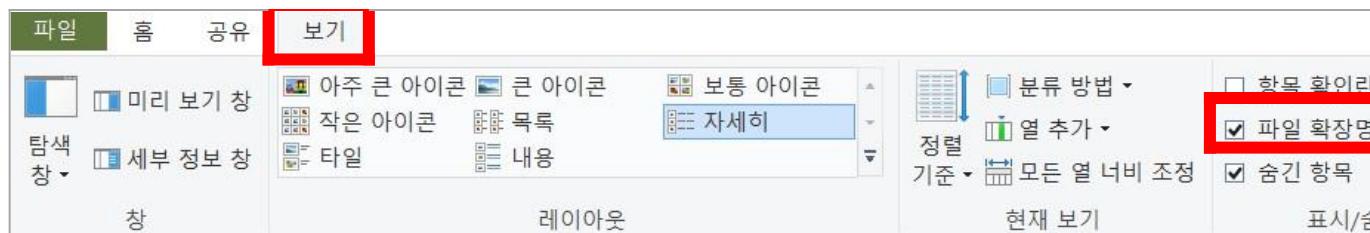
fine_name.bat로 변경

만약 확장자가 안 보인다면 탐색기 > 보기 > 파일 확장명 체크

2. 예 클릭

3. 파일선택 후 오른쪽 마우스 클릭

4. 편집 클릭



▶ 2-3 전체 프로세스 자동화 처리

- batch 파일 생성

```
① @echo off  
② pushd "%~dp0"  
③ file_name.exe  
④ pause
```

1. 필요한 출력만 표시
2. 현재 작업 디렉토리로 이동
3. 실행파일 실행
4. 출력 결과가 사라지지 않고 키를 입력해야 종료 : CMD창에서 실행할 때 차이가 남

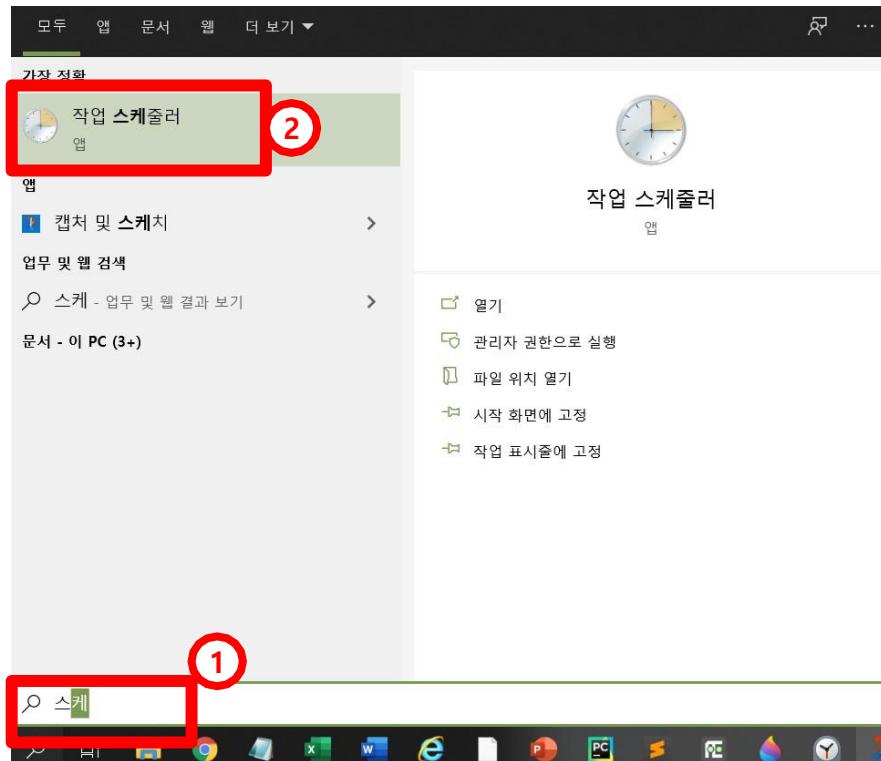
```
file_name.bat
```

▶ 2-3 전체 프로세스 자동화 처리

- 윈도우 스케줄러 실행

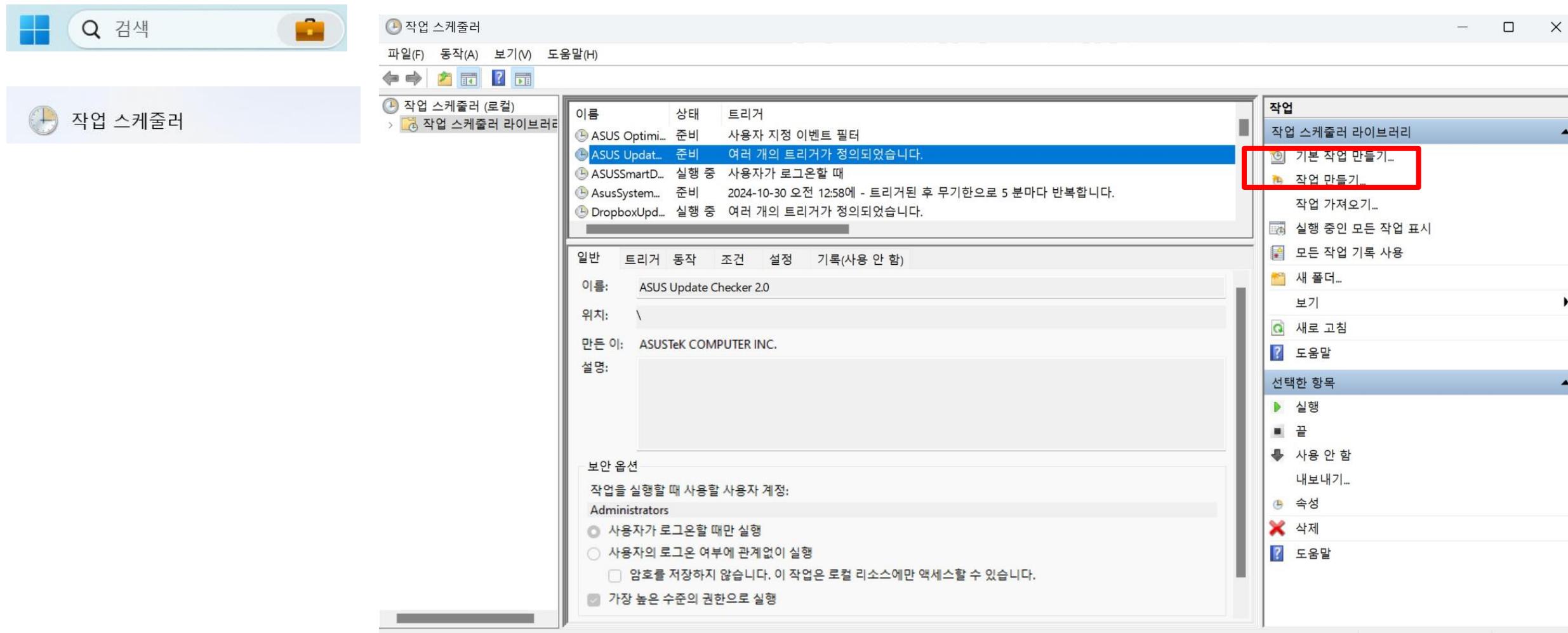
반복적인 작업을 일정에 맞추어 실행하는 윈도우 '작업 스케줄러'를 사용하여 수집작업을 규칙적으로 실행하겠습니다.

윈도우 검색에서 '스케줄러'입력 후 작업 스케줄러 실행

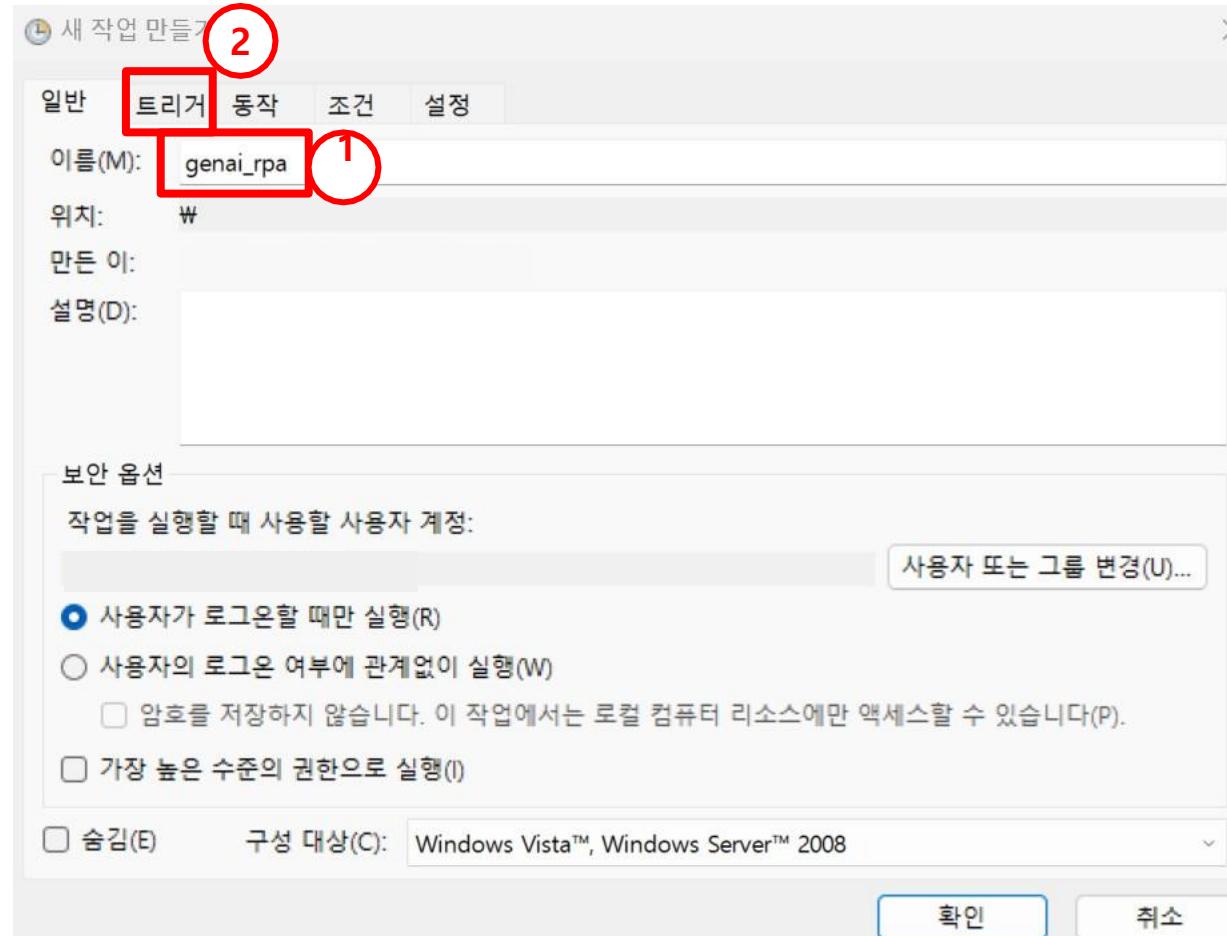


▶ 2-3 전체 프로세스 자동화 처리

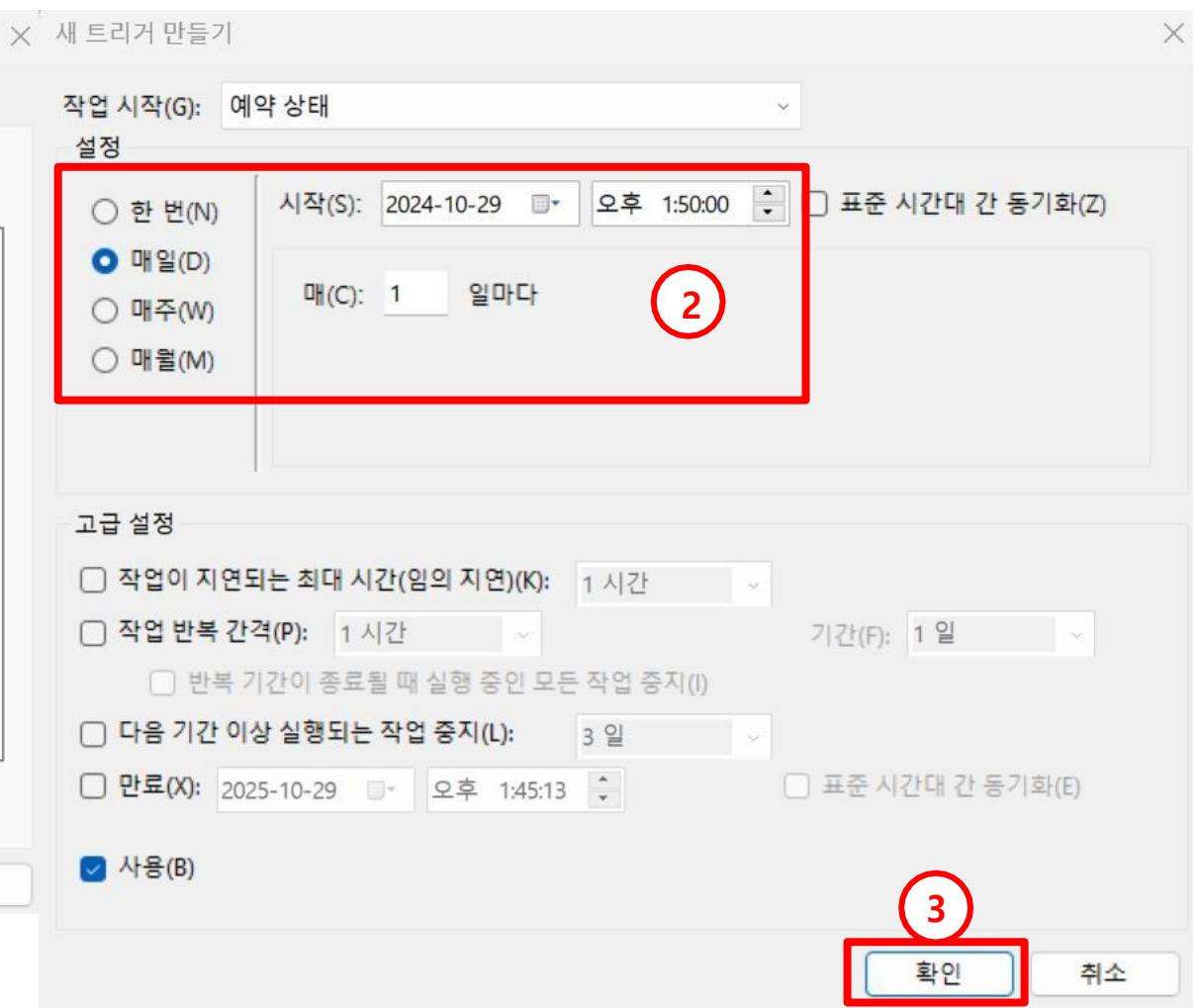
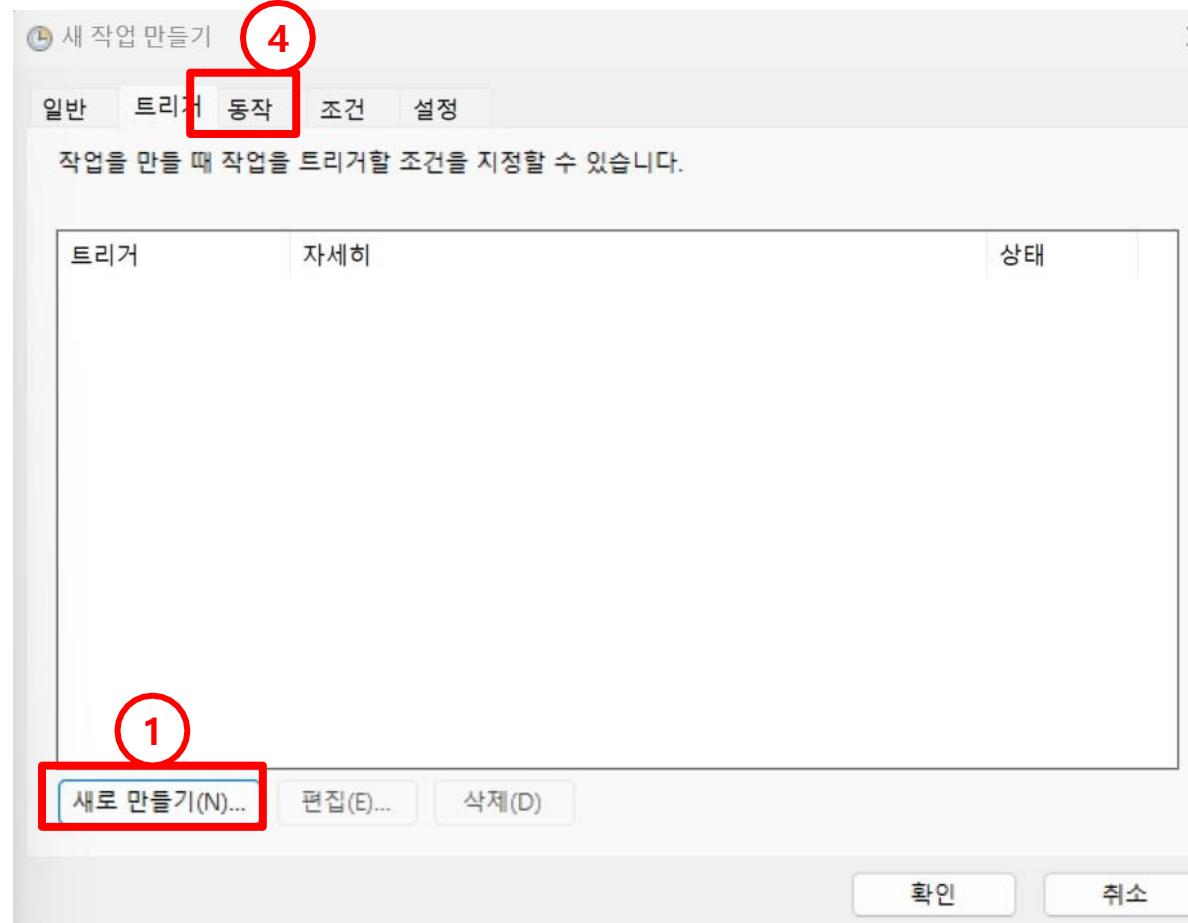
- 작업 스케줄러



▶ 2-3 전체 프로세스 자동화 처리



▶ 2-3 전체 프로세스 자동화 처리



▶ 2-3 전체 프로세스 자동화 처리

