

[KISTEP] 실무 데이터 분석 자동화 — Excel 고급 활용부터 AI 코딩까지

▼ 강사소개

🏆 로직플로우 대표

- 기능이 아닌, 시간과 성장을 제공
- 로직플로우 아키텍트(Logic Flow Architect)
- www.logicflow.ai.kr

▶ 일잘러스쿨(유튜브) 운영

- 엑셀, 한글, AI, 자동화 관련 교육 영상 업로드
- <https://www.youtube.com/@iljal-school>
- <https://open.kakao.com/o/gRZdvU6e>(오픈 카톡)
- 프밍쌤(오픈 프로필)
- 프밍쌤과 열혈 학생들(네이버 카페)
- 문의전화: 010-9088-8741

📖 전공

- 건국대학교 교육공학과 학부, 석사 졸업
- 건국대학교 교육공학과 박사 과정 수료

👩 강의

- ChatGPT와 파이썬으로 쉽게 시작하는 업무자동화(멀티캠퍼스)
- [Ctrl+AI] 2일만에 끝내는 파이썬 업무 자동화 프로젝트 with ChatGPT(멀티캠퍼스)
- 챗GPT를 활용한 업무 자동화(한국과학기술정보연구원(KISTI))
- 신입직원 OS역량 강화 교육
- AI활용 역량 강화를 위한 교직원 연수(공주교육대학교)
- AI활용 실무: 사무자동화 마스터 클래스(YMCA)
- AI활용 사무자동화 클래스(파이썬)
- AI활용 사무자동화 클래스(엑셀 VBA)
- 생성형 AI로 맛보는 파이썬 자동화
- 함수 없이 쉽게 배우는 엑셀
- 도서관 사서를 위한 엑셀 실무 교육
- 일잘러스쿨 영상 제작
 - AI와 함께하는 VBA
 - AI와 함께하는 파이썬
 - 함수 없이 배우는 엑셀
 - 선생님 질문 있어요!

⚙ 자동화 프로그램 개발

- 국내 대기업 DX프로젝트 자동화 프로그램 개발(VBA with chatGPT)
- 데이터 기반 공정지도 대시보드 개발(VBA with chatGPT)
- 나라장터 키워드 검색 및 트래킹 시스템 개발(파이썬 with Claude)
- 역량 측정 보고서 자동 생성 프로그램 개발(파이썬 with Cursor)
- 지출결의서 내역 비교하여 누락된 내역 찾기(파이썬 with Claude)
- 엑셀로 정리된 지출결의 내용 회사 시스템에 자동 업로드하기(VBA with chatGPT)
- 카드승인번호로 매출 전표에서 필요한 영수증 자동 캡처 저장(오토핫키, 파이썬)
- 교육평가결과 자동 계산 시스템 개발(엑셀)
- 교육 운영 후 통계 자료 자동 생성(엑셀)
- 주간 업무 보고 자료 자동 생성(엑셀)
- 만족도 조사 결과 보고서 내용 자동 작성(엑셀)
- 전체 수료증 자동 출력 파일(엑셀)
- 엑셀에 작성된 일정표를 구글 캘린더로 입력(오토핫키)
- 카드내역 전각/반각 문자 자동 변환(VBA with chatGPT)

▼ 강의 내용

Excel 고급 활용

- 표(Table)의 이해
- 피벗 테이블 기초
- 피벗 테이블 활용(고급)
- 동적 대시보드 만들기

AI 코딩(VBA)

- AI 기반 자동화 개념 이해(VBA개요)
- 반복 업무 자동화 사례 이해
- 자동화 코드 만들기(VBA)실습
- 나만의 치트키 Add-in 만들기

▼ Excel 고급 활용

학습 목표

1. 엑셀의 가치를 새롭게 인식한다.
2. 롱폼과 와이드폼을 구분 할 수 있다.
3. 피벗테이블을 이용하여 다양한 형태의 분석을하고 원하는 형식으로 가공할 수 있다.

▼ 0. 엑셀은 어떤 프로그램일까?

▼ 엑셀의 정의

일반 사용자가 데이터(표)를 쉽게 저장, 가공, 분석, 처리 할 수 있도록 만들어진 프로그램

▼ 1. 피벗 테이블

- 많은 데이터를 원하는 기준에 따라 분류하고 요약하는 기능
- 원본 데이터를 직접 수정하지 않고 분석 결과 생성
- 필드를 배치하는 방식에 따라 다양한 관점의 분석 가능
- 복잡한 수식을 작성하지 않고 합계, 개수, 평균 등의 결과 확인 가능

피벗테이블로 할 수 있는 분석

- **그룹별 집계**
 - 데이터를 특정 기준으로 나누어 합계, 평균, 개수, 최댓값, 최솟값 등을 계산
 - 하나의 원본 데이터에서 다양한 분류 기준 적용 가능
- **교차 분석**
 - 두 개 이상의 분류 기준을 행과 열에 배치하여 항목 간 관계 비교
 - 여러 조건이 결합된 결과를 표 형태로 확인
- **계층별 분석**
 - 상위 항목에서 하위 항목으로 세분화하여 분석
 - 연도 → 분기 → 월, 지역 → 지점과 같은 단계별 구조 표현
- **시계열 분석**
 - 날짜 데이터를 연도, 분기, 월 등의 기간 단위로 그룹화
 - 시간의 흐름에 따른 변화와 추세 확인
- **구성비 분석**
 - 각 항목이 전체, 행 합계 또는 열 합계에서 차지하는 비율 계산
 - 항목별 상대적 비중 비교
- **증감 분석**
 - 이전 기간이나 이전 항목과의 차이 계산
 - 증감액과 증감률을 이용한 변화 정도 확인
- **누계 분석**
 - 일정한 순서에 따라 값을 계속 누적하여 계산
 - 기간별 누적 실적이나 진행 현황 확인
- **순위 분석**
 - 집계 결과를 기준으로 항목별 순위 계산
 - 상위 항목과 하위 항목을 빠르게 구분
- **조건별 분석**
 - 필터와 슬라이서를 이용하여 특정 조건에 해당하는 데이터만 분석
 - 원본 데이터를 삭제하지 않고 분석 범위 변경
- **다중 지표 비교**

- 하나의 피벗테이블에서 합계, 평균, 개수 등 여러 계산 결과를 함께 표시
- 서로 다른 성격의 분석 지표를 동일한 분류 기준으로 비교

피벗테이블의 주요 영역

영역	역할	예시
행	데이터를 세로 방향으로 분류	지점, 상품군, 담당자
열	데이터를 가로 방향으로 분류	월, 분기, 판매채널
값	숫자 데이터를 계산하고 요약	매출액 합계, 주문 건수
필터	전체 분석 결과에 조건 적용	연도, 지역, 회원등급

- 같은 필드라도 배치 위치에 따라 역할이 달라짐
- 행과 열: 데이터를 구분하는 기준
- 값: 합계, 평균, 개수 등을 계산할 대상
- 필터: 전체 결과에서 원하는 조건만 선택

피벗테이블의 기본 구조

- 원본 데이터의 열 제목이 피벗테이블의 필드로 사용됨
- 열 제목을 행, 열, 값, 필터 영역에 배치하여 분석
- 필드 배치를 변경하면 분석 결과도 즉시 변경됨
- 하나의 원본 데이터로 여러 형태의 분석 가능

원본 데이터 예시

판매일자	지점	상품군	상품명	수량	매출액
2026-07-01	여주점	패션	셔츠	2	100,000
2026-07-01	파주점	잡화	가방	1	80,000
2026-07-02	여주점	패션	바지	3	150,000

지점별 매출 분석

- 행: 지점
- 값: 매출액
- 결과: 지점별 매출액 합계

지점별·상품군별 매출 분석

- 행: 지점
- 열: 상품군
- 값: 매출액
- 결과: 각 지점의 상품군별 매출액 비교

핵심 원리

- 피벗테이블의 분석 항목은 원본 데이터의 열을 기준으로 구성

- 분석하고 싶은 기준이 원본 데이터에서 하나의 열로 존재해야 함
- 원본 데이터의 구조가 분석 결과에 직접적인 영향을 줌

▼ 고급 스킬) 계산 필드

- 피벗테이블의 기존 필드를 이용해 새로운 계산 결과를 만드는 기능
- 원본 데이터에 계산 열을 추가하지 않고 피벗테이블 안에서 계산
- 일반 피벗테이블에서 사용

계산 필드 예시

원본 필드:

- 매출액
- 원가액

계산 필드:

이익액 = 매출액 - 원가액

계산 필드의 특징

- 원본 데이터의 필드 이름을 계산식에 사용
- 계산 결과가 피벗테이블의 새로운 값 필드로 추가
- 행, 열, 필터의 분석 기준에 따라 결과 집계
- 한 개의 원본 표를 이용한 간단한 계산에 적합
- 원본 데이터에 별도의 계산 결과 열을 만들지 않아도 됨

원본 계산 열과 계산 필드

원본 계산 열

- 원본 데이터의 각 행에서 결과 계산
- 피벗테이블 외부에서도 계산 결과 사용 가능
- 데이터 확인, 필터링, 다른 수식 참조 가능
- 테이블의 자동 채우기 기능 활용 가능

매출액	원가액	이익액
100,000	60,000	40,000

계산 필드

- 피벗테이블 안에서만 계산 결과 생성
- 원본 데이터에는 계산 결과가 저장되지 않음
- 분석 결과에 필요한 계산을 간단하게 추가
- 기존 피벗테이블 필드를 이용해 계산

▼ 2. 롱폼과 와이드폼

데이터는 배치 방식과 사용 목적에 따라 **롱폼**과 **와이드폼**으로 구분

- 롱폼: 데이터 분석에 적합한 데이터베이스형 자료
- 와이드폼: 사람이 결과를 확인하기 편한 보고서형 자료

I 데이터 구조의 두 가지 형태			
롱폼(Long Form)			와이드폼(Wide Form)
이름	월	매출	
홍길동	1월	100	
홍길동	2월	120	
홍길동	3월	130	
김철수	1월	90	
김철수	2월	95	
김철수	3월	110	

데이터 구조의 두 가지 형태		
구분	롱폼(Long Form)	와이드폼(Wide Form)
구조 개념	한 행(Row)이 하나의 관측값(Observation)을 의미	하나의 행(Row)에 여러 변수(Column)가 포함
형태	세로로 긴 형태	가로로 넓은 형태
용도	통계, 시각화, 그룹별 분석	사람이 보기 편한 보고용
파이썬 활용성	Pandas, Seaborn, Plotly 등에서 선호	일부 엑셀 작업에만 적합

롱폼: 데이터베이스형 자료

- 동일한 종류의 값을 하나의 열에 저장하는 구조
- 하나의 행에 하나의 거래 또는 기록을 입력
- 데이터가 아래 방향으로 계속 추가되는 형태
- 피벗테이블, 파워쿼리, 함수, 데이터 모델을 이용한 분석에 적합

상품군	지점	매출액
패션	여주점	55,000,000
패션	파주점	45,000,000
패션	시흥점	38,000,000
패션	부산점	32,000,000
식음	여주점	21,000,000
식음	파주점	18,000,000

로프의 특징

- 한 행은 하나의 기록
- 한 열은 하나의 항목
- 동일한 종류의 값은 하나의 열에 저장
- 첫 번째 행에 한 줄의 열 제목 사용
- 데이터 중간에 빈 행과 빈 열을 사용하지 않음
- 병합 셀을 사용하지 않음
- 원본 데이터에 중간 합계와 전체 합계를 넣지 않음
- 새로운 데이터는 아래쪽에 행으로 추가
- 하나의 셀에는 하나의 값만 입력

로프의 장점

- 항목별 필터링 가능
- 그룹별 합계, 평균, 개수 계산 가능
- 여러 기준을 이용한 교차 분석 가능
- 피벗테이블의 행, 열, 필터에 필드를 자유롭게 배치 가능
- 새로운 데이터가 추가되어도 기존 열 구조 유지
- 다른 테이블과 공통 항목을 기준으로 관계 설정 가능
- 파워쿼리를 이용한 반복적인 데이터 처리에 적합

와이드폼: 보고서형 자료

- 구분 항목이 가로 방향의 여러 열로 배치된 구조
- 집계된 결과를 사람이 한눈에 비교하기 위한 형태
- 출력, 보고, 공유 등 최종 결과물에 적합

상품군	여주점	파주점	시흥점	부산점
패션	55,000,000	45,000,000	38,000,000	32,000,000
식음	21,000,000	18,000,000	17,000,000	15,000,000

와이드폼의 특징

- 같은 종류의 항목이 여러 열 제목으로 나뉘
- 여러 항목의 결과를 가로 방향으로 비교하기 편리
- 중간 합계와 전체 합계를 포함할 수 있음
- 병합 셀, 빈 행, 빈 열 등을 이용해 시각적으로 구분할 수 있음
- 월, 지점, 연도 등의 항목이 증가할 때 새로운 열이 추가됨
- 사람이 읽기에는 편리하지만 데이터 분석의 원본으로는 제한이 있을 수 있음

와이드폼에서 발생할 수 있는 문제

- 동일한 종류의 값이 여러 열에 분산됨
 - 지점, 월 등의 항목 전체를 하나의 필드로 사용하기 어려움
 - 새로운 항목이 추가될 때마다 열 구조가 변경됨
 - 데이터가 여러 시트나 여러 영역으로 나뉘기 쉬움
 - 피벗테이블의 행과 열에 항목을 자유롭게 배치하기 어려움
 - 항목별 필터와 슬라이서를 만들기 어려움
 - 다른 테이블과 관계를 설정하기 어려움
-

▼ 3. 엑셀 테이블

- 일반 셀 범위를 데이터 관리에 적합한 구조로 변환하는 기능
- **Ctrl + T**를 이용해 생성 가능
- 피벗테이블과 파워쿼리의 원본 데이터 관리에 적합

테이블의 주요 기능

- 열 제목에 필터 자동 추가
- 데이터 추가 시 범위 자동 확장
- 표 서식 자동 적용
- 계산 수식 자동 채우기
- 구조화된 참조 사용
- 테이블 이름 지정
- 요약행과 슬라이서 활용
- 피벗테이블 원본 범위 관리

일반 범위와 테이블 비교

구분	일반 범위	테이블
데이터 추가	참조 범위를 직접 확인	범위 자동 확장
필터	별도로 설정	자동 설정
수식 복사	직접 복사	열 전체에 자동 적용
범위 이름	셀 주소 사용	테이블 이름 사용
새 데이터 서식	직접 적용	기존 서식 자동 적용
피벗 원본 관리	범위 변경 가능성 확인	테이블 범위 자동 반영

구조화된 참조

- 셀 주소 대신 테이블의 열 제목을 이용한 수식
- 수식의 의미를 쉽게 파악 가능
- 새로운 행이 추가될 때 수식 자동 적용

일반 셀 참조:

=F2-G2

테이블의 구조화된 참조:

=[@매출액] - [@원가액]

- **@**: 현재 행
- **[@매출액]**: 현재 행의 매출액
- **[@원가액]**: 현재 행의 원가액

▼ 4. 파워쿼리

- 데이터를 가져오고 정리하고 변환하는 엑셀 기능
- 데이터 탭에서 실행
- 반복적인 데이터 정리 과정을 단계별로 저장
- 원본 데이터는 유지하고 변환된 결과를 별도로 생성

파워쿼리의 주요 기능

- 불필요한 행과 열 제거
- 열 이름 변경
- 데이터 형식 변경
- 빈 값과 오류 처리
- 중복 데이터 제거
- 여러 파일 또는 여러 시트 통합
- 와이드폼을 롱폼으로 변환
- 하나의 열을 여러 열로 분할
- 하나의 열을 여러 행으로 분할
- 여러 표 병합 및 추가

적용된 단계

- 파워쿼리에서 수행한 작업이 순서대로 기록됨
- 각 단계의 작업 내용을 다시 확인하거나 수정 가능
- 원본 데이터가 변경되어도 같은 변환 과정을 반복 적용 가능
- 새로 고침을 통해 최신 데이터를 변환 결과에 반영

파워쿼리를 사용하는 이유

- 반복적인 데이터 정리 작업 감소
- 원본 데이터가 추가되어도 동일한 규칙 적용
- 수작업으로 인한 누락과 오류 감소
- 분석에 적합한 데이터 구조로 변환
- 정리 과정의 재사용 가능

▼ 언피벗

- 가로 방향으로 펼쳐진 여러 열을 세로 방향의 행으로 변환하는 작업
- 와이드폼 데이터를 롱폼 데이터로 변환할 때 사용
- 엑셀 파워쿼리에서 제공하는 데이터 변환 기능

언피벗 전

상품군	여주점	파주점	시흥점	부산점
패션	55,000,000	45,000,000	38,000,000	32,000,000

- 지점명이 각각의 열 제목으로 입력
- 지점별 매출액이 여러 열에 분산

언피벗 후

상품군	지점	매출액
패션	여주점	55,000,000
패션	파주점	45,000,000
패션	시흥점	38,000,000
패션	부산점	32,000,000

- 기존 열 제목이 **지점** 열의 값으로 변환
- 각 지점 열의 숫자가 **매출액** 열의 값으로 변환
- 동일한 종류의 값이 각각 하나의 열에 저장

언피벗의 변환 원리

- 여주점, 파주점, 시흥점, 부산점 → **지점** 열
- 각 지점에 입력된 숫자 → **매출액** 열
- 상품군처럼 변환하지 않은 열 → 각 행에 반복하여 유지

언피벗 후 가능한 분석

- 지점을 피벗테이블의 행 또는 열에 배치
- 지점별 매출액 필터링
- 상품군별·지점별 교차 분석
- 지점 슬라이서 생성
- 새로운 지점 데이터 추가
- 다른 지점정보 테이블과 연결

▼ 하나의 셀에 여러 값이 있는 데이터

- 데이터베이스형 자료의 기본 원칙: 하나의 셀에는 하나의 값
- 여러 값을 하나의 셀에 입력하면 각각의 항목을 독립적으로 분석하기 어려움

분석하기 어려운 구조

주문번호	상품
A001	셔츠, 바지
A002	가방
A003	신발, 모자, 양말

- 셔츠, 바지 전체가 하나의 텍스트로 인식
- 셔츠와 바지를 각각의 상품으로 인식하지 못함
- 쉽표로 구분되어 있어도 자동으로 분리되지 않음

발생하는 문제

- 상품별 주문 건수 계산 어려움
- 특정 상품만 정확하게 필터링하기 어려움
- 상품별 판매수량 집계 어려움
- 상품을 피벗테이블의 개별 행 항목으로 배치하기 어려움
- 상품정보 테이블과 정확한 관계 설정 불가
- 상품명에 입력되는 순서에 따라 서로 다른 값으로 인식

분석 가능한 구조

주문번호	상품
A001	셔츠
A001	바지
A002	가방
A003	신발
A003	모자
A003	양말

- 주문번호는 필요한 만큼 반복
- 하나의 셀에는 하나의 상품만 입력
- 상품별로 독립적인 필터와 집계 가능
- 피벗테이블에서 상품별 주문 건수 분석 가능

열로 분할과 행으로 분할

구분	변환 결과	적합한 상황
열로 분할	값이 오른쪽의 새로운 열에 배치	분리될 항목의 위치와 개수가 일정한 경우
행으로 분할	값이 아래쪽의 새로운 행에 배치	각 값을 동일한 분석 항목으로 사용할 경우

- 상품 개수가 주문마다 다르면 행으로 분할
- 각 상품을 같은 기준으로 집계하려면 행으로 분할
- 전화번호 앞자리·중간자리·뒷자리처럼 역할이 다르면 열로 분할

▼ 5. 여러 테이블 연결하기

- 서로 다른 테이블에 있는 정보를 공통된 코드로 연결하여 분석
- VLOOKUP이나 XLOOKUP으로 값을 가져오지 않고도 여러 테이블의 필드를 함께 사용
- 거래내역, 제품정보, 고객정보처럼 역할에 따라 나뉜 데이터를 그대로 유지

테이블 구성

- 거래내역 테이블
 - 실제 거래가 기록되는 표
 - 같은 제품코드나 고객코드가 여러 번 반복될 수 있음
- 기준정보 테이블
 - 코드별 상세정보를 관리하는 표
 - 제품코드, 고객코드가 한 번씩만 존재해야 함

연결 기준

- 두 테이블에 공통으로 존재하는 열 사용
- 예시
 - 판매내역[제품코드] ↔ 제품정보[제품코드]
 - 판매내역[고객코드] ↔ 고객정보[고객코드]

관계의 구조

- 일반적으로 일대다 관계 사용
- 일 쪽
 - 코드가 한 번만 존재
 - 제품정보, 고객정보
- 다 쪽
 - 같은 코드가 여러 번 반복
 - 판매내역

연결 방법

1. 각 데이터 범위를 엑셀 테이블로 변환
2. 각 테이블에 구분하기 쉬운 이름 지정
3. 피벗테이블을 만들 때 **이 데이터를 데이터 모델에 추가** 선택
4. 데이터 모델에 필요한 테이블을 모두 추가
5. 관계 관리에서 공통 코드 열을 기준으로 연결
6. 데이터 모델을 이용한 피벗테이블 생성
7. 서로 다른 테이블의 필드를 행, 열, 값 영역에 배치

관계 설정 시 확인할 내용

- 일 쪽 테이블의 연결 열에 중복값이 없는지 확인
- 두 연결 열의 데이터 형식이 같은지 확인
- 코드 앞뒤에 불필요한 공백이 없는지 확인
- 코드 표기 방식이 동일한지 확인
- 거래내역의 코드가 기준정보 테이블에 모두 존재하는지 확인

관계 설정 후 가능한 분석

- 제품정보의 상품군별 판매수량 분석
- 고객정보의 지역별 판매수량 분석
- 상품군과 회원등급을 함께 사용한 교차 분석
- 여러 테이블의 필드를 하나의 피벗테이블에서 함께 사용

▼ 고급스킬) 측정값

측정값

- 데이터 모델 피벗테이블에서 사용하는 계산식
- 피벗테이블의 행, 열, 필터 조건에 따라 결과가 자동으로 변경
- 여러 테이블의 값을 관계를 통해 함께 계산 가능

기본 측정값

```
총판매수량 := SUM(판매내역[수량])
```

측정값의 특징

- 피벗테이블의 값 영역에서 사용
- 그룹별 합계와 비율 계산에 적합
- 다른 측정값을 참조하여 추가 계산 가능
- 일반 피벗테이블의 계산 필드보다 여러 테이블 분석에 적합

핵심 정리

- 여러 테이블은 공통 코드로 연결
- 기준정보 테이블의 코드는 중복 없이 관리
- 거래내역 테이블에서는 같은 코드가 반복될 수 있음
- 데이터 모델에 테이블을 추가한 뒤 관계 설정
- 관계 설정 후 서로 다른 테이블의 필드를 피벗테이블에서 함께 사용
- 계산이 필요한 경우 측정값 추가

▼ AI코딩(VBA)

학습 목표

1. 자동화의 필요성을 인식하고 나의 업무 중 자동화 할 수 있는 일을 발견할 수 있다.
 2. AI로 내가 원하는 자동화 코드를 요청할 수 있다.
 3. VBA코드를 실행하고 오류 발생 시 수정할 수 있다.
-

▼ 자동화

자동화란?

내가 하기 귀찮은 일(반복적이고 규칙적인 작업)을 자동으로 처리할 수 있도록 하는 것

▼ 가장 편한 자동화는?



다른 사람에게 시키기

- 시킬 사람이 없으면 기계나 컴퓨터를 써야함
- 예전에는 컴퓨터(엑셀)이나 프로그래밍 언어(파이썬)를 모르면 자동화 하는 것이 어려웠음
- 자연어(말)로 일을 시키면 알아듣는 AI 등장
→ 자동화의 장벽이 낮아짐



컴퓨터에게 시키면 좋은 점

- 반복 작업을 질리지 않고 한다.
- 정해진 규칙대로 일관되게 실행한다.
- 일관적인 결과물이 나온다.
- 내가 안해도 된다.

▼ 우리가 해야 하는 자동화

반복되는 일을 규칙으로 바꾸어 컴퓨터가 대신 실행하게 만드는 것

어떤 업무를 자동화해야 할까?

내가 하기 너무 귀찮고 싫은 일인데 알바생에게 시켜도 될만한 것!

- 동일한 절차가 반복되고 예외 상황이 적은 업무
- 사람이 직접 수행하면 많은 시간을 소모하는 업무
- 오류 발생 가능성이 높은 업무

자동화에 적합하지 않은 업무

창의력과 판단이 요구되는 업무

- 기획, 전략 수립

변화가 잦고 복잡한 업무

- 규칙화하기 어려운 업무
- 예외가 많은 업무

자동화에 대한 두 가지 입장

내가 하는 일에는 자동화 할 만한 것이 없다.	자동화를 하면 모든 것이 버튼 한방에 해결될 것이다.
• 꼭 사람이 해야 하는 일이다. • 내가 하는 일은 너무 복잡하다.	• 어떻게 해야 하는지 모르겠다.



천 리 길도 한 걸음부터

소소하지만 확실한 자동화의 필요성

일상 업무의 누적된 시간 절약

- 1회 5분 소요되는 업무 × 하루 2회 × 연 250일 = 약 42시간 = 약 1주일 워킹 타임
- 자동화로 연간 수 일의 시간 확보 가능

정신적 피로 감소

- 단순 작업 반복 → 업무 만족도 하락
- 자동화로 더 중요한 업무에 집중 가능

오류 예방 효과

- 파일명 규칙, 폴더 정리, 숫자 입력 등에서 실수 잦음
- 자동화로 실수 원천 제거

왜 더 이상 자동화를 미루면 안될까?

- 예전에 비해 자동화 툴을 학습하는데 드는 시간이 획기적으로 줄어들음
- 비정형화된 내용도 자동화 할 수 있음
- 일의 맥락과 절차를 정확히 이해하는 사람과 아닌 사람의 격차가 크게 발생함
- 이제는 내가 무엇을 해야 하는지만 명확히 알면 프로그래밍을 할 수 있는 시대!



Input - Process - Output을 정확히 알면 할 수 있다.

- 스스로 자동화 프로그램을 만들 수 있으면 좋은 점!



나에게 꼭 맞는 프로그램을 만들 수 있다.



**나의 메인 업무부터 시작하기 보다는
쓸 데 없이 반복되는 반복 업무부터 시작하는 것을 추천**

☞ 하나의 작업을 작게 나눠서
각 단계에 대한 명확하고 구체적인 처리 방법을 생각한 뒤
적당한 단계로 합쳐서 완성

▼ 자동화 예시

- 항목별 차트 그리기

<https://youtu.be/fKAKXmOkh8o>

- 홈페이지에서 자료 다운 받기

https://youtu.be/tVL2f-_YVq4

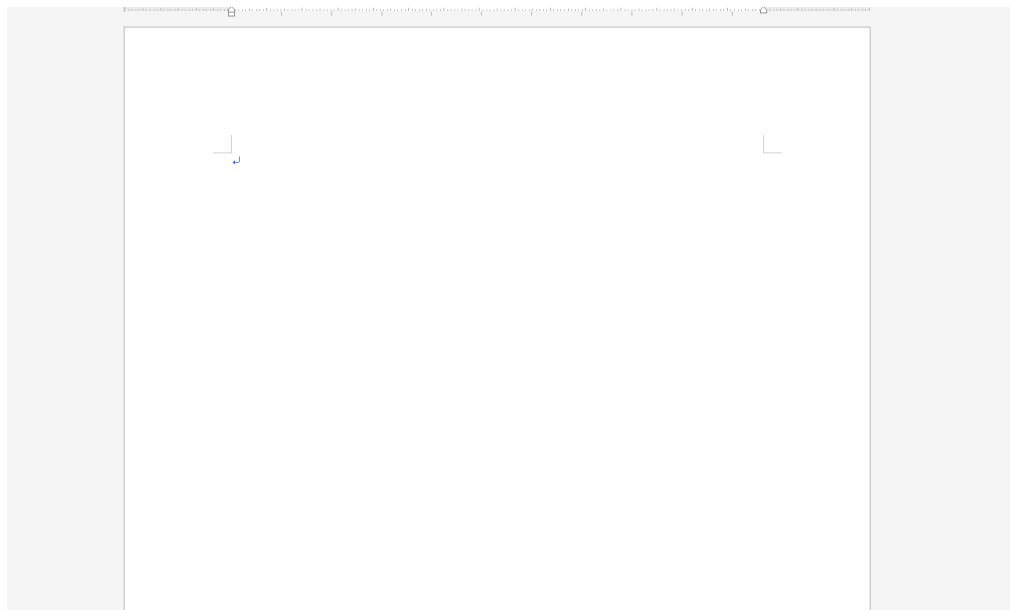
- 엑셀 데이터 한글로 옮기기

<https://youtu.be/aFyFth4GdYg>

- 회의록 데이터베이스 만들기

<https://youtu.be/Omzh2pcplU8>

- 내가 꿈꾸는 자동화
 - 28개 학과에 대한 "학생 역량 진단 보고서 자동 생성"



▼ 보고서 생성 프로그램을 만들기 위한 시간: 2주

- 데이터 전처리
- 로직 구상
- 한글 파일 템플릿 제작
- 기초 통계 분석
- 분석 결과 한글 표에 옮기기
- 차트 생성
- 차트 생성 이미지 한글 파일에 옮기기
- 학과별로 생성된 파일 하나로 합치기

▼ 엑셀 VBA란?

1. 정의

- **VBA (Visual Basic for Applications)**는 Microsoft Office 제품군(특히 Excel, Word, PowerPoint 등)에 내장된 **프로그래밍 언어**로, Microsoft의 **Visual Basic** 언어를 기반으로함

2. 주요 특징

특징	설명
자동화	반복적인 작업(예: 복사/붙여넣기, 보고서 생성 등)을 매크로로 자동화
이벤트 기반	버튼 클릭, 셀 변경 등 엑셀 이벤트에 반응하는 프로그래밍 가능
사용자 정의 함수 (UDF)	엑셀 기본 함수 외에 사용자가 직접 함수를 정의할 수 있음
다른 오피스 앱 제어	Excel을 통해 Word, Outlook 등 다른 Office 앱도 제어 가능
강력한 데이터 처리	반복문, 조건문 등 로직을 활용해 대규모 데이터 처리 가능

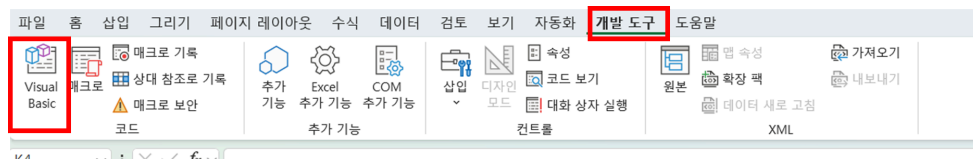
3. VBA를 배워야 하는 이유

- 반복 작업을 줄이고 **업무 효율 극대화**
- 기존 엑셀 기능으로는 할 수 없는 **복잡한 작업 자동화**
- 실무에서 요구되는 **맞춤형 자동화 솔루션** 제공 가능
- **프로그래밍적 사고 능력 향상**

4. VBA 사용 방법

1. Excel 열기

2. **Alt + F11** → **VBA 편집기(Visual Basic Editor)** 열기
or 개발 도구 > Visual Basic



개발도구가 없을 경우: 파일 > 옵션 > 사용자 리본 메뉴 > 개발 도구 체크

3. 메뉴 > **삽입** > **모듈** 선택
4. 원하는 코드 작성
5. **F5** 키 또는 실행 버튼 클릭

5. 주의사항

- **Ctrl+Z안됨**: 실수로 데이터가 삭제되거나 손상될 수 있으므로 **백업 필수**
- VBA코드가 삽입된 파일은 xlsm파일로 저장해야 코드가 함께 저장됨
- VBA는 **보안 경고**가 뜰 수 있으므로 공유시 미리 공지하는 것이 좋음

- Mac 버전 Excel에서는 일부 기능이 제한될 수 있음

6. 소소하지만 유용한 팁

- 나만의 VBA파일을 만들어보자
 - 대부분의 경우 ActiveSheet를 기준으로 작동하는데, 이 경우 VBA코드가 다른 파일에 저장되어있어도 실행이 가능하다.
- 버튼을 적극 활용하자
 - VBA를 실행 할 때 매번 VBA편집기를 열어서 실행하기 불편하다.
 - 단축키는 외우기 힘들고 제한이 있다.
 - 도형에 VBA코드를 쉽게 연결 할 수 있으므로 버튼을 적극 사용하자
- 서식 복사를 이용하자
 - 서식과 관련된 작업을 할 경우 원본 파일이 서식이 없는 상태였다면 아무 빈 셀을 선택하여 서식 복사를 해서 붙여넣으면 서식이 없는 상태로 되돌아간다.
- XLSM파일을 공유할때는 ZIP파일을 이용하자
 - VBA코드는 컴퓨터를 제어 할 수 있는 코드이기 때문에 나쁜 마음을 먹으면 바이러스를 심거나 다른 사람의 컴퓨터를 망쳐놓는 것도 가능하다.
 - 따라서 메일이나 메신저로 공유 할 때 형식이 제한이 되는 경우가 많다.
 - 이럴때 ZIP파일로 저장해서 송부 할 수 있다.
 - 상대방이 VBA를 쓸 줄 안다면 TEXT파일로 코드만 따로 보내는 것도 방법이다.

▼ VBA 코드 만들기 실습

▼ VBA 코드 생성을 위한 프롬프트 템플릿

- 주석은 컴퓨터 언어를 사람이 이해할 수 있는 해설이라고 생각하면됨
- 주석을 자세하게 요청하면 코드를 몰라도 어느 정도 수정이 가능함

너는 엑셀 **VBA**전문가야.
다음의 기능을 하는 **VBA**코드를 만들어줘.
주요 과정마다 주석으로 번호를 매겨주고 어떤 작업을 하는것인지 주석을 최대한 자세
히 남겨줘
오류 원인을 추적할 수 있도록 주요 단계별 실행 로그를 남겨줘.
내가 알려줘야 하는 내용이 있다면 물어봐서 작업해줘.
코드를 만들기 전에 로직을 설명해줘.

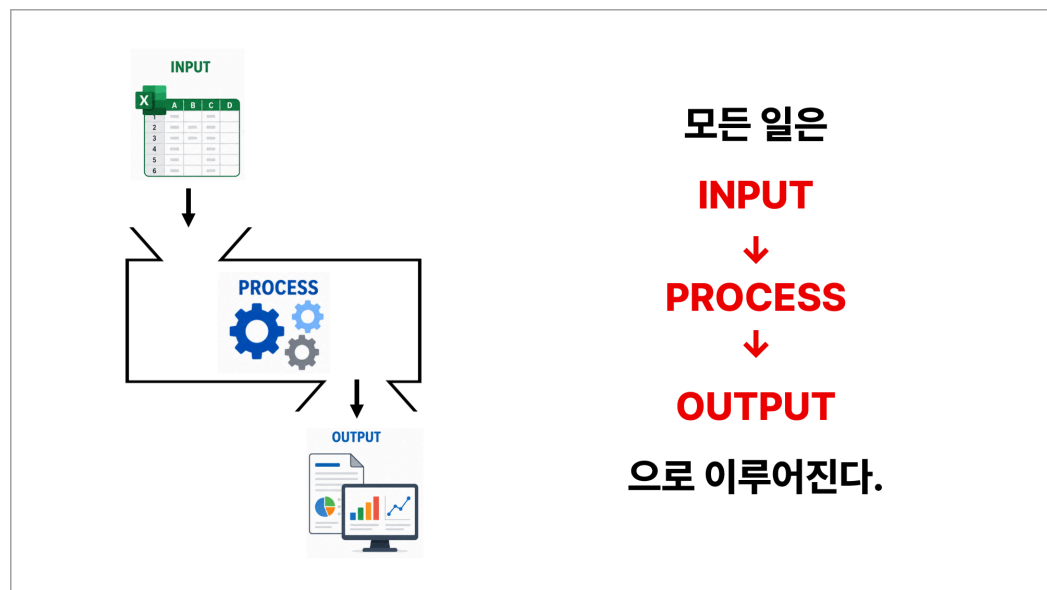
목적:

input

process

output

자동화의 기본 공식: Input → Process → Output



자동화는 복잡해 보여도 결국 세 가지 질문으로 정리할 수 있습니다.

구분	질문	예시
Input	무엇을 넣을 것인가?	엑셀 파일, 폴더, 문서, 메일
Process	어떤 규칙으로 처리할 것인가?	합치기, 분류하기, 계산하기, 추출하기
Output	무엇을 얻을 것인가?	새 엑셀 파일, 보고서, 차트, 메일 초안

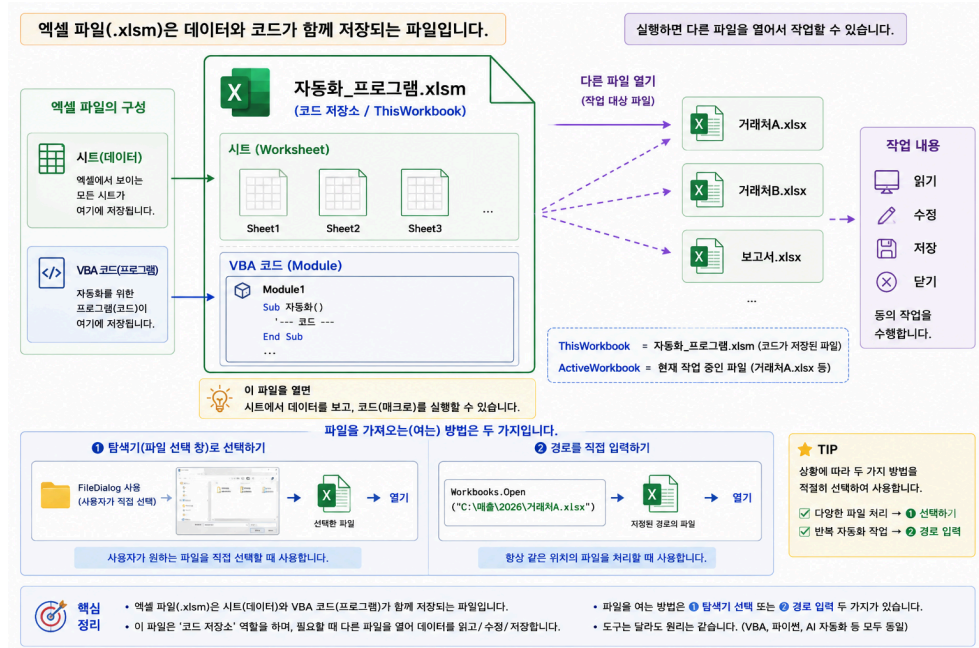
예를 들어 “여러 엑셀 파일을 하나로 합치고 싶다”는 업무는 이렇게 나눌 수 있습니다.

구분	내용
Input	폴더 안에 있는 여러 개의 엑셀 파일
Process	각 파일의 첫 번째 시트를 읽고, 같은 열 기준으로 아래에 붙이기
Output	하나로 합쳐진 통합결과.xlsx 파일

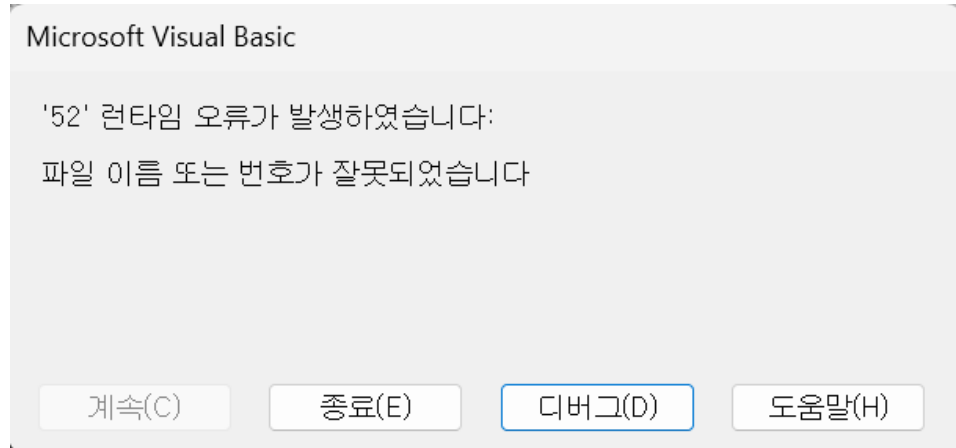
AI에게 자동화를 요청할 때도 결국 이 세 가지를 설명해야 합니다.

무엇을 넣고, 어떻게 처리하고, 어떤 결과를 원하는지가 명확할수록 AI가 더 정확한 코드를 만들어줍니다.

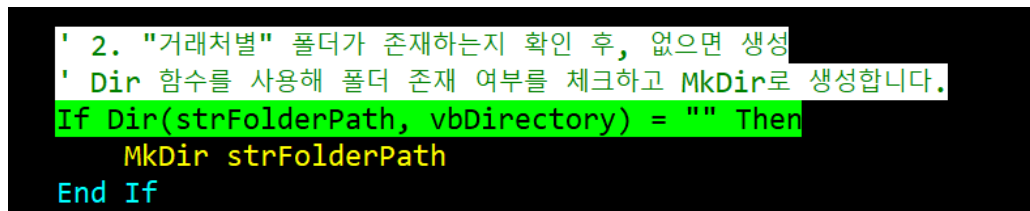
▼ VBA 기초



▼ 오류 발생 시 대응 방법



- win+shift+S(캡처)로 오류메세지를 캡처하여 채팅창에 붙여넣고 “디버그”를 누른 다음 하이라이트 된 부분을 복사해서



```
If Dir(strFolderPath, vbDirectory) = "" Then
```

이 부분에 첨부한 오류가 발생해

▼ 오류는 아닌데 내가 원하는 대로 작동하지 않을 경우

- 현재 작업이 일어난 결과의 상황을 설명하고 내가 원하는 상황을 다시 구체적으로 설명
- 내가 의도한 결과물과 현재 결과물을 이미지로 첨부하여 설명하면 더 쉽게 해결 가능

▼ 1 누락된 시트 찾기



누락된 엑셀 시트 이름 찾기 시연 영상

- 자료와 비교해서 누락된 시트 찾기
- [거래처 리스트]
삼성전자, LG화학, 현대모비스, 네이버, 카카오, 롯데제과, 한화에어로스페이스, CJ제일제당, 포스코인터내셔널, SK하이닉스

다음의 기능을 하는 VBA코드를 만들어줘.

주요 과정마다 주석으로 번호를 매겨주고 어떤 작업을 하는것인지 주석을 최대한 자세
히 남겨줘

오류 원인을 추적할 수 있도록 주요 단계별 실행 로그를 남겨줘.

내가 알려줘야 하는 내용이 있다면 물어봐서 작업해줘.

코드를 만들기 전에 로직을 설명해줘.

목적:

input

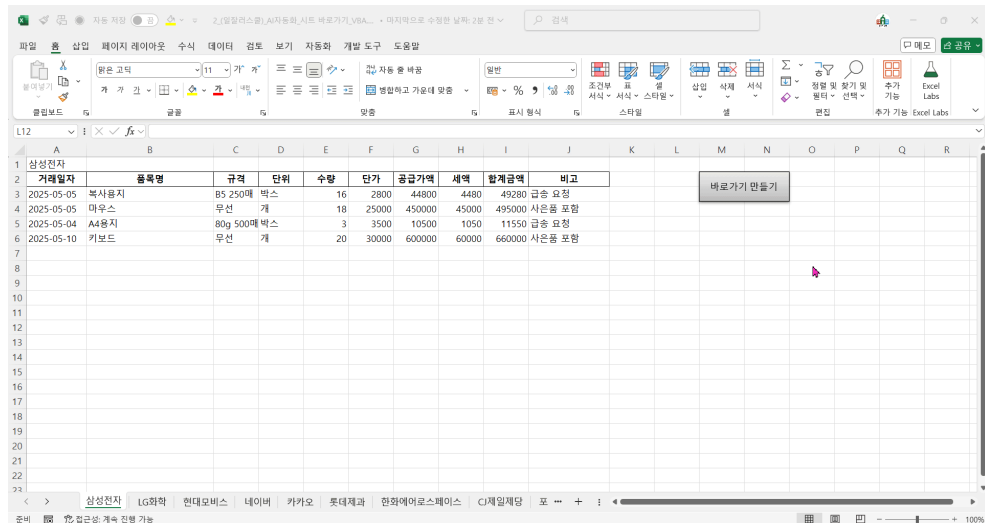
process

output

▼ 2_시트이동링크 만들기

시트 링크걸기 시연 영상

- 시트가 많은 파일에서 이동이 불편함
- 시트 맨 앞에 새로운 시트를 추가하고 시트명 입력
- 시트명을 누르면 해당 시트로 이동가능
- 모든 시트의 B1셀에 첫번째 시트로 이동하는 링크 추가



너는 엑셀 VBA전문가야.
다음의 기능을 하는 VBA코드를 만들어줘.
주요 과정마다 주석으로 번호를 매겨주고 어떤 작업을 하는것인지 주석을 최대한 자세
히 남겨줘
내가 알려줘야 하는 내용이 있다면 물어봐서 작업해줘.
코드를 만들기 전에 로직을 설명해줘.

목적: 시트간 이동을 쉽게 하기

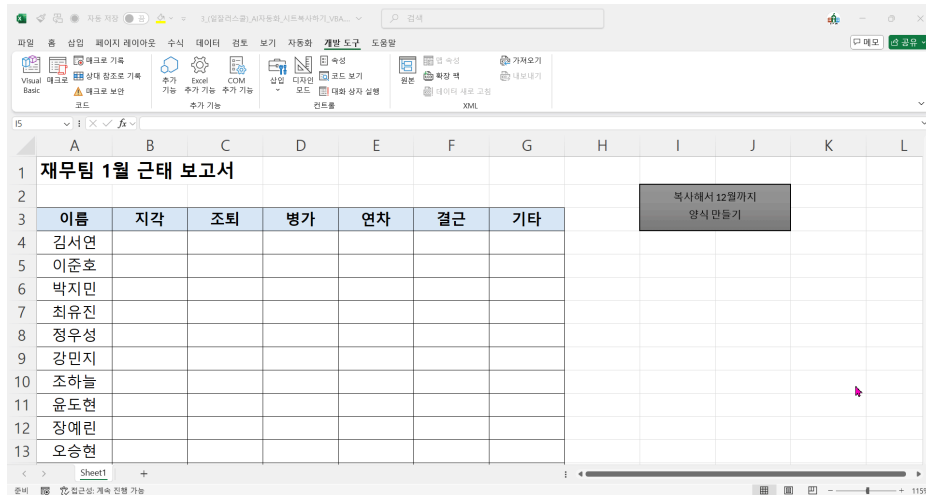
input

현재 파일

process

1. 시트 맨 앞에 새로운 시트 "목록"추가
2. "목록"시트의 A1셀에 "시트바로가기"입력
3. A2셀부터 세로로 시트의 이름 입력
4. 시트 이름에 해당하는 시트로 이동하는 링크 걸기
5. 모든 시트의 B2셀에 "목록"시트로 돌아오는 바로가기 만들기

▼ 3 시트 복사하기



- 같은 형식의 시트를 복사해서 1월부터 12월까지 개별 시트로 만들어보자.
 - A1셀에는 "재무팀 M월 근태 보고서"라는 텍스트가 들어가야 한다.
 - 모든 시트의 형식은 그대로 유지되어야 한다.
 - 시트명은 1월부터 12월까지 순차적으로 입력되어야한다.
 - (원본 시트는 삭제하지 말고 숨기기 처리 한다.)

다음의 기능을 하는 VBA코드를 만들어줘 .

주요 과정마다 주석으로 번호를 매겨주고 어떤 작업을 하는것인지 주석을 최대한 자세히 남겨줘

오류 원인을 추적할 수 있도록 주요 단계별 실행 로그를 남겨줘 .

내가 알려줘야 하는 내용이 있다면 물어봐서 작업해줘 .

코드를 만들기 전에 로직을 설명해줘 .

목적:

input

process

output

▼ 4 데이터합치기

- 같은 형식으로 되어있는 데이터를 세로 방향으로 합쳐보자.(어펜드)
 - 파일들은 하나의 폴더 안에 들어있다.
 - 모든 파일의 형식은 동일하다.
 - 거래일자 순으로 오름차순 정렬한다.
 - 사용된 모든 열의 폭을 자동 맞춤 한다.
 - 헤더는 하늘색으로 칠한다.

다음의 기능을 하는 VBA코드를 만들어줘 .
주요 과정마다 주석으로 번호를 매겨주고 어떤 작업을 하는것인지 주석을 최대한 자세히 남겨줘
오류 원인을 추적할 수 있도록 주요 단계별 실행 로그를 남겨줘 .
내가 알려줘야 하는 내용이 있다면 물어봐서 작업해줘 .
코드를 만들기 전에 로직을 설명해줘 .

목적:

input

process

output

▼ 5 데이터 필터링해서 나누기

- 거래처명 별로 데이터를 나누어보자.
 1. "취합데이터"시트에 있는 자료를 거래처명별로 필터링해서 각각의 시트로 저장한다.
 2. 해당 파일에 있는 모든 시트를 시트명을 파일명으로 하는 각각의 파일로 저장한다.

다음의 기능을 하는 VBA코드를 만들어줘 .
주요 과정마다 주석으로 번호를 매겨주고 어떤 작업을 하는것인지 주석을 최대한 자세히 남겨줘
오류 원인을 추적할 수 있도록 주요 단계별 실행 로그를 남겨줘 .
내가 알려줘야 하는 내용이 있다면 물어봐서 작업해줘 .
코드를 만들기 전에 로직을 설명해줘 .

목적:

input

process

output

▼ VBA를 추천하는 경우

1. 다중 시트/파일 자동 처리

추천 이유

- 엑셀의 기본 기능만으로는 조건에 따라 시트를 자동 생성하거나 숨기는 기능이 없음
- 파일 간 데이터 연결이나 처리 작업을 함수로 구현하려면 비효율적임
- 조건 분기와 반복문을 조합하여 다양한 조건별 자동화 로직 구현이 가능함
- 외부 파일을 열고, 특정 데이터를 불러와 저장하는 작업을 자동으로 수행 가능함

사용 예시

- 부서별로 나뉜 데이터를 각각의 시트로 자동 분류
- 특정 키워드를 포함한 시트만 선택하여 PDF로 저장
- 폴더 안의 여러 엑셀 파일을 자동으로 열어 지정 셀의 값만 수집

2. 이벤트 기반 자동 실행

추천 이유

- 셀 변경, 시트 선택, 파일 열기 등의 이벤트에 반응하는 기능은 함수로 구현 불가함
- 사용자의 액션에 따라 자동으로 동작하는 기능을 구현할 수 있음
- 조건 충족 시 자동 메시지, 팝업, 데이터 기록 등 다양한 반응을 설계할 수 있음
- 반복되는 실수나 누락을 이벤트 감지를 통해 방지할 수 있음

사용 예시

- 특정 열에 입력이 생기면 자동으로 날짜 기록
- 시트가 선택될 때 공지사항이나 안내 메시지 자동 표시
- 값이 변경되면 변경 내용을 로그 시트에 자동 저장

3. 반복 작업 자동화

추천 이유

- 수천 개의 셀을 한 번에 조건에 따라 처리하려면 함수보다 반복문이 효과적임
- 반복 계산이나 형식 변경 작업을 자동으로 수행 가능함
- 고정된 규칙에 따라 대량 작업을 간단한 코드로 처리 가능함
- 실수를 줄이고 일관된 결과를 도출할 수 있음

사용 예시

- 특정 단어가 포함된 행만 자동으로 강조 표시
- 각 직원별 보고서를 양식에 맞게 자동 채우기

- 대량 데이터 정리 및 변환 작업을 클릭 한 번으로 실행

4. 외부 앱 또는 파일 시스템 제어

추천 이유

- 엑셀 기능만으로는 Word, Outlook 등의 외부 프로그램을 제어할 수 없음
- 폴더 생성, 파일 이동, 파일명 변경 같은 작업을 엑셀 안에서 자동화할 수 있음
- 다양한 오피스 앱을 엑셀에서 제어하여 연동형 업무 자동화를 실현할 수 있음
- 사용자의 개입 없이 문서 생성, 저장, 메일 전송 등의 업무를 처리할 수 있음

사용 예시

- 견적서를 Word로 생성하고 자동으로 Outlook을 통해 메일 발송
- 특정 경로에 있는 파일을 날짜 기준으로 정리 및 백업
- 회의록 파일을 엑셀 내용 기반으로 자동 생성 후 저장

5. 행/열 추가 또는 삭제

추천 이유

- 엑셀 기본 기능에서는 여러 조건에 따라 동적으로 행이나 열을 추가/삭제하는 것이 매우 제한적임
- 특정 조건을 만족하는 데이터만 필터링하여 행 또는 열을 삽입하거나 제거하는 자동화가 필요할 때 유용함
- 수작업으로 하면 시간이 오래 걸리는 반복적인 구조 편집을 클릭 한 번으로 처리할 수 있음
- 반복문과 조건문을 활용해 다양한 패턴의 삽입/삭제 작업을 유연하게 제어할 수 있음

사용 예시

- 특정 값이 있는 행만 찾아 삭제하거나, 그 위에 요약 행 자동 삽입
- 격주마다 새로운 열을 삽입하고 제목 행을 자동 구성
- 데이터 입력이 완료된 마지막 행 아래에 계산 행 자동 추가
- 빈 셀이 일정 수 이상 존재하는 열은 자동으로 제거하여 보고서 정리

▼ VBA 애드인

애드인용코드.txt

애드인이란 무엇인가

- | 엑셀 기능을 확장하기 위해 추가로 로드하는 파일
- 엑셀 기본 기능 외에 사용자 정의 기능을 추가하기 위한 파일
- 보통 VBA 매크로, 사용자 함수(UDF), 자동화 기능을 포함
- 파일 확장자: .xlam (Excel Add-in)
- 일반 엑셀 파일(.xlsx, .xlsm)과 달리 직접 열어서 사용하는 것이 아니라 엑셀에 로드하여 사용
- 설치 후 엑셀을 실행할 때 자동으로 기능이 활성화 가능
- 조직 내에서 공통 매크로 배포할 때 많이 사용

애드인의 장점

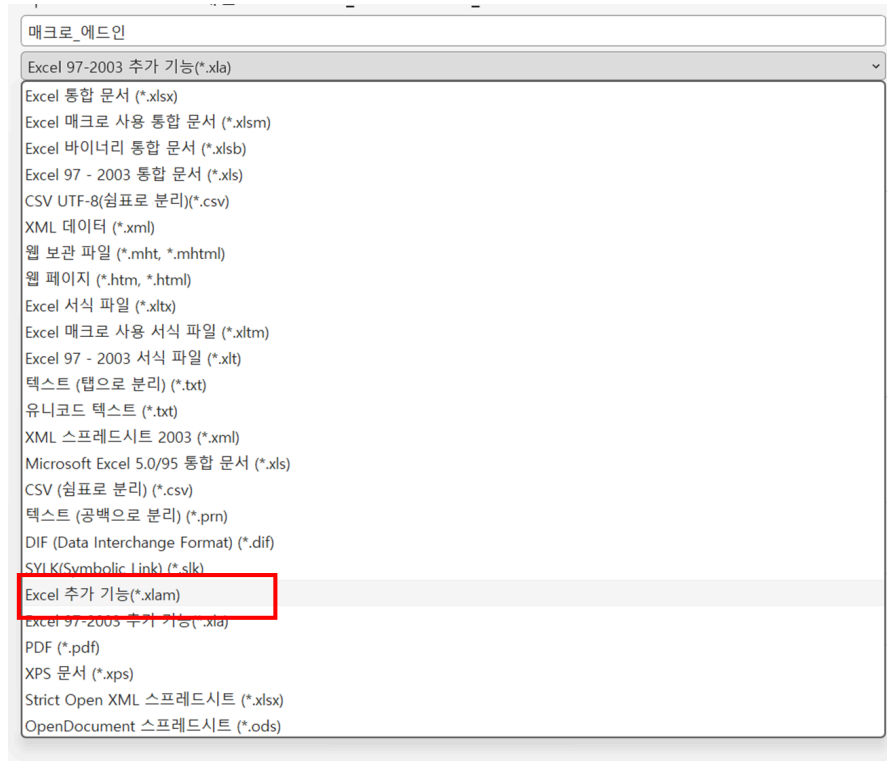
- | 매크로를 중앙화하여 여러 파일에서 재사용 가능
- 여러 엑셀 파일에서 동일한 VBA 기능 사용 가능
- 매크로 파일을 별도로 열 필요 없음
- 사용자 정의 함수(UDF)를 일반 함수처럼 사용 가능
- 코드 수정 시 애드인 파일만 수정하면 전체 적용
- 기업 환경에서 매크로 배포에 유리

VBA 코드 준비

- | 애드인으로 만들기 전에 VBA 코드가 포함된 엑셀 파일을 준비
- Excel에서 매크로 포함 파일(.xlsm) 생성
- 등록하려는 코드가 잘 작동하는지 확인

애드인 파일로 저장

- | 기존 매크로 파일을 Add-in 형식으로 저장
- Excel 메뉴 → 파일(File)
- 다른 이름으로 저장(Save As) 선택
- 파일 형식 선택
 - **Excel Add-in (*.xlam)**

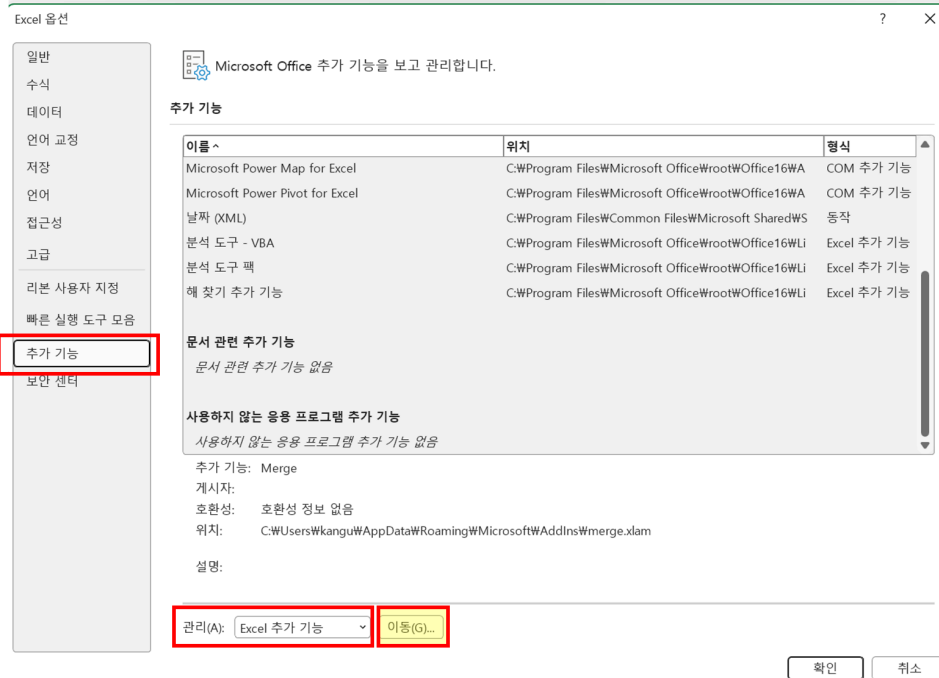


- 저장 위치
 - 보통 AddIns 폴더 권장
 - C:\Users\[사용자컴퓨터이름]\AppData\Roaming\Microsoft\AddIns
 - AppData폴더가 숨겨져 있을 수 있음
 - 윈도우 탐색기 > 보기 > 숨긴 항목

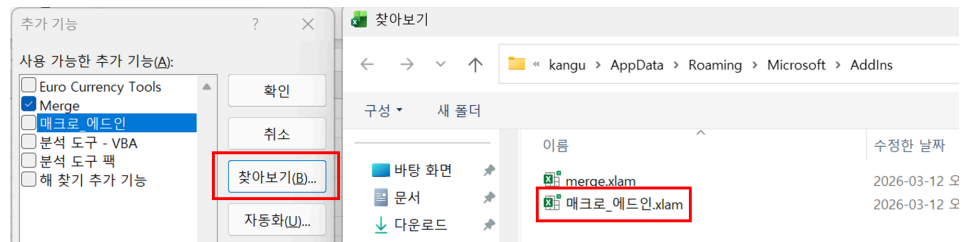
Excel 애드인 등록

| 저장한 .xlam 파일을 Excel에 등록

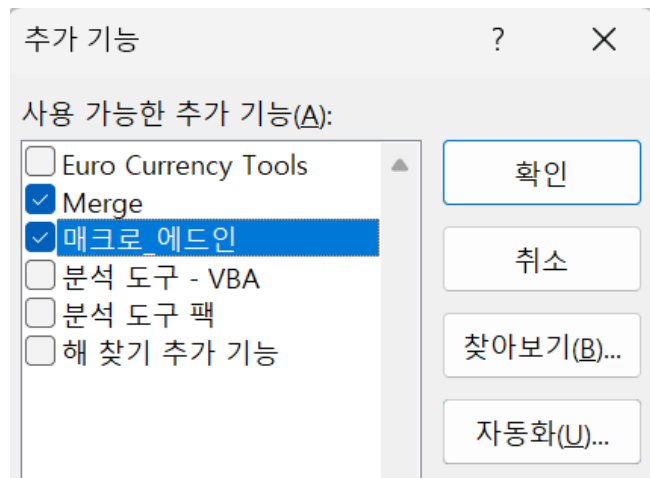
- Excel 실행
- 파일(File) → 옵션(Options)
- 추가 기능 → 관리 → "Excel 추가 기능"선택 → "이동"클릭



- “찾아보기 클릭” → 저장한 .xlam 파일 선택



- 체크박스 활성화



- 확인(OK) 클릭

Excel 애드인 리본메뉴 추가

추가한 매크로 기능을 리본메뉴에 추가

- Excel 실행
- 파일(File) → 옵션(Options)
- 리본 사용자 지정 → 새 탭 → 이름 바꾸기 → 새 그룹 → 이름 바꾸기
- 명령선택: 매크로
- 원하는 매크로 이름 선택 후 "추가"버튼 눌러서 원하는 위치에 추가
- 확인

