농업빅데이터조사원 실무 교안

데이터 처리 업무를 위한 AI 활용

수기 야장에서 컨설팅 자료까지, 반복 작업을 줄이는 법

수기 야장 입력·오류 검증·2주 주기 컨설팅 자료 작성의 반복 작업 줄이기. 실제 화면을 활용한 AI 실습과 **자동화 범위·검토 기준** 확인

일시	**2026년 9월 18일(금) 09:00 ~ 10:30**
대상	**농업빅데이터조사원**
주최	**경상북도농업기술원**
실습자료	**01_**데이터검증 · **02_**보고서자동화 · **03_**가상데이터 · **04_**한글변환

목차

1장 **시작하기 전에**
이 교안을 쓰는 법 · 우리가 다루는 데이터 · 파일 형식 · 준비물 확인

2장 **눈으로는 못 잡는 오류**
엑셀이 만드는 사고 · 센서가 만드는 사고 · 오류 8가지 유형

3장 **AI에 일을 시키는 세 가지 방법**
질문·코드 생성·파일 업로드의 차이와 데이터 전달 범위

4장 **ChatGPT 프로젝트로 반복 업무 묶기**
화면 살펴보기 · 프로젝트 만들기 · 지침 이해하기 · 2주 주기 돌리기

5장 **실습 A · 데이터 검증 자동화**
219건을 3초에 · 리포트 읽는 법 · 규칙 고치기 · 실패담

6장 **실습 B · 컨설팅 보고서 자동화**
현행 자료와 비교 · 달라진 7가지 · 기준 바꾸기

7장 **실습 C · 한글(HWP) 파일로 내보내기**
액션 테이블 971개 · 결재용 문서 자동 생성 · AI에게 시키는 법

8장 **개인정보와 AI**
우리 데이터의 개인정보 · 원칙 한 문장 · 안전한 질문법 · 체크리스트

부록 A **명령어 요약** — 복사해서 쓰는 실행 명령

부록 B **AI 프롬프트 모음** — 상황별로 그대로 쓰는 문장

부록 C **오류 유형 사전** — 검증기가 잡는 17가지

부록 D **실습 정답지** — 가상 농가에 심어 둔 오류 10건

부록 E **파일 구성** — 어느 파일을 언제 고치나

화면·수치와 실습 데이터의 출처

- 화면·분석 수치: 제공받은 샘플 데이터의 실행 결과
- 추가 실습 자료: 오류를 넣어 만든 가상 농가 데이터 포함
- 실행 결과 비교: 동일한 데이터·설정·실행 환경을 기준으로 확인

CHAPTER 01

시작하기 전에

교안 활용 방법과 실습 데이터 구조 확인

1.1 이 교안을 쓰는 법

- 학습 방식: **명령 실행 → 화면 변화 확인 → 결과 검토**
- 활용 방법: 설명을 읽은 뒤 실습 파일로 직접 실행

표시	뜻
해보기	직접 실행할 명령과 확인할 결과
주의	오류 발생 조건과 확인 사항
보안	개인정보 처리 시 준수 사항
이런 글씨	명령어·파일명·셀 주소

1.2 우리가 다루는 데이터

실습 대상: **오이·포도**의 농가정보·기상환경·생육기본·생육수확 자료

자료	수집 주기	한 파일의 크기	주요 항목
농가정보	1회성 (설문)	6~9행	농가명·면적·품종·정식일·재식밀도
기상환경	1시간 간격 (센서)	5,448행	온도·습도·CO2·지온·일사량
생육기본	1~2주 간격 (현장)	15,768행	초장·마디·엽장·엽폭·착과수
생육수확	수확기 (현장)	653행	과중·과장·과폭

주의 · 작목별 열 이름 차이

- 오이 열 구성: `농가명 | 조사일자` 로 시작
- 포도 열 구성: `시도 | 시군구 | 농가명 | 조사포장 주소 | 조사일` 로 시작
- 통합 전 확인: **조사일자 / 조사일** 등 같은 뜻의 열 이름 통일

1.3 파일 형식별 특징과 용도

확장자: 파일명 마지막 점 뒤의 형식 표시. 저장 형식에 따라 서식·수식·여러 시트의 지원 여부 차이

확장자	무엇인가	서식·수식	여는 곳	어디에 쓰나
`.xlsx`	여러 시트를 담는 엑셀 통합문서	있음	엑셀	야장 원본 · 검증 결과
`.xls`	옛 엑셀(2003 이전). 65,536행까지만 들어갑니다	있음	엑셀	필요 시 xlsx로 변환
`.csv`	쉼표로 값을 구분한 **텍스트 파일**. 하나의 표 저장	없음	엑셀 · 메모장	**AI에게 결과를 받을 때**
`.txt`	일반 텍스트 파일	없음	메모장	메모 · 실행 기록
`.hwp`	한글 문서	있음	한글	결재 문서 (7장)
`.pdf`	인쇄용 배치를 유지하는 문서	편집 도구 필요	어디서나	농가 배포용 (6장)
`.py`	파이썬 소스 코드(텍스트 파일)	없음	메모장 · VSCode	규칙 고치기 (5장)

주고받을 때는 CSV, 보관할 때는 XLSX

- **CSV**: 프로그램 간 표 데이터 교환에 활용
- 확인 사항: 가져오기 시 인코딩과 열별 자료형 지정
- **XLSX**: 색·수식·여러 시트를 포함한 최종본 보관

주의 · CSV 가져오기의 인코딩·자료형 확인

- 한글 깨짐 예방: 엑셀 **데이터 → 텍스트/CSV에서**로 가져오기 → 원본 인코딩에 맞게 지정(UTF-8 파일은 `65001: 유니코드(UTF-8)`)
- 번호 자동변환 예방: 가져오기 창에서 **개체번호 열을 텍스트로 지정** → 2.1절의 날짜 변환 사례 참고

1.4 실습 준비 — 파이썬 확인 및 필수 패키지 설치

최초 준비: 명령 프롬프트(`cmd`)에서 아래 명령 실행

```
# 파이썬이 깔려 있는지 확인
python --version

# 필요한 도구 설치 (엑셀 읽기 · 한글 조작)
pip install pandas openpyxl pywin32
```

- `python --version` 오류 발생 시: 파이썬 설치 여부와 실행 경로 확인
- 설치가 어려운 경우: **5.8절 엑셀 대체 방법**으로 범위·중복·자료형 검사

폴더 구조

```
실습자료/
├── 01_데이터검증/      ← 5장
├── 02_보고서자동화/    ← 6장
├── 03_가상데이터/      ← 실습용 자료 만들기
├── 04_한글변환/        ← 7장
└── 데이터_6농가/       ← 실습 데이터 (B=실제, 나머지=가상)
```

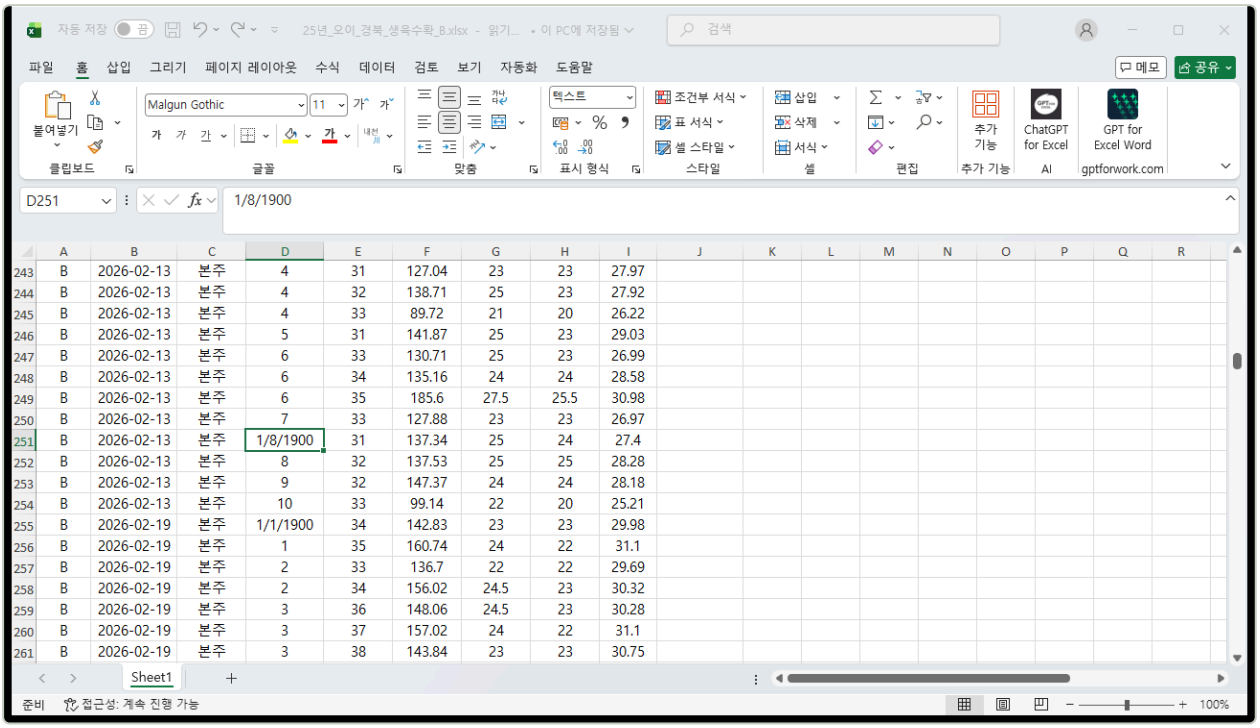
CHAPTER 02

눈으로는 못 잡는 오류

실제 데이터의 오류 사례와 수작업 검사의 한계 확인

> **관찰 과제:** 실제 제출된 오이 생육수확 파일에서 잘못 입력된 값 찾아보기

2.1 엑셀 서식 오류 — 개체번호의 날짜 변환



	A	B	C	D	E	F	G	H	I
243	B	2026-02-13	본주	4	31	127.04	23	23	27.97
244	B	2026-02-13	본주	4	32	138.71	25	23	27.92
245	B	2026-02-13	본주	4	33	89.72	21	20	26.22
246	B	2026-02-13	본주	5	31	141.87	25	23	29.03
247	B	2026-02-13	본주	6	33	130.71	25	23	26.99
248	B	2026-02-13	본주	6	34	135.16	24	24	28.58
249	B	2026-02-13	본주	6	35	185.6	27.5	25.5	30.98
250	B	2026-02-13	본주	7	33	127.88	23	23	26.97
251	B	2026-02-13	본주	1/8/1900	31	137.34	25	24	27.4
252	B	2026-02-13	본주	8	32	137.53	25	25	28.28
253	B	2026-02-13	본주	9	32	147.37	24	24	28.18
254	B	2026-02-13	본주	10	33	99.14	22	20	25.21
255	B	2026-02-19	본주	1/1/1900	34	142.83	23	23	29.98
256	B	2026-02-19	본주	1	35	160.74	24	22	31.1
257	B	2026-02-19	본주	2	33	136.7	22	22	29.69
258	B	2026-02-19	본주	2	34	156.02	24.5	23	30.32
259	B	2026-02-19	본주	3	36	148.06	24.5	23	30.28
260	B	2026-02-19	본주	3	37	157.02	24	22	31.1
261	B	2026-02-19	본주	3	38	143.84	23	23	30.75

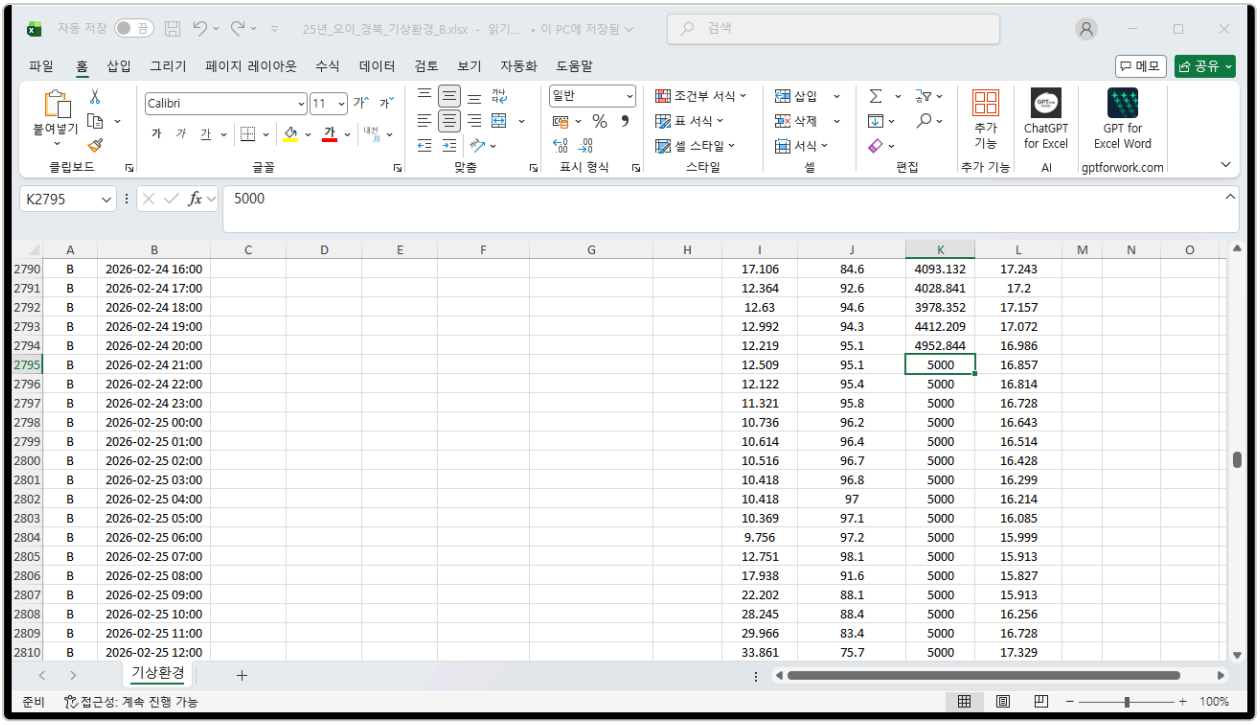
그림 2-1 원본 파일의 개체번호 열(D)에 표시된 `1/8/1900`. 주변 번호 순서와 비교해 확인

- 입력값: **개체번호 8**
- 셀 서식: 날짜 → 표시값 `1900년 1월 8일`
- 원인: 엑셀의 날짜 일련번호를 개체번호에 적용한 서식 오류

이 파일 한 개에만 14건

- 검출 건수: 전체 **653행 중 14건**
- 복원 후보: 날짜 일련번호를 **1~10의 개체번호**로 변환
- 확인 방법: 같은 조사일의 번호 순서·누락 번호·원본 야장과 대조

2.2 센서 이상 의심 — 상·하한값의 연속 기록



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
2790	B	2026-02-24 16:00							17.106	84.6	4093.132	17.243			
2791	B	2026-02-24 17:00							12.364	92.6	4028.841	17.2			
2792	B	2026-02-24 18:00							12.63	94.6	3978.352	17.157			
2793	B	2026-02-24 19:00							12.992	94.3	4412.209	17.072			
2794	B	2026-02-24 20:00							12.219	95.1	4952.844	16.986			
2795	B	2026-02-24 21:00							12.509	95.1	5000	16.857			
2796	B	2026-02-24 22:00							12.122	95.4	5000	16.814			
2797	B	2026-02-24 23:00							11.321	95.8	5000	16.728			
2798	B	2026-02-25 00:00							10.736	96.2	5000	16.643			
2799	B	2026-02-25 01:00							10.614	96.4	5000	16.514			
2800	B	2026-02-25 02:00							10.516	96.7	5000	16.428			
2801	B	2026-02-25 03:00							10.418	96.8	5000	16.299			
2802	B	2026-02-25 04:00							10.418	97	5000	16.214			
2803	B	2026-02-25 05:00							10.369	97.1	5000	16.085			
2804	B	2026-02-25 06:00							9.756	97.2	5000	15.999			
2805	B	2026-02-25 07:00							12.751	98.1	5000	15.913			
2806	B	2026-02-25 08:00							17.938	91.6	5000	15.827			
2807	B	2026-02-25 09:00							22.202	88.1	5000	15.913			
2808	B	2026-02-25 10:00							28.245	88.4	5000	16.256			
2809	B	2026-02-25 11:00							29.966	83.4	5000	16.728			
2810	B	2026-02-25 12:00							33.861	75.7	5000	17.329			

그림 2-2 기상환경 파일의 CO2 열. `5000`이 연속 기록된 구간으로, 센서 측정 범위와 상태 확인 대상

- 관찰값: **5000ppm 16회 연속·100ppm 23회 연속**
- 확인 사항: 센서의 측정 범위·교정 상태·수집 기록
- 판정 시 유의: 연속 고정값을 이상 의심 구간으로 분류한 뒤 원인 확인

왜 문제인가

- 영향: 이상값을 포함한 주차별 평균과 컨설팅 결과의 왜곡 가능성
- 검사 대상: 총 **5,448행**의 기상환경 자료
- 검사 방법: 범위 검사와 연속 고정값 검사를 함께 적용

2.3 주요 오류 유형 8가지

검사 결과: 제공받은 샘플 **10개에서 219건** 검출. 주요 유형별 분류는 아래 표 참고

유형	실제 사례	눈으로?
서식 자동변환	개체번호 → 1/8/1900	거의 불가
센서 고정	CO2 5000ppm × 16회	거의 불가
단위 불일치	비가림높이 정의서 cm ↔ 입력 m (100배)	어려움
자료형 혼입	추비수량 "10L", 관수량 "1시간반"	가능
날짜 논리	끝수확 05-11 < 정식 10-03	가능
표기 흔들림	"스마트팜" vs "스마트"	어려움
구조 오류	화학비료일자_2차 열이 2개	어려움
파일 간 모순	생육=적심 완료 ↔ 농가정보 적심일자 공란	불가

전체 검증 규칙 17가지: **부록 C** 참고

CHAPTER 03

AI에 일을 시키는 세 가지 방법

AI 활용 방식 3가지 비교: 질문·코드 생성·파일 업로드

	방식	예	농가 데이터	재사용
①	**물어보기**	“엑셀에서 중복 찾는 함수 알려줘”	안 나감	매번 질문
②	**코드를 짜게 하기**	“이런 형식의 파일에서 오류 찾는 코드 짜 줘”	안 나감	**계속 씀**
③	**데이터를 올리기**	엑셀 파일을 통째로 대화창에 첨부	나감	매번 업로드

학습 대상: ② 코드 생성

- 학습 방식: 원본 데이터 대신 **열 이름·자료형·판정 기준**을 전달해 코드 생성
- 데이터 처리: 생성한 코드를 검토한 뒤 내 PC에서 실행
- 재사용: 다음 회차의 데이터 형식과 기준을 확인한 뒤 다시 실행

3.1 데이터 형식과 판정 기준만 전달하기

질문 작성: 실제 데이터 행을 붙여 넣지 않고 열 구성과 검사 조건만 제시

```
# 이렇게 (안전)
오이 생육조사 엑셀 파일이 있습니다. 열 구성은 이렇습니다.
  농가명(문자) | 조사일자(날짜) | 개체번호(1~10 정수) | 초장(cm)
  | 마디번호(정수) | 엽장(cm) | 엽폭(cm)

개체번호가 1~10을 벗어나거나 숫자가 아닌 행을 찾아
엑셀로 목록을 내보내는 파이썬 코드를 짜 주세요.
```

```
# 이렇게는 하지 마세요 (위험)
[25년_오이_경북_생육기본_B.xlsx 파일 첨부]
여기서 오류 좀 찾아줘
```

주의 · 생성 코드의 사전 검증

- 사전 검증: **정답을 아는 연습 파일**로 실행
- 결과 대조: 누락된 오류·정상값의 오탐 여부 확인
- 불일치 시: 코드와 규칙을 수정한 뒤 재검증
- 연습 자료: **5.8절** 및 **부록 D 정답지** 참고

CHAPTER 04

ChatGPT 프로젝트로 반복 업무 묶기

공통 지침을 저장해 2주 주기 업무의 반복 설명 줄이기

> **프로젝트 활용 목적**: 데이터 형식·판정 기준·출력 형식을 함께 관리하고 회차별 요청에 재사용

어느 그림이 실제 화면인가

- 실제 화면: **그림 4-1·4-3·4-4·4-6**
- 개념도: 구조와 흐름을 설명한 **그림 4-2·4-5**
- 개인정보 처리: 계정·다른 프로젝트 이름 등을 회색으로 가림(8장 참고)
- 화면 확인: 버전·계정에 따라 메뉴 이름과 위치가 달라질 수 있음

4.1 화면 살펴보기

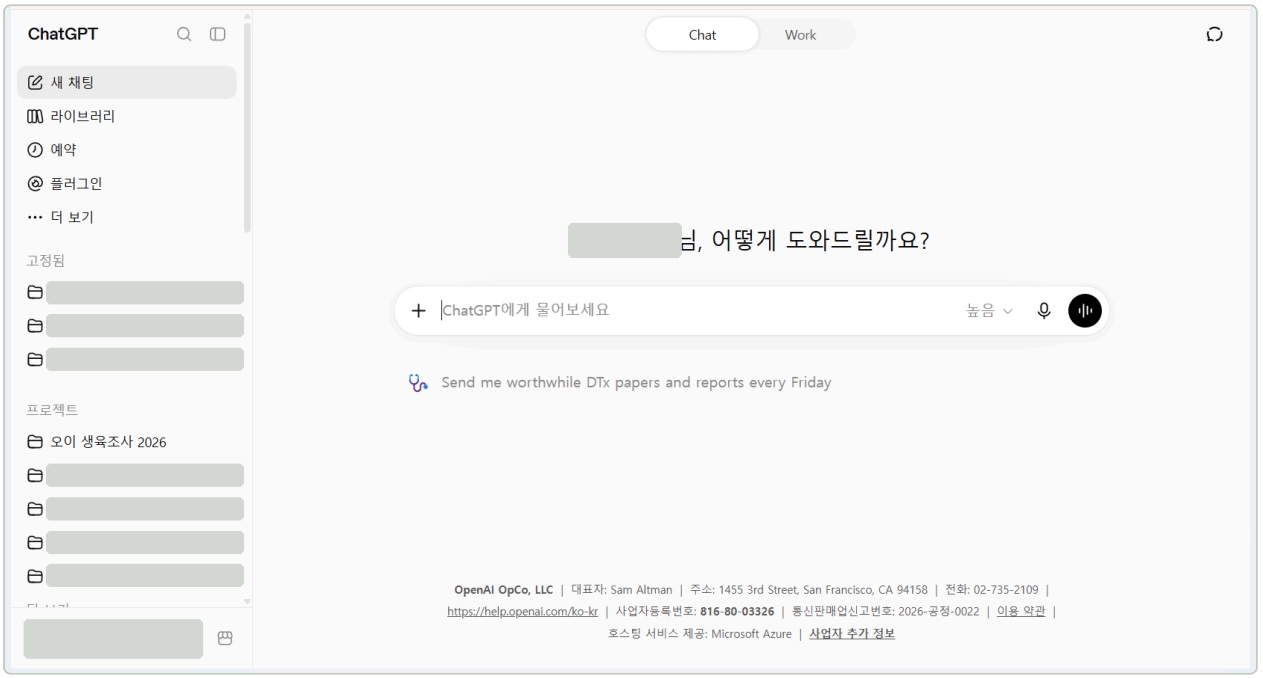


그림 4-1 촬영 당시 ChatGPT 화면. 왼쪽: 새 채팅·라이브러리·고정됨·프로젝트·채팅 / 가운데: 대화 / 아래: 입력창. 프로젝트 예시: `오이 생육조사 2026`. 개인 프로젝트·대화 제목·계정 정보는 회색 처리

회차별 공통 정보 관리

- 회차별 공통 정보: 데이터 형식·판정 기준·출력 형식
- 관리 방법: 공통 정보는 프로젝트 지침과 참고 파일에 정리
- 확인 사항: 이전 대화의 정보가 필요한 경우 해당 내용의 참조 여부 확인

시작 전 설정 보안

- 설정 확인: **설정 → 데이터 제어 → 모든 사용자를 위한 모델 개선** 끄기
- 기관 계정: 설정 변경 가능 여부를 담당 부서에 확인
- 자료 입력: 위 설정과 별도로 기관의 자료 반출 기준 준수

4.2 프로젝트 만들기

프로젝트: 관련 대화·참고 파일·공통 지침을 한곳에서 관리하는 작업 공간

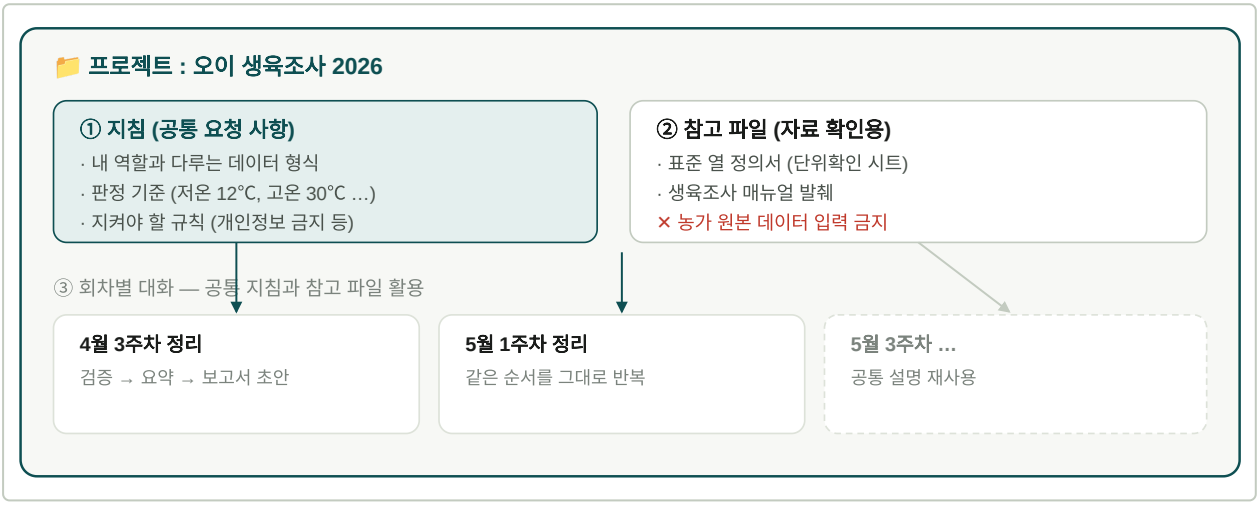


그림 4-2 (개념도) 프로젝트 구성: 공통 지침·참고 파일·회차별 대화. 공통 설명을 저장해 반복 입력 줄이기

만드는 순서 해보기

1. 왼쪽 목록에서 **프로젝트** 옆 **+** 를 누릅니다
2. 이름을 붙입니다 — `오이 생육조사 2026` 처럼 **작목과 연도**가 들어가면 나중에 찾기 쉽습니다
3. **지침**란에 4.3절의 예시를 붙여 넣고 우리 상황에 맞게 고칩니다
4. **파일**에 표준 열 정의서·매뉴얼 발췌를 올립니다. **농가 원본 데이터는 올리지 않습니다**
5. 이제부터 그 프로젝트 안에서 대화를 시작합니다

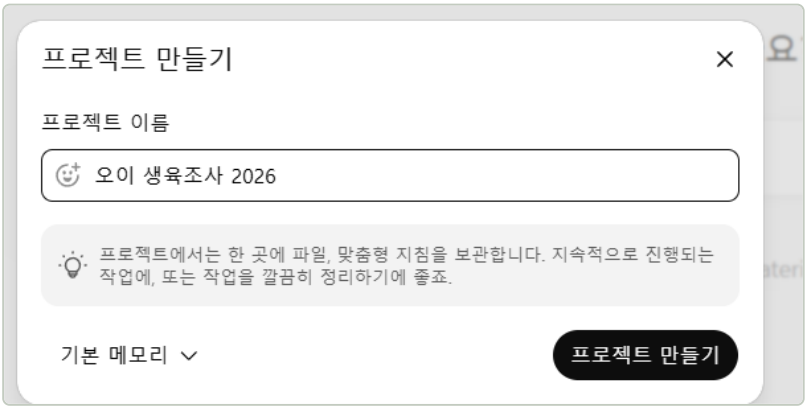


그림 4-3 프로젝트 만들기 창. 프로젝트 이름 입력 후 생성하고 파일·지침 추가

작목별 프로젝트 구성

- 분리 기준: 열 이름과 판정 기준이 다른 작목별로 프로젝트 구성
- 이름 예시: `오이 생육조사 2026` / `포도 생육조사 2026`

4.3 지침(프롬프트) 이해하기

지침: 프로젝트 대화에 적용할 공통 요청 사항. 역할·데이터 형식·판정 기준·준수 규칙의 네 항목으로 구성

담을 것	왜 필요한가	없으면
① **역할·맥락**	내가 누구고 무슨 일을 하는지	업무 맥락을 반영하지 못한 일반적인 답변
② **데이터 형식**	열 이름·단위·값의 범위	열 이름을 지어냅니다
③ **판정 기준**	무엇을 높다/낮다로 볼지	회차마다 판정이 달라집니다
④ **답변 형식**	결론 먼저, 근거 수치 포함	길고 두루뭉술해집니다

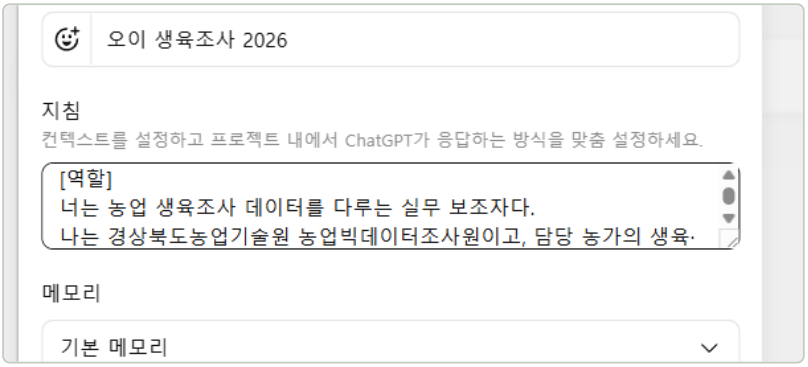


그림 4-4 촬영 당시 지침 입력 위치: 프로젝트 우상단 … → **프로젝트 설정** → **지침** . 아래 예시를 업무에 맞게 수정해 입력

그대로 쓰는 지침 예시

사용 방법: 프로젝트 지침란에 복사한 뒤 대괄호 안을 담당 업무에 맞게 수정

```
[역할]
너는 농업 생육조사 데이터를 다루는 실무 보조자다.
나는 경상북도농업기술원 농업빅데이터조사원이고,
담당 농가의 생육·환경·수확 데이터를 2주마다 정리한다.

[내가 다루는 파일 형식]
· 기상환경 : 농가명 | 측정시간(1시간 간격) | 온도_내부 | 상대습도_내부
              | 잔존CO2 | 토양온도
· 생육기본 : 농가명 | 조사일자 | 덩굴구분 | 개체번호(1~10) | 초장(cm)
              | 마디번호 | 엽장(cm) | 엽폭(cm) | 엽수
· 생육수확 : 농가명 | 조사일자 | 개체번호 | 마디번호 | 수확과중(g)
              | 과장곡선(cm) | 과장직선(cm) | 과폭(mm)
· 농가정보 : 농가명 | 품종 | 식부면적 | 재식밀도 | 재배유형 | 정식기일자
※ 포도는 열 이름이 다르다. 조사일자 → 조사일 이고,
   앞에 시도·시군구·조사포장 주소가 붙는다.
```

```
[판정 기준]  ← 기관 표준 기준으로 교체할 것
· 저온 : 12℃ 미만          · 고온 : 30℃ 초과 (심함 35℃)
· 다습 : 상대습도 90% 초과 · CO2 부족 : 08~16시 400ppm 미만
· 과중 등급 : 99g 이하 / 100~149g / 150~199g / 200g 이상

[지켜야 할 규칙]
1. 농가 성명·전화번호·주소는 다루지 않는다. 농가는 A·B·C 코드로만 부른다.
2. 내가 값을 주지 않으면 숫자를 지어내지 않는다. 모르면 모른다고 한다.
3. 코드는 파이썬으로 주고, 주석은 한국어로 단다.
4. 파일 경로는 내가 알려준다. 임의로 정하지 않는다.

[답변 형식]
· 결론을 맨 앞 세 줄로 먼저 말한다.
· 판단에는 근거 수치를 함께 적는다.
· 조치가 필요하면 우선순위를 매긴다.
```

이 지침이 하는 일

- ② 데이터 형식: 사용할 열 이름 명시
- ③ 판정 기준: 회차별로 적용할 기준 통일
- 규칙 1: 개인정보 입력 금지 명시
- 규칙 2: 근거 없는 수치 생성 금지 명시
- 결과 확인: 지침 준수 여부와 근거 수치를 별도로 검토

구분	지침에 쓰는 문장
나쁨	농업 데이터 분석을 잘 도와줘. 친절하게 설명해 줘.
좋음	개체번호는 1~10 정수다. 이 범위를 벗어나면 오류로 본다. 저온은 12℃ 미만으로 판정한다. 결론을 세 줄로 먼저 말한다.

지침에 넣으면 안 되는 것 보안

- 저장할 내용: **데이터 형식·판정 기준**
- 입력 금지: 농가 성명·연락처·주소 등 실제 개인정보

4.4 프로젝트를 활용한 2주 주기 업무

회차별 요청 항목: **분석 기간·검토된 요약 수치·필요한 산출물**

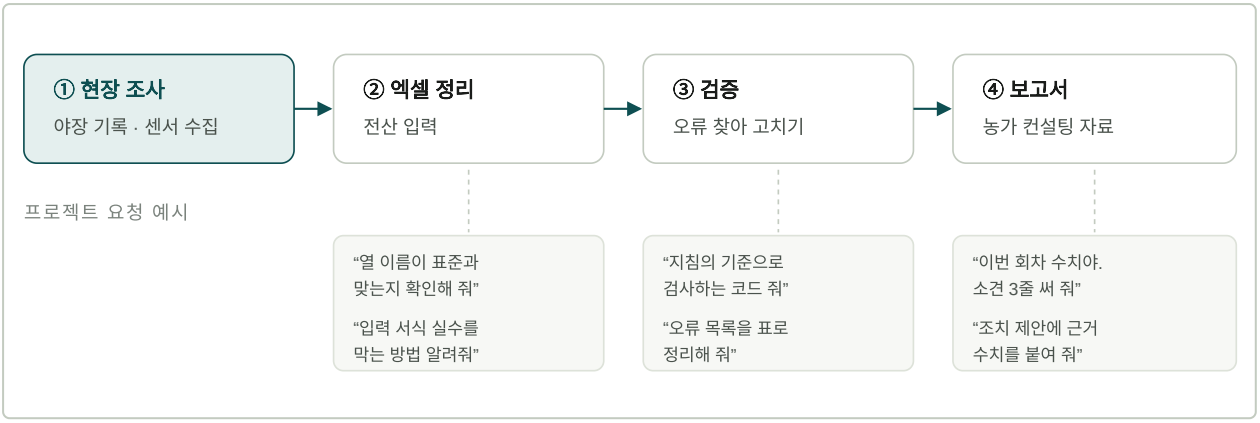


그림 4-5 (개념도) 2주 주기 업무와 단계별 요청. ① 현장 조사는 사람이 수행 / ②~④ 공통 설명은 프로젝트 지침으로 관리

회차별 요청 예시

1. **검증** – "이번 회차 파일이야. 지침의 기준으로 개체번호·온도·습도를 검사하는 파이썬 코드를 줘. 결과는 엑셀로 저장되게."
2. **요약** – "검사 결과 요약이야. (표 붙여넣기) 농가에 설명할 소견 세 줄로 정리해 줘."
3. **보고서** – "이번 기간 평균 온도 18.8℃, 저온 62시간, 고온 16시간이야. 지침 형식대로 조치 제안까지 써 줘."
4. **다듬기** – "두 번째 문장이 너무 기네. 한 줄로 줄여 줘."

그림 4-6 실제 대화 결과: 결론 세 줄 → 근거 수치 → 조치 우선순위(저온·다습·고온). 시간대별 자료 부족에 따른 추가 확인 사항 포함

회차가 쌓이면 이렇게 씁니다

- 형식 유지: 원하는 이전 결과나 출력 양식을 구체적으로 지정
- 회차 변경: 분석 기간·입력 수치·적용 기준 재확인
- 최종 검토: 지침에 맞는 출력 여부와 수치의 정확성 확인

표 결과를 CSV 파일로 받기

- 파일 요청: "이 표를 CSV 파일로 만들어 줘."
- 다운로드 링크가 없는 경우: "코드를 실행해서 파일로 만들어 줘."라고 추가 요청
- 결과 확인: 내려받은 파일의 열·행·수치를 원자료와 대조

엑셀 가져오기: **데이터 → 텍스트/CSV에서** 사용. 인코딩·자료형 지정 방법은 **1.3절** 참고

4.5 프로젝트와 코드의 역할 분담

상황	프로젝트만으로	권장
문장 다듬기·소견 쓰기	충분	프로젝트에서 바로
엑셀 수식·조건부 서식 만들기	충분	프로젝트에서 바로
표 몇 개 요약·비교	충분	값을 붙여넣어 처리
5,448행 전수 검사	부적합	**코드를 받아 내 PC에서** (5장)
매번 똑같은 처리 반복	비효율	**코드로 굳히기** (5·6장)
결과가 항상 같아야 할 때	불가	**코드** — 같은 입력이면 같은 출력
농가 원본 파일 업로드	금지	개인정보 열 제거 후에만 (8장)

가장 중요한 한계 · 매번 답이 조금씩 다릅니다

- 검사·계산: 검증한 코드로 실행
- 문장 작성: 프로젝트에서 소견 초안 작성
- 최종 판단: 담당자가 근거 수치와 업무 기준을 검토

이 장과 다음 장의 관계

- 프로젝트: `validate.py` · `make_report.py` 등의 코드 작성·수정 요청
- 내 PC: 코드 검토·시험 후 실제 데이터 처리
- 적용 원칙: 코드 생성과 원본 데이터 처리의 분리

다른 AI 도구 활용 시: 공통 지침 저장·참고 자료 관리·대화 연결의 지원 범위를 각각 확인

CHAPTER 05 · 실습 A

데이터 검증 자동화

명령 실행으로 데이터 일괄 검사 · 작목별 규칙과 판정 기준 수정

5.1 실행하기 해보기

1. 명령 프롬프트를 열고 `01_데이터검증` 폴더로 이동합니다
2. 아래 명령을 입력합니다

```
python validate.py "../데이터_6농가"
```

3. 몇 초 뒤 검출 결과가 화면에 나옵니다

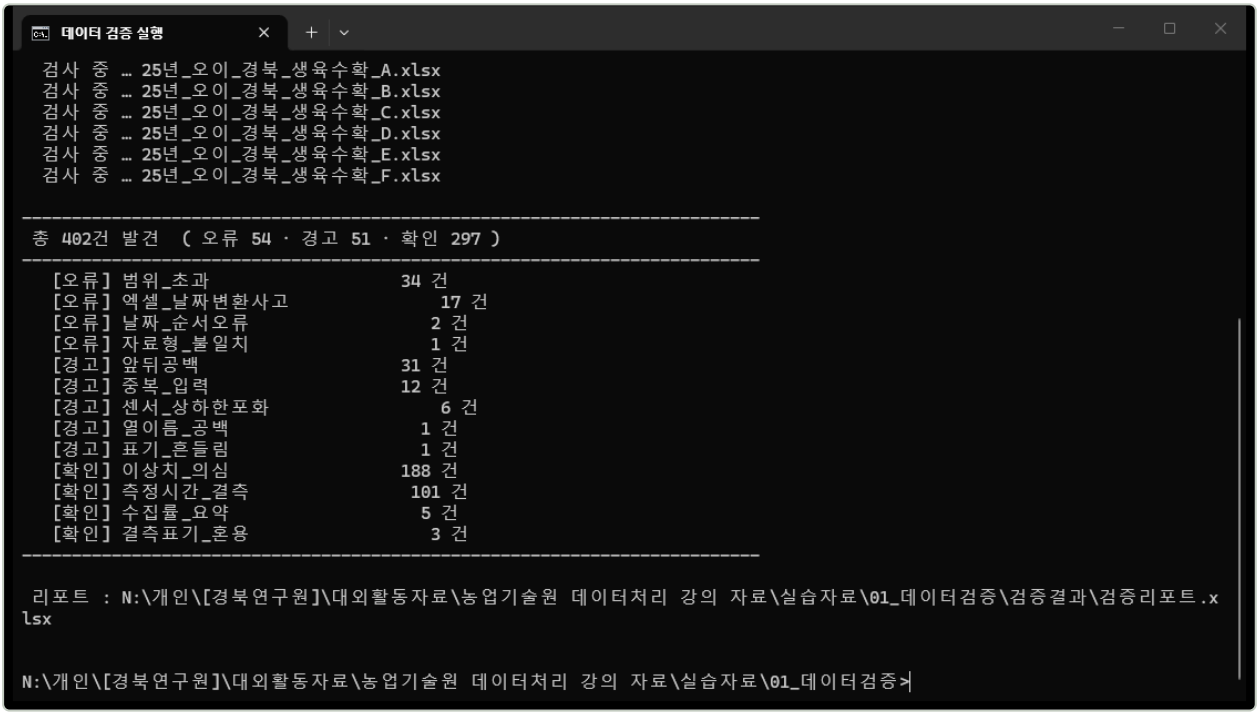
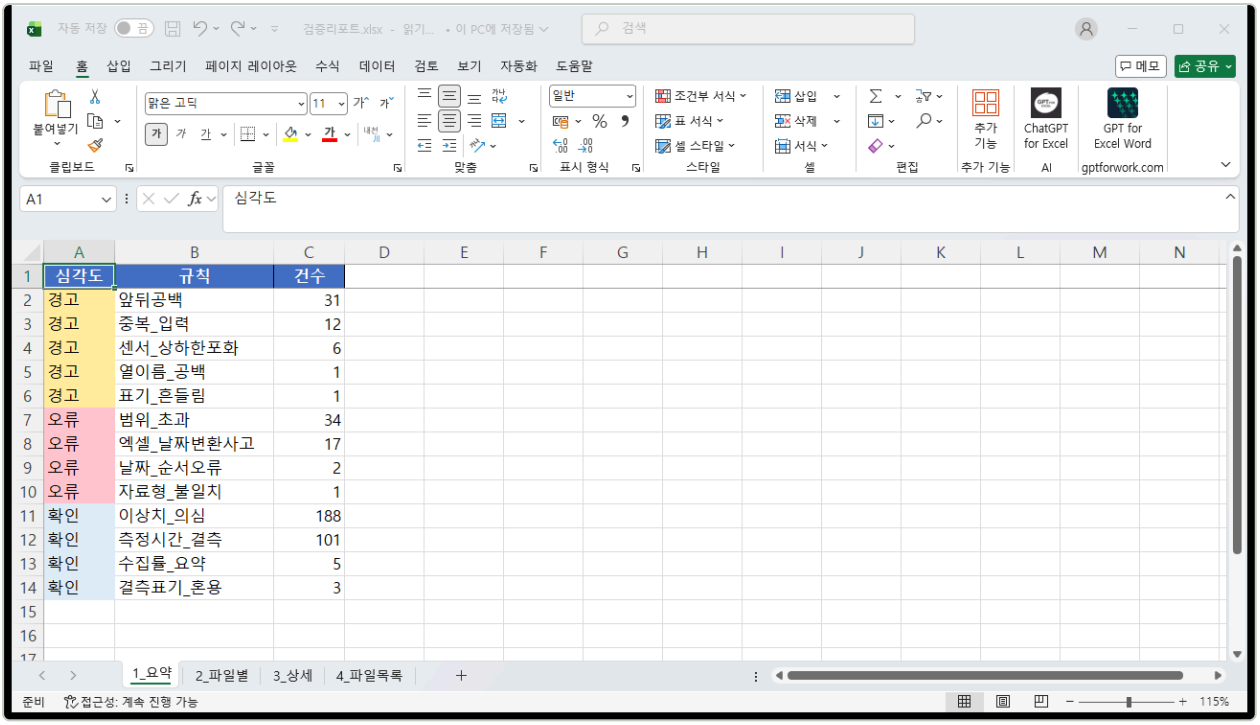


그림 5-1 검증 실행 결과: 19개 파일에서 **402건** 검출(오류 54·경고 51·확인 297). 하단에 결과 보고서 저장 경로 표시

5.2 결과 ① — 오류 유형별 건수

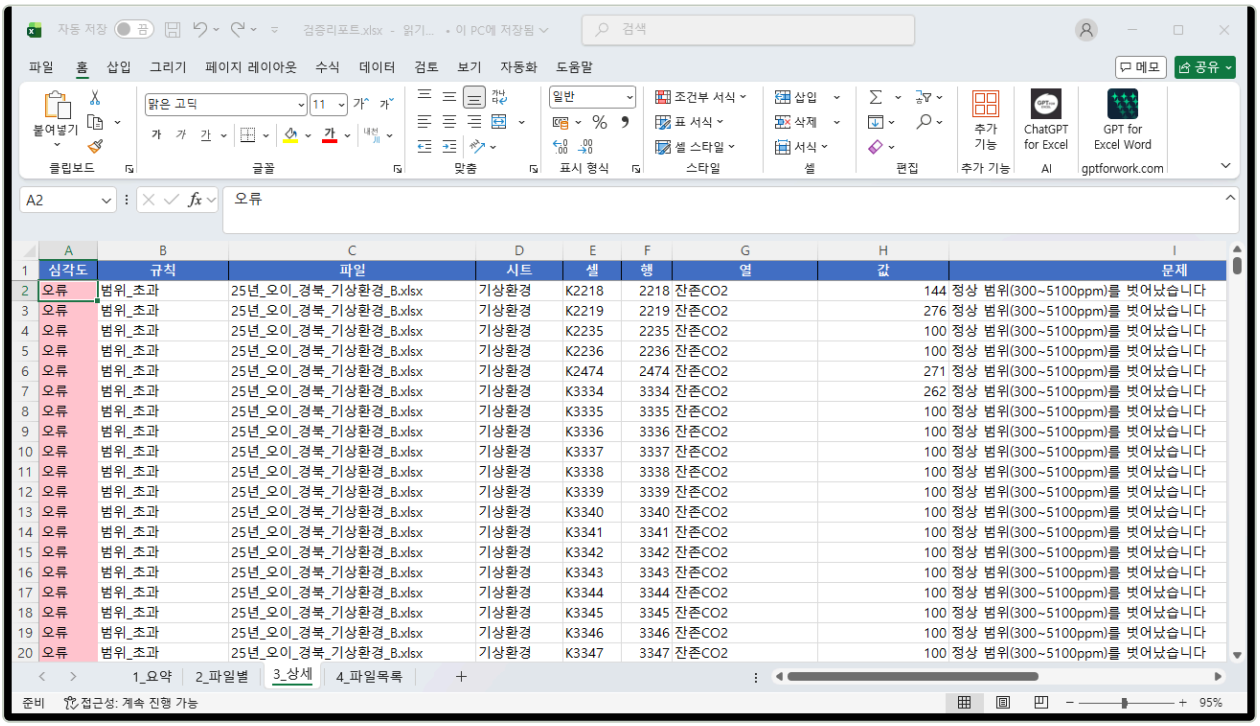


심각도	규칙	건수
경고	앞뒤공백	31
경고	중복_입력	12
경고	센서_상하한포화	6
경고	열이름_공백	1
경고	표기_흔들림	1
오류	범위_초과	34
오류	엑셀_날짜변환사고	17
오류	날짜_순서오류	2
오류	자료형_불일치	1
확인	이상치_의심	188
확인	측정시간_결측	101
확인	수집률_요약	5
확인	결측표기_혼용	3

그림 5-2 검증리포트.xlsx 의 **1_요약** 시트. 심각도별 규칙·건수 표시: 빨강(오류)·주황(경고)·파랑(확인)

- 우선 확인: **빨강(오류)**의 원인과 수정 필요 여부
- 후속 확인: 주황(경고)·파랑(확인)의 항목별 검토
- 요약 활용: 오류 유형별 건수와 집중 점검 대상 파악

5.3 결과 ② — 오류 셀 위치와 조치



심각도	규칙	파일	시트	셀	행	열	값	문제
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K2218	2218	잔존CO2	144	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K2219	2219	잔존CO2	276	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K2235	2235	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K2236	2236	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K2474	2474	잔존CO2	271	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3334	3334	잔존CO2	262	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3335	3335	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3336	3336	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3337	3337	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3338	3338	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3339	3339	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3340	3340	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3341	3341	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3342	3342	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3343	3343	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3344	3344	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3345	3345	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3346	3346	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다
오류	범위_초과	25년_오이_경북_기상환경_B.xlsx	기상환경	K3347	3347	잔존CO2	100	정상 범위(300~5100ppm)를 벗어났습니다

그림 5-3 **3_상세** 시트. 검출 항목별 파일·시트·셀주소·값·문제·조치·권장값 표시

확인 순서: **셀 주소 → 문제 → 조치**. 이동 예시: `Ctrl+G` → `D251` 입력

5.4 결과 ③ — 오류가 표시된 사본

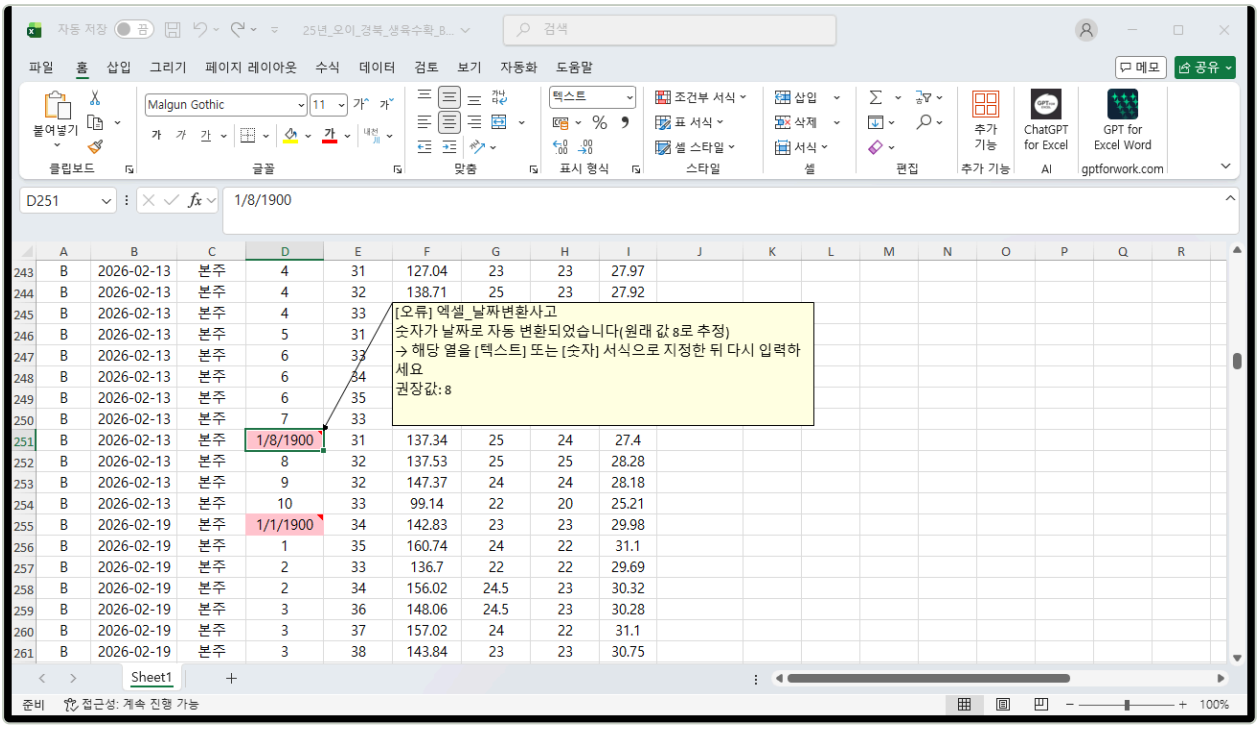


그림 5-4 `표시된_원본/` 폴더의 사본. 오류 셀 색상과 원인·조치·권장값 메모 표시. `D251` 예시: 원래 값 8로 추정 / 권장값 8

이게 왜 편한가

- 사본 검토: 표시된 셀의 원인·권장값을 원본 야장과 대조
- 수정 반영: 확인된 값만 수정하고 수정본 저장
- 원본 관리: 수정 전 파일을 보관하고 변경 내역 확인

5.5 검증 규칙 수정 — 기준값과 목록 변경

수정 대상: `rules.py`의 **숫자·목록**. 검증 프로그램 `validate.py`와 규칙 파일을 분리해 관리

```
# rules.py — 숫자 열의 정상 범위 : 열이름 → (하한, 상한, 단위)
NUMERIC_RANGES = {
    "온도_내부":     (-5, 55, "℃"),
    "상대습도_내부": (0, 100, "%"),
    "잔존CO2":       (300, 5100, "ppm"),   ← 이 숫자만 바꾸면 됩니다
    "수확과중":      (30, 500, "g"),
    "개체번호":      (1, 30, "번"),
}
```

새 작목의 규칙 추가 요청 예시

```
rules.py의 NUMERIC_RANGES에 참외 생육조사 항목을 추가해줘.
매뉴얼 기준으로 엽장 5~30cm, 엽폭 5~35cm, 과중 500~2500g이야.
```

5.6 규칙 파일 편집 도구

- 기본 편집: 메모장으로 `rules.py` 열기·수정
- 확인 사항: 따옴표·쉼표·들여쓰기 누락 여부
- 편집 보조: 코드 색상 구분과 오류 표시가 가능한 편집기 활용

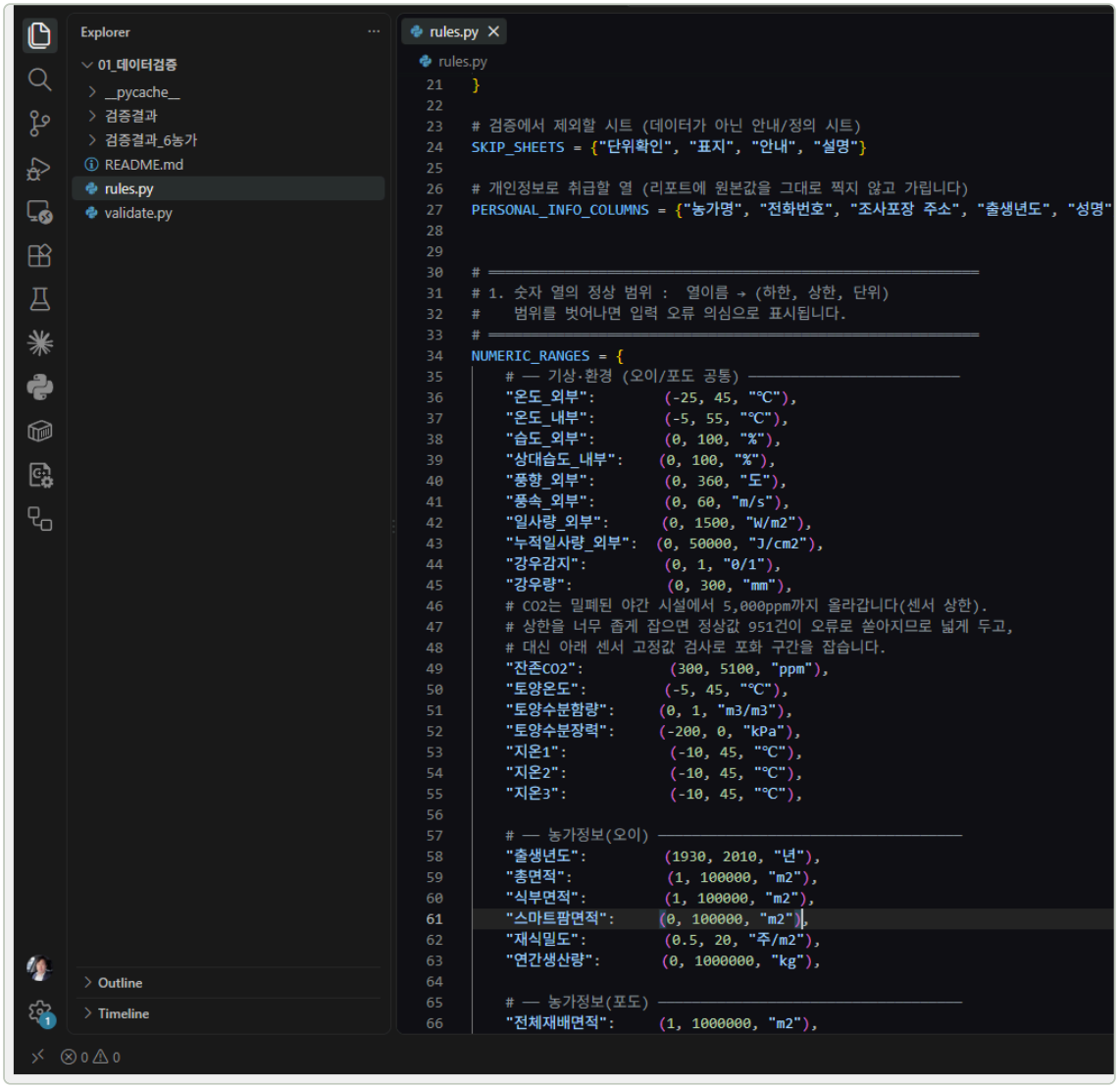


그림 5-5 VSCode로 연 `rules.py`. 문자·숫자의 색상 구분, 왼쪽 파일 목록, 중앙 주석의 CO2 상한 설정 근거 확인(5.7절 참고)

이런 점이 낫습니다	무슨 뜻인가
색이 다릅니다	열 이름은 주황, 숫자는 초록, 설명(주석)은 회색. **고칠 숫자만 눈에 들어옵니다**
오타를 미리 알려줍니다	쉼표나 따옴표를 빠뜨리면 **그 자리에 빨간 줄**이 그어지고 창 왼쪽 아래에 문제 개수가 뜹니다. 실행하기 전에 압니다
통합 명령창	`Ctrl` + `` ` ``(물결표 키)로 명령창 열기. 한 화면에서 **수정 → 실행 → 결과 확인**
파일이 한눈에	왼쪽 목록에서 `rules.py` · `validate.py` · `검증결과` 폴더 확인

설치

- 설치: `code.visualstudio.com` 에서 설치 파일 내려받기
- 한국어 메뉴: 한국어 언어 팩 추가
- 폴더 열기: **파일 → 폴더 열기 → 01_데이터검증** 선택

주의 · 꼭 설치해야 하는 것은 아닙니다

- 실행 환경: 편집기 설치와 별도로 파이썬 실행 환경 필요
- 메모장 편집: 저장 인코딩을 **UTF-8**로 지정
- 수정 후 확인: 파일 저장 → 실행 → 오류 메시지와 결과 점검

5.7 중요 판정 기준 설정과 오탐 점검

실제로 겪은 일

- 초기 설정: CO2 상한 **3000ppm**
- 초기 검사: **951건** 검출 후 기준의 적절성 재검토
- 검토 항목: 작목·시설·시간대별 환경과 센서 측정 범위

점검 원칙: 오류가 대량 검출되면 판정 기준·단위·자료형부터 재확인

- 기준 조정: 실습 코드의 상한을 **5100ppm**으로 변경 후 재검사
- 추가 확인: **5000ppm이 16회 연속** 기록된 구간(그림 2-2)
- 검토 방법: 범위 검사와 연속 고정값 검사를 구분하고 원자료·센서 상태 대조

5.8 연습 — 정답지가 있는 파일 해보기

실습 순서: `데이터_6농가` 의 **추가 오류 10건**을 직접 찾기 → 검증기 실행 → 결과 비교

농가	파일	힌트
C	기상환경	습도 허용 범위를 벗어난 값 2개
D	기상환경	온도 한 칸의 소수점 누락
D	생육수확	숫자 칸에 단위 문자 포함
E	생육기본	같은 농가명이 별도 항목으로 집계
F	생육수확	2장의 개체번호 날짜 변환 사례 3건
F	기상환경	CO2의 연속 고정값 구간
A	생육기본	같은 조사 기록의 중복 입력

정답 위치: **부록 D** 또는 `데이터_6농가/_안내_가상데이터.md`

파이썬 설치가 어려운 경우의 대체 방법

- 범위 검사: **조건부 서식**으로 범위 밖 값 표시
- 중복 검사: `COUNTIF` 활용
- 자료형 검사: `ISNUMBER` 로 문자 혼입 확인
- 입력 제한: **데이터 유효성 검사** 활용
- AI 활용: 필요한 엑셀 수식과 설정 방법 요청

컨설팅 보고서 자동화

기존 보고서 구성 검토 · 계산·소견·차트의 자동 생성

6.1 기존 컨설팅 자료의 구성과 개선 항목

오이 데이터 기반 컨설팅 자료

경상북도농업기술원

분석일자: 4월 농장주명:

1. 환경상황

가. 4월 4주차, 5월 1주차 환경상황

4주차

평균 온도는 18.8°C 로 농가평균과 비슷하였음.
평균 습도는 73.1% 로 농가평균과 비슷하였음.
평균 수분함량은 0.2475m³/m³ 로 농가평균보다 낮았음.
평균 토양온도는 21.3°C 로 농가평균과 비슷하였음.
누적 일사량은 1942.4J/cm² 로 농가평균보다 낮았음.
평균 CO2는 796.3ppm 로 농가평균보다 낮았음.

1주차

평균 온도는 20.1°C 로 농가평균과 비슷하였음.
평균 습도는 71.4% 로 농가평균보다 높았음.
평균 수분함량은 0.2494m³/m³ 로 농가평균보다 낮았음.
평균 토양온도는 22.1°C 로 농가평균과 비슷하였음.
누적 일사량은 1978.3J/cm² 로 농가평균보다 낮았음.
평균 CO2는 748.1ppm 로 농가평균보다 낮았음.

그림 6-1 기존 컨설팅 자료 1쪽. 주차별 해당 농가·농가평균 비교 구조. 농가주 성명은 가림 처리(8장 참고)

- 유지할 요소: 기존 보고서의 구성과 해당 농가·농가평균 비교 방식
- 개선할 요소: 분석일자·소견·결론 위치·수집률·판정 기준 표시

- 분석일자: 월 단위 표기로 **구체적인 일자 누락**
- 소견: 비교 결과에 따른 **조치 제안 부족**
- 배치: 마지막(4. 수량)의 결론을 **보고서 앞부분에 요약할 필요**
- 수집 상태: **데이터 수집률 표시 누락**
- 검토 근거: **판정 기준 미표시**

6.2 실행하기 해보기

1. `02_보고서자동화` 폴더로 이동합니다
2.
```
python make_report.py -d "../데이터_6농가"
```
3. 같은 폴더에 `보고서_B농가_....html` 이 생깁니다. 더블클릭해서 여십시오
4. 인쇄본이 필요하면 `Ctrl+P` → 대상을 **[PDF로 저장]**

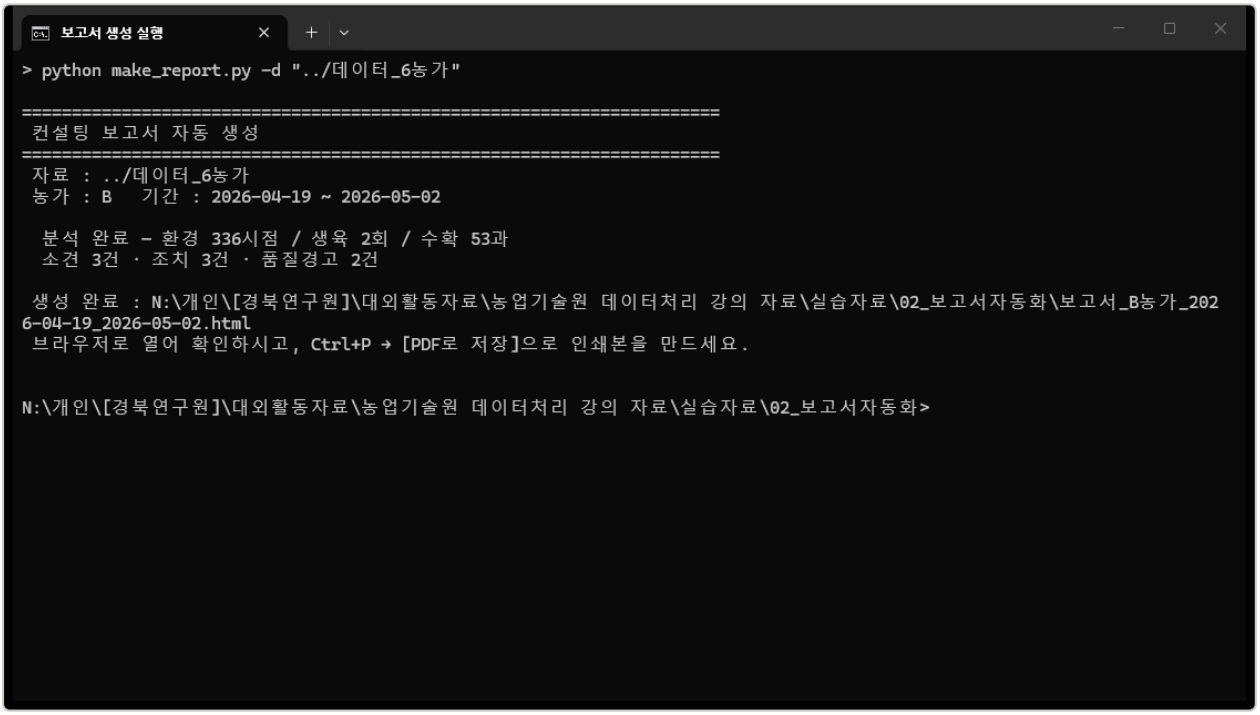


그림 6-2 보고서 생성 결과: 환경 336시점·생육 2회·수확 53과 분석 및 비교농가 5곳의 평균 계산

6.3 기존 구성을 유지한 개선 항목 7가지

#	현행	개선
1	분석일자가 “4월”까지만	생성 시각 자동 기입
2	“농가평균보다 낮았음”에서 끝	**원인 추정 + 조치 제안 + 근거 수치**
3	결론이 마지막에	**핵심 소견 3장이 맨 앞**
4	수집률 표시 없음	머리글에 **수집률·조사 횟수·확인필요**
5	판정 기준 비공개	**기준표를 그대로 인쇄**
6	2주치만	전 생육기 추세 위에 **이번 기간 음영**
7	이상 구간이 평균에 묻힘	**극값을 차트에 직접 라벨**

6.4 결과 읽기

그림 6-3 보고서 머리글의 환경 수집률 **100%(336/336시점)** . 회차 간 비교에 앞서 수집률 확인

그림 6-4 보고서 앞부분의 핵심 소견. 경고·주의·양호 구분과 근거 수치 표시

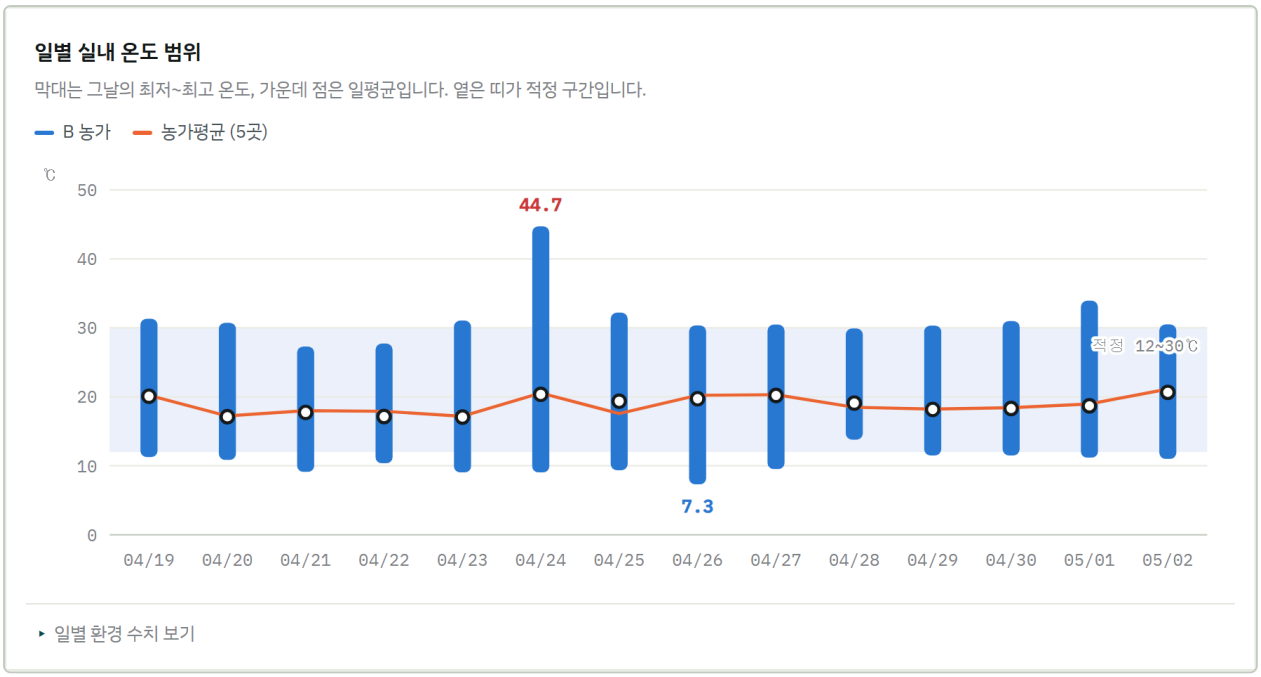
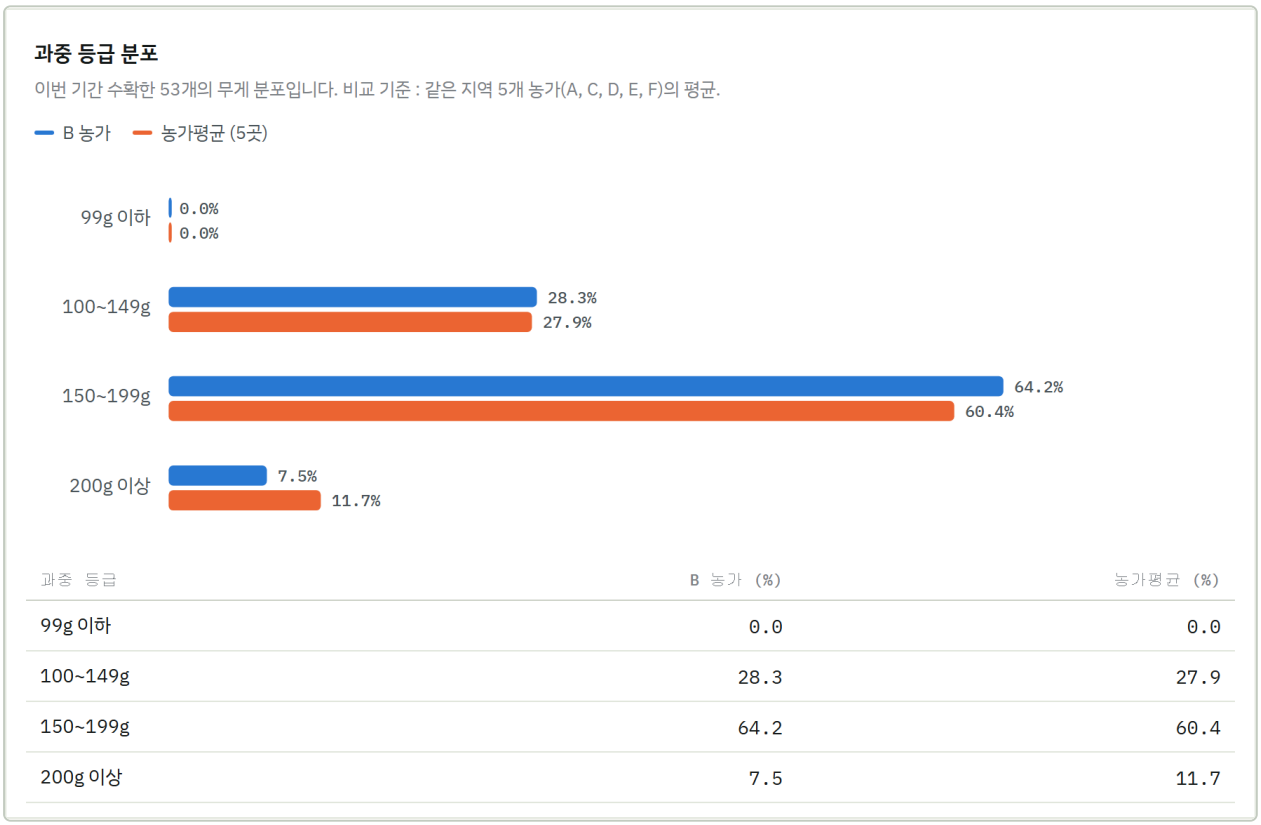


그림 6-5 일별 온도 범위: 막대(최저~최고)·중앙 점(일평균)·주황 선(농가평균). 4월 24일의 최고 44.7°C와 일평균 20.4°C 비교



과중 등급	B 농가 (%)	농가평균 (%)
99g 이하	0.0	0.0
100~149g	28.3	27.9
150~199g	64.2	60.4
200g 이상	7.5	11.7

그림 6-6 해당 농가와 농가평균의 과중 등급 분포 비교. 차트 아래에 동일 수치의 표 제공

02 조치 제안

아래 순서로 조치하시면 다음 2주 구간의 환경 편차를 줄일 수 있습니다. 각 항목의 근거가 되는 수치를 함께 적었습니다.

1순위 **야간 보온 강화**

해가 진 뒤 실내온도가 12℃ 아래로 내려가는 시간이 길었습니다. 보온커튼 개폐 시각을 앞당기거나 다겹보온커튼 추가를 검토해 보시기 바랍니다.

근거 · 12℃ 미만 62시간 · 최저 7.3℃

2순위 **환기 시작온도 하향**

측창·천창 환기 설정온도를 1~2℃ 낮추면 순간 고온 구간을 줄일 수 있습니다. 특히 맑은 날 오전 중 급상승 구간을 확인해 보시기 바랍니다.

근거 · 30℃ 초과 16시간 · 최고 44.7℃

3순위 **야간 제습**

상대습도 90% 초과가 길어지면 노균병·잿빛곰팡이병 발생 위험이 높아집니다. 야간 최소환기 또는 난방과 병행한 제습을 검토해 보시기 바랍니다.

근거 · 습도 90% 초과 44시간

그림 6-7 조치 제안의 우선순위와 판단 근거 수치 표시

07 판정 기준

이 보고서가 "높다·낮다"를 가른 기준입니다. 기준을 공개해야 농가가 결과를 검증할 수 있습니다.

항목	기준	출처
저온 한계	12°C 미만	강의 시연용 예시값 — 기관 표준으로 교체 필요
고온 한계	30°C 초과 (심함 35°C)	강의 시연용 예시값
다습 한계	상대습도 90% 초과	강의 시연용 예시값
CO2 부족	광합성 시간대 400ppm 미만	강의 시연용 예시값
주야 온도차	6~18시 평균 − 나머지 시간 평균	계산 정의
과중 등급	99g 이하 / 100~149g / 150~199g / 200g 이상	현행 컨설팅 자료와 동일 구간

그림 6-8 결과 검토를 위한 판정 기준 명시

6.5 판정 기준값 변경

기준값 관리 파일: `thresholds.py`

```
# thresholds.py
COLD = 12          # 이 온도 미만이면 저온 스트레스
HOT = 30           # 이 온도 초과면 고온
WET = 90           # 이 습도 초과면 다습 (병해 위험)
CO2_LOW = 400      # 광합성 시간대에 이 농도 미만이면 부족
```

실습: `COLD = 12`를 `10`으로 변경 → 재실행 → 소견 문장과 조치 순위의 변화 확인

> **주의 · 시연용 예시 기준값**
>
> - 실습 기준: `thresholds.py`의 숫자는 **강의 시연용 예시값**
> - 실무 적용: 농촌진흥청·농업기술원의 작목별 기준을 확인해 교체
> - 보고서 표시: 7절에 예시값임을 명시

6.6 데이터·기준·서식의 분리 관리

바뀐 것	고치는 파일	걸리는 시간
데이터 (2주 뒤 새 조사)	**없음** — 그냥 다시 실행	3초
판정 기준 (12℃ → 10℃)	`thresholds.py` 숫자 1개	10초
디자인 (색·글꼴·배치)	`template.html`	—
계산 항목 추가	`analyze.py`	—

- 구성 원칙: 데이터 처리·판정 기준·문서 서식을 별도 파일로 관리
- 기존 도구: 정상 작동하는 VBA 매크로는 업무 적합성을 확인해 활용
- 전환 검토: 기준 변경이나 기능 추가에 드는 유지보수 부담을 비교

CHAPTER 07 · 실습 C

한글(HWP) 파일로 내보내기

결재용 한글 문서 생성 · 기존 양식 활용 · 출력 결과 확인

7.1 한글 자동화의 기본 명령과 액션 ID

기본 호출 예시: 액션 ID를 지정해 한글 기능 실행

```
hwp.Run("액션ID")
```

액션 ID: 한글 자동화 기능을 식별하는 이름. `ActionTable_2504.pdf` 에서 기능별 ID와 설명 확인

Action ID	ParameterSet ID	Description	비고
AddHanjaWord	+	한자단어 등록	
AllReplace	FindReplace*	모두 바꾸기	
AQcommandMerge	UserQCommandFile*	입력 자동 명령 파일 저장/로드 (훈글메뉴의 [도구-빠른 교정-빠른 교정내용]에서 [입력 자동 명령 사용자 사전] 대화상자) ParameterSet을 직접 조작하여 사용함.	
AutoChangeHangul	-	낱자모 우선	
AutoChangeRun	-	동작	
AutoSpell Run	-	맞춤법 – 메뉴에서 맞춤법 도우미 동작 On/Off	
AutoSpellSelect1 ~ 16	-	맞춤법 도우미를 통해 선택된 어휘로 변경 (어휘는 1에서 최대 16까지)	
Average	Sum	블록 평균	
BackwardFind	FindReplace*	뒤로 찾기	
Bookmark	BookMark	책갈피	
BookmarkEditDialog		북마크 편집 대화상자 호출 액션 - 책갈피 작업창에서 편집 대화상자를 호출하기 위한 액션	
BreakColDef	-	단 정의 삽입	
BreakColumn	-	단 나누기	
BreakLine	-	line break	
BreakPage	-	쪽 나누기	
BreakPara	-	문단 나누기	
BreakSection	-	구역 나누기	
BulletDlg	ParaShape	글머리표 대화상자	
Cancel	-	ESC	
CaptionPosBottom	ShapeObject	캡션 위치-아래	

2

hancom 한글과컴퓨터

그림 7-1 `ActionTable_2504.pdf` : 52쪽·971개 액션. 예시: `InsertText` (텍스트 삽입), `TableCreate` (표 만들기)

찾는 방법: PDF에서 `Ctrl+F` → **한글 기능 이름** 검색. 예시: 쪽 번호 → `InsertPageNum`

7.2 실행하기 해보기

1. `04_한글변환` 폴더로 이동합니다

2

```
python to_hwp.py --charts
```

3 한글이 잠깐 실행됐다가 닫히고, `컨설팅자료_B농가_....hwp` 가 생깁니다

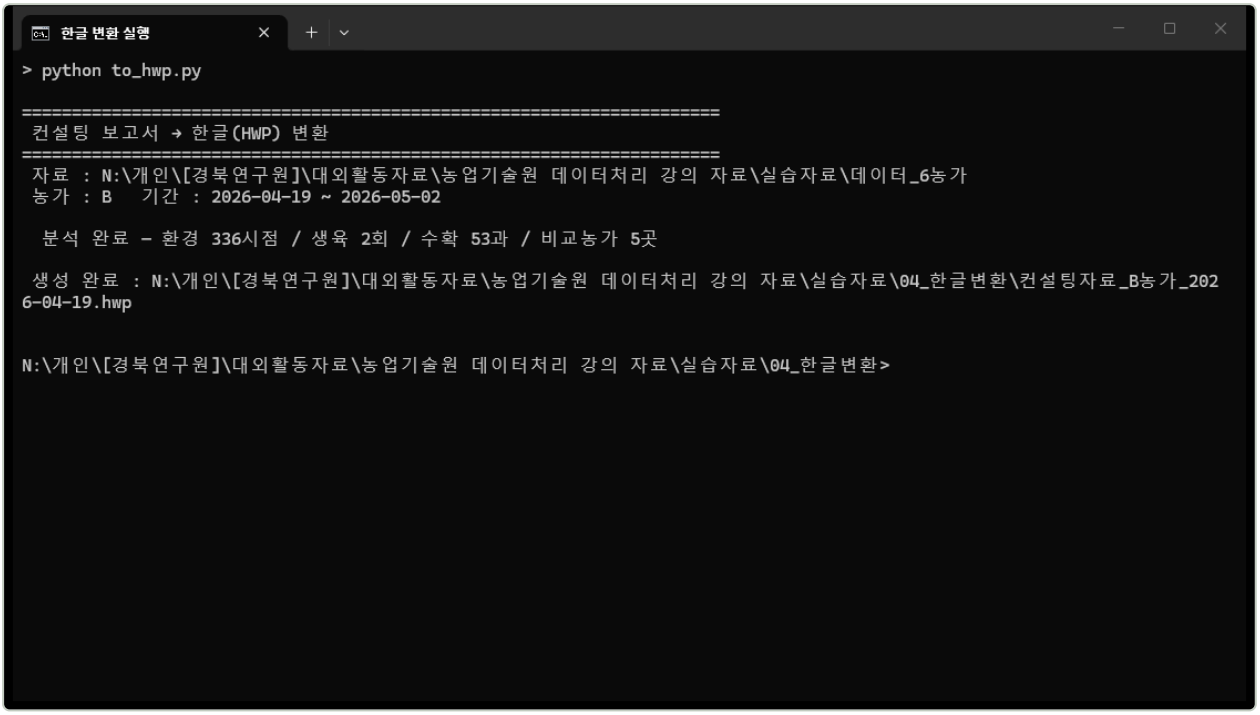


그림 7-2 한글 변환 완료 후 `.hwp` 저장 경로 표시

7.3 결과

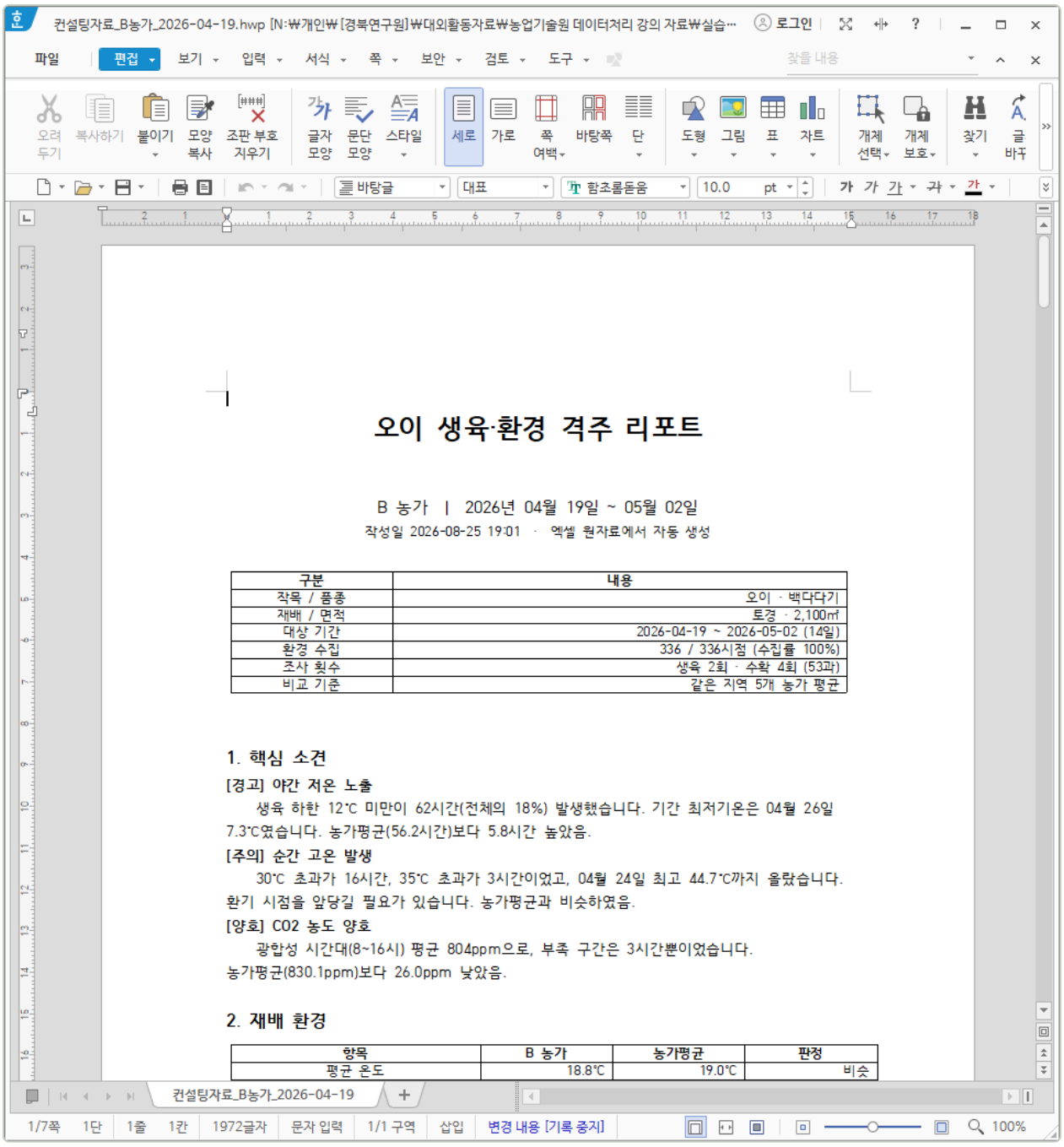


그림 7-3 자동 생성한 한글 문서: 제목·개요표·핵심 소견·환경 표(우리농가/농가평균/판정). 화면의 상태 표시: 1/7쪽

그림 7-4 문서 2쪽: 표 아래 차트 이미지 삽입. 내용·서식 검토 후 수정 및 결재 상신

실제로 확인한 것

- 확인 환경: 한글 2020
- 확인 기능: 제목·문단·표·차트 그림·쪽 나누기
- 저장 형식: `.hwp` · `.hwpx` · `.pdf`
- 추가 확인: 실행 환경별 파일 저장 보안 모듈 설정

7.4 한글 자동화 코드 요청 — 액션 설명서 첨부

액션 설명서에 근거한 코드 생성

- 오류 유형: 실제로 없는 액션 ID를 AI가 제시할 가능성
- 요청 방법: `ActionTable_2504.pdf` 를 첨부하고 문서에 있는 액션만 사용하도록 지정
- 검증 방법: 설명서 대조 및 생성 코드 시험 실행

```
ActionTable_2504.pdf를 첨부합니다.
파이썬 pywin32로 한글 문서를 만들려고 합니다.
첨부 문서에 실제로 있는 액션 ID만 사용해서 다음 코드를 짜 주세요.

1. 새 문서를 만들고
2. 가운데 정렬 18pt 굵은 글씨로 제목을 넣고
3. 3열 5행 표를 만들어 값을 채우고
4. .hwp와 .pdf로 저장

주의: 파일 저장할 때 보안 팝업이 뜨지 않게 해 주세요.
```

7.5 환경별 한글 문서 생성 방법 4가지

방법	한글 필요	서식	언제
① **COM 자동화** (오늘 실습)	필요	★★★	기관 PC. 표·머리말까지 제대로
② HWPX 직접 생성	불필요	★★	한글 없는 PC, 서버 처리
③ HTML을 한글에서 열기	필요	★	급할 때. 서식이 깨질 수 있음
④ 기존 양식 + 값만 채우기	필요	★★★	**결재 양식이 이미 있을 때**

기존 결재 양식 활용: ④ **양식의 값만 채우기**. `{{농가명}}` 등의 자리표시자를 실제 값으로 치환

```
한글 서식 파일 '컨설팅양식.hwp'에 {{농가명}}, {{평균온도}} 같은
자리표시자가 들어 있습니다.
pywin32로 이 파일을 열어 자리표시자를 실제 값으로 바꾸고
다른 이름으로 저장하는 코드를 짜 주세요.
```

7.6 자주 막히는 곳

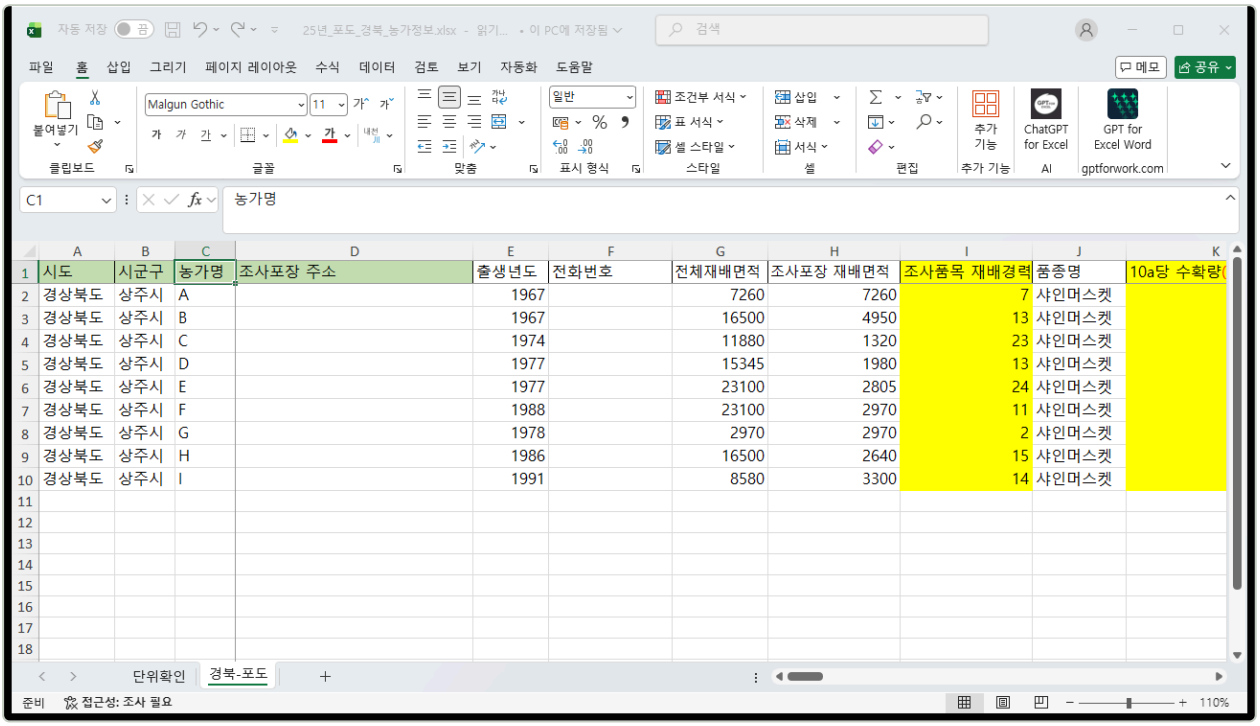
증상	해결
파일 저장 때 **보안 팝업**	`RegisterModule("FilePathCheckDLL", "FilePathCheckerModule")`
다른 대화상자가 떠서 멈춤	`SetMessageBoxMode(0x00020000)`
실행 후 **Hwp.exe가 남음**	`with Hwp() as h:` 형태로 쓰면 자동 정리
글자 크기가 이상함	HWPUNIT 단위 확인: **1pt = 100**
명령창에서 한글이 깨짐	`sys.stdout.reconfigure(encoding="utf-8")`
`Run()` 했는데 아무 일도 안 남	ParameterSet에 `*` 표시가 있는지 확인 후 필요한 매개변수 지정

CHAPTER 08

개인정보와 AI

농가 자료의 개인정보 항목 확인 · AI 요청과 외부 공유 시 점검 사항

8.1 농가 자료의 개인정보 항목



시도	시군구	농가명	조사포장 주소	출생년도	전화번호	전체재배면적	조사포장 재배면적	조사품목 재배경력	품종명	10a당 수확량
경상북도	상주시	A		1967		7260	7260	7	샤인머스켓	
경상북도	상주시	B		1967		16500	4950	13	샤인머스켓	
경상북도	상주시	C		1974		11880	1320	23	샤인머스켓	
경상북도	상주시	D		1977		15345	1980	13	샤인머스켓	
경상북도	상주시	E		1977		23100	2805	24	샤인머스켓	
경상북도	상주시	F		1988		23100	2970	11	샤인머스켓	
경상북도	상주시	G		1978		2970	2970	2	샤인머스켓	
경상북도	상주시	H		1986		16500	2640	15	샤인머스켓	
경상북도	상주시	I		1991		8580	3300	14	샤인머스켓	

그림 8-1 포도 농가정보 파일의 개인정보 항목: 농가명·출생년도·전화번호·조사포장 주소

- 정의서 위치: `단위확인` 시트
- 농가명: 농장주 성명 / 전화번호: 농장주 휴대폰 연락처
- 확인 사항: 이름·연락처·주소가 함께 포함된 파일의 취급 기준

8.2 코드 생성과 원본 데이터 처리의 분리

> 코드는 밖에서 만들고,
> 데이터는 안에서 처리한다.

- AI 요청: 실제 데이터 대신 작업 조건과 파일 형식을 설명해 코드 생성
- 실행 위치: 검토한 코드로 **내 PC에서** 농가 자료 처리
- 도구 점검: 외부 전송 기능 여부와 실행 환경 확인

8.3 안전한 질문법

구분	실제 문장
위험	`[생육기본_B.xlsx 첨부] 여기서 이상한 값 찾아줘`
안전	`열이 농가명·조사일자·개체번호(1~10)·초장(cm)입니다.` `개체번호가 범위를 벗어난 행을 찾는 코드를 짜 주세요.`

8.4 자료 업로드 전 점검

1. **개인정보 열을 지웁니다** — 성명·전화번호·주소·출생년도
2. **농가명을 코드로 바꿉니다** — `홍길동` → `A` (제공받은 샘플이 이미 이렇게 되어 있습니다)
3. **몇 줄만 발췌합니다** — 전체를 올릴 이유가 없습니다
4. **도구 설정을 확인합니다** — 대화 내용을 학습에 쓰지 않도록 설정

8.5 한글의 개인정보 가림 기능

개인 정보 찾아 숨기기

- 기능: 한글의 **개인 정보 찾아 숨기기**
- 자동화 액션: `ReplacePrivateInfoDlg`
- 외부 공유 전: 가림 처리 후 문서 전체의 누락 여부 확인

교안 적용 사례: **그림 6-1의 농가주 성명 가림 처리**

8.6 체크리스트 보안

활용 방법: 인쇄 후 업무 전·후 확인

- [ ] AI 대화창에 **농가 성명·연락처·주소**를 붙여넣지 않았는가
- [ ] 엑셀 파일을 **통째로 첨부**하지 않았는가
- [ ] 꼭 올려야 한다면 **개인정보 열을 지우고 코드로 치환**했는가
- [ ] AI에게는 **데이터가 아니라 "형식"**을 알려주었는가
- [ ] 받은 코드는 **내 PC에서** 돌리고 있는가
- [ ] 결과 파일을 외부로 보내기 전 **개인정보를 가렸는가**
- [ ] 업무망 반출 규정과 기관 정보보안 지침을 확인했는가

마무리 — 업무에 적용할 실습 선택

적용 과제: 아래 세 가지 중 업무 여건과 숙련도에 맞게 선택

1. **AI에게 엑셀 수식 요청** — 담당 업무에 필요한 수식 작성·검증
2. **검증 스크립트 실행** — 담당 농가 자료의 검출 결과 확인
3. **규칙을 내 작목에 맞게 고치기** — `rules.py`의 숫자를 바꾸는 것부터

APPENDIX

부록

현장 참고용 명령어·프롬프트·검증 규칙·정답·파일 구성

부록 A · 명령어 요약

```
# 최초 1회
pip install pandas openpyxl pywin32

# 실습 데이터 만들기 (6농가)
cd 03_가상데이터
python make_farms.py

# 데이터 검증
cd 01_데이터검증
python validate.py                      # 기본 샘플
python validate.py "C:\내데이터"          # 내 폴더
python validate.py "폴더" --no-highlight  # 색칠본 생략(빠름)

# 보고서 생성
cd 02_보고서자동화
python make_report.py -d "../데이터_6농가"
python make_report.py -f D               # 농가 지정
python make_report.py -s 2026-05-03 -e 2026-05-16   # 기간 지정

# 한글 문서 만들기
cd 04_한글변환
python to_hwp.py --charts                # 차트 포함
python to_hwp.py --pdf --hwpx            # 다른 형식으로도
```

부록 B · AI 프롬프트 모음

상황	그대로 쓰는 문장
새 작목 추가	rules.py의 NUMERIC_RANGES에 참외 항목을 추가해줘. 엽장 5~30cm, 엽폭 5~35cm, 과중 500~2500g이야.
오탐이 많을 때	CO2 상한을 5100에서 6000으로 올려줘. 지금 정상값이 오류로 잡히고 있어.
판정 기준 변경	thresholds.py에서 오이 저온 한계를 12°C에서 10°C로 바꿔줘.
지표 추가	analyze.py에 '일교차가 15°C를 넘은 날 수'를 세는 지표를 추가하고 환경 타일에 넣어줘.
기관 색 적용	template.html의 강조색을 우리 기관 색(#003C7D)으로 바꿔줘. 밝은 화면과 어두운 화면 둘 다 맞춰줘.
한글 자동화	ActionTable_2504.pdf를 첨부합니다. 첨부 문서에 실제로 있는 액션 ID만 사용해서 코드를 짜 주세요.
기존 양식 채우기	'컨설팅양식.hwp'의 {{농가명}} 같은 자리표시자를 실제 값으로 바꾸고 다른 이름으로 저장하는 코드를 짜 주세요.
엑셀만으로	파이썬을 못 씁니다. 개체번호가 1~10을 벗어난 셀을 조건부 서식으로 색칠하는 방법을 알려주세요.

부록 C · 오류 유형 사전

심각도	규칙	무엇을 잡나
오류	엑셀_날짜변환사고	숫자가 `1/8/1900`으로 둔갑 — **원래 값까지 복원**
오류	자료형_불일치	숫자 칸에 문자 (`10L` , `1시간반` , `전체`)
오류	범위_초과	설정된 허용 범위를 벗어난 값
오류	단위_불일치	열 전체가 다른 단위 (정의서 cm ↔ 실제 m)
오류	날짜_순서오류	끝수확일자가 정식일자보다 빠름
오류	미래_날짜	연도 오타로 미래 날짜
오류	필수값_누락 / 필수열_누락	반드시 있어야 할 항목이 빔
오류	열이름_중복	같은 이름의 열이 두 개
경고	센서_상하한포화	센서가 측정 한계에 걸려 같은 값만 반복
경고	센서_값고정	통신 장애·배터리 방전 의심 구간
경고	표기_흔들림	`스마트팜` / `스마트` 처럼 같은 뜻 다른 표기
경고	계산_불일치	과립중 × 과립수 ≠ 과방중
경고	중복_입력	같은 조사가 두 번
경고	앞뒤공백 / 정수아님 / 허용값_아님	겉보기엔 같은데 다르게 집계되는 값
확인	이상치_의심	범위 안이지만 유독 튀는 값 (사분위 기준)
확인	측정시간_결측 / 수집률_요약	기상데이터가 빠진 시각, 전체 수집률
확인	결측표기_혼용	빈칸 대신 `X` , `-` , `없음`

부록 D · 실습 정답지

5.8절 실습 정답: 직접 검토하고 검증기를 실행한 뒤 대조

농가	파일	위치	심어 둔 오류
C	기상환경	J1202 / J2402	습도 105% — 물리적으로 불가능
D	기상환경	I3002	온도 185℃ — 소수점 입력 누락 (18.5)
D	생육수확	F122	숫자 칸에 단위 문자 `120g`
E	생육기본	A2~A82	농가명 뒤 공백 `"E "` — 집계가 둘로 쪼개짐
F	생육수확	D32 / D79 / D142	개체번호가 날짜로 변환 (`1/4/1900` 등)
F	기상환경	K2002~K2022	CO2 5000ppm 21회 연속 — 센서 포화
A	생육기본	6행	같은 조사일·개체·마디가 두 번 입력

기존 자료와 가상 자료의 검출 항목 구분

- **B 농가**: 제공받은 실제 자료. 기존 검출 항목(개체번호 날짜변환 14건·CO2 연속 고정 구간 등) 포함
- **A·C·D·E·F 농가**: 실습용 가상 데이터. 실제 농가 자료가 아님

부록 E · 파일별 역할과 수정 시점

파일	역할	고칠 일
01_데이터검증/rules.py	검증 규칙 (범위·허용값·필수열)	**자주**
01_데이터검증/validate.py	검증 프로그램	거의 없음
02_보고서자동화/thresholds.py	판정 기준 (12℃·30℃·90%)	**자주**
02_보고서자동화/analyze.py	엑셀 → 숫자 계산	지표 추가할 때
02_보고서자동화/template.html	보고서 디자인	서식 변경 시
02_보고서자동화/charts.py	숫자 → 차트	거의 없음
04_한글변환/hwp_helper.py	한글 조작 도우미	거의 없음
04_한글변환/to_hwp.py	보고서 → 한글 문서	문서 구성 바꿀 때
03_가상데이터/make_farms.py	실습용 가상 농가 만들기	실습 준비할 때만

- 화면·분석 수치: 2026년 8월 제공받은 샘플 데이터의 처리 결과
- 개인정보: 그림 속 농가주 성명 가림 처리
- 판정 기준: `thresholds.py` 의 시연용 예시값을 실무 적용 전 기관 표준 기준으로 교체
- 가상 자료: A·C·D·E·F 5개 농가 데이터는 실습용으로 대외 인용 불가