

첨부 4. AI와 함께한 개발 일지 — 코드 수정 이력 발췌

배차 자동화 스크립트(파이썬, 총 4,411줄)에 남아 있는 날짜별 수정 주석 기록

아래는 실제 운영 중인 배차 자동화 스크립트의 주석에서 발췌한 수정 이력입니다. 현장에서 예외 상황이 생길 때마다 생성형 AI(Claude)와 대화하며 그날그날 코드를 고쳤고, 그 기록이 코드 안에 날짜별 주석으로 남아 있습니다. 총 86곳의 수정 주석이 2026-04-22 ~ 2026-06-05 사이 11일에 걸쳐 기록되어 있습니다. 회사명 · 거래처명 · 담당자명은 가림 처리(■■■)했습니다.

날짜별 수정 주석 분포

수정 일자	주석 수	주요 내용
2026-04-22	34	카카오 지도 동선 묶음 도입, 사업장 구분 로직, 누적 이력 방식 전환 등 대규모 개선
2026-04-23	12	혼합 배송(자차/택배) 처리, 자재 적송 파일 지원
2026-04-28	19	제외 거래처 규칙, 담당자 요청 수정
2026-04-29	8	수출 건 지원, 주소 파일 신규 포맷 대응
2026-04-30	2	세부 수정
2026-05-06	1	신규 기능 추가
2026-05-19	2	세부 수정
2026-05-22	2	ERP 재다운로드 파일명 대응, 이력 초기 세팅 자동화
2026-05-27	1	세부 수정
2026-06-04	3	세부 수정
2026-06-05	2	연휴 낀 납기일 누락 버그 수정

발췌 1 — 스크립트 맨 위의 수정 이력 주석 (원문, 회사명 · 업체명 가림)

```
# =====
# ■■■ 자동 배차 + 수요예측 스크립트 (수정본)
# 수정사항:
# 1. 자차/용차 혼합 인도조건 → 용차로 분류 (자차 호차 불필요 증가 방지)
# 2. 오늘출하 현재고=0 → 출하이력 최근 현재고로 자동 보완
# 3. 적송요청입력_YYYYMMDD.xlsx 파일 감지 시 → 배차계획 1시트에 '적송'건으로 자동 추가
# 4. [2026-04-22 변경] ERP에서 울산/워주 통합 파일로 조회 가능해짐.
# → 별도 '출하출하의뢰품목조회_■■■_*' 파일 병합 로직 제거
# → 통합된 '출하의뢰품목조회_■■■_YYYYMMDD.xlsx' 하나만 사용
# → 인도조건 '(출하출하)'는 기존대로 제외 (워주에서 나가지 않음)
# → 인도조건 '(워주출하)' 포함 건은 '출발지=울산'으로 표기 (울산→워주 이동건)
# → 그 외 워주 출발 건은 '출발지=워주'
# 5. [2026-04-22 변경] 배차계획 시트 가독성 개선 (담당자 요청)
# → 배차구분~가용재고(1~13열) 기본 배경 회색으로 변경
# → 특이사항 폰트 8pt → 10pt, bold 적용 (행 높이도 특이사항 길이에 맞춰 자동 확장)
# → 의미있는 강조색(특이사항 노랑, 재고부족 빨강, 마킹 노랑)은 유지
# 6. [2026-04-22 변경] 인도조건 '자차/택배' 혼합건 처리
# → 자차 동선 지역이면 자차로 편입 (가는 길에 들르면 되니까)
# → 자차 동선이 아니면 택배로 처리
# 7. [2026-04-22 변경] 자차 편입 로직 강화 (S사 층주 케이스)
# → REGION_ROUTE_ZONES: 지역별 동선상 편입 가능 권역 매핑 추가
```

· 충주 → 위주근방/충청권/수도권남부 (평택 동선)
· 세종/천안/아산/안성 → 수도권남부/충청권
→ 자차후보가 기본 권역에 편입 못해도 동선상 인접 권역 모두 시도
→ 명시 매핑 없으면 MERGE_CANDIDATES 구조로 대체 탐색
→ 편입 시 '동선편입' 표시로 기본 권역과 구분
→ REGION_TO_ZONE 충주 중복 정의 정리 (위주근방으로 통일)
8. [2026-04-22 변경] 배차계획 시트 테두리/폰트 스타일 개선 (담당자 요청)
→ 셀 테두리 색 회색(CCCCCC) → 검정(000000) 으로 변경 (칸 구분 잘 보이게)
→ 데이터 행 모든 컬럼 bold 적용 (기존엔 조건부로 일부만 굵었음)
9. [2026-04-22 변경] 누적 이력 파일 방식으로 전환 (성능 개선)
→ 기존: 매일 ERP에서 1년치 통째로 조회 → 느리고 렉
→ 변경: 하루치 파일만 올리면 스크립트가 자동으로 1년치 파일에 누적
→ 파일 구조:
· 출하의뢰품목조회_■■■■_YYYYMMDD.xlsx (하루치, 내일 배차 대상)
· 출하이력_1년.xlsx (누적 이력, 스크립트가 관리)
→ 동작:
1) 하루치 파일로 내일 배차 짜기
2) 하루치 → 누적 이력에 자동 병합 (덮어쓰기 방식)
3) 수요일예측·주기분석은 누적 이력에서 로드
→ 병합 정책: ★덮어쓰기★
· 같은 건(같은 키)이면 기존 삭제 후 새 내용으로 교체
· 담당자가 수량·특이사항·인도조건 등을 수정하면 이력에 즉시 반영
· 하루에 여러 번 수정해도 항상 최신 상태 유지
→ 중복 판단 키:
· 1순위: 배차등록번호 + 품목내부코드 + 납품처코드
· 2순위(등록번호 없을 때): 납기일 + 품명 + 납품처 + 출하의뢰일
· ※ 수량은 키에 포함 안 함 (수량 바뀌어도 같은 건으로 인식)
10. [2026-04-22 변경] 카카오 지도 기반 동선 묶음 로직 추가 (담당자 요청)
→ 권역이 달라도 '가는 길'에 있는 납품처끼리 자동 묶음
· 예: A사(서산, 충청권) + B사(평택, 수도권남부)
→ 권역은 다르지만 위주에서 서쪽 방향 동선 → 같은 자차에 묶음
→ 필요 파일:
· kakao_api_key.txt (카카오 REST API 키, 한 번만 발급)
· 납품처_좌표캐시.xlsx (스크립트가 자동 관리)
→ 동작:
1) 납품처주소 → 카카오 API로 좌표 변환 (한 번만, 캐시 저장)
2) 자차후보가 권역 매칭 실패 시 → 지도 기반 동선 판단
3) 활성 자차 중 '같은 동선' 자차에 편입 (거리+방위각 기준)
→ 판단 기준 (DONGSEON_CONFIG 변수에서 조정 가능):
· MAX_DIST_KM: 두 납품처 간 직선거리 (기본 20km)
· MAX_BEARING_DIFF_DEG: 위주 기준 방위각 차이 (기본 25°)
→ API 키 없거나 오류 시 → 자동으로 기존 권역 로직만 사용 (안전)
11. [2026-04-22 변경] 사업단위 컬럼 기반 위주/울산 판별 (담당자 요청)
→ 기존: 인도조건의 '(울산출하)' 텍스트만 보고 제외
· 문제: 인도조건에 아무 표시 없는 울산건(C사 등)은 위주건으로 오인
→ 변경: '사업단위' 컬럼 우선 활용 (ERP에서 새로 제공)
· 사업단위='(주)■■■■ 위주' → 위주 출발 (배차 대상)
· 사업단위='(주)■■■■ 울산' + 인도조건 '(위주출하)' → 울산→위주 이동건 (배차 대상)
· 사업단위='(주)■■■■ 울산' + 그 외 → 위주에서 안 나감 (제외)
→ 사업단위 컬럼 없는 파일은 기존 방식(인도조건 텍스트)으로 자동 fallback
→ 출발지 태깅도 동일 규칙 적용 (울산 사업단위 → 출발지=울산)

```
# =====
```

발췌 2 — 한 달 뒤에도 계속된 개선 기록 (2026-05-22)

```
# -----
# [2026-05-22 수정] 출하의뢰품목조회_■■■■_YYYYMMDD.xlsx 자동 탐색 (하루치 파일)
#
# 기존 문제:
# ERP에서 파일을 재다운로드할 때 자동으로 붙는 접미사
# 예) 출하의뢰품목조회_■■■■_20260522_1.xlsx ← 기존 패턴에 걸리지 않아 스크립트 종료
#
# 수정 내용:
# ① 탐색 패턴을 re.fullmatch 로 변경
# → 날짜 8자리 뒤에 바로 .xlsx 인 '정상 파일명' 우선 탐색
# ② 정상 파일이 없으면 접미사(_숫자_, _1_ 등) 붙은 파일도 허용 (폴백)
# → 이 경우 ERP 재다운로드 파일임을 경고 출력
#
# ★ 1년치 이력 파일 자동 세팅 로직 추가:
# 폴더에 출하의뢰품목조회 파일이 있고 출하이력_1년.xlsx 가 없는 경우,
# 해당 파일이 1년치 이력 데이터(데이터 건수 > 500) 라고 판단되면
# → 출하이력_1년.xlsx 로 자동 복사 후 종료 (초기 세팅 안내)
# -----
```

발췌 3 — 연휴 배차 누락 버그 수정 기록 (2026-06-05)

```
# [2026-06-05 수정] 내일~다음 영업일 구간에 납기가 하나도 없는 경우:
# 예) 파일명이 목요일 날짜 + 금요일 납기 0건 + 토/월 납기 존재
# 기존: 가장 가까운 납기일 '하루만' 반환 → 토요일만 잠히고 월요일 누락!
# 수정: 가장 가까운 미래 납기일(_primary)부터 '그 이후 첫 영업일'까지 묶음
# → 토/일/월(공휴일이면 그 다음 영업일까지) 전부 포함
if not _targets:
    _check_date = _primary
    while not _is_workday(_check_date):
        _check_date += timedelta(days=1)
    _limit = pd.Timestamp(_check_date)
    _targets = [d for d in _future_sorted if d <= _limit]
return _targets if _targets else [_primary]
```

※ 위 발췌는 실제 운영 코드의 주석 원문이며, 회사명 · 거래처명 · 담당자명만 가림 처리했습니다.