

☐ 아래의 내용을 검토한 다음 DB 작업 및 프로그램 작업을 나누어 각 작업별 프로세스를 설명해줘.

필수컬럼 => 코드/거래처명/담당자이메일/이메일발송여부/영업 담당 이메일/이메일발송여부

```
CUST_CD      VARCHAR(10)
CUST_MAIN_EMAIL  VARCHAR(100)
CUST_SENDMAIL_YN  VARCHAR(1)
SALESREP_EMAIL   VARCHAR(100)
SALESREP_SENDMAIL_YN VARCHAR(1)
INID           VARCHAR(10)
INDATE         DATETIME2
MOID           VARCHAR(10)
MODATE         DATETIME2
CONSTRAINT PR_CUST_CD PRIMARY KEY (CUST_CD)
```

☐ E-Order Admin – 거래처관리 – 주문 메일 알람(New)

특정 고객 담당자 및 담당 영업사원에게 D+1(익일차) 주문에 대하여 특정시간에 메일(내용+첨부파일(엑셀))을 전달하는 기능(Daily)

- **O_CUSTOMER의 정보를 그대로 전달 받을 수 있도록 하는 신규 테이블 생성**
 - 담당자 이메일 / 영업 담당 이메일이 수정을 할 수 있어야 함(필요에 따라 O_CUSTOMER와 달라질 수 있음(담당자가 여러명인 거래처 등))
 - (O_CUSTOMER의 테이블에 신규 정보 입력 시 그대로 가져오도록 필요 => 동기화 시점 고려)
- **체크박스를 제공하여 체크 된 담당자/영업담당자에게만 메일이 전송 되도록 처리 필요**
 - 정렬은 체크된 거래처가 우선으로 보이게 함(체크된 거래처 -> 코드순 정렬 / 가장 상위로 변경)
 - 체크 박스가 두개인 이유는 필요에 따라 담당자 혹은 영업담당자만 메일을 받아야 할 수도 있기 때문
- **메일 형식은 제공된 메일 및 엑셀과 동일하게 전송 필요**
 - 제목 및 내용은 형식에 맞춰서 전송 필요
- **메일 발송 시각 설정**
 - 메일 발송 시각은 필요에 따라 변경 될 수 있기에 사용자가 능동적으로 입력 할 수 있도록 설정(1시간 단위)
**
 - 메일 발송 시각의 관리 방법이 필요
- **메일 발송 로그 기록**
 - 발송 성공 여부 등

좋아~ 아래 내용을 기준으로 해서 작업을 **DB 작업**과 **프로그램 작업**으로 나눠서 정리해줄게. 각 항목은 **단계별 프로세스**로 정리했어.

✓ 1. DB 작업

[1-1] 신규 테이블 생성

O_CUSTOMER와 유사하되, 메일 발송 제어용 테이블 생성

```
CREATE TABLE O_CUSTOMER_MAILINFO (  
    CUST_CD VARCHAR(10) PRIMARY KEY,  
    CUST_MAIN_EMAIL VARCHAR(100),  
    CUST_SENDEMIL_YN VARCHAR(1),          -- 담당자 이메일 발송 여부 (Y/N)  
    SALESREP_EMAIL VARCHAR(100),  
    SALESREP_SENDEMIL_YN VARCHAR(1),      -- 영업 담당 이메일 발송 여부 (Y/N)  
    LAST_SYNC_DATETIME DATETIME          -- O_CUSTOMER와 마지막 동기화 시점  
);
```

△ O_CUSTOMER 와 별도 관리되는 이유:

- 이메일 수신자 수동 변경 가능성 있음
- 다중 담당자 등 유연한 관리 필요

[1-2] 동기화 프로세스 (초기 데이터 이관 + 주기적 동기화)

- O_CUSTOMER → O_CUSTOMER_MAILINFO 로 신규 거래처 데이터 반영
- 담당자/영업 담당자 정보가 없으면 NULL 또는 기본값
- 방법:
 - A. INSERT + ON DUPLICATE KEY UPDATE (최초 또는 주기적 배치로 동기화)
 - B. 트리거(TRIGGER)로 자동 동기화 (단, 유연성 고려 시 A권장)

예시 쿼리:

```
MERGE INTO O_CUSTOMER_MAILINFO AS TARGET  
USING O_CUSTOMER AS SRC  
ON (TARGET.CUST_CD = SRC.CUST_CD)  
WHEN NOT MATCHED THEN  
    INSERT (CUST_CD, CUST_MAIN_EMAIL, SALESREP_EMAIL, CUST_SENDEMIL_YN, SALESREP_SENDEMIL_YN)  
    VALUES (SRC.CUST_CD, SRC.CUST_MAIN_EMAIL, SRC.SALESREP_EMAIL, 'Y', 'Y', GETDATE());
```

[1-3] 메일 발송 로그 테이블

```
CREATE TABLE O_CUSTOMER_MAIL_LOG (  
    LOG_ID INT IDENTITY(1,1) PRIMARY KEY,  
    CUST_CD VARCHAR(10),  
    TARGET_EMAIL VARCHAR(100),  
    EMAIL_TYPE VARCHAR(10), -- 'CUST' or 'SALESREP'  
    SEND_RESULT VARCHAR(10), -- 'SUCCESS', 'FAIL', 'SKIPPED'  
    SEND_DATETIME DATETIME,  
    ERROR_MSG VARCHAR(500) NULL  
);
```

[1-4] 발송 스케줄 관리 테이블

```
CREATE TABLE O_CUSTOMER_MAIL_SCHEDULE (  
    ID INT PRIMARY KEY CHECK (ID = 1), -- 단일 관리용  
    SEND_HOUR INT CHECK (SEND_HOUR BETWEEN 0 AND 23), -- 발송 시각 (시간 단위)  
    LAST_UPDATED DATETIME  
);
```

✓ 2. 프로그램 작업

[2-1] 화면 기능 - 관리자 화면 구성

위치: E-Order Admin > 거래처관리 > 주문 메일 알람(New)

- O_CUSTOMER_MAILINFO 조회 화면 구성
- 필드: CUST_CD, 거래처명, CUST_MAIN_EMAIL, CUST_SENDMAIL_YN (체크박스), SALESREP_EMAIL, SALESREP_SENDMAIL_YN (체크박스)
- 체크박스 정렬 기능 구현:
 - 우선순위: 체크된 항목 → 코드순

[2-2] 메일 발송 처리 로직 (Batch/Job)

- 실행 조건:
 - 현재 시각 == 설정된 발송 시각 (O_CUSTOMER_MAIL_SCHEDULE.SEND_HOUR)
 - 오늘 기준 D+1 배송 예정 주문이 존재하는 거래처
- 처리 순서
 1. O_CUSTOMER_MAILINFO 에서 * _SENDMAIL_YN = 'Y' 인 거래처 추출
 2. 해당 거래처의 D+1 배송예정 주문 조회

3. 메일 본문/첨부파일 생성 (엑셀)
4. 메일 전송
5. 결과 `O_CUSTOMER_MAIL_LOG` 에 기록

[2-3] 발송 시각 수정 기능

- 관리자 화면에서 **발송 시각(시간)**을 수정 가능하게 구현 (1~23시)
- 변경 시 `O_CUSTOMER_MAIL_SCHEDULE` 업데이트

[2-4] 메일 포맷

- 제목/본문/엑셀파일 양식은 지정된 템플릿에 맞춰 개발
- 메일 내용: 거래처명, 주문요약, 배송일 등 포함
- 첨부파일: 엑셀 다운로드 가능한 형식

✓ 요약 정리

구분	작업 내용
DB 작업	- <code>O_CUSTOMER_MAILINFO</code> , 로그, 스케줄 테이블 생성 - <code>O_CUSTOMER</code> 와 동기화 쿼리 작성
화면 작업	- 거래처 메일 설정 UI - 체크박스 필터 및 정렬 - 발송 시각 설정 화면
메일 발송	- 배치작업 구현 (D+1 주문 확인 → 메일 전송) - 엑셀 생성 및 메일 템플릿 적용 - 발송 로그 기록

필요하면 각 항목별 샘플 쿼리/메일 양식 샘플/엑셀 포맷 등도 도와줄게. 다음 단계로 넘어가고 싶으면 말해줘 :)