AI에 보내기 전에 지운 것

시설민원을 AI로 분류하면서, 민원 글이 국외로 나가기 직전에
무엇을 지우고 무엇을 남겼는지 적었습니다.

민원인이 쓴 글	제2공학관 **27501호** 에어컨 고장입니다. 연락처 ~~010-1234-5678~~ , ~~김철수~~ 교수님 연구실.
↓ 보내기 직전에 지움	이메일 · 주민등록번호 · 학번 · 전화번호 · 호칭이 붙은 이름, 이 다섯 가지
AI에게 실제로 나간 글	제2공학관 **27501호** 에어컨 고장입니다. 연락처 [전화] , [이름] 교수님 연구실.

호실번호와 증상은 **일부러 남깁니다.** 그게 분류의 근거이기 때문입니다. 그래서 공개해 둔 검사 52개 중 **26개**는 「지워졌는지」가 아니라 **「지우면 안 되는 걸 지우지 않았는지」**를 봅니다.

성균관대학교 자산관리처 관리팀 **장상민** · 시설통합포털(FMIS) 민원 분류
2026년 10월 · 예시의 **이름과 번호는 검사용 가짜**이고, 호실 형식은 실제와 같습니다.
지우는 규칙과 검사 설명은 공개 저장소의 2026년 10월 27일 판 기준이고, 같은 날 운영에도 같은 파일을 넣었습니다.

실제로 이렇게 나갑니다

지우는 코드는 공개 데모로 올려 뒀습니다. 아래가 그 화면이고(예시 1), 이어지는 두 건은 공개 저장소의 자동 검사 파일에 그대로 들어 있는 예시입니다.

`llm-mask.js · 263줄 · 의존성 0` — ● 브라우저에서 실행 중 · 전송 없음

민원인이 쓴 글

제2공학관 27501호 에어컨 고장입니다. 연락처 010-1234-5678, 김철수 교수님 연구실.

AI에게 실제로 나가는 글

제2공학관 27501호 에어컨 고장입니다. 연락처 `[전화]`, `[이름]` 교수님 연구실.

전화 1 · 주민번호 0 · 메일 0 · 학번 0 · 이름 1 · 지운 개수 2

예시 1 · 공개 데모 「직접 해 보세요」 — 머리에 `llm-mask.js · 263줄 · 의존성 0`, 오른쪽에 「브라우저에서 실행 중 · 전송 없음」. 왼쪽이 민원인이 쓴 글, 오른쪽이 AI에게 나가는 글입니다. 전화번호와 「김철수」가 **[전화]·[이름]**로 바뀌고 **27501호는 남았습니다.** 배지는 전화 1 · 이름 1, 지운 개수 2. 저장소 설명서(README) 첫머리 예시와 같은 문장입니다.

예시 2 · 위치와 증상은 그대로 남습니다

민원인이 쓴 글

제2공학관 27501호 에어컨에서 물이 떨어집니다. 연락처 ~~010-1234-5678~~ 입니다.

실제로 나간 글

제2공학관 27501호 에어컨에서 물이 떨어집니다. 연락처 `[전화]` 입니다.

검사가 확인하는 것은 **제2공학관 · 27501호 · 에어컨 · 물이 떨어집니다** 네 가지가 모두 살아 있는지입니다. 하나라도 사라지면 검사가 「분류 근거 소실」로 실패합니다

예시 3 · 지울 게 없으면 손대지 않습니다

민원인이 쓴 글

3동 5층 복도 천장에서 물이 떨어집니다

실제로 나간 글

3동 5층 복도 천장에서 물이 떨어집니다

동 번호와 층수는 사람을 가리키지 않습니다. 지운 개수는 0

왜 이렇게 나눴나

민원 글에서 위치와 증상이 지워지면 AI가 판단할 재료가 없어집니다. 분류가 틀리면 담당이 잘못 잡히고, 그 불편은 곧바로 민원인에게 돌아갑니다. 그래서 **확실할 때만 지운다**를 원칙으로 잡았습니다.

확실할 때만 지운다는 것은 **못 지우고 나가는 것이 있다**는 뜻이기도 합니다. 무엇이 안 지워지는지는 6쪽에 그대로 적어 뒀습니다.

지우는 것과 남기는 것

지우는 항목마다 바뀌는 말을 다르게 뒀습니다. 「연락처를 남겼다」는 문맥까지 사라지면 AI가 긴급도를 잘못 볼 수 있어서입니다.

무엇을 지우고 무엇을 남기나

바뀌는 말을 항목마다 다르게 뒀습니다. 전부 [개인정보]로 뭉개면 "연락처를 남겼다"는 문맥까지 사라집니다.

지웁니다	바뀌는 말	예
이메일	[메일]	test.user@example.com
주민등록번호	[주민번호]	900101-1234567 · 900101 1234567
학번	[학번]	2019123456
전화번호	[전화]	010-1234-5678 · 02)760-1234 · +82 10-1234-5678
호칭이 붙은 이름	[이름]	김철수 교수님 · 홍길동님께

남깁니다		왜
호실번호 · 건물 · 동 · 층	27501호	어느 방인지 알아야 담당을 정합니다
증상 서술	물이 떨어집니다	이게 분류의 본문입니다
날짜 · 시간 · 금액 · 접수번호	14:30	사람을 가리키지 않습니다

같은 표가 공개 페이지에 그대로 있습니다 — 데모 아래쪽 「무엇을 지우고 무엇을 남기나」입니다. 위가 지우는 다섯 가지와 바뀌는 말, 아래가 남기는 것과 그 이유입니다.

화면의 「예」 칸에는 몇 가지만 적었습니다. 실제로는 전화 0507-1234-5678 · 070-8888-9999, 하이픈 앞뒤를 띄운 010 - 1234 - 5678, 하이픈 없는 13자리 주민등록번호(0512313456789)와 점으로 이은 850315.1234567, gildong+fms@mail.example.co.kr 같은 메일까지 봅니다. **표기가 조금 달라도 같이 봅니다.** 하이픈 대신 쓴 - · — · —, 폭이 넓은 전각 숫자(０１０), 숫자 사이에 숨은 보이지 않는 공백 문자, 맥에서 복사해 자음·모음이 쪼개진 한글도 잡습니다. 지운 자리 말고는 글자를 바꾸지 않아서 전각으로 쓴 ２층 ３０５호는 그대로 나갑니다(쪼개진 한글만 보통 한글로 모아 내보냅니다). 남기는 쪽의 날짜·시간·금액·접수번호는 6쪽 표에 예문 그대로 적어 뒀습니다.

> **어디에 두고 어디로 보내나**
>
> 민원 원문과 연락처가 저장되는 데이터베이스는 **서울에 있는 서버(구글 클라우드 서울 데이터센터)**에 있습니다. 민원 글 가운데 AI로 나가는 것은 지운 뒤의 본문뿐입니다.
>
> 그 전송은 개인정보처리방침에 적었습니다. (7쪽)

순서가 정해져 있고, 건너뛸 수 없습니다

이메일은 반드시 맨 먼저 지웁니다. 그리고 이 처리를 거치지 않고 민원 분류에서 내보내는 길은 코드에 없습니다.

이메일이 먼저입니다

1. **이메일**부터. 통째로 지워야 도메인이 안 남습니다. 늦게 지우면 2026012345@example.ac.kr이 [학번]@example.ac.kr로 나갑니다.
2. **주민등록번호**는 13자리. 띄우거나 점으로 이은 것은 앞 6자리가 날짜일 때만 봅니다.
3. **학번**은 20으로 시작하는 10자리. 날짜(20260826)와 접수번호(FR-20260826-001)는 자릿수가 달라 안 걸립니다.
4. **전화번호**는 0으로 시작하는 9~12자리와 +82 국제전화 표기.
5. **이름**이 마지막. 앞에서 바뀐 [전화] 같은 말은 이름 규칙에 걸리지 않습니다.

2~4번이 서로 엉키지 않는 것은 **앞뒤 숫자 경계** 덕분입니다. 번호 바로 앞이나 뒤에 숫자가 또 붙어 있으면 그 번호로 보지 않는다는 조건입니다. 이게 없으면 긴 숫자 안쪽이 번호로 잘려 나갑니다. 3000000000원은 3[전화]원이 되고, 12자리 주문번호 202012345678은 [학번]78이 됩니다. 그래서 **이메일만 맨 앞에 두면 나머지는 순서를 바꿔도 결과가 같습니다.** 순서 120가지를 문장 5천여 개에 돌려 확인했습니다.

이름은 조건 세 개를 모두 만족해야 지웁니다

조건	내용
① 흔한 성씨로 시작	한 글자 성씨 39자 + 두 글자 성씨(남궁·황보·제갈·사공·선우·독고)
② 전체 두~네 글자	성 한~두 자 + 이름 한~두 자
③ 호칭이 반드시 붙음 세 글자 이상 이름(김철수 · 남궁민수)	교수·선생·박사·조교·연구원·주임·대리·과장·팀장·부장·처장·학생(님이 있든 없든), 조교수·부교수·명예교수 같은 교수 직함, 책임·선임연구원, **님이 붙을 때만** 강사·책임·선임·수석·실장·소장·원장·차장·주무관·행정관·센터장·본부장·학장·서무, 그리고 님·씨. 뒤에 하고·으로부터·까지·처럼·네, 「교수실」의 실 같은 말이 붙어도 봅니다
③ 호칭이 반드시 붙음 두 글자 이름(김혁)	처음 규칙 그대로: 앞의 12개 직함과 님·씨, 이·가·께·에게·입니다 같은 조사·어미만. 「우리」·「조금」 같은 흔한 낱말과 겉모양이 같아서입니다

성씨로 시작하지만 이름이 아닌 낱말 151개(고객 · 강의실 · 전기 · 우리 · 신임 · 이모 · 강사 · 서무 · 이미 등)는 막아 뒀고, 직함·부서로 끝나거나(정교수 · 주무관 · 전기팀) 「에·과·처·층·앞」처럼 이름 끝에 쓰지 않는 글자로 끝나는 말(최근에 · 전산과 · 정문앞)도 이름으로 보지 않습니다. 호칭과 함께 잡힌 세 글자 이상 이름은 같은 글에서 다시 나와도 지웁니다 (김철수 교수님 … 김철수에게 → [이름] 교수님 … [이름]에게). 「홍길동전」처럼 다른 낱말 안쪽은 그대로입니다.

건너뛸 수 없는 이유

민원 분류에서 **AI를 부르는 경로는 코드에 한 곳**이고, 그 길목에 지우는 처리가 **조건 없이** 있습니다. 켜고 끄는 설정값도 없습니다. 이 처리는 **2026년 9월 8일**부터 운영에 적용돼 있고, **10월 27일**에는 운영의 지우는 파일을 공개본과 같은 파일로 바꿨습니다. 아래는 운영 서버 llm-classify.js에서 AI를 부르기 직전 부분입니다.

```
/* [법규] LLM 전송 전 개인정보 마스킹 — 국외(OpenAI, 미국) 전송 경계는 «여기»다.
 *    · 바깥으로 나가는 것은 masked.text 뿐이다. … */
const masked = mask.maskForLlm(text);          // ← 여기를 지나지 않고는 못 나간다(설명으로 붙인 말)
const piiMasked = mask.totalHits(masked.hits);
const input = { text: masked.text, campusHint, buildingCandidates: undefined, ruleResult };
```

지운 결과에는 **원문이 어디에도 담기지 않습니다.** 돌아오는 값은 바뀐 글과 **지운 개수**(전화 1, 이름 1 …)뿐입니다.

검사 52개 중 26개는 「지우면 안 되는 것」 검사

공개 저장소에 올린 검사 52개입니다. 저장소에서 `node --test` 한 줄로 돌아가고, 52개 모두 통과합니다.

무엇을 보는 검사인가	개수	검사 제목 기준으로 센 내역
지워야 할 것이 지워졌는가	22	전화 6 · 주민등록번호 4 · 이메일 1 · 학번 1 · 이름 9 · 긴 입력 1
지우면 안 되는 것을 안 지웠는가	**26**	**잘못 지우지 않는지 보는 검사 24개**(검사 제목에는 「위양성」이라고 적혀 있습니다): 호실 · 연도 · 금액 · 동/층 · 날짜 · 접수번호 · 시간 · 일반명사 · 직함만 있는 호칭 · 낱말이 붙는 호칭 · 번호 매김 · 자릿수 · 이름 없이 호칭 뒤에 붙는 말 · 이름 없이 쓴 직함 · 띄운 하이픈이 든 날짜·시간 · 전각 숫자와 이모지 · 띄어 쓴 숫자 둘 · 다른 낱말 안쪽 · 사람 앞에 흔한 낱말 · 100만 자 · 쪼개진 한글 · 숫자 경계 · 장소·부서 낱말 · 시간 낱말. **이름 잘못 지우기 방지 2개**: 호칭 없는 이름 · 어미 붙은 일반명사
실제 민원처럼 생긴 문장	2	연락처만 지워지고 위치·증상은 남는지
빈 입력·돌려주는 값 방어	2	빈 값·null이 들어와도 안 터지는지 · 돌려주는 값에 원문이 안 담기는지

공개 저장소 `test/llm-mask.test.js`의 검사 52개를 제목별로 센 것입니다(22+26+2+2). 「긴 입력」과 「100만 자」는 1초 안에 끝나는지도 봅니다. 실제 전송 경로가 지운 글만 내보내는지 보는 통합 검사 2개는 운영 저장소의 다른 모듈이 있어야 돌아가서 공개본에는 넣지 않았습니다. 운영에서는 이 2개를 더한 54개가 모두 통과합니다.



예문 「지우면 안 되는 것들」을 누른 화면 — 아래 단추 줄에 있습니다. 왼쪽과 오른쪽이 글자 하나까지 같습니다. 날짜 2026-08-26, 시간 14:30, 접수번호 FR-20260826-001, 「학생회관 2층」이 그대로 남았고 배지는 모두 0, **지운 개수 0**입니다. 위 표의 날짜 · 시간 · 접수번호 · 「낱말이 붙는 호칭」(학생 + 회관) 검사가 이런 문장을 봅니다.

「지우면 안 되는 것」 검사가 실제로 보는 문장들

검사 26개마다 들어 있는 문장을 한두 개씩 옮겼습니다. 그리고 아직 못 잡는 것도 그대로 적어 둡니다.

검사	지우지 않아야 하는 예
호실 · 연도 · 금액 · 동/층	제2공학관 27501호 에어컨 고장 / 2026년 2학기부터 계속 그렇습니다 / 수리비 1,500,000원 견적 / 3동 5층 복도 천장에서 물이 떨어집니다
날짜 · 접수번호 · 시간	2026-08-26 부터 증상이 있습니다 / 접수번호 FR-20260826-001 조회했습니다 / 14:30 경, 09:05 에도 동일 증상
호칭 같은 일반명사 · 직함만	고객님 응대 관련 문의드립니다 / 전기 과장 부재중이었습니다 / 시설팀 주임님께 전달했습니다
호칭 + 낱말 · 어미 붙은 명사	학생회관 2층 화장실 누수 (학생 + 회관) / 강의실 학생입니다만 / 전기실 담당자입니다
번호 매김 · 자릿수 혼동	항목 1) 0층 조명 점검 / 2019123456(학번) · 01012345678(전화) · 2020123456789(주민번호)를 서로 구분
호칭 없는 이름	담당자 정하늘 확인요망 / 김철수 010-1234-5678 → 전화만 지웁니다
이름 없는 호칭 · 직함	교수님하고 통화했습니다 / 교수실 에어컨이 고장났습니다 / 강사님 휴게실 정수기 고장
사람 앞에 흔한 낱말	우리 교수님 연구실 누수 / 청소 이모님께서 알려주셨어요
장소·부서 · 시간 낱말	전기실 과장님하고 통화했어요 / 최근에 교수님께서 말씀하셨습니다
띄운 하이픈 · 띄어 쓴 숫자	2026 - 10 - 27 점검 예정 / 단가 150000 1500000원 견적
숫자 경계 · 낱말 안쪽	예산 3000000000원 편성 / 주문번호 202012345678 / 홍길동 교수님께서 홍길동전 연구를 하십니다 → 「홍길동전」은 그대로
전각 숫자 · 쪼개진 한글 · 100만 자	２층 ３０５호 형광등 교체 / 맥에서 복사해 쪼개진 「고객님 응대 관련 문의드립니다」 / 개인정보 없는 글 100만 자 → 1초 안에 그대로

못 잡는 것

호칭이 없으면 이름을 못 거릅니다. 「담당자 정하늘 확인요망」, 「신고자 김철수」가 그렇고, 「교수 김철수 연구실」처럼 호칭이 앞에 오는 것, 성 없이 쓴 「철수 씨」도 못 거릅니다.

성씨 목록 39자 밖의 성은 호칭이 붙어도 지나갑니다. 나·민·진·지·엄·채·원·천·방·공·현·함·변·염·여·추·도·소·석·선·설·마·길·연·위·표 같은 성이라 「민경훈 님」, 「천지훈 교수님」은 그대로 나갑니다. 목록을 늘려 보니 「지도 교수님」·「현재 교수님」 같은 말이 이름으로 지워져서 늘리지 않았습니다.

넓힌 호칭·조사로는 안 보는 이름이 있습니다. 두 글자 이름(「김혁 강사님」, 「김혁 교수님하고」), 끝 글자가 실·장·관·소·한 등인 이름(「김은실 선생님하고」), 막아 둔 낱말로 시작하는 이름(「이미진 교수님하고」)은 처음 규칙(「교수님이」, 「선생님께」)으로만 잡고, 「김수한무 교수님」은 못 잡습니다.

목록에 없는 호칭과 다른 표기는 지나갑니다. 기사님 · 이사님 · 대표님 · 총장님 · 반장님 · 여사님 · 선배, 「학생들」 · 「님들」, 붙여 쓴 「담당자김철수 교수님」, 「김철수(교수)」, 영문 이름, 슬래시 · 물결 · 줄바꿈으로 끊은 번호(010/1234/5678), 한글로 쓴 숫자(「공일공」), test @ example.com, 끊어 쓴 학번(2019-312345)이 그렇습니다. 주소 · 계좌번호 · 차량번호 · 카드번호 · 생년월일은 처음부터 지우지 않습니다.

반대쪽도 뚫립니다. 쉼표 없는 10자리 금액 2050000000원은 [학번]원, 0으로 시작하는 계량기 번호 012345678은 [전화], 241027 1234567은 [주민번호]가 됩니다. 「조금 교수님께」, 「전기실 과장님이」처럼 목록에 없는 낱말 뒤에 호칭이 오면 [이름]으로 지웁니다. 그래도 멀쩡한 말을 지우는 쪽이 더 위험하다고 봤습니다. 이름이 아닌 말을 지우면 민원 내용이 무너집니다.

민원인에게 알렸습니다

개인정보처리방침 제6조(국외 이전)에 적은 내용입니다. 「개인정보 보호법」 제28조의8이 정한 고지 항목을 갖춰 처리방침에 공개했습니다.

항목	내용
이전받는 자	OpenAI, L.L.C. (privacy@openai.com)
이전 국가	미국
이전 항목	민원 본문 텍스트 — 자유롭게 작성하는 항목이므로 **이름·전화번호·호실 등 개인정보가 포함될 수 있습니다.**
일시 및 방법	웹 접수 화면에서 쓴 글이 HTTPS(TLS)로 암호화된 응용프로그램 인터페이스(API) 호출로 전송
이용 목적	민원 내용의 유형·긴급도 자동 분류 보조
보유 기간 (이전받는 쪽 OpenAI)	전송받은 데이터를 모델 학습에 이용하지 않으며, 오류·오용 점검 목적으로 **최대 30일** 보관한 뒤 삭제 (전송받은 자의 응용프로그램 인터페이스 데이터 처리 정책에 따름)

보관 기간은 두 군데를 따로 봐야 합니다

우리 쪽. 민원 기록은 민원이 끝난 뒤 **3년**, 첨부 사진은 **1년**, 접수 때 남는 접속 기록(IP 해시)과 AI에 보낸 글의 관측 기록은 **90일**이 지나면 매일 새벽 도는 파기 작업이 지웁니다.

OpenAI 쪽. 나간 글은 **최대 30일** 뒤 삭제되며, 이전받는 쪽 정책이라 우리 쪽 기간과 서로 대체하지 않습니다.

누구 이름으로 공개했나

이 방침은 **자산관리처 명의**로 공개돼 있습니다. 포털을 만들고 쓰는 것은 팀장·처장께 보고해 승인을 받았습니다.

방침에 **Google LLC**를 국외 이전 사업자로 적은 것은 백업·장애복구 때 국외 데이터센터로 옮겨질 수 있어서이고, **민원 데이터베이스와 서버 함수는 모두 서울 데이터센터**에 있습니다.

사업자(국가)	이용 서비스	전송되는 정보	목적
OpenAI, L.L.C.(미국) ●●●●@●●●.●●	통합민원·AI 문서창고	민원 본문 텍스트, 업로드 문서의 본문 텍스트	민원 자동 분류 보조, 문서 요약·검색 색인 생성

위 화면은 **포털 전체 개인정보처리방침의 국외 이전 표**에서 OpenAI 행만 잘라 온 것입니다(실제 공개 화면). 포털 전체 방침이라 여러 서비스가 한 행에 같이 적혀 있고, 위 표는 그중 통합민원 방침 제6조입니다. 회색 굵은 줄 두 개는 다른 사업자 행과 아래 각주를 잘라 낸 자리입니다.

공개 검사 파일 첫머리에 적어 둔 것

★ 테스트 데이터는 전부 가짜다. 실재 번호·실재 인물 이름을 넣지 말 것
(테스트 파일도 저장소에 남고, 저장소는 개인정보 보관 장소가 아니다).

이 문서의 예시는 공개 저장소(llm-mask-ko)의 검사 파일에서, 4쪽 코드 인용은 시설통합포털 저장소의 실제 파일에서 가져왔습니다. 번호와 이름은 검사용 가짜 데이터이며 호실 형식은 실제와 같습니다. 줄 수·성씨 39자·검사 개수·전화 자릿수와 예문 결과는 2026년 10월 27일 공개 저장소 파일(커밋 15cc140)로 다시 센 값입니다.