	Backend / Devops developer
	손석영
	Phone: 010-9404-4182
	Email Git: https://github.com/coder-ssy
	Blog: https://coder-ssy.tistory.com
SKILLS	
Back-End	JAVA SPRING BOOT PYTHON Django
DataBase	MySQL PostGreSQL
Devops	Nginx Docker Redis Jenkins GithubActions K6
	AWS(EC2, S3)
Monitoring	Grafana Promethus InfluxDB
project	
Techeer-Picture	상업용 이미지 비디오 자동 생성 프로그램 #생성형 AI #인스타 연동
	Batch Insert를 활용한 대량 데이터 삽입 속도 개선
	데이터가 많아질수록 트랜잭션 오버헤드 증가 우려 / 응답 속도 지연 및 DB 부하
	MySQL InnoDB 기반으로 트랜잭션 단위로 실행하여 초당 처리량(TPS): 350건 → 2850건으로 성능 향상
	Redis를 활용한 비동기 처리로 API 응답 속도 개선
	외부 API를 동기적으로 호출하면서 평균 3~5초의 응답 지연 발생, 서버 리소스 점유율이 증가하고 전체 처리 속도가 느려짐
	Redis List(Queue) 구조를 활용하여 작업을 큐에 저장, 외부 API를 비동기 처리 외부 API 응답 체감 시간 3.5초 → 0.1초로 단축
	분산 SQL 데이터베이스 적용
	EC2 기반 MySQL의 경우 단일 노드 한계로 인해 쓰기 작업이 집중되면 병목 현상 발생, 수평 확장이 어려워 효율성 저하 예상 ↓
	대량 쓰기 처리 속도 (10,000건 기준) 28~30초 → 5~6초 약 83% 감소 / TPS 300~400 → 1000 이상으로 약 2.5~3배 증가

Spring Cloud Gateway 도입을 통한 로드 밸런싱 및 트래픽 관리 최적화 각각의 컨트롤러 엔드포인트로 접근, 요청이 특정 인스턴스 또는 서비스에 몰릴 경우 트래픽 불균형 및 응답 속도 저하가 발생 Gateway → Service Layer 구조로 구성, 트래픽 분산 효과로 평균 응답 시간 1.2초 \rightarrow 0.7초 수준으로 약 42% 향상 Lazy Loading 적용을 통한 데이터 로딩 최적화 S3 자원에 바로 접근하거나 전체 객체를 불러오기 때문에 응답 시간이 길어짐, 실제 필요하지 않은 리소스까지 매번 반환 @ManyToOne(fetch = FetchType.LAZY) 적용, 리스트 렌더링 시 네트워크 전송량 100% → 40~50% 정도 약 50~60% 감소 **Activties** Techeer [테커] 2024.09 ~ 현재 실리콘밸리 개발자의 SW 개발자 커리어 그룹 프로젝트, 스터디, 네트워킹 행사 등 다양한 활동 참여 부트캠프 2024.06 ~ 2024.08 2024 하계 실리콘밸리 SW 부트캠프에서 우수상 수상 해커톤 2025.03.02 ~ 2025.03.03 미국 새크라멘토 UC Davis대학(캘리포니아 주도)에서 개최하는 SACHACKS VI에 참가