

# 이력서



이 름	이준수	영 문	Lee Junsu
생년월일	2001년 08월 19일 (만 25세)		
휴대전화	010-7129-8901	e-mail	rjaekawpxm@naver.com
GitHub	<a href="https://rjaekawpxm1-netizen.github.io/junsu-portfolio">https://rjaekawpxm1-netizen.github.io/junsu-portfolio</a>		
주 소	충청북도 괴산군 칠성면 괴강로 두천 4길 55-10		

## □ 학력사항

재학기간	학교명	전공	구분
20.03.01~26.02	청주대학교	인공지능소프트웨어학과	졸업
17.03.01~20.02	괴산고등학교	이공계열	졸업

## □ 병역사항

복무기간	군필 여부	군별 병과	기타
20.12.29~22.06.28	군필	육군 / 수송	만기 전역

## □ 보유 자격증

취득일	자격증명 (등급)	발행처
2025.11	ADSP	한국데이터산업진흥원
2025.12	GAIQ	구글
2020.01	운전면허증(1종보통)	충청북도경찰청장

## □ 교육 사항

교육명	K-Digital Training: 데이터 분석 데브코스
교육기관	프로그래머스
교육기간	25.12.30 ~ 26.04.21 (총 4개월) - 수료중
교육내용	▶ 데이터 분석 기본기 & 문제해결 역량

▶ SQL 기반 데이터 추출·가공 및 분석 역량

▶ Python 기반 통계·시각화·분석 역량

▶ 대시보드 및 분석 결과 시각화 역량

# 프로젝트 소개서

(프로젝트 상세내용은 포트폴리오 안에 있습니다)

## □ 프로젝트 소개서 1

프로젝트명	SECOM 반도체 생산 공정 데이터 기반 불량 예측 분석 (개인프로젝트)
프로젝트 내용	<p>SECOM 반도체 생산 라인에서 발생하는 센서 데이터를 활용하여 실제 공정상의 불량 제품을 예측하는 모델을 구축했습니다.</p> <p>수많은 센서 데이터 중 유의미한 변수를 추출하기 위해 데이터 전처리와 변수 선택 알고리즘을 적용했습니다.</p> <p>데이터 불균형 문제를 해결하기 위해 오버샘플링 기법을 활용하였으며, 이를 통해 불량 탐지 성능을 높이는 데 집중했습니다.</p> <p>분석 결과, 특정 센서 수치와 불량 발생 사이의 상관관계를 파악하여 공정 효율화의 근거를 마련할 수 있었습니다.</p>

## □ 프로젝트 소개서 2

프로젝트명	Bosch 생산라인 공정 데이터 불량 예측 분석 (개인프로젝트)
프로젝트 내용	<p>Bosch 생산라인 데이터를 기반으로 공정 과정에서 발생하는 불량 제품의 패턴을 분석하고 예측 모델을 구축했습니다.</p> <p>제품이 통과한 공정 시간, 방문 공정 수, 데이터 결측 패턴 등을 기준으로 공정 흐름과 불량 발생 관계를 분석했습니다.</p> <p>단순 공정 시간 비교를 넘어 특정 공정 라인 방문 여부와 공정 조합이 불량 발생에 어떤 영향을 미치는지 확인하는 데 집중했습니다.</p>

그 결과 특정 공정 구간 및 공정 방문 패턴이 불량 발생과 밀접하게 연결된다는 점을 확인하고, 이를 활용한 불량 예측 모델을 구축할 수 있었습니다.

### □ 프로젝트 소개서 3

프로젝트명	AI 스트레스 분석 앱 (팀프로젝트) - 팀장
프로젝트 내용	<p>사용자가 자신의 스트레스 상태를 객관적으로 인지하기 어렵다는 문제점에서 출발하여 분석 모델을 설계했습니다.</p> <p>입력 데이터(설문·음성·표정 등)를 기반으로 스트레스 상태를 다각도로 분석하고, 그 결과를 앱 화면으로 시각화하여 제공했습니다.</p> <p>분석 결과를 단순 수치로 보여주는 것에 그치지 않고, 사용자가 직관적으로 이해하고 실제 행동(휴식, 명상 등)으로 이어질 수 있도록 표현 방식에 집중했습니다.</p> <p>데이터 분석 결과를 실제 서비스 형태로 연결하는 전체 프로세스를 경험하며 팀장으로서 프로젝트를 리딩했습니다.</p>

### □ 프로젝트 소개서 4

프로젝트명	ROS 2 기반 로봇 시스템 노드 통신 및 제어 구현 (개인프로젝트)
프로젝트 내용	<p>ROS 2 환경에서 로봇 시스템의 기본이 되는 퍼블리셔와 서브스크라이버 노드 간의 통신 구조를 설계하고 구현했습니다.</p> <p>사용자 정의 메시지 타입을 생성하여 로봇 상태 정보를 효율적으로 전달하고, 패키지 의존성 관리 및 빌드 시스템을 활용한 개발 환경을 구축했습니다.</p> <p>Python 을 활용해 노드별 로직을 작성하였으며, 런타임에서 발생하는 패키지 가시성 및 실행 오류를 디버깅하며 안정적인 통신 시스템을 완성했습니다.</p> <p>이 과정을 통해 분산형 로봇 운영체제의 아키텍처를 이해하고, 시스템 엔지니어링의 기초가 되는 미들웨어 활용 역량을 쌓을 수 있었습니다.</p>