

1. 자기소개
2. 스킬
3. 해외활동
4. 근무
5. 프로젝트



# 이력서

## Contact

Name :   
Phone :   
Email :

## Channel

Github :   
Blog :

자기소개

## Introduction

### Write and Share 기록과 공유

Conclusion  
: 다른 사람이 이해하기 쉽도록  
내용을 재구성하는 능력을 기렀다

자신의 개발관련 성격 기술  
또 장점을 소개

기술 블로그를 운영하면서 알고리즘 풀이와 CS 지식, 개발 관련 글을 꾸준히 올려 다른 사람들과 공유하였습니다.

문제 풀이 해결 능력 향상을 위해 알고리즘 문제를 꾸준히 풀어 깃허브와 블로그에 해당 문제 풀이를 올렸습니다. 풀이를 작성하면서 문제 복기를 한 번 더 할 수 있는 기회가 되었고 문제 풀이에 어려움을 가진 다른 사람들의 시각에서 쉽게 이해될 수 있도록 시각 자료(배열의 값 변화 등)를 만들어 풀이 내용을 작성하였습니다. 이러한 과정을 통해서 문제의 접근 방식을 빠르게 판단 할 수 있게 되었습니다.

풀이 진행방식을  
다른 사람이 이해할 수  
있도록

서울시 공공데이터 API를 활용하여 현재 위치 기준으로 WIFI 정보를 가져오는 프로젝트 진행한 적이 있습니다. 공공데이터 API 관련 글을 찾던 중 원하던 자료가 없었고 여러 자료를 참고하여 진행하였습니다. 이때 한 번에 볼 수 있는 자료가 있으면 해당 자료를 찾는 사람들에게 편리할 것 같다는 생각에 Java로 JSON 파싱하는 과정에 대해 글을 작성하여 공유하였습니다. → 공유의 결과/Content의 질에 대한 평가가 있어야

위키 내용

### Repeat and Check 반복과 확인

한 개의 기능을 개발 완료 후 다음 기능 개발로 넘어가지 않고, 완성된 기능에 대해 다양한 사례로 점검 후 정확성이 입증되면 다음 기능 개발로 진행하였습니다.

두 좌표에 대해서 거리 계산을 하여 응답하는 기능을 개발하던 중 Google Map API 중에서 두 지점 사이의 거리를 계산하는 distance API가 있다는 것을 알고 distance API를 이용하여 기능을 구현하였습니다. 기능 구현 완료 후 특정 지점 A와 B의 거리 계산을 시도하고 원하는 값이 나왔습니다. 하지만 여기서 멈추지 않고 해당 값이 정확하게 나왔는지 다시 한번 더 검사를 해보며, 다른 지역 C와 D도 테스트해 보면서 해당 기능의 정확성을 검증하였습니다.

다양한 시나리오  
error 점검  
왜 check  
일관성 check

이 과정 중에 특정 지역에서 거리 계산이 안 된다는 오류를 발견하였고 기존에 사용하였던 Google Map API 중 distance API 사용 대신에 두 좌표의 위도와 경도를 이용한 거리 계산 공식을 이용하여 거리 계산을 하였습니다. 이때 다양한 테스트에서도 정확한 값을 응답하는 것을 확인 후 다음 기능 개발을 진행하였습니다.

오류 발견 →  
해결방안 작성

## Skills

- Languages : Java Spring boot Javascript Python
- Database : MySQL Mariadb Redis
- Tool : Git Github Postman Intelli J Docker DataGrip

→ 스킬이  
살짝 작아져 눈에 안들어옴

## Overseas Experience

### Onikon (캐나다 인턴쉽)

부동산 마케팅 회사

2018.07 ~ 2018.08 (8주)

- 교내 해외 인턴쉽 프로그램 참여하였습니다.
- 부동산 마케팅 관련 회사에서 대량의 지역별 학교 데이터에서 각 학교별 최신 데이터 추출 업무 담당하였습니다. 이때 정렬을 통하여 최신 정보를 추출 할 수 있었습니다.

- 첫 현업의 분위기를 직접 느낄 수 있었으며, 현업에서 업무 진행하는 방식에 대해서 경험 할 수 있었습니다.
- 학교에서 배운 지식을 업무에 적용해보면서 개발 직무에 관심이 가는 계기가 되었습니다.

## Work Experience

### 와스콘

플랜트 현장에 필요한 밸브를 조립하여 납품하는 회사

2020.01 ~ 2022.07 (2년 7개월)

영업지원부 / 사원

- 프로젝트 사양서를 참고하여 고객사에서 원하는 스펙으로 밸브 견적서 작성 및 승인문서 작성하였습니다.
- 밸브에 관한 지식이 없는 상태로 입사하여 입사 초반에 직무를 수행하는데 많은 어려움이 있었습니다. 업무를 잘 수행하기 위해서 퇴근 후 선배들에게 얻은 자료를 바탕으로 공부를 하였고, 모르는 부분에 대해서는 선배들에게 도움을 청해 업무를 수행할 수 있었습니다. 그 결과 업무 수행 속도가 빨라졌습니다.
- 업무 진행에 대해서 정확도와 속도가 상승되면서 담당하는 프로젝트의 규모가 커지며 주주 확률이 높아졌습니다.
- 처음 경험해보는 업무에서도 잘 적응하여 업무를 수행 할 수 있다는 자신감이 생겼습니다.

· 잘 모르는 사항은 바로 물어보려고 함

### Project → 다양한 업무 / 취득한 자질을 재능으로

#### A+uction

- 프론트엔드와 협업 프로젝트 한 팀 프로젝트
  - 기간 : 23.03.24 ~ 23.05.02 (약 5주)
  - 소개 : 온라인 경매 프로그램
  - 인원 : 6명 (FE : 2명 / BE : 4명)
  - 기술 스택 : Spring boot, Java, JPA, Maria db, Elastic search, Docker, AWS
  - 담당 → 수행 내용
    - Elasticsearch 을 이용하여 경매 이름, 카테고리 검색 기능을 구현하였습니다.
    - Member 테이블과 Auction 테이블을 wish 테이블 각각 join 하여 wish list (관심 목록)을 구현하였습니다.
    - 회원, 경매 정보 수정 기능을 구현하였습니다. 경매 수정은 경매 입찰 시작 전에만 가능 하도록 구현하였습니다.
    - 프로젝트 진행 과정, API 명세서 등 프로젝트와 관련된 내용을 노선에 서류화 하는 작업을 하였습니다.
  - 프로젝트 통해서 경험 및 취득한 점
    - 검색 엔진인 Elasticsearch를 사용하면서 Elasticsearch의 구조를 익힐 수 있었으며, 경매 아이템, 카테고리 검색 기능을 구현 할 수 있었습니다. 또한 테스트를 통해 응답값이 정렬되지 않은 상태로 응답하는 것을 알게 된 후 정렬 조건을 추가하여 원하는 응답 값을 받을 수 있었습니다.
    - Github flow 로 Git 관리 하였으며, 개발 진행 중에 문제가 될 부분 혹은 추가 구현할 부분을 Issue로 등록하여 팀원들과 공유 하였습니다.
    - BDD 방식으로 Test 코드 작성하여 코드 작성에서 놓친 부분과 오류를 찾아 해결 하였습니다. 또한 test 코드를 통해 팀원들에게 작성한 AP에 대해서 신뢰성을 높일 수 있었으며, 완성도를 높일 수 있었습니다.
- 깃허브 : [redacted]

#### Order-drink

- 스타벅스 사이렌 오더 백엔드 api (개인 프로젝트 - 클론코딩)
  - 기간 : 2023.03.03 ~ 2023.03.27 (약 4주)
  - 기술 스택 : JAVA, Spring boot, JPA, Mysql, Google map API, SSE, Docker, Swagger
  - 주요 기능

- Google map API 을 이용하여 사용자 위치에서 가까운 지점과 거리를 제공합니다.
- 음료 제작이 완료 되면 SSE 을 이용하여 음료 주문자에게 음료 제작 완료 알림을 제공합니다.
- 관리자는 각 지점별 매출을 조회 할 수 있습니다.
- 프로젝트 통해서 경험 및 체득한 점
  - 외부 API를 사용 할 때는 주로 rest template을 사용한다는 것을 알게 되어 Google map API사용 시에 rest template을 적용하여 구현하였습니다.
  - Google map API를 사용하여 사용자 위치의 좌표를 응답 받을 수 있었으며, 좌표를 통해 거리 계산을 하여 사용자 위치에서 가까운 지점을 응답 할 수 있었습니다.
  - 음료 제작 완료 후에 user 에게 제작 완료 알림을 SSE 를 통하여 구현하였습니다.
  - BDD 방식으로 test 코드를 작성하면서 발견 못했던 오류를 발견하였으며, test 코드의 중요성을 알게 되었고 이로 인해 프로젝트의 완성도를 높일 수 있었습니다. → 오랜 잡내고 해결하는 능력
- 깃허브 : [redacted]

## SOOB

- 학교 졸업작품 (캡스톤 디자인)
  - 기간 : 2019.03 ~ 2019.06 (4개월)
  - 소개 : 실시간 검색어를 기반으로 키워드의 카테고리 분류 및 키워드의 정보를 확인하는 웹페이지
  - 인원 : 3명
  - 기술 스택 : Python, MySQL, CSS3, HTML5, JavaScript, PHP, Naver blog API, Google chart API, AWS
  - 담당 → 팀에서 코드로 역할, 커뮤니케이션 능력
    - 팀 내에서 팀장을 맡아 팀원들과의 의견 조율, 프로젝트를 이끌어서 학과 내에서 2등이라는 성과를 얻었습니다.
    - 파이썬을 이용하여 인스타그램 게시글을 크롤링하였으며, 크롤링을 통해 얻은 한글 데이터를 형태소 분석하였습니다.
  - 프로젝트 통해서 경험 및 체득한 점
    - 크롬드라이버를 통해 인스타그램을 파이썬으로 크롤링하였습니다.
    - KoNLPy 라이브러리를 사용하여 한글 형태소 분석을 할 수 있었습니다.
  - 깃허브 : [redacted]

## Education

### 제로베이스 백엔드 스쿨

- 기간 : 2022.11 ~ 2023.05
- 학습 내용 : Java, Spring boot, 자료구조, CS, 프로젝트 진행
- 온라인 부트캠프 스쿨

- 기간 : [redacted]
- 학과 : [redacted]
- 학점 : [redacted]

## Certificate

### 정보처리기사

- 취득일 : 2019.08.16
- 기관 : 한국산업인력공단

### SQLD (SQL개발자)

- 취득일 : 2022.12.02
- 기관 : 한국데이터산업진흥원



# PORTFOLIO



## INDEX

1. A\_uction : 온라인 경매 시스템
  - a. Elasticsearch 구현
2. Order drink : 음료 주문 백엔드 API
  - a. Google map API, SSE 구현
3. SOOB : 실시간 검색어를 기반으로 키워드의 카테고리 분류 및 키워드의 정보를 확인하는 웹페이지
  - a. Instagram crawling, 한글 형태소 분석



## A+uction

온라인 경매 시스템

기간 : 2023.03.31 ~ 2023.05.02 (5주)

인원 : 6명 (FE : 2 명 / BE : 4 명)

깃허브 :



ERD



아이템 이름 검색



wish list

간편하고 큰 화면의 수정은 모바일로 잘 돌아감

## 사용스택

Java Spring boot v2.7.10 JPA Mariadb Elasticsearch Swagger

IntelliJ Postman Docker DataGrip Git Github RESTful API

## 담당

Elastic search Wish list Member, Auction information modify 프로젝트 서류화(Notion)

- Elasticsearch 을 이용하여 경매 이름, 카테고리 검색 기능을 구현하였습니다.
- Member 테이블과 Auction 테이블을 wish 테이블 각각 join 하여 wish list (관심 목록)을 구현하였습니다.
- 회원, 경매 정보 수정 기능을 구현하였습니다. 경매 수정은 경매 입찰 시작 전에만 가능 하도록 구현하였습니다.

## 프로젝트 통해서 경험 및 체득한 점

### Elasticsearch 사용하여 경매 물품, 카테고리 등 검색 기능 구현

- 검색 엔진인 Elasticsearch를 처음 적용해보면서 Elasticsearch의 구조에 대해서 학습할 수 있었습니다. Elasticsearch는 역색인 구조를 사용하며, 데이터는 JSON 구조, RDB 에서 부르는 명칭과 Elasticsearch에서 부르는 명칭이 다르다는 것(RDB: Database / Elasticsearch: Index etc.)을 알게 되었습니다. 처음 사용하는 것이라 프로젝트에 적용하는 데 시간이 걸렸지만 다양한 참고문서를 참고하여 프로젝트에 적용할 수 있었습니다.
- Elasticsearch는 score를 계산하여 결과값을 응답하는데 이것을 테스트해 보면서 경매를 검색할 때 응답이 정렬되지 않는 상태로 응답하는 것을 알게 되었습니다. 이것을 해결하기 위해서 쿼리 문에 정렬 조건을 넣어서 해결할 수 있었습니다.

### Github flow 로 Git 관리

- Pull request를 통하여 팀원의 코드를 리뷰할 수 있었고 팀원이 남긴 comment를 참고하여 코드를 수정할 수 있었습니다. 이것을 통하여 서로의 코딩 작성 방식을 배울 수 있었습니다.
- Pull request 작성 시에 최대한 부드러운 단어를 선택하여 작성해야 한다는 것을 알게 되었습니다. 문서의 특성상 읽는 사람에 따라서 다르게 의미를 받아들여 오해가 생기기 때문입니다.
- 프로젝트 진행 중 추가로 필요한 기능 혹은 개선 되어야 할 점은 Issue에 해당 내용을 기재하여 팀원 모두 확인 할 수 있게 하였습니다.

### BDD 방식으로 Test 코드 작성

- 작성한 코드에 대해서 BDD 방식으로 test 코드를 작성하였습니다. 테스트 코드 작성을 통하여 코드 작성에서 놓친 부분을 찾을 수 있었습니다. 경매 수정 시 시간과 관련된 정책(경매 입찰 시간은 경매 종료 시각 이전 / 경매 종료 시각은 경매 입찰 시간 이후 / 경매 입찰 시간은 경매 수정 시점 이후) 중에서 경매 입찰 시간이 등록일 이전에도 수정이 되는 것을 발견하여 해당 부분을 수정할 수 있었습니다.
- Test 코드를 작성하여 팀원들에게 구현한 API의 신뢰성을 높일 수 있었으며, 완성도를 높일 수 있었습니다.

### Docker 사용

- Docker를 통하여 팀원들과 프레임워크 버전을 일치시켰으며, AWS를 통해 해당 프로젝트를 배포할 수 있었습니다.



## Order drink

스타벅스 사이렌오더 클론한 Back-end API 입니다.

기간 : 2023.03.06 ~ 2023.03.31 (4주)

인원 : 1인 (개인)

깃허브 :



입력받은 주소 기준으로 제일 가까운 지점을 거리 순으로 2개 응답 (Google Map API 사용)



특정 기간내에 모든 지점의 매출액 확인



음료 제작이 완료 되면 주문자에게 제작 완료 알림 전송 (SSE 사용)



ERD

## 사용스택

Java Spring boot v2.5.6 JPA Gradle Mysql Google map API SSE Swagger

IntelliJ Postman Docker DataGrip Git Github

RESTful API

## 주요 기능

- Google map API 을 이용하여 사용자 위치에서 가까운 지점과 거리를 제공합니다.
- 음료 제작이 완료 되면 SSE 을 이용하여 음료 주문자에게 음료 제작 완료 알림을 제공합니다.
- 관리자는 각 지점별 매출을 조회 할 수 있습니다.

## 프로젝트 통해서 경험 및 체득한 점

### Google Map API 사용

- Google map API 중에서 distance API 가 있다는 것을 알게 된 후 두 지점의 거리 계산을 distance API를 이용하여 계산하려고 하였으나. 하지만 특정 두 지점에 대해서 거리 계산이 불가능하였습니다. 데이터의 정확성을 위해서 두 좌표 거리 계산하는 공식을 가져와서 계산 하였습니다.
- Google map API를 이용하면서 웹브라우저(크롬)에서는 한글로 나오지만 IDE(IntelliJ)에서는 응답이 영문으로 나오는 것을 API 파라미터값에 `language=ko` 을 추가하여 한글로 응답을 받을 수 있었습니다.
- 외부 API 사용 시에는 rest template 을 사용한다는 것을 알게 되었습니다. Rest template 을 사용하면 new로 새로 인스턴스 사용할 때보다 비용이 절약된다는 것을 알게 되었습니다. 이것을 반영하여 google map API 사용한 부분을 수정하였습니다.

### SSE를 통하여 알림 서비스 구현

→ 다양한 방법을 찾고 최상의 방법을 선택

- 음료 제작 완료 후 알림 서비스 구현 방법에 대해서 web socket과 SSE 중 어떤 것을 사용할지에 대해 고민하였습니다. 주문 완료 알림은 사용자에게 단방향으로 진행되며, web socket과 같이 양방향성 로직은 오버 스택이라 판단하여 단방향인 SSE를 사용하였습니다.

### Native Query 사용

- 각 지점의 매출 합을 구하기 위해서는 group by를 사용하여 구현하고 싶었지만, JPA에서는 group by가 지원하지 않아 다른 방법으로 접근해야 했습니다. 그중 한 가지 방법인 native query를 사용하여 구현하였습니다.

### Swagger UI 사용하여 API 명세서 작성

- API를 문서화하면서 동시에 테스트가 가능한 swagger가 적절하다고 판단이 되어 API 명세서를 swagger 통하여 작성하였습니다.
- 작성한 API를 한눈에 볼 수 있다는 장점이 있었으며, Restful 형식에 맞게



## SOOB

실시간 검색어를 기반으로 키워드의 카테고리 분류 및 키워드의 정보를 확인하는 웹페이지

기간 : 2019.03 ~ 2019.06 (4개월)

인원 : 3명

깃허브 : 



## 사용스택

Python MySQL Naver blog API PyCharm Visual Studio Code AWS

## 담당

팀장 Database Instagram Crawling

- 팀 내에서 팀장을 맡아 팀원들과의 의견 조율, 프로젝트를 이끌어서 학과 내에서 2등이라는 성과를 얻었습니다.
- 파이썬을 이용하여 인스타그램 게시글을 크롤링하였으며, 크롤링을 통해 얻은 한글 데이터를 형태소 분석하였습니다.

## 프로젝트 통해서 경험 및 체득한 점

### Python 을 활용하여 웹 크롤링

- 실시간 검색어, 블로그, 인스타그램 정보를 가져오기 위하여 크롤링 과정을 진행하였습니다. 프로젝트를 진행하면서 처음 크롤링의 개념을 접하게 되었으며, 크롤링하기 위하여 웹페이지의 구조를 먼저 파악해야 한다는 점을 알게 되었습니다. 또한 크롤링을 파이썬으로 진행하면서 파이썬을 처음 접하게 되었습니다.
- 파이썬을 이용해 인스타그램 게시글을 크롤링하였는데 이때 크롬드라이버를 이용하였습니다.

### 형태소 분석을 위해 KoNLPy 라이브러리 사용

- 한국어 형태소 분석하는 데는 KoNLPy 라이브러리가 대표적이라는 것을 알게 되었습니다. KoNLPy 라이브러리를 통해 한글 형태소 분석을 하였습니다. 이때 트위터(Twitter) 모듈을 이용하여 프로젝트에서 필요한 형태소(명사, 형용사, 동사)를 분리하였습니다.