

# 데이터베이스 프로젝트 보고서

<항공권 가격 변동 정보 추적>



컴퓨터공학과 박진우 19011566

데이터사이언스학과 이하은 20011830

## 목차

|                            |    |
|----------------------------|----|
| I. 프로젝트 동기 .....           | 3  |
| II. 현황 및 문제점 분석 .....      | 4  |
| III. 프로젝트 목적 .....         | 5  |
| IV. 기대효과 .....             | 5  |
| V. 요구사항 분석 .....           | 6  |
| VI. 개념적 설계 .....           | 7  |
| VII. 논리적 설계 .....          | 8  |
| VIII. 데이터 수집 및 확보 방안 ..... | 10 |
| IX. 물리적 설계 및 GUI 구현 .....  | 11 |
| X. 향후 개발 계획 .....          | 16 |

## I. 프로젝트 동기

### A. 항공권의 과거 가격 변동 정보 파악의 어려움

2023 년 5 월, 신종 코로나바이러스 감염증(코로나 19) 팬데믹이 공식적으로 종료되었고 여행 정보가 해제되었습니다. 2 년 만에 해제된 경보에 전 세계적으로 여행 수요가 증가하고 있습니다. 특히 해외여행 재개로 인해 항공권의 가격은 점점 상승하고 있습니다. 이러한 상황 속에서 소비자들 사이에서는 저렴한 항공권 정보에 대한 수요가 늘어나고 있습니다.

대부분의 소비자들은 스카이스캐너, 네이버항공, 구글 플라이트와 같은 항공권 조회 사이트에서 가격 정보를 찾습니다. 위 사이트에서는 검색 당시의 항공권 가격 정보만을 제공합니다. 그러나 합리적인 소비를 하기 위해서는 과거 항공권 가격의 변동 정보까지 정확하게 알아야 합니다. 즉, 소비자들은 검색 당시의 가격과 과거 가격 변동의 추이를 모두 알고 싶어 하지만, 기존의 사이트에서는 그러한 정보를 제공하지 않는 실정입니다.

### B. 최저가 항공권의 가격 및 예약 시기 파악의 필요성



<그림 1>

<그림 2>

모든 경제주체들은 적은 비용으로 최상의 결과를 얻는, 합리적인 선택을 추구합니다. 항공권을 구매하려는 소비자들 역시 마찬가지로, 항공권을 최저가에 구매하고 싶어 합니다. <그림 1>, <그림 2>에서 알 수 있듯이 노선마다 차이가 있지만, 대부분 3~4 개월 전 예약하는 항공권이 제일 저렴하고, 2 개월 미만으로 남은 시점부터는 항공권 가격이 점차 상승합니다. 그렇다면 3~4 개월 전에 항공권을 예약하는 것이 가장 저렴할까요?

꼭 그렇지만은 않습니다. 항공권은 다양한 변수에 의해 가격이 자주 변동하므로 가격 변동 추이에 따라 가장 합리적인 예약 시기를 파악해야 합니다. 위의 정보들은 공신력 있는 자료가 아닌 일종의 참고자료에 불과합니다. 그렇기에 항공권 가격 변동 추이를 나타내는 정보가 있다면, 최근 몇 달간 특정 노선 또는 항공여정에 대해 최저가를 파악하기에 수월할 것입니다. 또한 여행 일정에 따른 최적의 항공권 예약 시기도 파악할 수 있습니다. 이 말은 미래의 항공권 가격이 어떻게 변할지는 알 수 없지만, 과거 변동 이력에 따른 어느 정도의 예측은 가능할 것이라는 뜻입니다.

위와 같은 이유로 과거 항공권 가격의 변동 정보를 수집하여 제공한다면, 많은 사람들이 항공권을 예약할 때 편리함을 느낄 수 있다고 생각했기 때문에 본 프로젝트를 구상하게 되었습니다.

## II. 현황 및 문제점 분석

### A. 과거 항공권 가격의 변동 정보 미제공



<그림 1>

<그림 2>

현재 존재하는 항공권 조회 사이트는 과거 항공권 가격의 변동 정보를 제공하지 않습니다. <그림 1>과 같이 '구글 플라이트'에서 가격 변동 내역을 제공하지만, 해당 정보는 특정 항공편이 아닌 조회된 노선 전체의 평균치를 제공합니다. 또한 제한적인 몇몇 노선에 대해서만 제공하며 60 일보다 오래된 가격 정보는 확인할 수 없습니다. 세부 내역 파악을 위한 구글 플라이트 api 도 제공하지 않아 수집된 데이터를 자세히 알 수도 없습니다.

사용자가 가격 정보의 변동을 알고 싶어 한다면, <그림 2>와 같이 특정 항공노선의 가격 정보의 변동이 있을 때 사용자에게 메일을 보내 알림을 주는 기능을 몇몇 사이트에서 제공합니다. 그러나 이러한 기능은 변화된 가격에 따른 알림일 뿐 적절한 예약 시기를 파악하는데 도움을 주지 않습니다

## III. 프로젝트 목적

### A. 항공권의 과거 가격 변동 정보를 구체적으로 제공

현재 존재하는 많은 사이트는 과거 항공권 가격의 변동 정보를 제공하지 않으며, 사용자가 원하는 정보를 찾는 데 어려움이 있습니다. 그래서 사용자들은 항공권 가격을 비교하고 분석하는 과정에서 제한된 정보로 인해 최적의 선택을 하지 못합니다. 따라서 저희는 사용자들이 필요로 하는 정보들을 지속적으로 수집하여, 사용자가 원할 때 활용할 수 있는 시스템이 필요하다고 판단하였고, 이를 구축하고자 했습니다. 이를 통해 사용자들은 노선, 항공사, 날짜 등 원하는 모든 조건의 항공권들의 가격 변동 정보를 알 수 있습니다.

### B. 적절한 항공권 구매 시기 및 최저가 정보 파악

항공권 가격은 계절, 휴일, 항공사의 운영 정책, 수요와 공급의 변화 등 여러 변수에 의해 변동됩니다. 과거 항공권 가격의 변동 정보를 수집하여 추이를 살펴본다면, 가까운 미래에 어떠한 변수가 있을지 추측할 수 있을 것입니다. 이를 통해 사용자가 항공권을 구매하기에 가장 적절한 시기와 최저가 정보를 파악할 수 있습니다.

## C. GUI 를 제공하여 사용자 경험 최적화

양질의 데이터를 수집하는 것은 물론이고, 수집된 데이터를 활용하는 것 역시 편리해야 합니다. 예상 고객층은 공학 전공자가 아닌 일반인이므로, 복잡한 SQL 문을 이용하여 조회하지 못합니다. 그래서 저희는 DBMS 를 이용한 조회 뿐 아니라 GUI 를 활용하여 누구나 쉽게 항공권의 가격 변동 정보를 조회하는 방법을 구상해 보았습니다.

# IV. 기대 효과

## A. 사용자의 필요에 맞는 세부적인 항공권 가격 변동 정보 파악 가능

본 프로젝트를 통해 사용자들은 세부적인 검색 조건을 통해 항공권 가격 변동 정보를 파악할 수 있습니다. 앞서 언급했듯 기존의 항공권 조회 웹사이트에서는 과거 항공권 가격 변동 정보를 일체 제공하지 않습니다. '구글 플라이트'에서 대략적인 정보를 제공하고 있으나, 여러 문제점들로 인해 사용자가 활용하기 어렵습니다. 반면, 저희는 과거의 항공권 가격 변동 정보를 직접 수집하였기에, 소비자가 세부적인 검색 조건을 지정할 수 있습니다. 가격, 날짜, 항공사, 동일 항공권의 가격 변동 내역 등 '네이버 항공권' 사이트에 존재하는 모든 정보를 검색 조건으로 사용할 수 있습니다. 추가적으로 해당 정보를 가공한 국내 항공사, 해외 항공사, 국가 정보, 공항 정보 등의 검색 조건도 적용 가능합니다.

## B. 최적의 예약 정보 제공을 통한 경제적 이익 및 시간 절약

사용자가 직접 수집한 데이터로 변동 추이를 살펴보기 때문에 최적의 항공권 예약시기를 파악할 수 있습니다. 또한 항공권 가격의 변동 정보를 통해 최근 며칠, 몇 달의 최저가를 파악할 수 있고 추가적인 분석을 통해 항공권 가격 예측도 가능할 것이라 생각합니다. 결과적으로 최저가 항공권과 최적의 예약시기를 파악함으로써 경제적 이익을 수반합니다. 항공권과 정보를 검색하는데 사용되는 시간도 줄일 수 있습니다.

## C. 사용자의 항공권 조회 프로그램 활용 가능성 극대화

현재 수집된 데이터는 국토교통부에서 제공한 취항률 상위 18 개 노선을 대상으로 180 일간의 정보를 수집합니다. 사용자가 다른 노선을 조회하거나, 노선을 줄이거나, 조회 기간을 늘리는 등의 수정도 쉽게 이루어집니다. 크롤링 프로세스의 자동화로 인해, 새로운 데이터를 수집하려는 노력 없이 원하는 공항의 공항코드만으로 수집이 이루어집니다. 필요한 정보를 검색하는 것이 아니기 때문에 누구나 프로그램을 수정하여 입맛에 맞게 사용할 수 있습니다. 또한 수집된 정보는 GUI 를 이용하여 조회할 수 있으므로 SQL 질의문을 모르는 사용자도 편하게 이용할 수 있습니다.

## V. 요구사항 분석

### A. 요구사항 명세서 작성

1. 항공권은 항공권 ID, 항공권을 검색한 날짜, 출발 날짜, 출발 시각, 도착 시각, 운항 시간, 가격, 결제 혜택 정보, 등의 정보를 갖는다.
2. 항공권은 하나의 항공사에 의해 운영된다.
3. 항공권은 하나의 노선을 운항한다.
4. 노선은 노선 ID, 예상 운항 시간, 운항하는 항공사의 수를 정보로 갖는다.
5. 노선 정보는 출발 공항에서 도착 공항을 목적지로 한다.
6. 공항 정보는 공항 코드, 터미널의 수, 규모를 정보로 갖는다.
7. 공항은 하나의 국가에 위치한다.
8. 국가는 국가 코드, 국내총생산, 연 관광객 수, 공항 수를 정보로 갖는다.
9. 항공사 정보는 항공사 코드, 사업 모델을 정보로 갖는다.
10. 항공사는 하나의 국가를 국적으로 가진다

## VI. 개념적 설계

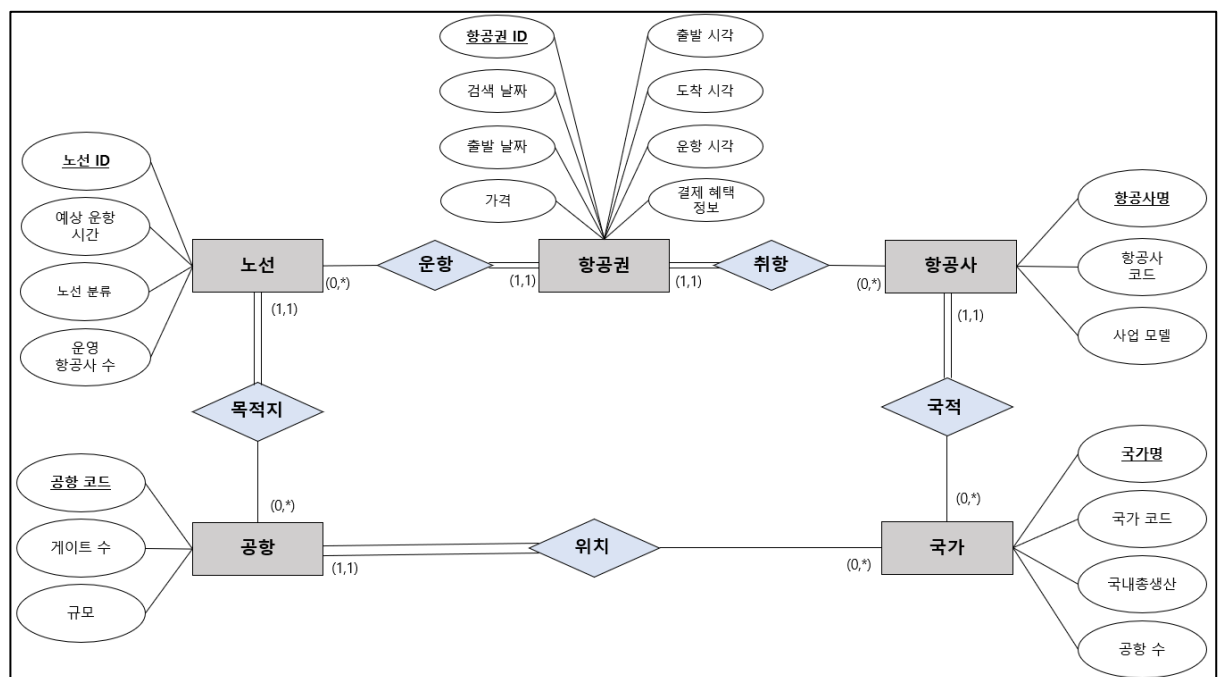
### A. 개체 및 속성 추출

| 개체  | 속성  |
|-----|---|
| 항공권 | 항공권 ID, 수집 날짜, 출발 날짜, 출발 시각, 도착 시각, 운항 시간, 가격, 결제 혜택 정보 |
| 노선  | 노선 ID, 예상 운항 시간, 노선 분류, 운항 항공사 수                        |
| 항공사 | 항공사명, 항공사 코드, 사업 모델                                     |
| 공항  | 공항 코드, 터미널 수, 규모  |
| 국가  | 국가명, 국가 코드, 국내총생산, 공항 수                                 |

### B. 관계 추출

| 관계    | 관계에 참여하는 개체         | 관계 유형 |
|-------|---------------------|-------|
| 취항    | 항공권 (필수) - 항공사 (선택) | 일대다   |
| 운항    | 항공권 (필수) - 노선 (선택)  | 일대다   |
| 목적지   | 노선 (필수) - 공항 (선택)   | 일대다   |
| 위치    | 공항 (필수) - 국가 (선택)   | 일대다   |
| 국적 소속 | 항공사 (필수) - 국가 (선택)  | 일대다   |

### C. E-R 다이어그램 작성



## VII. 논리적 설계

### A. 릴레이션 스키마 변환

|          |        |          |       |       |       |       |    |          |       |      |
|----------|--------|----------|-------|-------|-------|-------|----|----------|-------|------|
| 항공권 릴레이션 | 항공권 ID | 수집 날짜    | 출발 날짜 | 출발 시각 | 도착 시각 | 운항 시간 | 가격 | 결제 혜택 정보 | 노선 ID | 항공사명 |
| 노선 릴레이션  | 노선 ID  | 예상 비행 시간 | 노선 분류 | 항공사 수 | 출발 공항 | 도착 공항 |    |          |       |      |
| 항공사 릴레이션 | 항공사명   | 항공사 코드   | 사업 모델 | 소속 국가 |       |       |    |          |       |      |
| 공항 릴레이션  | 공항 코드  | 게이트 수    | 규모    | 소속 국가 |       |       |    |          |       |      |
| 국가 릴레이션  | 국가명    | 국가 코드    | GDP   | 공항 수  |       |       |    |          |       |      |

### B. 테이블 명세서 작성

#### 1. Flight Ticket 테이블

| 테이블명(한글) |          | 항공권              |             | 테이블 설명 |     | 수집 날짜에 따른 항공권 가격 정보 |              |    |
|----------|----------|------------------|-------------|--------|-----|---------------------|--------------|----|
| 테이블명(영어) |          | Flight Ticket    |             |        |     |                     |              |    |
| 순번       | 컬럼명(한글)  | 컬럼명(영어)          | 자료형         | 널 여부   | 기본키 | 외래키                 | 설명           | 비고 |
| 1        | 항공권 ID   | Flight Ticket ID | INT         | NN     | PK  |                     | 항공권 정보 식별 번호 |    |
| 2        | 수집 날짜    | Searching Date   | VARCHAR(50) | NN     |     |                     | 항공권을 검색한 날짜  |    |
| 3        | 출발 날짜    | Departure Date   | VARCHAR(50) | NN     |     |                     | 항공권 출발 날짜    |    |
| 4        | 노선 ID    | Route ID         | INT         | NN     |     | FK                  | 노선 식별 번호     |    |
| 5        | 출발 시각    | Departure Time   | VARCHAR(50) | NN     |     |                     | 항공권 출발 시각    |    |
| 6        | 도착 시각    | Arrive Time      | VARCHAR(50) | NN     |     |                     | 항공권 도착 시각    |    |
| 7        | 운항 시간    | Flight Time      | VARCHAR(50) | NN     |     |                     | 항공권 운항 시간    |    |
| 8        | 항공사명     | Airline          | VARCHAR(50) | NN     |     | FK                  | 항공사 이름       |    |
| 9        | 가격       | Price            | INT         | NN     |     |                     | 가격           |    |
| 10       | 결제 혜택 정보 | Card Benefit     | VARCHAR(50) | Y      |     |                     | 카드 혜택 결제 정보  |    |

#### 2. Route 테이블

| 테이블명(한글) |          | 노선                   |             | 테이블 설명 |     | 항공 노선 정보 |           |         |
|----------|----------|----------------------|-------------|--------|-----|----------|-----------|---------|
| 테이블명(영어) |          | Route                |             |        |     |          |           |         |
| 순번       | 컬럼명(한글)  | 컬럼명(영어)              | 자료형         | 널 여부   | 기본키 | 외래키      | 설명        | 비고      |
| 1        | 노선 ID    | Route ID             | INT         | NN     | PK  |          | 노선 식별 번호  |         |
| 2        | 예상 비행 시간 | Expected Flight Time | VARCHAR(50) | NN     |     |          | 예상 비행 시간  |         |
| 3        | 노선 분류    | Route Classification | VARCHAR(50) | NN     |     |          | 노선 분류     | 국제선/국내선 |
| 4        | 항공사 수    | Number of Airlines   | INT         | NN     |     |          | 노선 식별 번호  |         |
| 5        | 출발 공항    | Departure Airport    | VARCHAR(50) | NN     |     | FK       | 항공권 출발 시각 |         |
| 6        | 도착 공항    | Arrive Airport       | VARCHAR(50) | NN     |     | FK       | 항공권 도착 시각 |         |



### 3. Airline 테이블

| 테이블명(한글) |         | 항공사            | 테이블 설명      |      | 항공사 정보 |     |              |         |
|----------|---------|----------------|-------------|------|--------|-----|--------------|---------|
| 테이블명(영어) |         | Airline        |             |      |        |     |              |         |
| 순번       | 컬럼명(한글) | 컬럼명(영어)        | 자료형         | 널 여부 | 기본키    | 외래키 | 설명           | 비고      |
| 1        | 항공사명    | Airline Name   | VARCHAR(50) | NN   | PK     |     | 항공사 이름       |         |
| 2        | 항공사 코드  | Airline Code   | VARCHAR(50) | Y    |        |     | 항공사의 ICAO 코드 |         |
| 3        | 사업 모델   | Business Model | VARCHAR(50) | Y    |        |     | 항공사의 사업 모델   | LCC/FSC |
| 4        | 소속 국가   | Nation         | VARCHAR(50) | NN   |        | FK  | 항공사의 국적      |         |

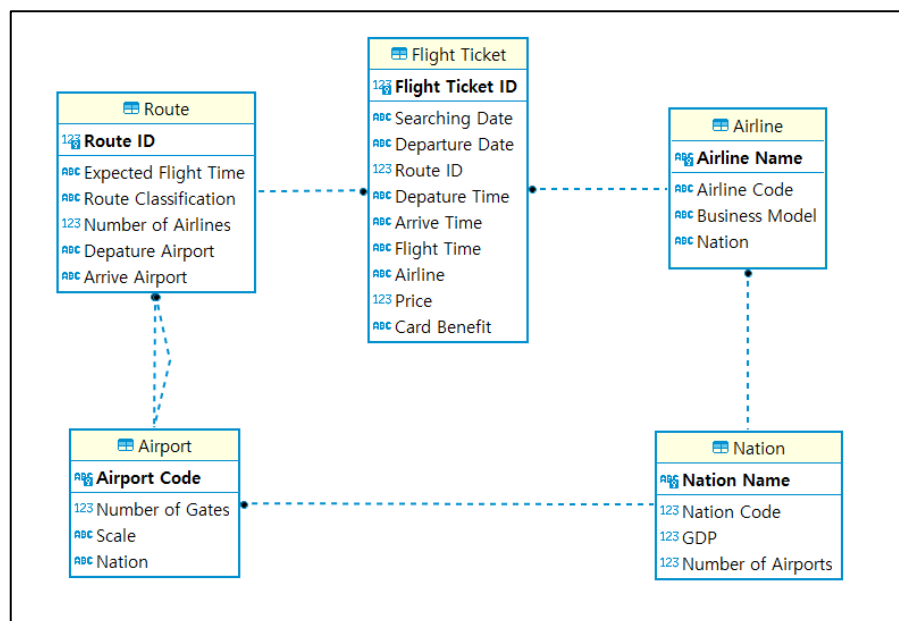
### 4. Airport 테이블

| 테이블명(한글) |         | 공항              | 테이블 설명      |      | 공항 정보 |     |             |              |
|----------|---------|-----------------|-------------|------|-------|-----|-------------|--------------|
| 테이블명(영어) |         | Airport         |             |      |       |     |             |              |
| 순번       | 컬럼명(한글) | 컬럼명(영어)         | 자료형         | 널 여부 | 기본키   | 외래키 | 설명          | 비고           |
| 1        | 공항 코드   | Airport Code    | VARCHAR(50) | NN   | PK    |     | 공항의 IATA 코드 |              |
| 2        | 게이트 수   | Number of Gates | INT         | Y    |       |     | 공항의 게이트 수   |              |
| 3        | 규모      | Scale           | VARCHAR(50) | Y    |       |     | 공항의 규모      | Large/medium |
| 4        | 소속 국가   | Nation          | VARCHAR(50) | NN   |       | FK  | 공항의 소속 국가   |              |

### 5. Nation 테이블

| 테이블명(한글) |         | 국가                 | 테이블 설명      |      | 국가 정보 테이블 |     |                |      |
|----------|---------|--------------------|-------------|------|-----------|-----|----------------|------|
| 테이블명(영어) |         | Nation             |             |      |           |     |                |      |
| 순번       | 컬럼명(한글) | 컬럼명(영어)            | 자료형         | 널 여부 | 기본키       | 외래키 | 설명             | 비고   |
| 1        | 국가명     | Nation Name        | VARCHAR(50) | NN   | PK        |     | 국가의 이름         | 한글표기 |
| 2        | 국가 코드   | Nation Code        | INT         | Y    |           |     | ITU-T 표준 국가 번호 |      |
| 3        | 국내총생산   | GDP                | INT         | Y    |           |     | 국가의 국내총생산      | 갱신없음 |
| 4        | 공항 수    | Number of Airports | INT         | Y    |           |     | 국가 내 공항 수      | 민간공항 |

## C. E-R 다이어그램



## VIII. 데이터 수집 및 확보 방안

### A. 데이터 수집 방법

항공권 조회 웹 사이트에서 1 일 1 회 지정된 18 개 노선의 항공권 정보를 최대 183 일 이후까지 크롤링을 통해 수집합니다. 하루 평균 수집 데이터의 개수는 약 52000 개이며, 평균 소요시간은 약 10 시간입니다. 소요시간이 매우 오래 걸리므로, 후술할 클라우드 컴퓨팅 방식을 이용하여 크롤링 서버를 구축하였습니다.

수집하는 정보는 조사 날짜, 출발 날짜, 노선 ID, 출발 시각, 도착 시각, 비행 시간, 항공사, 가격, 카드 정보이며 'Flight Ticket' 테이블의 속성과 일치합니다.

수집된 정보는 flights.csv 파일에 매일 누적되며, 날짜 별 로그파일을 통해 프로그램 이상을 파악합니다.

### B. 데이터 수집 소스 선정

항공권 조회 웹 사이트 한 군데에서, 크롤링을 통해 필요한 정보를 저장합니다. 여러 후보군 중 선정 기준에 따라 정해진 크롤링 대상 페이지는 '네이버 항공권'이며, 다음과 같은 기준에 의하여 선정하였습니다.

- 봇 탐지 프로세스가 존재하지 않는 사이트
- IP 차단이 발생할 경우 타 Proxy IP 사용으로 우회 가능한 사이트
- 또는 HTTP headers 의 user-agent 속성 값 조작 또는 지정으로 우회 가능한 사이트
- 크롤링 속도 향상을 위해 동기적으로 작동하는 웹사이트

'네이버 항공권' 페이지는 봇 탐지 프로세스가 존재하지 않아 별다른 우회 조치가 필요 없고, 동기적으로 작동하는 항공권 조회 사이트는 존재하지 않으므로 후술할 방식으로 크롤링 속도를 향상하였습니다.

### C. 개발 환경 구축 - 크롤링 서버

아마존에서 제공하는 클라우드 컴퓨팅 시스템인 AWS EC2 를 사용하였습니다.

- 운영체제 : Ubuntu Server 22.04 LTS (HVM), SSD Volume Type
- 인스턴스 : t2.micro
- vCPU : 1, 메모리 : 1GiB, 네트워크 : 1Gbps, IMDSv2 : Optional
- FTP : FileZila - EC2 인스턴스 코드 배포 및 데이터 회수
- SSH Client : XShell

## D. 프로세스 자동화 및 스케줄링

비동기 크롤링의 특성상 많은 시간이 소요되므로, 이를 해결하기 위해 다음과 같은 시도를 하였습니다.

- ssh 세션 종료 이후 프로세스를 백그라운드로 실행 - nohup POSIX 명령어 사용
- selenium webdriver 멀티스레딩 미지원 이슈 - client level 멀티스레딩
- Unix Job Scheduler - crontab 을 사용한 크롤링 스크립트 자동화
- 파싱 라이브러리 혼용 - bs4, selenium

## E. 데이터 검증 및 자체 평가

1. 신뢰할 수 있는 데이터인가?

크롤링을 통해 실제 가격 정보를 가져오므로 신뢰할 수 있습니다.

2. 지속적으로 확보할 수 있는 데이터인가?

자동화 프로세스 구축으로 매일 데이터를 수집하여 갱신합니다.

3. 데이터의 양은 충분한가?

한 번의 스크립트로 수집되는 데이터의 양은 약 52000 개이며, 최대 183 일 이후까지 정보를 수집하므로 활용할 수 있는 데이터는 최대 950 만개입니다. 이는 18 개의 노선에 대한 정보만을 포함하므로, 사용자가 노선을 추가할 경우 더 많은 데이터를 수집 할 수 있습니다.

## IX. 물리적 설계 및 GUI 구현

### A. 테이블 생성

1. Flight Ticket 테이블

```
CREATE TABLE public."Flight Ticket" (  
  "Flight Ticket ID" int4 NOT NULL,  
  "Searching Date" varchar(50) NOT NULL,  
  "Departure Date" varchar(50) NOT NULL,  
  "Route ID" int4 NOT NULL,  
  "Departure Time" varchar(50) NOT NULL,  
  "Arrive Time" varchar(50) NOT NULL,  
  "Flight Time" varchar(50) NOT NULL,  
  Airline varchar(50) NOT NULL,  
  Price int4 NOT NULL,  
  "Card Benefit" varchar(50) NULL,  
  CONSTRAINT flight_ticket_pk PRIMARY KEY ("Flight Ticket ID"),  
  CONSTRAINT flight_ticket_information_fk FOREIGN KEY (Airline)  
    REFERENCES public."Airline"("Airline Name"),  
  CONSTRAINT flight_ticket_information_route_fk FOREIGN KEY ("Route ID")  
    REFERENCES public."Route"("Route ID")  
);
```

## 2. Route 테이블

```
CREATE TABLE public."Route" (  
  "Route ID" int4 NOT NULL,  
  "Expected Flight Time" varchar(50) NULL,  
  "Route Classification" varchar(50) NULL,  
  "Number of Airlines" int4 NULL,  
  "Depature Airport" varchar(50) NOT NULL,  
  "Arrive Airport" varchar(50) NOT NULL,  
  CONSTRAINT route_pk PRIMARY KEY ("Route ID"),  
  CONSTRAINT route_fk_arrive FOREIGN KEY ("Arrive Airport")  
    REFERENCES public."Airport"("Airport Code"),  
  CONSTRAINT route_fk_departure FOREIGN KEY ("Depature Airport")  
    REFERENCES public."Airport"("Airport Code")  
);
```

## 3. Airline 테이블

```
CREATE TABLE public."Airline" (  
  "Airline Name" varchar(50) NOT NULL,  
  "Airline Code" varchar(50) NULL,  
  "Business Model" varchar(50) NULL,  
  "Nation" varchar(50) NOT NULL,  
  CONSTRAINT airline_pk PRIMARY KEY ("Airline Name"),  
  CONSTRAINT airline_fk FOREIGN KEY ("Nation") REFERENCES public."Nation"("Nation Name")  
);
```

## 4. Airport 테이블

```
CREATE TABLE public."Airport" (  
  "Airport Code" varchar(50) NOT NULL,  
  "Number of Gates" int4 NULL,  
  "Scale" varchar(50) NULL,  
  "Nation" varchar(50) NOT NULL,  
  CONSTRAINT airport_pk PRIMARY KEY ("Airport Code"),  
  CONSTRAINT airport_fk FOREIGN KEY ("Nation") REFERENCES public."Nation"("Nation Name")  
);
```

## 5. Nation 테이블

```
CREATE TABLE public."Nation" (  
  "Nation Name" varchar(50) NOT NULL,  
  "Nation Code" int4 NULL,  
  "GDP" int4 NULL,  
  "Number of Airports" int4 NULL,  
  CONSTRAINT nation_pk PRIMARY KEY ("Nation Name")  
);
```

## B. 기능 별 SQL 문 출력 예시

### 1. 노선 별 항공권 가격 변동 내역 조회

```
SELECT *
FROM "Flight Ticket" ft
      JOIN "Route" r ON r."Route ID" =ft."Route ID"
WHERE r."Depature Airport" ='ICN' AND r."Arrive Airport" ='DAD'
ORDER BY ft."Price"
```

| Flight Ticket ID | nsc Searching Date | nsc Departure Date | 123 Route ID | nsc Departure Time | nsc Arrive Time | nsc Flight Time | nsc Airline | 123 Price |
|------------------|--------------------|--------------------|--------------|--------------------|-----------------|-----------------|-------------|-----------|
| 453,154          | 2023-06-04         | 2023-06-26         | 10           | 22:45              | 01:25           | 04:40           | 비엠허항공       | 80,141    |
| 242,460          | 2023-05-31         | 2023-06-04         | 10           | 22:45              | 01:25           | 04:40           | 비엠허항공       | 83,293    |
| 611,576          | 2023-06-07         | 2023-06-13         | 10           | 20:25              | 23:30           | 05:05           | 에어서울        | 94,100    |
| 611,526          | 2023-06-07         | 2023-06-10         | 10           | 20:25              | 23:30           | 05:05           | 에어서울        | 94,100    |
| 611,593          | 2023-06-07         | 2023-06-14         | 10           | 20:25              | 23:30           | 05:05           | 에어서울        | 94,100    |
| 611,541          | 2023-06-07         | 2023-06-11         | 10           | 20:25              | 23:30           | 05:05           | 에어서울        | 94,100    |
| 664,625          | 2023-06-08         | 2023-06-14         | 10           | 20:25              | 23:30           | 05:05           | 에어서울        | 94,100    |
| 558,928          | 2023-06-06         | 2023-07-09         | 10           | 21:55              | 00:30           | 04:35           | 제주항공        | 102,200   |
| 558,947          | 2023-06-06         | 2023-07-10         | 10           | 21:20              | 00:30           | 05:10           | 제주항공        | 102,200   |
| 453,181          | 2023-06-04         | 2023-06-27         | 10           | 21:55              | 00:30           | 04:35           | 제주항공        | 102,400   |

### 2. 항공권 별 가격 변동 내역 조회

```
SELECT *
FROM "Flight Ticket" ft
      JOIN "Route" r ON r."Route ID" =ft."Route ID"
WHERE r."Depature Airport" ='ICN' AND r."Arrive Airport" ='DAD' AND ft."Airline" ='진에어'
      AND ft."Departure Date" ='2023-08-01' AND ft."Departure Time" ='17:15'
ORDER BY ft."Price"
```

| Flight Ticket ID | nsc Searching Date | nsc Departure Date | 123 Route ID | nsc Departure Time | nsc Arrive Time | nsc Flight Time | nsc Airline | 123 Price |
|------------------|--------------------|--------------------|--------------|--------------------|-----------------|-----------------|-------------|-----------|
| 294,832          | 2023-06-01         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 341,900   |
| 191,856          | 2023-05-30         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 347,300   |
| 243,402          | 2023-05-31         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 347,300   |
| 140,663          | 2023-05-29         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 347,400   |
| 34,773           | 2023-05-27         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 347,400   |
| 89,410           | 2023-05-28         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 347,400   |
| 769,886          | 2023-06-10         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 368,300   |
| 665,447          | 2023-06-08         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 368,300   |
| 559,333          | 2023-06-06         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 368,400   |
| 612,412          | 2023-06-07         | 2023-08-01         | 10           | 17:15              | 20:00           | 04:45           | 진에어         | 368,400   |

### 3. 국가 별 항공권 가격 변동 내역 조회

```
SELECT *
FROM "Flight Ticket" ft
      JOIN "Route" r ON r."Route ID" =ft."Route ID"
      JOIN "Airport" a ON r."Arrive Airport" =a."Airport Code"
WHERE a."Nation" ='베트남'
ORDER BY ft."Price"
```

| Flight Ticket ID | nsc Searching Date | nsc Departure Date | 123 Route ID | nsc Departure Time | nsc Arrive Time | nsc Flight Time | nsc Airline | 123 Price |
|------------------|--------------------|--------------------|--------------|--------------------|-----------------|-----------------|-------------|-----------|
| 453,154          | 2023-06-04         | 2023-06-26         | 10           | 22:45              | 01:25           | 04:40           | 비엠허항공       | 80,141    |
| 300,374          | 2023-06-01         | 2023-09-11         | 12           | 06:25              | 08:55           | 04:30           | 비엠허항공       | 82,230    |
| 242,460          | 2023-05-31         | 2023-06-04         | 10           | 22:45              | 01:25           | 04:40           | 비엠허항공       | 83,293    |
| 249,099          | 2023-05-31         | 2023-09-15         | 12           | 06:25              | 08:55           | 04:30           | 비엠허항공       | 84,890    |
| 669,871          | 2023-06-08         | 2023-06-12         | 12           | 21:10              | 23:45           | 04:35           | 제주항공        | 88,100    |
| 774,587          | 2023-06-10         | 2023-07-08         | 12           | 21:10              | 23:45           | 04:35           | 제주항공        | 88,100    |
| 669,952          | 2023-06-08         | 2023-06-18         | 12           | 21:10              | 23:45           | 04:35           | 제주항공        | 88,100    |
| 670,049          | 2023-06-08         | 2023-06-25         | 12           | 21:10              | 23:45           | 04:35           | 제주항공        | 88,100    |
| 774,341          | 2023-06-10         | 2023-06-20         | 12           | 21:10              | 23:45           | 04:35           | 제주항공        | 88,100    |
| 669,966          | 2023-06-08         | 2023-06-19         | 12           | 21:10              | 23:45           | 04:35           | 제주항공        | 88,100    |

## C. GUI 기능 정의 및 결과 화면

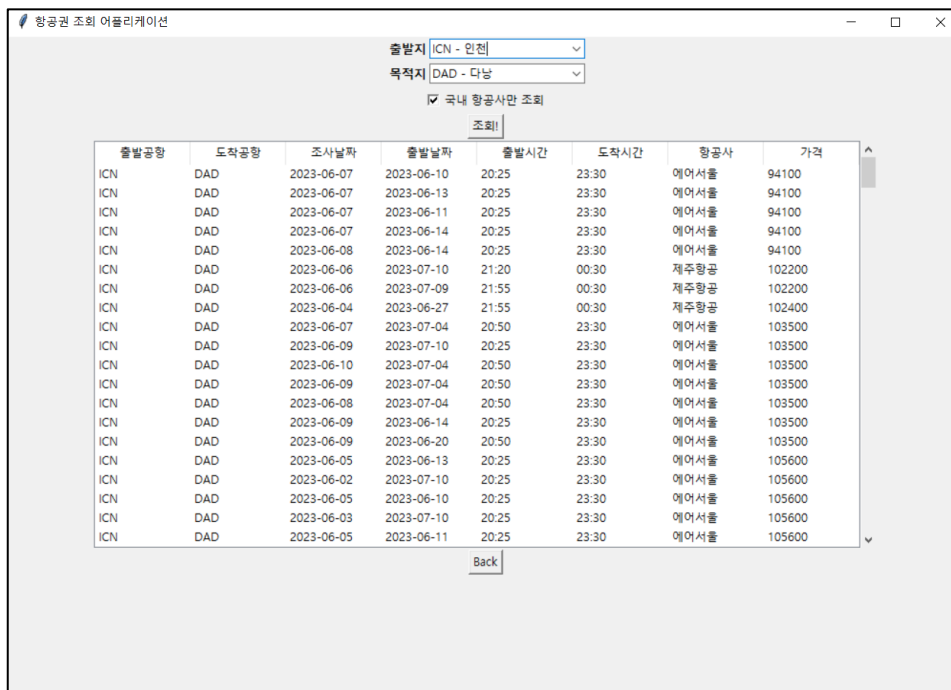
### 1. 홈 화면

항공권 조회 프로그램 처음 실행 시 나오는 메인 화면입니다. 3 개의 자주 사용되는 기능과 사용자가 검색 조건을 설정할 수 있는 기능까지 총 4 개의 버튼으로 이루어집니다.



### 2. 노선 별 항공권 가격 변동 내역 조회 화면

노선 별로 항공권의 가격 변동 내역을 조회할 수 있는 화면입니다.



### 3. 항공권 별 가격 변동 내역 조회 화면

동일한 항공권의 가격 변동 내역을 조회할 수 있는 화면입니다.

항공권 조회 어플리케이션

출발지 ICN - 인천  
목적지 DAD - 다낭  
출발 날짜 2023-08-01  
항공사 이름 진에어  
출발 시각 17:15

조회

| 출발공항 | 도착공항 | 조사날짜       | 출발날짜       | 출발시간  | 도착시간  | 항공사 | 가격     |
|------|------|------------|------------|-------|-------|-----|--------|
| ICN  | DAD  | 2023-05-30 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 347300 |
| ICN  | DAD  | 2023-05-31 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 347300 |
| ICN  | DAD  | 2023-06-01 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 341900 |
| ICN  | DAD  | 2023-06-02 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 374400 |
| ICN  | DAD  | 2023-06-03 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 372200 |
| ICN  | DAD  | 2023-06-04 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 372200 |
| ICN  | DAD  | 2023-06-05 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 372200 |
| ICN  | DAD  | 2023-06-06 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368400 |
| ICN  | DAD  | 2023-06-07 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368400 |
| ICN  | DAD  | 2023-06-08 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368300 |
| ICN  | DAD  | 2023-06-09 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368400 |
| ICN  | DAD  | 2023-06-10 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368300 |
| ICN  | DAD  | 2023-06-11 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368300 |
| ICN  | DAD  | 2023-06-12 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368300 |
| ICN  | DAD  | 2023-06-13 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368000 |
| ICN  | DAD  | 2023-06-14 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368000 |
| ICN  | DAD  | 2023-06-15 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 370600 |
| ICN  | DAD  | 2023-06-16 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 372400 |
| ICN  | DAD  | 2023-06-17 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 380600 |
| ICN  | DAD  | 2023-06-18 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 380600 |

Back

### 4. 국가 별 항공권 가격 변동 조회 화면

특정 국가 행 항공권의 가격 변동 내역을 조회할 수 있는 화면입니다.

항공권 조회 어플리케이션

여행 국가 미국

조회

| 출발공항 | 도착공항 | 조사날짜       | 출발날짜       | 출발시간  | 도착시간  | 항공사      | 가격     |
|------|------|------------|------------|-------|-------|----------|--------|
| ICN  | HKG  | 2023-06-09 | 2023-08-28 | 13:50 | 16:40 | 그레이터베이항공 | 115100 |
| ICN  | HKG  | 2023-06-07 | 2023-08-31 | 13:50 | 16:40 | 그레이터베이항공 | 115100 |
| ICN  | HKG  | 2023-06-09 | 2023-09-25 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-05 | 2023-08-31 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-02 | 2023-08-22 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-02 | 2023-07-06 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-09 | 2023-08-31 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-03 | 2023-09-19 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-09 | 2023-09-12 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-02 | 2023-09-21 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-08 | 2023-07-13 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-06 | 2023-09-25 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-10 | 2023-09-14 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-04 | 2023-08-22 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-04 | 2023-09-19 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-05 | 2023-08-24 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-07 | 2023-09-07 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-02 | 2023-08-31 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-06 | 2023-09-07 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |
| ICN  | HKG  | 2023-06-06 | 2023-09-05 | 13:50 | 16:40 | 그레이터베이항공 | 116600 |

Back

## 5. 사용자 지정 항공권 가격 변동 조회 화면

사용자가 모든 검색 조건을 지정할 수 있습니다. 출발지, 목적지, 출발 날짜, 항공사, 출발시각 등의 조건을 비워 두거나 입력하여 검색 조건으로 활용할 수 있습니다. 또한 정렬 순서와 정렬 방식을 설정할 수 있습니다.

| 출발공항 | 도착공항 | 조사날짜       | 출발날짜       | 출발시간  | 도착시간  | 항공사 | 가격     |
|------|------|------------|------------|-------|-------|-----|--------|
| ICN  | DAD  | 2023-06-02 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 374400 |
| ICN  | DAD  | 2023-06-06 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 368400 |
| ICN  | DAD  | 2023-05-31 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 380700 |
| ICN  | DAD  | 2023-06-07 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 368400 |
| ICN  | DAD  | 2023-06-03 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 372200 |
| ICN  | DAD  | 2023-06-08 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 368300 |
| ICN  | DAD  | 2023-05-30 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 380700 |
| ICN  | DAD  | 2023-06-09 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 368400 |
| ICN  | DAD  | 2023-06-04 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 372200 |
| ICN  | DAD  | 2023-06-01 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 375300 |
| ICN  | DAD  | 2023-05-27 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 380800 |
| ICN  | DAD  | 2023-06-05 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 372200 |
| ICN  | DAD  | 2023-05-29 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 380800 |
| ICN  | DAD  | 2023-06-10 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 368300 |
| ICN  | DAD  | 2023-05-28 | 2023-08-01 | 20:50 | 23:30 | 진에어 | 380800 |
| ICN  | DAD  | 2023-06-10 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 368300 |
| ICN  | DAD  | 2023-05-28 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 347400 |
| ICN  | DAD  | 2023-06-05 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 372200 |
| ICN  | DAD  | 2023-05-31 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 347300 |
| ICN  | DAD  | 2023-06-01 | 2023-08-01 | 17:15 | 20:00 | 진에어 | 341900 |

## X. 향후 개발 계획

### A. 데이터 수집 프로세스 완전 자동화

현재 데이터를 수집하는 크롤링 프로세스는 자동화가 이루어졌습니다. 하지만 자동화를 위한 명령어를 입력하는 과정이나, 수집된 파일을 가져오는 과정은 FTP를 이용하여 직접 가져와야 합니다. 이러한 과정들 전부 자동화하여 사용자의 편의성을 극대화할 예정입니다.

### B. UI를 통한 노선 정보 및 수집 정보 변경

현재 노선 정보를 변경하거나 수집 정보를 변경하려면 코드를 수정해야 합니다. 이러한 과정 또한 UI를 통해 접근할 수 있게 구현함으로써 역시 사용자의 편의성을 극대화할 것입니다.

### C. 웹앱 및 모바일 어플리케이션 구축을 통한 온라인 접속 및 실생활 적용

현재 제공하는 GUI는 파이썬의 tkinter 라이브러리를 사용한 스크립트 파일입니다. 해당 스크립트를 실행파일로 만들 뿐 아니라, 크롤링 및 백엔드 서버를 구축하여 사용자가 온라인으로 접속하여 활용할 수 있도록 적용 범위를 늘릴 예정입니다.

### D. 시각자료 제공 등 UX 개선

사용자가 정보를 한눈에 파악하기 쉽도록 가격 변동 그래프 등의 시각자료를 제공함으로써 GUI 제공의 목적을 살릴 예정입니다

프로젝트에 사용된 소스코드 및 개발 환경은 <https://github.com/j2noo/DatabaseTermProject> 에서 확인하실 수 있습니다.