

2025년 상반기 K-디지털 트레이닝

JDBC 프로그래밍 - Travel

[KB] IT's Your Life



🗸 프로젝트 개요

○ 관광지 정보를 담고 있는 travel.csv 파일을 데이터베이스로 임포트

	Α	В	С	D	Е	F	G	Н
1	no	district	title	description	address	phone		
2	1	수도권	인천 개항장역사문	1883년 인천 개항	인천광역시 중구	인천 중구 문화유신	팀 032-760)-6448
3	2	수도권	인천 월미문화의거	인천의 대표적인	인천광역시 중구	032-764-0842		
4	3	수도권	인천 연안부두해양	서해 도서로 운항	인천광역시 중구	032-777-1330		
5	4	수도권	인천 하나개해수욕	하나개해수욕장은	인천광역시 중구	032-751-8833		
6	5	수도권	양평 세미원	세미원 연꽃박물관	경기도 양평군 양	031-775-1835		
7	6	수도권	시흥 갯골생태공원	시흥갯골생태공원	경기도 시흥시 동	031-488-6900		

○ 각 관광지의 이미지 파일 정보를 데이터베이스로 임포트

■ travel-image 폴더에 준비

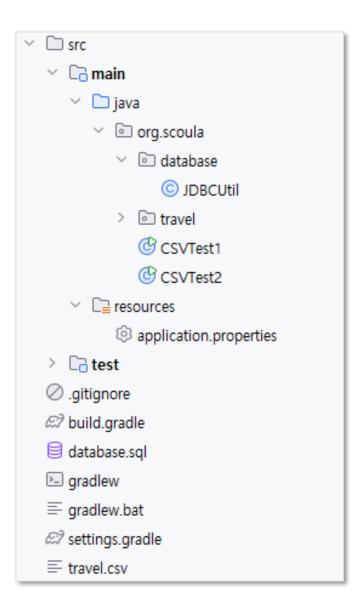
- 파일명: 관광지no-순번.jpg

- 예: 007-4.jpg

ਂ 프로젝트 만들기

o 프로젝트: travelapp

o 빌드 툴: gradle



build.gradle

```
dependencies {
   implementation 'com.opencsv:opencsv:5.5'

implementation 'com.mysql:mysql-connector-j:8.3.0'

compileOnly 'org.projectlombok:lombok:1.18.30'
   annotationProcessor 'org.projectlombok:lombok:1.18.30'

testCompileOnly 'org.projectlombok:lombok:1.18.30'

testAnnotationProcessor 'org.projectlombok:lombok:1.18.30'

testImplementation platform('org.junit:junit-bom:5.10.0')
   testImplementation 'org.junit.jupiter:junit-jupiter'
}
```

O opencsv

■ 자바로 CSV 파일을 다루는 라이브러리

resources::/application.properties

driver=com.mysql.cj.jdbc.Driver
url=jdbc:mysql://127.0.0.1:3306/jdbc_ex
id=scoula
password=1234

CSV 파일 다루기

- o CSVReader 클래스
 - 생성자에 CSV 파일 리더 지정
 - 예: new CSVReader(new FileReader("travel.csv"))
- String[] readNext()
 - ','을 기준으로 끊어서 String 배열로 리턴

org.scoula.CSVTest1.java

```
public class CSVTest1 {
   public static void main(String[] args) throws Exception {
        CSVReader csvReader = new CSVReader(new FileReader("travel.csv"));
        String[] line;
        while ((line = csvReader.readNext()) != null) {
              System.out.println(String.join(",", line));
        }
   }
}
```

```
org.scoula
database
travel
CSVTest1
CSVTest2
```

1,수도권,인천 개항장역사문화공간,1883년 인천 개항장 126년 세월의 흔적이 쌓인 개항기 역사와 문화의 만남의 장소이다. 이곳 2,수도권,인천 월미문화의거리,인천의 대표적인 관광지 중의 하나인 월미도에는 월미도 문화의 거리, 월미공원, 월미테마파크 등 3,수도권,인천 연안부두해양광장,서해 도서로 운항하는 여객선의 출발지이다. 싱싱한 생선과 젓갈 등을 저렴한 가격으로 구입할 4,수도권,인천 하나개해수욕장,하나개해수욕장은 샘꾸미 선착장에서 차로 10분이면 도착할 수 있다. 무의도에서 가장 큰 갯벌이 5,수도권,양평 세미원,세미원 연꽃박물관은 연꽃이라는 단일한 테마 아래 연꽃관련 생활 용품, 고서, 음식 등의 유물이 전시된 6,수도권,시흥 갯골생태공원,시흥갯골생태공원은 경기도 유일의 내만갯벌과 옛염전의 정취를 느낄 수 있는 아름다운 공원이다. 0 7,수도권,고양 행주산성,덕양산 정상에 축조된 행주산성(사적)은 임진왜란 당시 의병과 승명을 포함한 2천3백명으로 왜군 3만 0

🥑 CSV 파일 다루기

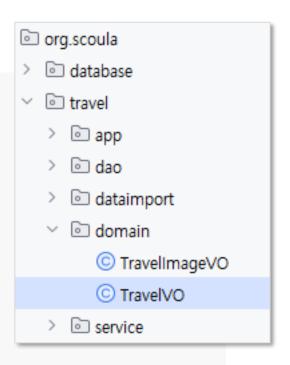
- o CsvToBeanBuilder<T> 클래스
 - csv 파일의 한 라인을 읽어 지정한 T 타입의 인스턴스로 변환해주는 클래스
 - csv 파일의 헤더(첫 번째 줄) 명과 같은 T 클래스의 필드로 매핑

클래스로 변환하기

```
List<T> t = new CsvToBeanBuilder<T>(new FileReader("csv 파일 경로"))
.withType(T.class)
.build()
.parse();
```

org.scoula.travel.domain.TravelVO.java

```
@Data
@Builder
@NoArgsConstructor
@AllArgsConstructor
public class TravelVO {
    private Long no;
    private String district;
    private String title;
    private String description;
    private String address;
    private String phone;
}
```



O csv 파일의 첫 번째 줄에 있는 각 헤더명과 동일하게 필드명 정의

no, district, title, description, address, phone

org.scoula.CSVTest2.java

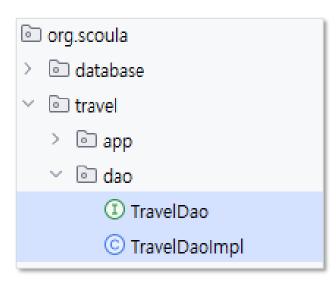
```
org.scoula

oditabase

travel
CSVTest1
```

TravelVO(no=1, district=수도권, title=인천 개항장역사문화공간, description=1883년 인천 개항장 126년 세월의 흔적이 쌓인 개항기 역사와 문화의 만남의 장소이다. 0 TravelVO(no=2, district=수도권, title=인천 월미문화의거리, description=인천의 대표적인 관광지 중의 하나인 월미도에는 월미도 문화의 거리, 월미공원, 월미테마파크 TravelVO(no=3, district=수도권, title=인천 연안부두해양광장, description=처해 도서로 운항하는 여객선의 출발지이다. 싱싱한 생선과 젓갈 등을 저렴한 가격으로 구입할 TravelVO(no=4, district=수도권, title=인천 하나개해수욕장, description=하나개해수욕장은 샘꾸미 선착장에서 차로 10분이면 도착할 수 있다. 무의도에서 가장 큰 갯벌인 TravelVO(no=5, district=수도권, title=양평 세미원, description=세미원 연꽃박물관은 연꽃이라는 단일한 테마 아래 연꽃관련 생활 용품, 고서, 음식 등의 유물이 전시된 TravelVO(no=6, district=수도권, title=시흥 갯골생태공원, description=시흥갯골생태공원은 경기도 유일의 내만갯벌과 옛염전의 정취를 느낄 수 있는 아름다운 공원이다. TravelVO(no=7, district=수도권, title=고양 행주산성, description=덕양산 정상에 축조된 행주산성(사적)은 임진왜란 당시 의병과 승명을 포함한 2천3백명으로 왜군 3만 TravelVO(no=8, district=수도권, title=고양 행주산성역사공원, description=한강을 직접 만져볼 수 있는 친수공간(빨랫돌머리) 군초소 전망대(행호정), 철책 포토존, 바

TravelDao 준비하기

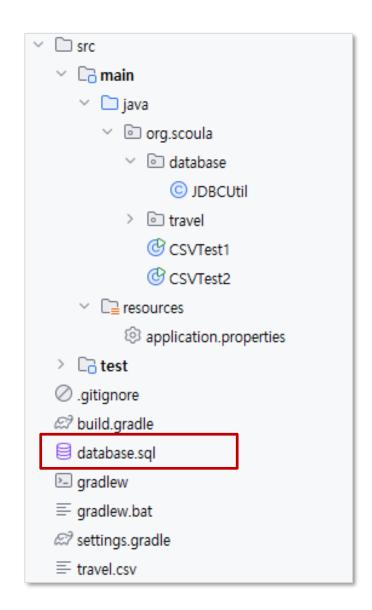


```
package org.scoula.travel.dao;
import org.scoula.travel.domain.TravelVO;
public interface TravelDao {
    void insert(TravelVO travel);
}
```

```
package org.scoula.travel.dao;
public class TravelDaoImpl implements TravelDao {
    Connection conn = JDBCUtil.getConnection();
    @Override
    public void insert(TravelVO travel) {
        String sql = "insert into tbl_travel(no, district, title, description, address, phone) values(?,?,?,?,?)";
        try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
            pstmt.setLong(1, travel.getNo());
            pstmt.setString(2, travel.getDistrict());
            pstmt.setString(3, travel.getTitle());
            pstmt.setString(4, travel.getDescription());
            pstmt.setString(5, travel.getAddress());
            pstmt.setString(6, travel.getPhone());
            int count = pstmt.executeUpdate();
        } catch (SQLException e) {
            throw new RuntimeException(e);
```

🥝 travel.csv 파일을 데이터베이스로 임포트 하기

○ 테이블 준비



org.scoula.travel.dataimport.ImportTravelData.java

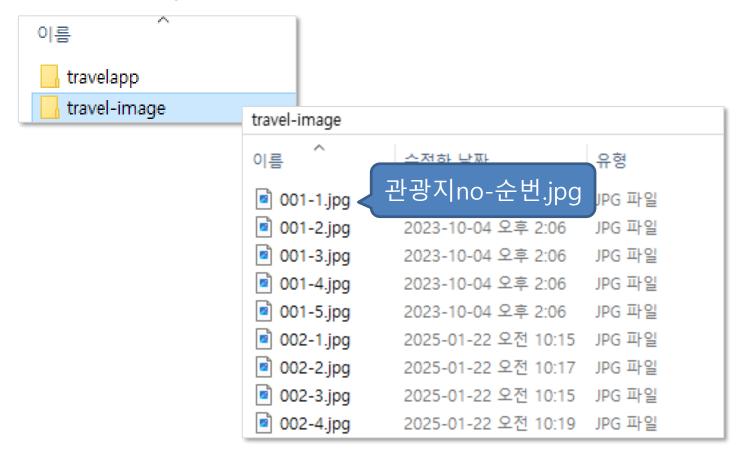
```
public class ImportTravelData {
    public static void main(String[] args) throws CsvValidationException, IOException {
        TravelDao dao = new TravelDaoImpl();
        List<TravelVO> travels = new CsvToBeanBuilder<TravelVO>(new FileReader("travel.csv"))
                .withType(TravelV0.class)
                .build()
                .parse();
        travels.forEach(travel -> {
            System.out.println(travel);
            dao.insert(travel);
        });
        JDBCUtil.close();
```

org.scoula

org.sc

😕 각 관광지의 이미지 파일 정보를 데이터베이스로 임포트

- o travel-image 폴더 준비
 - 프로젝트와 동일한 레벨에 준비



tbl_travel_image 테이블

org.scoula.travel.dao.TravellmageVO.java

```
@Data
@Builder
@NoArgsConstructor
@AllArgsConstructor
public class TravelImageVO {
    private Long no;
    private String filename;
    private Long travelNo;
}
```

org.scoula

org.sc

```
package org.scoula.travel.dao;
import org.scoula.travel.domain.TravelVO;
public interface TravelDao {
    void insert(TravelVO travel);

    void insertImage(TravelImageVO image);
}
```

```
package org.scoula.travel.dao;
public class TravelDaoImpl implements TravelDao {
    Connection conn = JDBCUtil.getConnection();
    @Override
    public void insertImage(TravelImageV0 image) {
        String sql = "insert into tbl_travel_image(filename, travel_no) values(?,?)";
        try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
            pstmt.setString(1, image.getFilename());
            pstmt.setLong(2, image.getTravelNo());
            int count = pstmt.executeUpdate();
        } catch (SQLException e) {
            throw new RuntimeException(e);
```

org.scoula.travel.dataimport.ImportImageData.java

```
public class ImportImageData {
    public static void main(String[] args) {
        TravelDao dao = new TravelDaoImpl();
        File dir = new File("../travel-image");
       File[] files = dir.listFiles();
       for (File file : files) {
           String filename = file.getName();
            long travelNo = Long.parseLong(filename.split("-")[0]); // 관광지 no 얻기
            TravelImageV0 image = TravelImageV0.builder()
                    .filename(filename)
                    .travelNo(travelNo)
                    .build();
            System.out.println(image);
            dao.insertImage(image);
        JDBCUtil.close();
```

org.scoula

org.sc

TravelDao에 기능 추가하기

- 전체 데이터 건수 얻기
- 권역 목록 얻기
- 전체 목록 얻기
- 페이지별 목록 얻기
- 특정 권역의 목록 얻기
- 관광지 정보 하나 얻기
 - Join을 이용해서 해당 이미지 정보를 같이 추출하기
 - Optional 로 리턴하기

org.scoula.travel.dao.TravelDao.java

```
package org.scoula.travel.dao;
import org.scoula.travel.domain.TravelImageVO;
import org.scoula.travel.domain.TravelVO;
import java.util.List;
import java.util.Optional;
public interface TravelDao {
   void insert(TravelVO travel);
   void insertImage(TravelImageV0 image);
   int getTotalCount();
   List<String> getDistricts();
                               // 권역 목록 얻기
   List<TravelVO> getTravels(); // 목록 얻기
   List<TravelVO> getTravels(int page); // 페이지별 목록 얻기
   List<TravelVO> getTravels(String district); // 해당 권역의 목록 얻기
   Optional<TravelVO> getTravel(Long no); // 특정 관광지 정보 얻기
```

org.scoula

org.sc

org.scoula.travel.dao.TravelDaoImpl.java

```
public class TravelDaoImpl implements TravelDao {
    Connection conn = JDBCUtil.getConnection();
    @Override
    public int getTotalCount() {
        String sql = "select count(*) from tbl_travel";
        try (PreparedStatement pstmt = conn.prepareStatement(sql);
             ResultSet rs = pstmt.executeQuery()) {
            rs.next();
            return rs.getInt(1);
        } catch (SQLException e) {
           throw new RuntimeException(e);
```

org.scoula

database

travel

app

dao

TravelDao

TravelDaolmpl

dataimport

domain

service

```
// 권역 목록 얻기
@Override
public List<String> getDistricts() {
    List<String> districts = new ArrayList<>();
    String sql = "SELECT DISTINCT(district) FROM tbl_travel ORDER BY district";
    try (PreparedStatement pstmt = conn.prepareStatement(sql);
         ResultSet rs = pstmt.executeQuery()) {
       while (rs.next()) {
            districts.add(rs.getString("district"));
    } catch (SQLException e) {
        throw new RuntimeException(e);
    return districts;
```

```
private TravelVO map(ResultSet rs) throws SQLException {
     return TravelVO.builder()
             .no(rs.getLong("no"))
             .district(rs.getString("district"))
             .title(rs.getString("title"))
             .description(rs.getString("description"))
             .address(rs.getString("address"))
             .phone(rs.getString("phone"))
             .build();
@Override
public List<TravelVO> getTravels() {
     List<TravelVO> travels = new ArrayList<>();
     String sql = "select * from tbl_travel order by district, title";
     try (PreparedStatement pstmt = conn.prepareStatement(sql);
          ResultSet rs = pstmt.executeQuery()) {
        while (rs.next()) {
             TravelV0 travel = map(rs);
             travels.add(travel);
     } catch (SQLException e) {
         throw new RuntimeException(e);
     return travels;
```

```
@Override
public List<TravelVO> getTravels(int page) {
    List<TravelVO> travels = new ArrayList<>();
    String sql = "select * from tbl_travel order by district, title limit ?,?";
    try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
        int count = 10;
       int start = (page - 1) * count;
        pstmt.setInt(1, start);
        pstmt.setInt(2, count);
        try (ResultSet rs = pstmt.executeQuery()) {
            while (rs.next()) {
                TravelVO travel = map(rs);
                travels.add(travel);
    } catch (SQLException e) {
        throw new RuntimeException(e);
    return travels;
```

```
@Override
public List<TravelVO> getTravels(String district) {
    List<TravelVO> travels = new ArrayList<TravelVO>();
    String sql = "select * from tbl_travel where district = ?";
    try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
        pstmt.setString(1, district);
        try (ResultSet rs = pstmt.executeQuery()) {
            while (rs.next()) {
                TravelVO travel = map(rs);
                travels.add(travel);
    } catch (SQLException e) {
        throw new RuntimeException(e);
    return travels;
```

💟 관광지 정보 하나 얻기

○ Join을 이용해서 해당 이미지 정보를 같이 추출하기

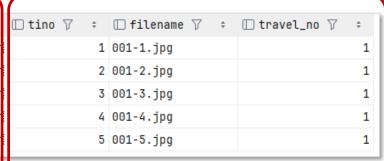
```
select t.*, ti.no as tino, ti.filename, ti.travel_no
from tbl_travel t
    left outer join tbl_travel_image ti
        on t.no = ti.travel_no
where t.no = 1;
```

	□ no ▽	÷	\square district $ abla$	□.	title 🎖 ‡	□ de	scription	n 7	‡	□ addres	s 7	‡	□ pl	none	₹ \$	□ tino ▽ ÷ □ filename ▽ ÷ □]travel_no ▽ ÷
1		1	수도권	인천	! 개항장역사문회	1883	크 인천 개형	항장 12	26년 세	인천광역시	중구	제물량로 <mark>232</mark>	인천	중구	문화유산	1 001-1.jpg	
2		1	수도권	인천	! 개항장역사문회	1883	크 인천 개형	항장 12	26년 세	인천광역시	중구	제물량로 <mark>232</mark>	인천	중구	문화유산	2 001-2.jpg	
3		1	수도권	인천	! 개항장역사문회	1883	크 인천 개형	항장 12	26년 세	인천광역시	중구	제물량로 <mark>232</mark>	인천	중구	문화유산	3 001-3.jpg	
4		1	수도권	인천	! 개항장역사문회	1883	크 인천 개형	함장 12	26년 세	인천광역시	중구	제물량로 <mark>232</mark>	인천	중구	문화유산	4 001-4.jpg	
5		1	수도권	인천	! 개항장역사문회	1883	크 인천 개형	항장 12	26년 세	인천광역시	중구	제물량로 <mark>232</mark>	인천	중구	문화유산	5 001-5.jpg	

TravelVO

List < Travellmage VO >





TravelVO

List < TravelImageVO >

```
TravelVO(
no=1,
district=수도권,
title=인천 개항장역사문화공간,
description=1883년 인천 개항장 126년 세월의 ...,
address=인천광역시 중구 제물량로232번안길 27,
phone=인천 중구 문화유산팀 032-760-6448,
images=[
TravellmageVO(no=1, filename=001-1.jpg, travelNo=1),
TravellmageVO(no=2, filename=001-2.jpg, travelNo=1),
TravellmageVO(no=4, filename=001-4.jpg, travelNo=1),
TravellmageVO(no=5, filename=001-5.jpg, travelNo=1),
TravellmageVO(no=5, filename=001-5.jpg, travelNo=1)]
```

org.scoula.travel.domain.TravelVO.java

```
@Data
@Builder
@NoArgsConstructor
@AllArgsConstructor
public class TravelVO {
    private Long no;
    private String district;
    private String title;
    private String description;
    private String address;
    private String phone;

private List<TravelImageVO> images;
}
```

```
private TravelImageV0 mapImage(ResultSet rs) throws SQLException {
    return TravelImageV0.builder()
            .no(rs.getLong("tino"))
            .filename(rs.getString("filename"))
            .travelNo(rs.getLong("travel_no"))
            .build();
@Override
public Optional<TravelVO> getTravel(Long no) {
    TravelV0 travel = null;
    String sql = """
                select t.*, ti.no as tino, ti.filename, ti.travel_no
                from tbl_travel t
                         left outer join tbl_travel_image ti
                                         on t.no = ti.travel no
                where t.no = ?;
           .....
```

```
try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
    pstmt.setLong(1, no);
   try (ResultSet rs = pstmt.executeQuery()) {
       if (rs.next()) {
           travel = map(rs);
           List<TravelImageV0> images = new ArrayList<>();
           try {
                do {
                   TravelImageVO image = mapImage(rs);
                   images.add(image);
               } while (rs.next());
           } catch (SQLException e) {
                // 이미지가 없는 경우 발생
           travel.setImages(images);
           return Optional.of(travel);
       } else {
           return Optional.empty();
} catch (SQLException e) {
    throw new RuntimeException(e);
```

💟 다음 기능들을 실행하는 서비스 객체 정의하기

- 전체 관광지 목록 출력하기
 - no, district, title만 출력
- 특정 권역의 관광지 목록 출력하기
 - 권역목록을 출력
 - 출력할 권역을 입력 받음
 - no, district, title만 출력
- 특정 페이지의 관광지 목록 출력하기
 - 전체 페이지를 출력
 - 출력할 페이지 번호를 입력 받음
 - no, district, title만 출력
- 관광지 상세 보기
 - 출력할 관광지 번호를 입력 받음
 - 이미지 정보를 포함한 모든 정보 출력하기

💟 전체 관광지 목록 출력하기

○ 권역명, 제목 순으로 정렬

- 101 강원권 강릉 경포해변
- 105 강원권 강릉 솔향수목원
- 100 강원권 강릉 연곡해변캠핑장
- 106 강원권 강릉 정동진 모래시계공원
- 103 강원권 강릉 커피거리
- 104 강원권 강릉 통일공원
- 102 강원권 강릉 허균?허난설헌 기념공원
- 107 강원권 동해 망상해수욕장
- 98 강원권 속초 아바이마을
- 99 강원권 속초 해수욕장관광지
- 108 강원권 정선 삼탄아트마인
- 95 강원권 춘천 남이섬

:

♥ 특정 권역의 관광지 목록 출력하기

- 1] 강원권
- 2] 경상권
- 3] 수도권
- 4] 전라권
- 5] 제주권
- 6] 충청권

선택: 5

- 109 제주권 제주 사려니 숲길삼나무숲
- 110 제주권 제주 서귀포 치유의숲
- 111 제주권 제주 붉은 오름 자연휴양림
- 112 제주권 제주 서귀포 천지연폭포

특정 페이지의 관광지 목록 출력하기

총 12 페이지

페이지 선택(1~12): 3

- 78 경상권 경주 보문관광단지
- 79 경상권 고령 대가야 역사테마파크
- 87 경상권 고성 당항포관광지
- 85 경상권 김해 가야테마파크
- 83 경상권 김해 낙동강 레일파크
- 84 경상권 김해 봉하마을
- 86 경상권 김해한옥체험관
- 73 경상권 대구 근대골목
- 75 경상권 대구 비슬산군립공원
- 74 경상권 대구 사문진주막촌

☑ 관광지 상세 보기

관광지 No: 1

권역: 수도권

제목: 인천 개항장역사문화공간

설명: 1883년 인천 개항장 126년 세월의 흔적이 쌓인 개항기 역사와 문화의 만남의 장소이다. 이곳은 옛 일본영사관인 중구청과 일본 제1은행 인천지점, 일본 제18은행 인천지점, 일본 제58은행 인천지점 등 근대 건축을 그대로 볼 수 있는 장소이다. 이뿐아니라 일본우선주식회사나 인천의 아트플렛폼, 제물포구락부 등 역사의 흔적들을 직접 눈으로 볼 수 있고 귀로 들을 수있는 곳이다. 이곳은 인천문화 관광해설사가 배치되어 예약을 받아 해설을 하고 있다. 보다 깊은 해설을 들으며 도보관광을 즐길 수 있어 가족 단위나 여행단체의 큰 호응을 얻고 있다.

주소: 인천광역시 중구 제물량로232번안길 27

전화번호: 인천 중구 문화유산팀 032-760-6448

사진들

[1] 001-1.jpg

[2] 001-2.jpg

[3] 001-3.jpg

[4] 001-4.jpg

[5] 001-5.jpg

org.scoula.travel.service.TravelService.java

```
package org.scoula.travel.service;

public interface TravelService {

    // 전체 관광지 목록 출력하기
    void printTravels();

    // 특정 권역의 관광지 목록 출력하기
    void printTravelsByDistrict();

    // 특정 페이지의 관광지 목록 출력하기
    void printTravelsByPage();

    // 광광지 상세보기
    void printTravel();
}
```

org.scoula

odatabase

travel
odao
odataimport
odomain
oservice
TravelServiceImpl

org.scoula

JDBC 프로그래밍 - Travel

org.scoula.travel.service.TravelServiceImpl.java

```
database
@RequiredArgsConstructor

∨ In travel

public class TravelServiceImpl implements TravelService {
    final TravelDao dao;
                                                                                                    > 🖭 app
                                                                                                    > 🖭 dao
    private int getNumber(String prompt) {
        System.out.print(prompt);
                                                                                                      dataimport
        Scanner sc = new Scanner(System.in);
                                                                                                    domain
        int num = Integer.parseInt(sc.nextLine());
        return num;
                                                                                                      service
                                                                                                         TravelService
    private void printDistricts(List<String> districts) {
                                                                                                         © TravelServiceImpl
        for (int i = 0; i < districts.size(); i++) {</pre>
            System.out.printf("%d] %s\n", i + 1, districts.get(i));
    private void printTravels(List<TravelVO> travels) {
        for (TravelVO travel : travels) {
            System.out.printf("%3d %6s %s\n", travel.getNo(), travel.getDistrict(), travel.getTitle());
```

org.scoula.travel.service.TravelServiceImpl.java

```
@Override
public void printTravels() {
   List<TravelVO> travels = dao.getTravels();
    printTravels(travels);
@Override
public void printTravelsByDistrict() {
   List<String> districts = dao.getDistricts();
    printDistricts(districts);
   int ix = getNumber("선택: ");
   String district = districts.get(ix - 1);
   List<TravelVO> travels = dao.getTravels(district);
    printTravels(travels);
@Override
public void printTravelsByPage() {
    int totalPage = (int) Math.ceil(dao.getTotalCount() / 10.0);
    System.out.printf("총 %d 페이지\n", totalPage);
    int page = getNumber(String.format("페이지 선택(1~%d): ", totalPage));
   List<TravelVO> travels = dao.getTravels(page);
    printTravels(travels);
```

org.scoula.travel.service.TravelServiceImpl.java

```
@Override
public void printTravel() {
    long no = getNumber("관광지 No: ");
    TravelVO travel = dao.getTravel(no).orElseThrow(NoSuchElementException::new);
    System.out.println("권역: " + travel.getDistrict());
    System.out.println("제목: " + travel.getTitle());
    System.out.println("설명: " + travel.getDescription());
    System.out.println("주소: " + travel.getAddress());
    System.out.println("주소: " + travel.getPhone());
    System.out.println("사진들");
    for (TravelImageVO image : travel.getImages()) {
        System.out.printf(" [%3d] %s\n", image.getNo(), image.getFilename());
    }
}
```

💟 어플리케이션 만들기

이 메뉴 운영

- 선택 메뉴 실행
 - 해당 메뉴에 해당하는 서비스 기능 호출
- 메뉴와 실행할 서비스 기능 간의 관계 관리 필요
 Menultem 클래스로 캡슐화

☑ 1) 전체목록

```
1) 전체목록 2) 페이지목록 3) 권역별목록 4) 상세보기 5) 종료
선택: 1
101 강원권 강릉 경포해변
105 강원권 강릉 솔향수목원
100 강원권 강릉 연곡해변캠핑장
106 강원권 강릉 정동진 모래시계공원
103 강원권 강릉 커피거리
104 강원권 강릉 통일공원
102 강원권 강릉 허균?허난설헌 기념공원
107 강원권 동해 망상해수욕장
98 강원권 속초 아바이마을
99 강원권 속초 해수욕장관광지
108 강원권 정선 삼탄아트마인
95 강원권 춘천 남이섬
```

☑ 2) 페이지목록

1) 전체목록 2) 페이지목록 3) 권역별목록 4) 상세보기 5) 종료 선택: 2 총 12 페이지 페이지 선택(1~12): 3 78 경상권 경주 보문관광단지 79 경상권 고령 대가야 역사테마파크 87 경상권 고성 당항포관광지 85 경상권 김해 가야테마파크 83 경상권 김해 낙동강 레일파크 84 경상권 김해 봉하마을 86 경상권 김해한옥체험관 73 경상권 대구 근대골목 75 경상권 대구 비슬산군립공원 74 경상권 대구 사문진주막촌

◎ 3) 권역별목록

1) 전체목록 2) 페이지목록 3) 권역별목록 4) 상세보기 5) 종료 선택: 3 1] 강원권 2] 경상권 3] 수도권 4] 전라권 5] 제주권 6] 충청권 선택: 5 109 제주권 제주 사려니 숲길삼나무숲 110 제주권 제주 서귀포 치유의숲 111 제주권 제주 붉은 오름 자연휴양림 112 제주권 제주 서귀포 천지연폭포

🗸 4) 상세보기

1) 전체목록 2) 페이지목록 3) 권역별목록 4) 상세보기 5) 종료

선택: 4

관광지 No: 1 권역: 수도권

제목: 인천 개항장역사문화공간

설명: 1883년 인천 개항장 126년 세월의 흔적이 쌓인 개항기 역사와 문화의 만남의 장소이다. 이곳은 옛 일본영사관인 중구청과 일본 제1은행인천지점, 일본 제18은행인천지점,일본 제58은행인천지점 등 근대 건축을 그대로 볼 수 있는 장소이다. 이뿐아니라 일본우선주식회사나인천의 아트플렛폼, 제물포구락부 등 역사의 흔적들을 직접 눈으로 볼 수 있고 귀로 들을 수있는 곳이다. 이곳은 인천문화 관광해설사가 배치되어 예약을 받아 해설을 하고 있다. 보다 깊은 해설을 들으며 도보관광을 즐길 수 있어 가족 단위나 여행단체의 큰 호응을 얻고 있다.

주소: 인천광역시 중구 제물량로232번안길 27

전화번호: 인천 중구 문화유산팀 032-760-6448

사진들

[1] 001-1.jpg

[2] 001-2.jpg

[3] 001-3.jpg

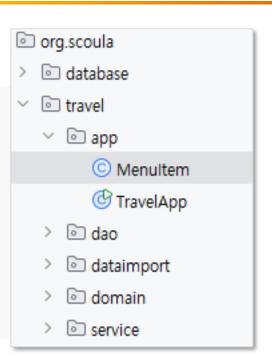
[4] 001-4.jpg

[5] 001-5.jpg

org.scoula.travel.app.MenuItem.java

```
package org.scoula.travel.app;
import lombok.AllArgsConstructor;
import lombok.Data;

@Data
@AllArgsConstructor
public class MenuItem {
    String title;
    Runnable command;
}
```



org.scoula.travel.app.TravelApp.java

```
package org.scoula.travel.app;
import org.scoula.database.JDBCUtil;
import org.scoula.travel.dao.TravelDao;
import org.scoula.travel.dao.TravelDaoImpl;
import org.scoula.travel.service.TravelService;
import org.scoula.travel.service.TravelServiceImpl;
import java.util.Scanner;
public class TravelApp {
    TravelService service;
    MenuItem[] menu;
    Scanner sc = new Scanner(System.in);
    public TravelApp() {
        TravelDao dao = new TravelDaoImpl();
        service = new TravelServiceImpl(dao);
       menu = new MenuItem[]{
               new MenuItem("전체목록", service::printTravels),
               new MenuItem("페이지목록", service::printTravelsByPage),
               new MenuItem("권역별목록", service::printTravelsByDistrict),
               new MenuItem("상세보기", service::printTravel),
               new MenuItem("종료", this::exit),
       };
```

org.scoula

org.scoula

database

fravel

org.scoula

height database

org.scoula

height database

org.scoula

or

org.scoula.travel.app.TravelApp.java

```
public void exit() {
   sc.close();
   JDBCUtil.close();
   System.exit(0);
public void printMenu() {
   System.out.println("=============");
   for (int i = 0; i < menu.length; i++) {
      MenuItem menuItem = menu[i];
      System.out.printf("%d) %s ", i + 1, menuItem.getTitle());
   System.out.println();
   System.out.println("=============");
public int getMenuIndex() {
   System.out.print("선택: ");
   String line = sc.nextLine();
   int ix = Integer.parseInt(line) - 1;
   return ix;
```

org.scoula.travel.app.TravelApp.java

```
public void run() {
    while (true) {
        printMenu();
        int ix = getMenuIndex();
        Runnable command = menu[ix].getCommand();
        command.run();
    }
}

public static void main(String[] args) {
    TravelApp app = new TravelApp();
        app.run();
}
```