

# PORTFOLIO

기록하고 경험하는 것을  
중요시 하는 백엔드 개발자 김가율 입니다.

이름                    김가율

생년월일            1991.02.27

이메일                    gayulz@kakao.com

Git                            <https://github.com/gayulz>

Tistory                    [https://yurizzo.tistory.com/](https://yurizzy.tistory.com/)



# 응급구조사 응급실 연계 플랫폼

Springframework 사용 경험 및 숙련도 향상을 위해 진행함

## EMBAD project



EMBAD

응급실 찾기 병원 접수 내역 정보 변경 로그아웃

### 의료 서비스

필요한 응급실을 찾아드립니다

입원 병상 선택

증환자실 목록

소아/유아 격리포함

수술 및 기타

필수 의료 장비

격리/감염병/코호트

**병원검색**

Copyright © Developer  
Contact Us :: yurizz

1/14

12km

313

322

311

309

42

110

40

308

307

306

305

304

303

302

301

300

299

298

297

296

295

294

293

292

291

290

289

288

287

286

285

284

283

282

281

280

279

278

277

276

275

274

273

272

271

270

269

268

267

266

265

264

263

262

261

260

259

258

257

256

255

254

253

252

251

250

249

248

247

246

245

244

243

242

241

240

239

238

237

236

235

234

233

232

231

230

229

228

227

226

225

224

223

222

221

220

219

218

217

216

215

214

213

212

211

210

209

208

207

206

205

204

203

202

201

200

199

198

197

196

195

194

193

192

191

190

189

188

187

186

185

184

183

182

181

180

179

178

177

176

175

174

173

172

171

170

169

168

167

166

165

164

163

162

161

160

159

158

157

156

155

154

153

152

151

150

149

148

147

146

145

144

143

142

141

140

139

138

137

136

135

134

133

132

131

130

129

128

127

126

125

124

123

122

121

120

119

118

117

116

115

114

113

112

111

110

109

108

107

106

105

104

103

102

101

100

99

98

97

96

95

94

93

92

91

90

89

88

87

86

85

84

83

82

81

80

79

78

77

76

75

74

73

72

71

70

69

68

67

66

65

64

63

62

61

60

59

58

57

56

55

54

53

52

51

50

49

48

47

46

45

44

43

42

41

40

39

38

37

36

35

34

33

32

31

30

29

28

27

26

25

24

23

22

21

20

19

18

17

16

15

14

13

12

11

10

9

8

7

6

5

4

3

2

1

0

1km

2km

3km

4km

5km

6km

7km

8km

9km

10km

11km

12km

13km

14km

15km

16km

17km

18km

19km

20km

21km

22km

23km

24km

25km

26km

27km

28km

29km

30km

31km

32km

33km

34km

35km

36km

37km

38km

39km

40km

41km

42km

43km

44km

45km

46km

47km

48km

49km

50km

51km

52km

53km

54km

55km

56km

57km

58km

59km

60km

61km

62km

63km

64km

65km

66km

67km

68km

69km

70km

71km

72km

73km

74km

75km

76km

77km

78km

79km

80km

81km

82km

83km

84km

85km

86km

87km

88km

89km

90km

91km

92km

93km

94km

95km

96km

97km

98km

99km

100km

101km

102km

103km

104km

105km

106km

107km

108km

109km

110km

111km

112km

113km

114km

115km

116km

117km

118km

119km

120km

121km

122km

123km

124km

125km

126km

127km

128km

129km

130km

131km

132km

133km

134km

135km

136km

137km

138km

139km

140km

141km

142km

143km

144km

145km

146km

147km

148km

149km

150km

151km

152km

153km

154km

155km

156km

157km

158km

159km

160km

161km

162km

163km

164km

165km

166km

167km

168km

169km

170km

171km

172km

173km

174km

175km

176km

177km

178km

179km

180km

181km

182km

183km

184km

185km

186km

187km

188km

189km

190km

191km

192km

193km

194km

195km

196km

197km

198km

199km

200km

201km

202km

203km

204km

205km

206km

207km

208km

209km

210km

211km

212km

213km

214km

215km

216km

217km

218km

219km

220km

221km

222km

223km

224km

225km

226km

227km

228km

229km

230km

231km

232km

233km

234km

235km

236km

237km

238km

239km

240km

241km

242km

243km

244km

245km

246km

247km

248km

249km

250km

251km

252km

253km

254km

255km

256km

257km

258km

259km

260km

261km

262km

263km

264km

265km

266km

267km

268km

269km

270km

271km

272km

273km

274km

275km

276km

277km

278km

279km

280km

281km

282km

283km

284km

285km

286km

287km

288km

289km

290km

291km

292km

293km

294km

295km

296km

297km

298km

299km

300km

301km

302km

303km

304km

305km

306km

307km

308km

309km

310km

311km

312km

313km

314km

315km

316km

317km

318km

319km

320km

321km

322km

323km

324km

325km

326km

327km

328km

329km

330km

331km

332km

333km

334km

335km

336km

337km

338km

339km

340km

341km

342km

343km

344km

345km

346km

347km

348km

349km

350km

351km

352km

353km

354km

355km

356km

357km

358km

359km

360km

361km

362km

363km

364km

365km

366km

367km

368km

369km

370km

371km

372km

373km

374km

375km

376km

377km

378km

379km

380km

381km

382km

383km

384km

385km

386km

387km

388km

389km

390km

391km

392km

393km

394km

395km

396km

397km

398km

399km

400km

401km

402km

403km

404km

405km

406km

407km

408km

409km

410km

411km

412km

413km

414km

415km

416km

417km

418km

419km

420km

421km

422km

423km

424km

425km

426km

427km

428km

429km

430km

431km

432km

433km

434km

435km

436km

437km

438km

439km

440km

441km

442km

443km

444km

445km

446km

447km

448km

449km

450km

451km

452km

453km

454km

455km

456km

457km

458km

459km

460km

461km

462km

463km

464km

465km

466km

467km

468km

469km

470km

471km

472km

473km

474km

475km

476km

477km

478km

479km

480km

481km

482km

483km

484km

485km

486km

487km

488km

489km

490km

491km

492km

493km

494km

495km

496km

497km

498km

499km

500km

501km

502km

503km

504km

505km

506km

507km

508km

509km

510km

511km

512km

513km

514km

515km

516km

517km

518km

519km

520km

521km

522km

523km

524km

525km

526km

527km

528km

529km

530km

531km

532km

533km

534km

535km

536km

537km

538km

539km

540km

541km

542km

543km

544km

545km

546km

547km

548km

549km

550km

551km

552km

553km

554km

555km

556km

557km

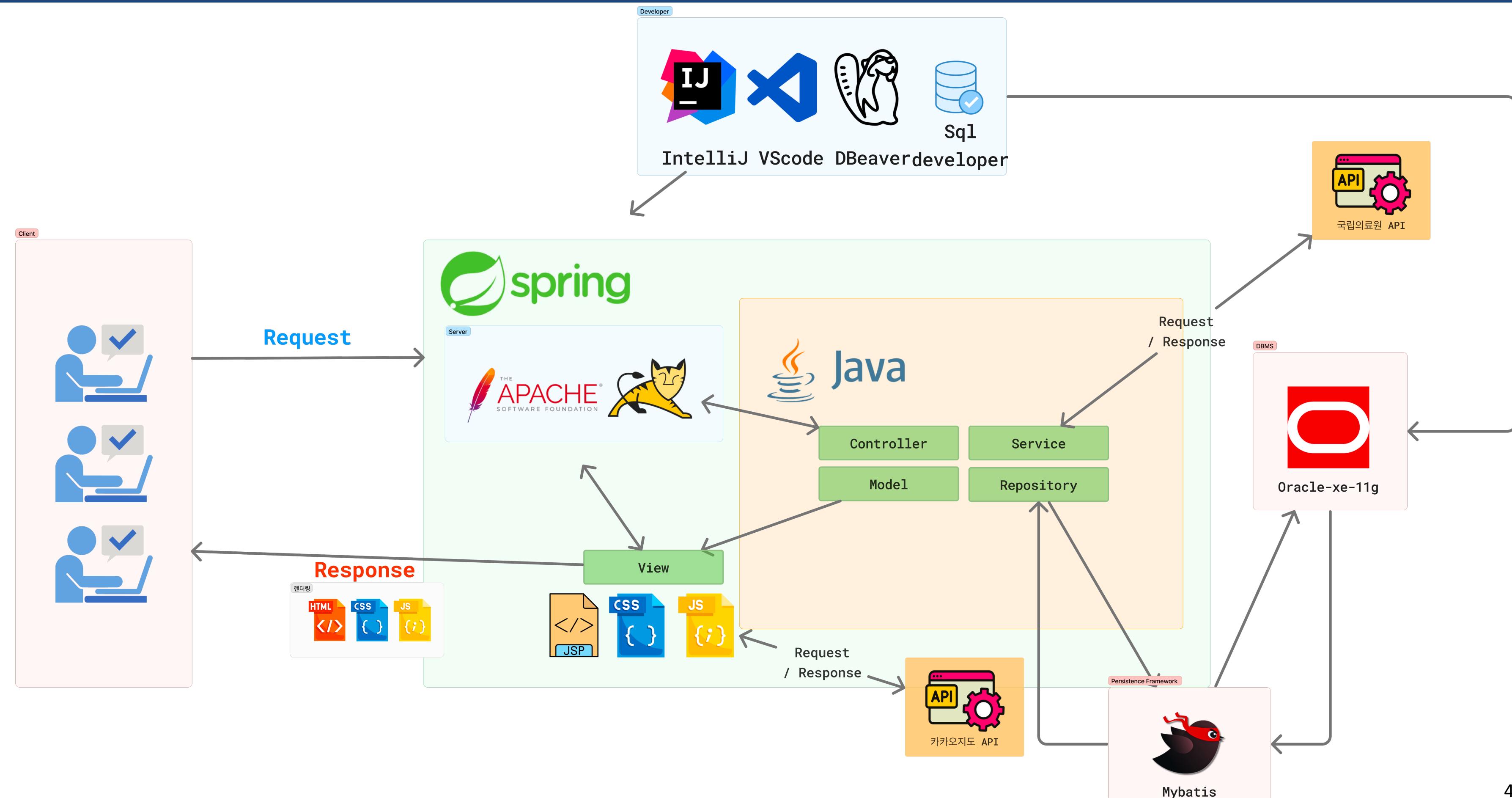
5

# 응급구조사 응급실 연계 플랫폼

Springframework 사용 경험 및 숙련도 향상을 위해 진행함

## Introduce project

작업 기간	2023.12.20 ~ 2024.01.17 ( 29일 )
인력 구성 (기여도)	1인 개발 (기여도 100%)
프로젝트 목적	PL/SQL 문법과 프로시저, 시퀀스를 활용한 자동 증가 처리로 PL/SQL 숙련도 향상 JAVA 인터페이스 활용으로 프로그래밍 경험 강화 Spring Framework와 MVC 패턴 적용으로 전반적인 웹 개발 숙련도 향상 동기/비동기 통신을 통한 네트워크 프로그래밍 이해도 증진 Open API 활용 및 통합으로 API 사용 경험 및 숙련도 향상
프로젝트 내용	1) 응급구조사가 필요한 의료 서비스를 검색하면, 해당 서비스를 제공하고 병상이 있는 응급실을 지도로 보여주는 웹 서비스 개발 2) 공공 데이터 포털의 국립중앙의료원 정보를 활용한 Open API 기반의 데이터 접속 웹 서비스 구현 3) 응급구조사의 접수 목록 확인 및 병원 관계자가 근무 병원에 접수된 응급환자 정보 관리 가능한 의료 서비스 구축
주요 업무 및 상세 역할	1) 데이터베이스 설계, 테이블 생성 및 관리 2) 프로시저 사용하여 Open API 데이터의 안전한 저장 및 관리 3) 시퀀스를 통한 직원번호 및 업체번호 자동 증가 로직 구현 4) 프론트 엔드 구조 및 디자인 구현 5) Spring Framework 환경 구축 및 의존성 관리 6) MVC 패턴을 활용한 비즈니스 로직 구현 7) 동기/비동기 통신 로직 개발 8) 이메일 인증 및 6자리 랜덤 숫자 생성 로직 구현
사용언어 및 개발 환경	Front : HTML / CSS / Javascript Server : apache / tomcat Back : SpringFramework / mybatis / JAVA / JSP DB : oracle Tools : IntelliJ / Vscode / Dbeaver / Sqldeveloper Build : Maven SCM : github
참고 자료	<a href="https://github.com/gayulz/EMBAD_project">https://github.com/gayulz/EMBAD_project</a>



### 의료 서비스

필요한 응급실을 찾아드립니다

[입원실] 외상 전용

중환자실 목록

소아/유아 격리포함

수술 및 기타

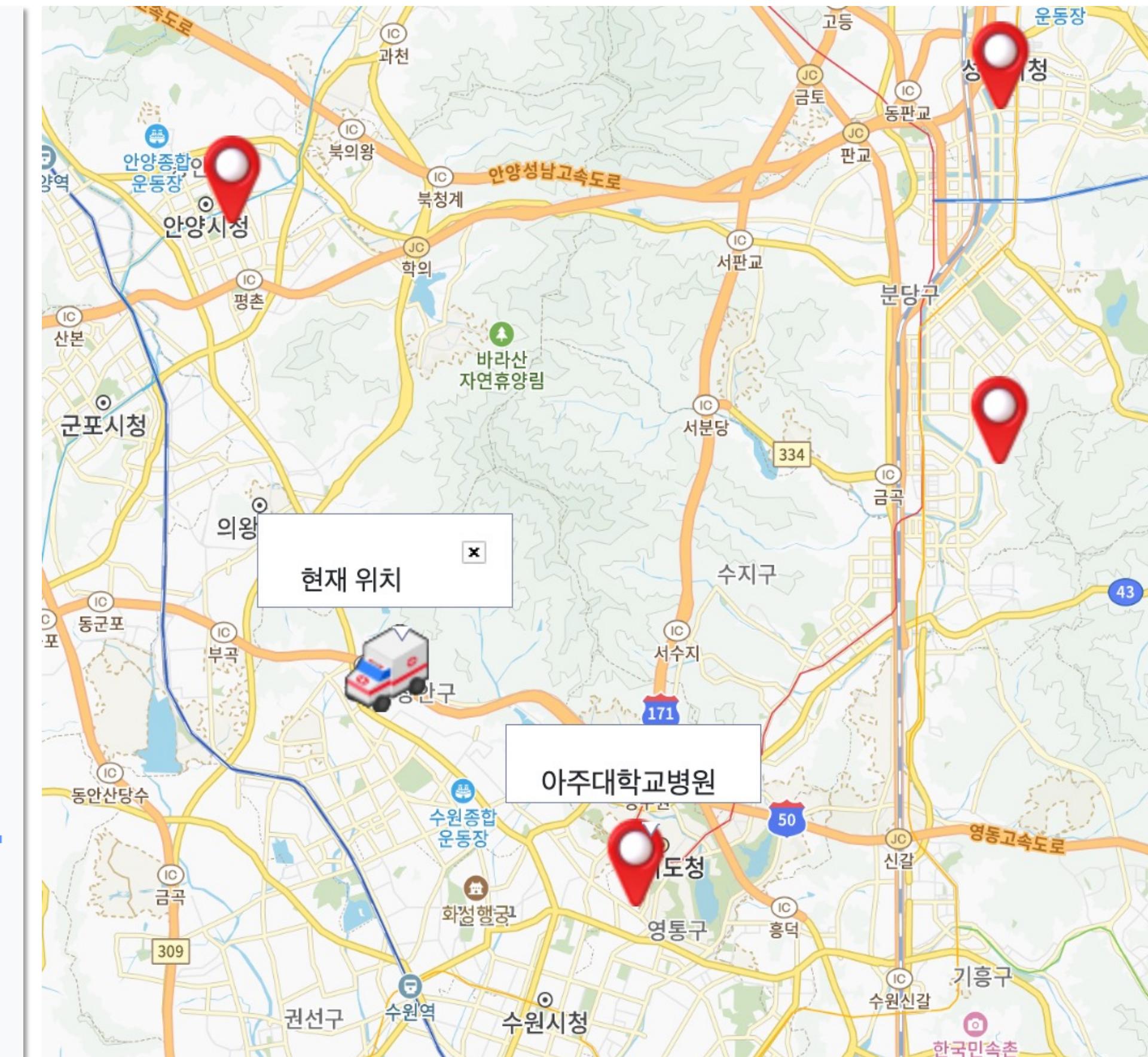
필요 의료 장비

격리/감염병/코호트

[입원실] 일반

[입원실] 외상 전용

병원검색



### \* 기능 소개

- 1) 사용자로부터 요청받은 의료 서비스를 입력 받아, 선택된 의료 서비스를 검색 버튼 상단에 표시하여 사용자가 선택한 서비스를 확인할 수 있게 하는 기능
- 2) 입력된 데이터를 기반으로 서울 및 경기 지역 응급실 중, 요청받은 의료 서비스를 제공하는 병원을 찾아서 결과를 뷰(view)로 전달하는 기능
- 3) 뷰로 전달될 때, API로부터 받은 XML 형식의 응답 데이터 중 JSP에서 필요한 데이터만을 추출하여, JSON 형식으로 뷰단에 전달하는 기능
- 4) JSP에서 응답을 받은 경우, 자바스크립트 기능을 이용하여 카카오 맵 API로 지도를 랜더링하는 기능
- 5) 지도 위에 사용자의 현재 위치와 병원의 위치를 이미지로 표시하여 사용자에게 제공하는 기능
- 6) 표시된 병원 이름 위에 마우스가 이동하면 병원이름이 사용자에게 보여 지는 기능

자세한 소스 코드 및 프로젝트 소개는

 [https://github.com/gayulz/EMBAD\\_project](https://github.com/gayulz/EMBAD_project)에서 확인하실 수 있습니다.

# 반려동물과 함께 가는 숙소 찾기

JAVA Swing 라이브러리를 적용, 다양성과 상속을 적용하여 숙련도 향상을 목표로 진행 함

## 너만가냥 나도가개 project



# 반려동물과 함께 가는 숙소 찾기

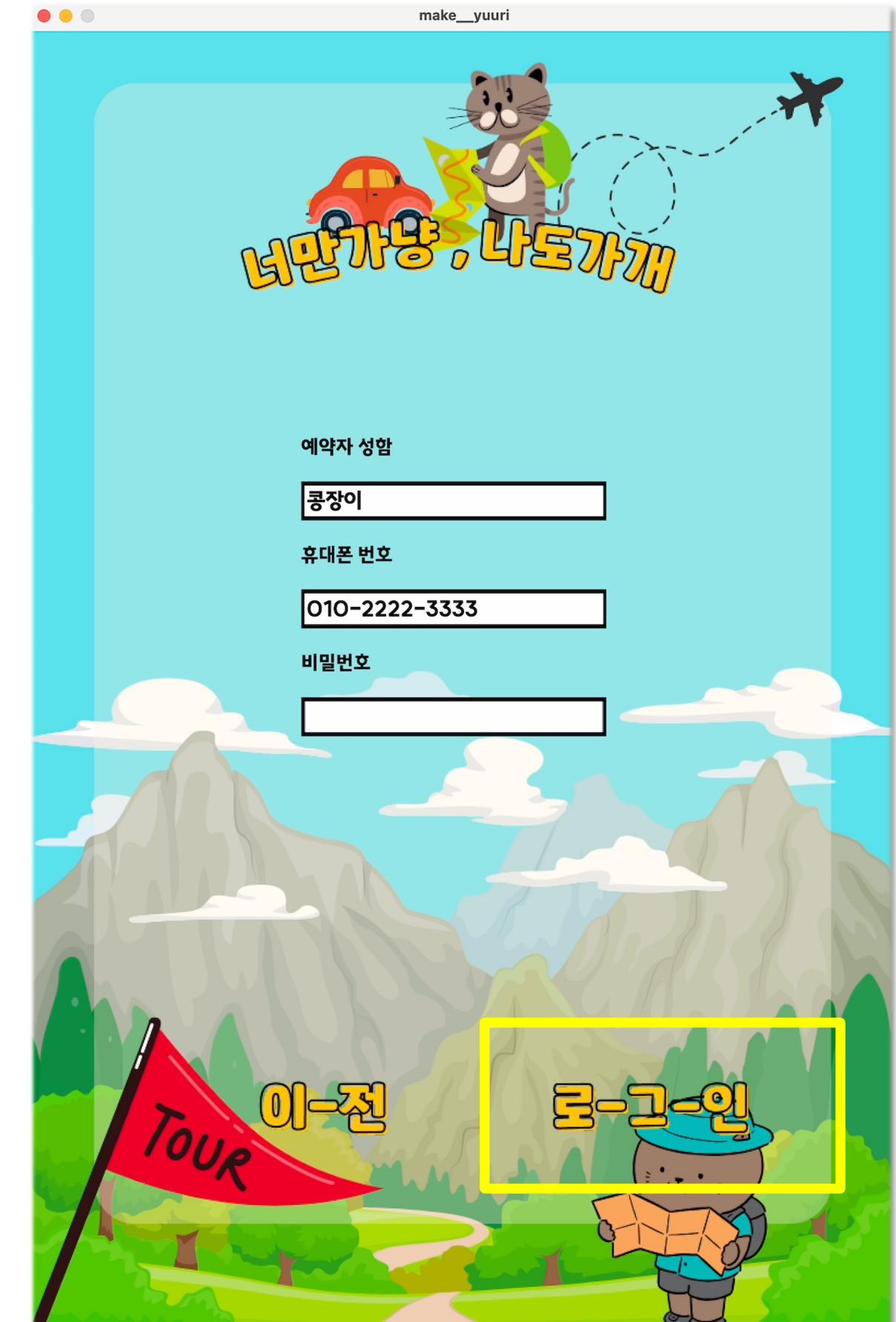
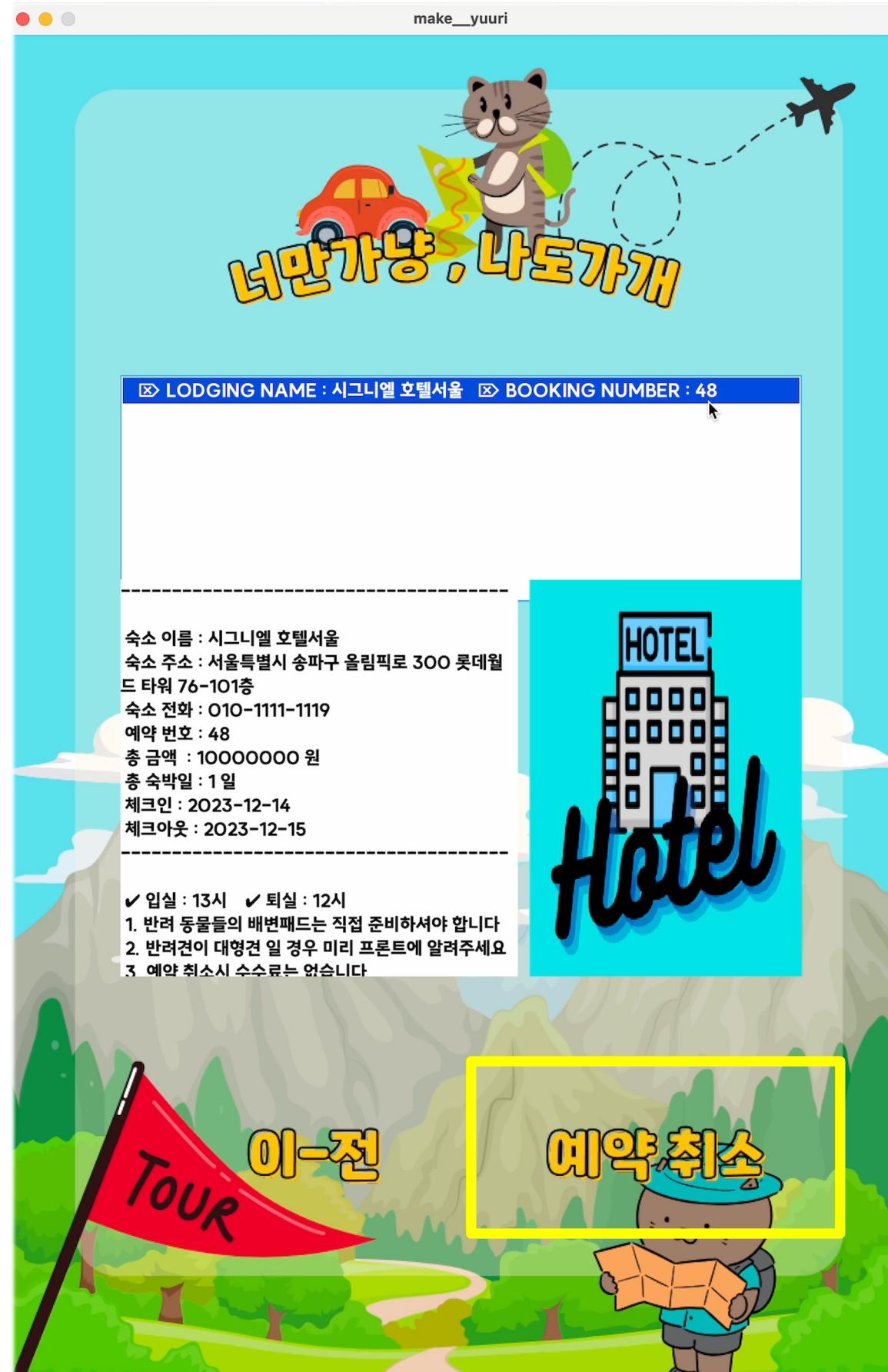
JAVA Swing 라이브러리를 적용, 다형성과 상속을 적용하여 숙련도 향상을 목표로 진행 함

## Introduce project

작업 기간	2023.11.20 ~ 2023.12.01 ( 11일 )
인력 구성 (기여도)	1인 개발 (기여도 100%)
프로젝트 목적	<ul style="list-style-type: none"><li>1) PL/SQL 문법과 프로시저 처리를 통한 숙련도 향상</li><li>2) 객체지향 Java 언어의 Class, 상속, 인터페이스 활용으로 이해도 증진</li><li>3) 디자인 패턴 적용을 통한 개발 경험 강화</li><li>4) Java와 oracle DB의 연동을 통한 CRUD 작업 경험</li><li>5) Java Swing와 javax.awt 라이브러리를 활용한 GUI 개발 경험</li></ul>
프로젝트 내용	<ul style="list-style-type: none"><li>1) 반려동물 동반 가능 숙박업체 검색 및 연계 프로그램 개발</li><li>2) 사용자의 숙소 선호도에 따른 분류, 지역, 날짜 입력 기능 및 Oracle DB연동으로 일치하는 숙소 정보 제공</li><li>3) 숙소 예약, 예약자 개인정보 입력 후 로그인을 통한 예약 정보 확인 기능</li><li>4) 예약 취소 가능한 예약 시스템 구현</li></ul>
주요 업무 및 상세 역할	<ul style="list-style-type: none"><li>1) 데이터베이스 설계 및 테이블 생성, 관리</li><li>2) 제약 조건 설정 및 회원 정보 관리</li><li>3) Java를 사용한 GUI 설계 및 구현</li><li>4) State패턴 적용을 통한 MVC패턴 구현</li><li>5) ojdbc 드라이브를 활용 한 DB연결 및 CRUD 기능 구현</li><li>6) 상속을 이용한 DB연결 관련 드라이버 및 연결 리소스 관리</li><li>7) 액션 리스너를 활용한 JPanel교체 기능 개발</li></ul>
사용언어 및 개발 환경	JAVA , Oracle IntelliJ , SqlDeveloper , Dbeaver
참고 자료	<a href="https://github.com/gayulz/JAVAGUI_project">https://github.com/gayulz/JAVAGUI_project</a>

# 반려동물과 함께 가는 숙소 찾기

Main work - JPanel 별 버튼에 대한 리스너 처리



### \* 기능 소개

- 1) 복잡한 사용자 인터페이스를 가진 Java GUI 어플리케이션에서 각 JButton이 수행해야 할 독립된 기능을 명확히 구별하고 관리할 수 있도록, 액션 리스너를 별도의 클래스로 분리하여 효율적으로 구성함.
- 2) 사용자의 다양한 상호작용에 따라 동일한 JPanel 내의 버튼들이 서로 다른 작업을 수행할 수 있도록, State 패턴을 적용하여 상태에 따라 컨텍스트의 행동이 결정되고 변경될 수 있도록 함.
- 3) State 패턴을 활용하여 애플리케이션의 다양한 상태(예: 검색 화면, 예약 화면 등) 간 전환을 원활하게 관리하며, 각 상태에 적합한 사용자 인터페이스를 제공하여 사용자 경험을 향상시킴.
- 4) 코드의 가독성과 재사용성을 향상시키기 위해, JButton 별로 수행해야 할 기능을 메서드로 분리하여 모듈화함.
- 5) 액션 리스너 내에서 정의된 메서드들은 사용자의 입력에 반응하여 다양한 데이터 접근 및 변화를 처리하며, 이를 통해 예약 추가, 예약 취소, 예약 조회와 같은 사용자 요구를 실시간으로 반영함.

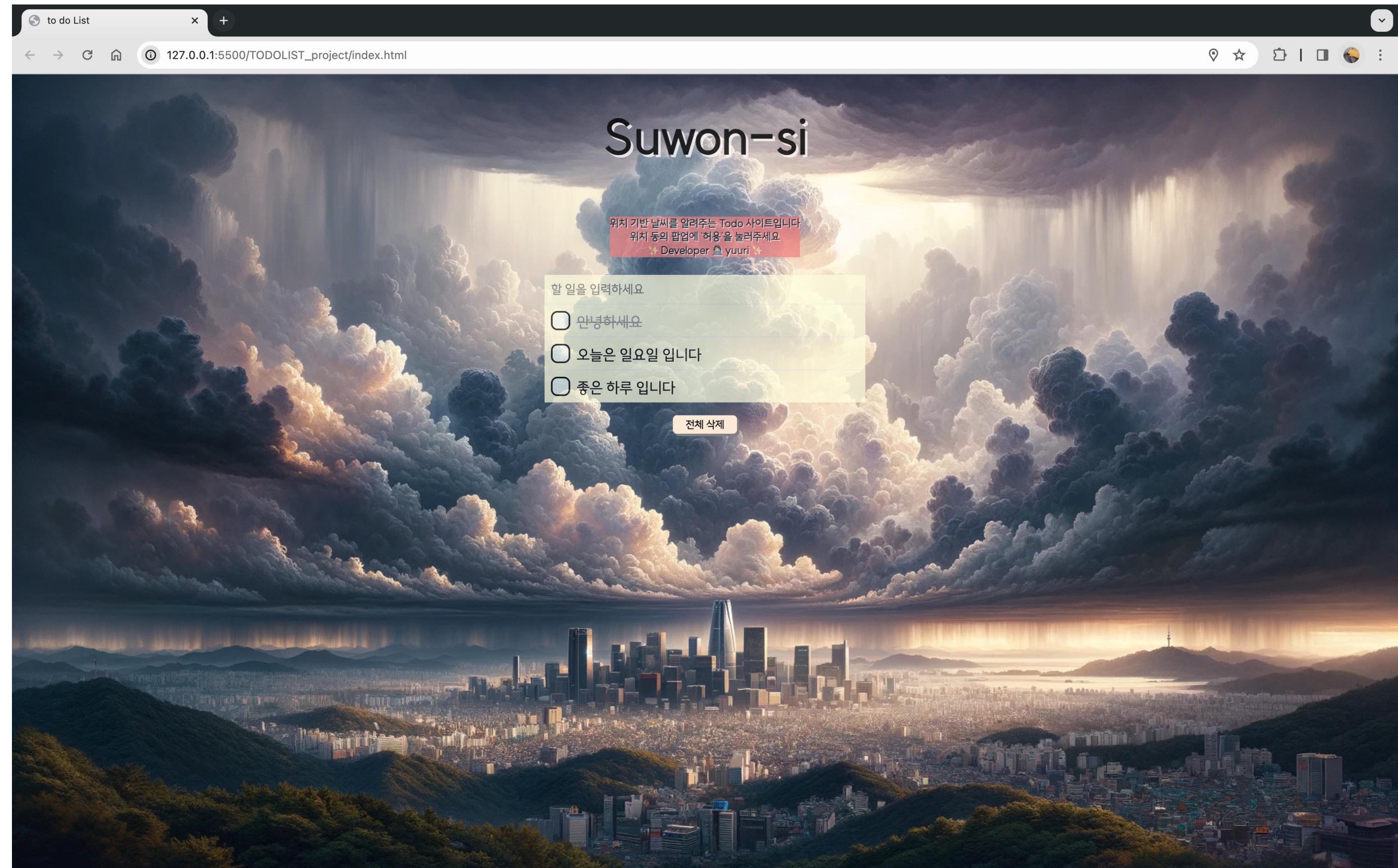
자세한 소스 코드 및 프로젝트 소개는

 [https://github.com/gayulz/JAVAGUI\\_project](https://github.com/gayulz/JAVAGUI_project)에서 확인하실 수 있습니다.

# 웹 스토리지 날씨 TODO-List

개인 사이드 프로젝트로 Javascript와 NodeJs 사용 경험 및 숙련도 향상을 목표로 진행

## weather TODO-Web



# 웹 스토리지 날씨 TODO-List

개인 사이드 프로젝트로 Javascript와 Node.js 사용 경험 및 숙련도 향상을 목표로 진행

## Introduce project

작업 기간	2023.12.23 ~ 2023.12.25 ( 3일 )
인력 구성(기여도)	1인 개발 (기여도 100%)
프로젝트 목적	<ul style="list-style-type: none"><li>1) Javascript 및 Node.js를 활용한 프로그래밍 경험 증진</li><li>2) Geolocation API와 OpenWeather API 사용으로 API통합 개발 스킬 향상</li><li>3) 동기 및 비동기 API 통신에 대한 이해도 강화 및 실제 적용</li><li>4) Web Storage를 사용한 데이터 관리 기법 습득</li></ul>
프로젝트 내용	<ul style="list-style-type: none"><li>1) 사용자의 실시간 위치를 Geolocation API로 파악한 후, 해당 위치를 기반으로 OpenWeather API를 통해 날씨 정보를 조회하고, 그 결과를 시각적으로 표현하는 웹사이트 개발</li><li>2) Web Storage를 사용하여, 웹 페이지나 PC 재부팅 후에도 Todo-List 데이터를 유지하는 기능 개발</li><li>3) 사용자가 관리하는 Todo-List 항목을 개별적으로 삭제하거나, 완료 체크박스를 통해 관리할 수 있는 사용자 친화적인 웹사이트 구축</li></ul>
주요 업무 및 상세 역할	<ul style="list-style-type: none"><li>1) Node.js 및 JavaScript로 소스 코드 작성</li><li>2) 프론트엔드 구현</li><li>3) API 호출 및 데이터 통신 처리</li><li>4) npm 및 yarn을 사용하여 라이브러리 관리</li><li>5) Express와 dotenv 라이브러리를 이용한 백엔드 환경 구축</li><li>6) .env 파일을 이용한 API 키 보안 관리</li><li>7) Express를 통해 백엔드에서 API 키를 호출하고 프론트엔드로 전달하는 기능 구현</li></ul>
사용언어 및 개발 환경	HTML5, CSS3, JavaScript, Node.js Geolocation API, OpenWeather API VS Code Web Storage npm, yarn, express, dotenv
참고 자료	<a href="https://github.com/gayulz/TODOLIST_project">https://github.com/gayulz/TODOLIST_project</a>

# 웹 스토리지 날씨 TODO-List

## Main work - API 통신 처리 로직

The screenshot shows the Chrome DevTools Application tab interface. The left sidebar lists storage types: Manifest, Service workers, and Local storage. Local storage is expanded, showing items at `http://127.0.0.1:5500`. The main area displays key-value pairs for the `saved-items` object.

**Storage Types:**

- 매니페스트
- Service workers
- 저장용량

**Local Storage:**

- 로컬 스토리지 (`http://127.0.0.1:5500`)
  - 키: saved-items
  - 값: [{"contents": "안녕하세요", "complete": true}, {"contents": "오늘은 일요일입니다", "complete": false}]
    - 0: {"contents": "안녕하세요", "complete": true}
    - 1: {"contents": "오늘은 일요일입니다", "complete": false}
    - 2: {"contents": "좋은 하루입니다", "complete": false}
- 세션 저장소
- IndexedDB
- 웹 SQL
- 쿠키
- 비공개 상태 토큰
- 관심분야 그룹
- Shared Storage
- 캐시 스토리지

**Application Tab:**

- 요소
- 콘솔
- 소스
- 네트워크
- 성능
- 메모리
- 애플리케이션 (highlighted)
- 보안
- Lighthouse
- 녹음기
- »
- ×

**Filter:** 필터

**Address:** http://127.0.0.1:5500

**Source:** http://127.0.0.1:5500

**Key:**

- saved-items
- \_\_mantle\_tile\_meta\_data
- saved-weather

**Value:**

```
[{"contents": "안녕하세요", "complete": true}, {"contents": "오늘은 일요일입니다", "complete": false} {"tileTypes": ["basic", "satellite", "terrain"], "version": 1} {"location": "Suwon-si", "weather": "Clouds"}]
```

**Background Services:**

### \* 기능 소개

- 1) 사용자의 편의성을 고려하여 데이터베이스 대신 Web Storage를 사용하여 데이터를 로컬에 저장함
- 2) 웹 사이트가 재부팅 되거나 PC가 재 시작 되어도 사용자의 정보와 설정이 유지 됨
- 3) 할일 목록에 대한 생성, 완료 표시, 삭제기능을 단일 페이지 어플리케이션(SPA)내에서 자연스럽게 처리할 수 있도록 함수를 이용해 기능을 세분화하고 모듈화 했음
- 4) 웹 사이트에 접속 할 때마다 Web Storage에 저장된 할 일 목록과 날씨 데이터를 자동으로 불러와 사용자에게 즉각 표시하도록 하여, 사용자 경험을 향상 시켰습니다.

자세한 소스 코드 및 프로젝트 소개는

 [https://github.com/gayulz/TODOLIST\\_project](https://github.com/gayulz/TODOLIST_project)에서 확인하실 수 있습니다.