**1. 디렉토리 구조**

**src/main/java**

* com.yourcompany.projectname: 기본 패키지
  + controller: 컨트롤러 클래스
  + service: 서비스 클래스
  + repository: 데이터 접근 클래스
  + model: 도메인/엔티티 클래스
  + dto: 데이터 전송 객체
  + config: 설정 파일

**src/main/resources**

* static: 정적 리소스 (CSS, JavaScript, 이미지 등)
  + css: CSS 파일
  + js: JavaScript 파일
  + images: 이미지 파일
* templates: HTML 템플릿 파일
* application.properties: Spring 설정 파일

**2. 자바 파일**

**클래스 이름**

* 파스칼 케이스(PascalCase)
  + UserService, UserController, UserRepository, UserDto

**메서드 이름**

* 카멜 케이스(camelCase)
  + getUserById, createUser, updateUser, deleteUser

**변수 이름**

* 카멜 케이스(camelCase)
  + userId, userName, userEmail

**3. 자바스크립트 파일**

**파일 이름**

* 카멜 케이스(camelCase) 또는 케밥 케이스(kebab-case)
  + main.js, userManagement.js, dataService.js

**함수 이름**

* 카멜 케이스(camelCase)
  + fetchUserData, submitForm, handleClick

**변수 이름**

* 카멜 케이스(camelCase)
  + userId, userName, isFormValid

**4. HTML 파일**

**파일 이름**

* 케밥 케이스(kebab-case)
  + index.html, user-profile.html, user-list.html

**클래스 및 ID**

* 케밥 케이스(kebab-case)
  + class="user-profile", id="main-header"

**5. CSS 파일**

**파일 이름**

* 케밥 케이스(kebab-case)
  + styles.css, user-profile.css, layout.css

**클래스 및 ID**

* 케밥 케이스(kebab-case)
  + class="btn-primary", id="header-container"

**6. 변수**

**자바 변수**

* 카멜 케이스(camelCase)
  + int userId;, String userName;

**자바스크립트 변수**

* 카멜 케이스(camelCase)
  + let userId;, const userName;

**7. 기타 규칙**

* **상수**: 모든 대문자, 단어 사이에 밑줄
  + public static final String BASE\_URL = "https://api.example.com";
* **패키지 이름**: 소문자, 단어 사이에 점
  + com.yourcompany.projectname.controller

**예시**

**디렉토리 구조 예시**

src/

└── main/

├── java/

│ └── com/

│ └── yourcompany/

│ └── projectname/

│ ├── controller/

│ ├── service/

│ ├── repository/

│ ├── model/

│ ├── dto/

│ └── config/

├── resources/

│ ├── static/

│ │ ├── css/

│ │ ├── js/

│ │ └── images/

│ └── templates/

└── application.properties

클래스 예시

// UserService.java

package com.yourcompany.projectname.service;

public class UserService {

public UserDto getUserById(int userId) {

// ...

}

}

자바스크립트 예시

// main.js

function fetchUserData(userId) {

// ...

}

HTML 예시

<!-- user-profile.html -->

<div id="user-profile">

<h1 class="profile-header">User Profile</h1>

<!-- ... -->

</div>

CSS 예시

/\* styles.css \*/

.profile-header {

font-size: 2em;

color: #333;

}

\*주석 작성 예시

**1. Java 파일**

**주석 종류**

* **단일 행 주석**: //로 시작하며, 주로 간단한 설명이나 메모에 사용됩니다.
* **다중 행 주석**: /\* ... \*/로 감싸며, 여러 줄의 설명을 작성할 때 사용됩니다.
* **Javadoc 주석**: /\*\* ... \*/로 감싸며, 클래스, 메서드, 필드 등에 대한 문서화에 사용됩니다. JavaDoc 도구로 HTML 문서를 생성할 수 있습니다.

**예시**

java

코드 복사

package com.yourcompany.projectname.service;

/\*\*

\* UserService 클래스는 사용자 관리 기능을 제공합니다.

\* 사용자 생성, 조회, 업데이트, 삭제 기능을 포함합니다.

\*/

public class UserService {

/\*\*

\* 사용자 ID로 사용자를 조회합니다.

\*

\* @param userId 조회할 사용자의 ID

\* @return 조회된 사용자 정보

\*/

public UserDto getUserById(int userId) {

// TODO: 데이터베이스에서 사용자 조회 구현

// 이 부분에서 userId를 사용하여 사용자 정보를 검색합니다.

return new UserDto();

}

/\*\*

\* 새로운 사용자를 생성합니다.

\*

\* @param userDto 생성할 사용자 정보

\* @return 생성된 사용자 정보

\*/

public UserDto createUser(UserDto userDto) {

// 데이터베이스에 사용자 정보를 저장

return new UserDto();

}

}

**2. JavaScript 파일**

**주석 종류**

* **단일 행 주석**: //로 시작하며, 한 줄짜리 설명에 사용됩니다.
* **다중 행 주석**: /\* ... \*/로 감싸며, 여러 줄에 걸친 설명에 사용됩니다.

**예시**

javascript

코드 복사

// main.js

/\*\*

\* 주어진 사용자 ID로 사용자 데이터를 가져옵니다.

\*

\* @param {number} userId - 조회할 사용자의 ID

\*/

function fetchUserData(userId) {

// TODO: 서버에서 사용자 데이터 가져오기

// 이 부분에서 AJAX 호출을 사용하여 사용자 데이터를 가져옵니다.

console.log("Fetching data for user with ID:", userId);

}

/\*\*

\* 폼 데이터를 서버로 전송합니다.

\*

\* @param {Event} event - 폼 제출 이벤트

\*/

function submitForm(event) {

// 폼 제출 기본 동작을 막습니다.

event.preventDefault();

// TODO: 폼 데이터 수집 및 서버로 전송

const formData = new FormData(event.target);

console.log("Form data:", formData);

}

**3. HTML 파일**

**주석 종류**

* **HTML 주석**: <!-- ... -->로 감싸며, HTML 코드 내에서 설명이나 메모를 작성할 때 사용됩니다.

**예시**

html

코드 복사

<!-- user-profile.html -->

<!-- 사용자 프로필 섹션 -->

<div id="user-profile">

<!-- 프로필 헤더 -->

<h1 class="profile-header">User Profile</h1>

<!-- 사용자 정보 폼 -->

<form id="user-form">

<!-- 사용자 이름 입력 -->

<label for="user-name">Name:</label>

<input type="text" id="user-name" name="userName">

<!-- 제출 버튼 -->

<button type="submit">Save</button>

</form>

</div>

**4. CSS 파일**

**주석 종류**

* **CSS 주석**: /\* ... \*/로 감싸며, CSS 코드 내에서 설명이나 메모를 작성할 때 사용됩니다.

**예시**

css

코드 복사

/\* styles.css \*/

/\* 사용자 프로필 스타일 \*/

#user-profile {

margin: 0 auto;

width: 50%;

padding: 20px;

}

/\* 프로필 헤더 스타일 \*/

.profile-header {

font-size: 2em;

color: #333;

}

/\* 사용자 이름 입력 스타일 \*/

#user-name {

width: 100%;

padding: 10px;

margin-top: 10px;

}