|  |
| --- |
|  |
| 주석 표준 가이드 |
|  |
|  |
| **김민지** |
|  |
|  |

|  |
| --- |
|  |

목차

[1 프로젝트 주석 표준의 의미 2](#_Toc77604763)

[2 프로젝트 주석 작성 표준안 3](#_Toc77604764)

[2.1 Method Header 3](#_Toc77604765)

[3 소스 코드 내부에서만 사용되는 주석 4](#_Toc77604766)

[3.1 In-Line Comment 4](#_Toc77604767)

[4. GitHub Commit Message 작성 5](#_Toc77604768)

[4.1 GitHub Commit Message 5](#_Toc77604770)

# 프로젝트 주석 표준의 의미

주석 표준의 사용은 필수적으로 사용해야 합니다. 프로젝트에 사용되는 각각의 메서드(또는 함수)등 해당 기능에 대한 내용을 설명함으로써 유지보수 또는 다른 사람에게 자신이 만든 메서드 등을 다른 개발자에게 내용을 전달할 수 있도록 하는데 있습니다.

주석 표준은 크게 2가지로 구분합니다.

1. 내부에서만 접근하여 사용되는 주석
2. 외부에서 접근할 수 있는 주석

내부에서만 사용되는 주석

* 하나의 기능을 Private 으로 선언된 메서드 또는 함수에 사용되는 주석
* 메서드에서 반복문 또는 분기문에 사용되는 주석

외부에서 접근할 수 있는 주석

* Public 등 외부에서 접근할 수 있는 한정자를 사용할 경우

내부와 외부의 정확한 사용 범위를 기준으로 주석을 사용하여야 합니다. 모듈 프로젝트1에서는 주석의 내용을 생성한 클래스 및 메서드 등의 모든 주석을 Visual Studio Code를 이용하여 정의 합니다.

# 프로젝트 주석 작성 표준안

## Method Header

* + 서비스 내용
  + 최초작성자
  + Python 주석

Summary: 서비스 내용, 최초 작성자

param: 파라미터 이름, 내용

returns: 리턴값 내용

|  |
| --- |
| # Summary:  # 서비스 내용, 최초 작성자  # param:  # 파라미터 이름, 내용  # returns:  # 리턴 값 내용 |

# 소스 코드 내부에서만 사용되는 주석

## In-Line Comment

* + 변수 선언 시
  + 비교/루프 구문 사용 시
    - if:
    - while:
    - for:
  + 가급적 블록 단위로 주석을 작성
  + 하드코딩은 권장하지 않음

예) 변수일 경우

|  |
| --- |
| # 카운터에 사용  totalCount = 100 |

예) 반복문 등을 경우

|  |
| --- |
| # CSV 읽기 후 내용을 정제하여 JSON에 저장  for i in data: |

# 4. GitHub Commit Message 작성



## GitHub Commit Message

GitHub의 Commit을 활용하여 Message를 작성하여 소스 코드의 공유에 활용합니다.

주요 내용

* 작성자
* Commit Type

Commit Type

* feat: 새로운 기능 추가
* fix: 버그 수정
* docs: 문서 수정
* style: 코드 포맷팅, 코드 변경이 없는 경우
* refactor: 코드 리펙토링
* test: 테스트 코드, 리펙토링 테스트 코드 추가
* chore: 빌드 업무 수정