

Coding Standard: Naming Convention

버전	일자	변경 내역	작성자
Ver 1.0	2024/11/19	최초 버전	김세윤
Ver 2.0	2024/11/23	JSON 내 파라미터 이름 camel case 로 변경 및 메소드명 변수명 제한 삭제	김세윤
Ver 3.0	2024/12/7	API Path 통일	김준호
Ver 4.0	2024/12/13	최종 반영 및 포매팅	최수아

UOSense 명명 규칙

- 상수는 모두 대문자
- 클래스와 인터페이스 이름
 - 대문자로 시작(명사 또는 명사구)
 - Pascal Case
- 변수 이름
 - 일반적으로 소문자 시작
 - 최소 1 자, 최대 20 자 길이의 제한을 둘 것
 - 모호한 이름 사용하지 않기
- 메소드 이름
 - Camel Case
 - 일반적으로 소문자 시작
 - 매개변수는 전치사구로 표현
 - 웬만하면 By로 통일
 - e.g., findByCategory
 - 조건 묻는 boolean 함수 이름은 "is"로 시작
 - 메소드명의 경우 글자 수 제한 삭제

5. 패키지 이름

- Camel Case

6. Json 내 파라미터 이름

- Camel Case

7. 블록은 일반적인 명명 규칙을 따름

- if-else문 같은 경우 아래 예시와 같이 블록을 이어서 작성

```
if (flag) {  
    validate();  
    update();  
} else {
```

8. 문장과 수식

- 문장이나 수식은 메소드나 클래스로 패키지화
- 블록 문장
- 제어구조 중첩되었을 때 발생하는 혼란 방지
- 수식
 - 괄호를 이용하여 오퍼레이션의 순서를 명확히

9. 오류 처리

- 오류 발생 시 주석 처리

10. 주석

- `/**` 혹은 `//` 추가

11. 메서드 주석

- 매개 변수와 리턴 값, 예외 표시 가능
- 메소드 상단에 `/* */` 사용
- 필수 작성 사항 아님

```
/*  
    @param username  
    @throws IllegalArgumentException
```

```
@return UserEntity
*/
```

12. Swagger 문서 주석

- 아래 예시와 같이 tag, operation, API Response 작성
- tag : 컨트롤러 역할 표기
 - @Tag(name = "댓글 관리")
- operation : 컨트롤러 메소드 기능 표기
 - @Operation(summary = "댓글 삭제", description = "기존 댓글을 삭제합니다.")
- ApiResponse : 응답 코드에 따른 설명 표기
 - 같은 코드는 설명 형식 통일
 - 예시
 - 200: ~이 성공적으로 ~되었습니다
 - 400: 잘못된 요청입니다.
 - 500: 서버 오류입니다 / 네트워크 연결에 문제가 있습니다.

```
@ApiResponses(value = {
    @ApiResponse(responseCode = "200", description = "성공적으로 삭제되었습니다"),
    @ApiResponse(responseCode = "404", description = "댓글이 존재하지 않습니다"),
    @ApiResponse(responseCode = "400", description = "잘못된 요청입니다"),
    @ApiResponse(responseCode = "500", description = "서버 오류입니다 / 네트워크 연결에 문제가 있습니다")
})
```

13. 컨트롤러 및 서비스의 CRUD 메서드

- Controller
 - 컨트롤러 클래스 안에서 메서드 명을 작성할 때는 아래와 같은 동사로 시작
 - create: 등록 유형의 controller 메서드
 - get: 조회 유형의 controller 메서드
 - 전체목록일 경우 뒤에 list를 붙임. e.g. getListByDoorType
 - update: 수정 유형의 controller 메서드

- delete: 삭제 유형의 controller 메서드
- Service
 - 서비스 클래스 안에서 메서드 명을 작성할 때는 아래와 같은 동사로 시작
 - register: 등록 유형의 service 메서드
 - find: 조회 유형의 service 메서드
 - 특정 파라미터로 조회할 경우 findBy{파라미터} 형식으로 명명
 - 단, 파라미터가 기본적인 값일 경우 By~ 생략
 - edit: 수정 유형의 service 메서드
 - remove: 삭제 유형의 service 메서드
- 행위 대상이 default 하위일 경우 동사 뒤에 붙이고 default 일 경우 생략한다.
 - e.g. saveImage(사진 저장), remove(식당 삭제)

14. API Path

- **default path:** `/api/v1/도메인`
- HTTP 요청 방식 유형 별로 아래와 같은 형식을 따름
- GET
 - `/도메인/get`
- POST
 - `/도메인/create`
- PUT
 - `/도메인/update`
- DELETE
 - `/도메인/delete`