

자주 사용하는 IntelliJ 단축키 정리

OS : macOS, Windows

- ⇧ : shift
 - ^ : controll
 - ⌘ : option
 - ⌘ : command
 - Home : fn + ←
 - End : fn + →
 - Page Up : fn + ↑
 - Page Down : fn + ↓
-

일반

- ⌘ 0~9 (Alt + [0-9]) : 각 도구창 토글
 - ⌘+1 : Project
 - ⌘+2 : Favorite
 - ⌘+3 : Search
 - ⌘+4 : Run
 - ⌘+5 : Debug
 - ⌘+6 : Problems
 - ⌘+7 : Structure
 - ⌘+8 : Services
 - ⌘+9 : VCS
 - ⌘ F12 : Terminal
 - ⌘ ⇧ F12 (Ctrl + Shift + F12) : 편집기 영역 최대화 토글
 - ⌘ , (Ctrl + Alt + S) : 설정 (Settings)
 - ⌘ ; (Ctrl + Alt + Shift + S) : 프로젝트 구조 (Project Structure)
-

****탭, 그룹****

- **⌘ [/]** (Alt + ← / →) : 탭 이동
-

****검색****

- **⌘ F** (Ctrl + F) : 찾기
 - **⇧ ⌘ F** (Ctrl + Shift + F) : 찾기
 - **⌘ R** (Ctrl + R) : 바꾸기
 - **⇧ ⌘ R** (Ctrl + Shift + R) : 바꾸기
 - **⌘ R / ⇧ ⌘ R** (F3 / Shift + F3) : 다음 찾기 / 이전 찾기
 - **⇧ ⇧** (Shift + Shift) : 전체 검색 (파일, 클래스, 액션 등)
 - **⌘ E** (Ctrl + E) : 최근 사용한 파일 목록 조회
 - **⇧ ⌘ E** (Ctrl + Shift + E) : 최근에 수정한 파일 목록 조회
 - **⌘ F7** (Alt + F7) : 해당 항목이 사용된 위치 검색
-

****네비게이션****

- **⌘ B** (Ctrl + B) : 정의로 이동, 정의인 경우 사용되는 목록 출력
 - **⌘ ⌘ B** (Ctrl + Alt + B) : 해당 코드의 구현부로 이동
 - **⌘ P** (Ctrl + P) : 현재 위치한 메서드의 파라미터 정보를 조회
 - **⌘ Space** (Ctrl + Shift + I) : 구현된 코드를 조회 (Quick Definition)
 - **F2 / ⇧ F2** (F2 / Shift + F2) : 오류, 경고 영역으로 이동
 - **⌘ + [/]** (Alt + ← / →) : 이전 위치 / 다음 위치
 - **⌘ L** (Ctrl + G) : 지정한 라인으로 이동
 - **^ M** (Ctrl + Shift + M) : Go to matching bracket
 - **⌘ ⌘ [/]** (Ctrl + [/]) : Go to opening/closing tag
-

****편집****

- **⌘ D** (Ctrl + D) : 라인 복제

- ⌘ Delete (Ctrl + Y) : 라인 삭제
 - ⌘ ⌥ ↑ / ↓ (Alt + Shift + ↑ / ↓) : 라인 이동
 - ⌘ ⌘ ↑ / ↓ (Ctrl + Shift + ↑ / ↓) : 구문 이동
 - ⌘ ← / → (Fn + ← / →) : 포커스를 끝으로 이동
 - ⌘ ← / → (Ctrl + ← / →) : 포커스를 단어 단위로 이동
 - ⌘ ↑ / ↓ (Ctrl + W, Shift + Ctrl + W) : 선택 영역 확장/축소
 - ⌘ ⌘ V : 클립보드 확인하여 붙여넣기
 - ⌘ ⌘ Click (Alt + Shift + Click) : 클릭한 위치에 커서 삽입
-

****코딩 편의****

- ⌘ / (Ctrl + /) : 라인 주석
 - ⌘ ⌘ / (Ctrl + Shift + /) : 블록 주석
 - ⌘ ⌘ L (Ctrl + Alt + L) : 코드 정렬
 - ^ Space (Ctrl + Space) : 기본 자동완성
 - ^ ⌘ Space (Ctrl + Shift + Space) : 스마트 자동완성
 - ⌘ N, ^ ⌘ ↵ (Alt + Insert) : 생성자, Getter/Setter, Override 등의 코드 생성
 - ⌘ O (Ctrl + O) : Override 메소드 자동 생성
 - ⌘ I (Ctrl + I) : Implement 가능한 메서드 자동 생성
 - ⌘ ↵ (Alt + Enter) : 퀵픽스 제안
 - ⌘ F6 (Shift + F6) : 이름 바꾸기
 - ⌘ F6 (Ctrl + F6) : 접근제어자, 반환타입, 이름 등 주요 정보 변경
 - ^ T (Ctrl + Alt + Shift + T) : 리팩토링 관련 전체 항목을 조회
 - ^ ⌘ O (Ctrl + Alt + O) : import 정리
 - ⌘ P (Ctrl + P) : 파라미터 보기
 - F1 (Ctrl + Q) : Doc 보기
 - ⌘ . (Ctrl + .) : 접기 / 펼치기 토글
-

****Live Template****

- ⌘ J (Ctrl + J) : Live template 목록 조회
- psf : public static final
- prsf : private static final
- psvm : public static void main(String[] args)

- thr : throw new
- fori : for 문
- iter : for-each 문
- todo : 날짜와 함께 todo 주석 ex) // TODO: 2019/12/31
- sout : System.out.println();

Live Template 추가 등록

Preferences → Editor → Live Templates → others → +

빌드, 실행, 디버그

- ⌘ F9 (Ctrl + F9) : 프로젝트 빌드(Make project)
 - ^ ⌵ R (Shift + F10) : 실행 (Run)
 - ⌘ D (Shift + F9): 디버그 (Debug)
 - ⌘ F2 (Ctrl + F2) : 실행 종료
 - ⌘ F8 (Ctrl + F8) : 브레이크 포인트
 - ⌘ F9 (F9) : Resume (다음 브레이크 포인트로)
 - F8 (F8) : Step Over (다음 라인으로)
 - F7 (F7) : Step Into (안으로 들어가기)
 - ⇧ F8 (Shift + F8) : Step Out (밖으로 나가기)
-

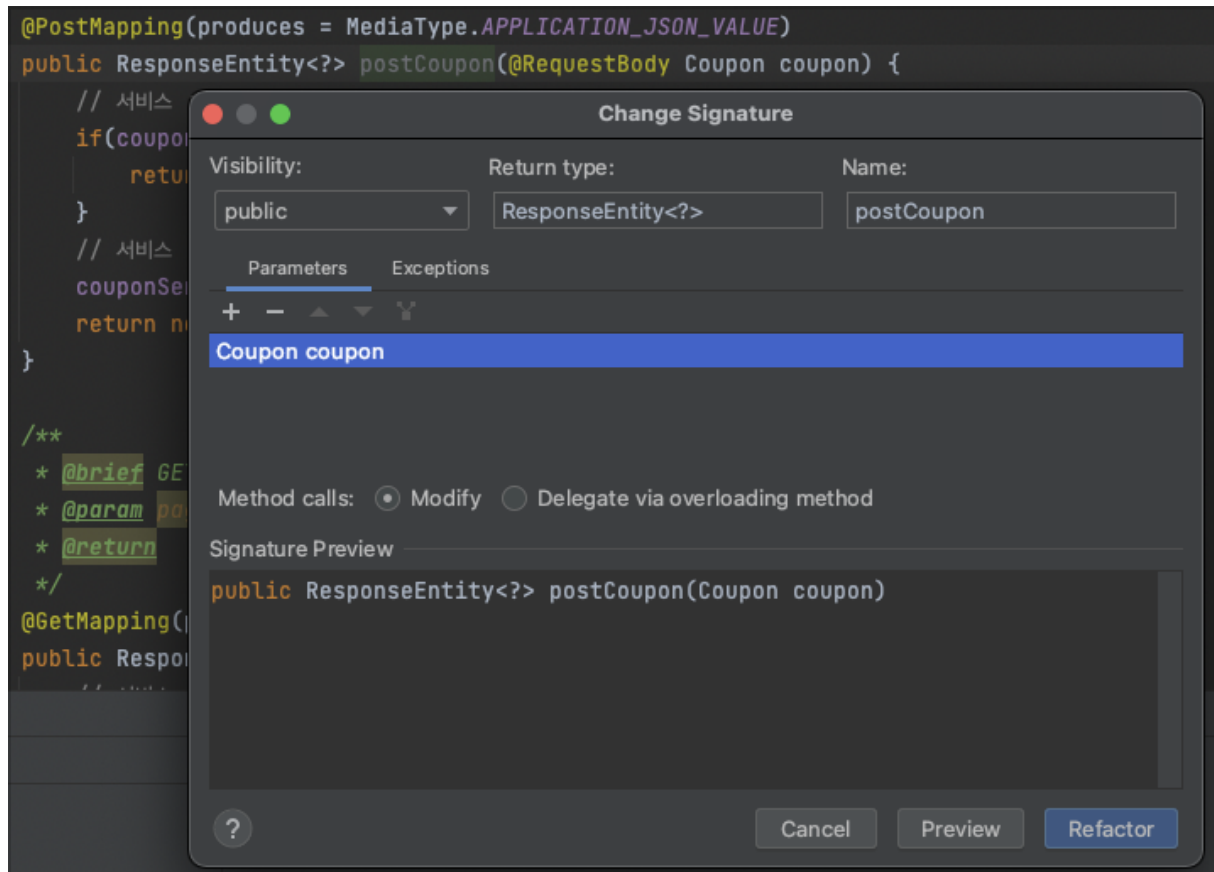
버전관리

- ^ V (Alt + BackQuote) : VCS Operations
 - ⌘ T (Ctrl + T) : Update
 - ⌘ K (Ctrl + K) : 커밋
 - ⌘ ⇧ K (Ctrl + Shift + K) : Push
-

리팩토링

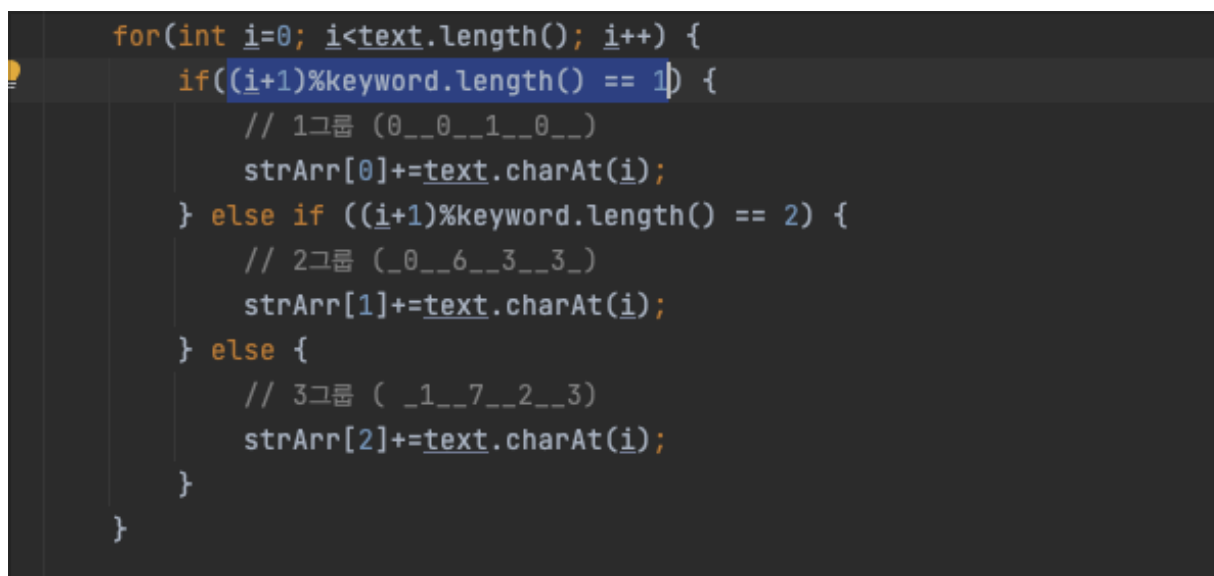
- ^ T (Ctrl + Alt + Shift + T) : 리팩토링 관련 전체 항목을 조회
- ⇧ F6 (Shift + F6) : 이름 바꾸기

- ⌘ F6 (Ctrl + F6) : 파라미터, 접근제어자, 리턴타입, 익셉션, 이름 등을 편집



리팩토링

- 해당 구문 메소드로 추출 (Extract Methods)



```

for(int i=0; i<text.length(); i++) {
    if(isaBoolean * (keyword, i)) {
        // 1그룹 (0__0__1__0__)
        strArr[0]+=text.charAt(i);
    } else if ((i+1)%keyword.length() == 2) {
        // 2그룹 (_0__6__3__3_)
        strArr[1]+=text.charAt(i);
    } else {
        // 3그룹 ( _1__7__2__3)
        strArr[2]+=text.charAt(i);
    }
}

private static boolean isaBoolean(String keyword, int i) {
    return (i+1)%keyword.length() == 1;
}

```

- 해당 값 로컬 변수로 추출 (Introduce Variable)

```

couponService.createCoupon(coupon);
return new ResponseEntity<>( body: "{}", HttpStatus.CREATED);
}

```

```

couponService.createCoupon(coupon);
String s = "{}";
return new ResponseEntity<>( body: s, HttpStatus.CREATED);
}

```

- 해당 값 상수로 추출 (Introduce Constant)
- 해당 값 파라미터로 추출 (Introduce Parameter)