

# 자기주도 PJT

코딩 컨벤션 해설

코딩 컨벤션의 요소에 해당하는 설명을 참고해서 작성하시기 바랍니다

## 1. 들여쓰기

space 와 tab 을 섞어서 사용하지 않는다

사용자는 개발 환경에 따라 space 또는 tab 의 들여쓰기가 다르게

보일 수 있기 때문에 이를 통일하지 않으면 가독성이 떨어진다

## 2. 문장의 종료

한 줄에 하나의 문장만 허용하고, 종료 시 세미콜론을 사용한다

## 3. 명명 규칙

변수와 함수의 이름을 정할 때는 FE 개발랩에서는 camel case 를 사용한다

예약어를 사용하지 않는다

```
// Bad
let class;
let enum;
let extends;
let super;
let const;
let export;
let import;
```

상수는 영문 대문자 snake case 를 사용한다

```
SYMBOLIC_CONSTANTS;
```

생성자는 대문자 camel case 를 사용한다

```
class ConstructorName {  
    ...  
};
```

변수, 함수에는 camel case 를 사용한다

```
// 숫자, 문자, 불린  
let dog;  
let variableName;  
  
// 배열 - 배열은 복수형 이름을 사용  
const dogs = [];  
  
// 정규표현식 - 정규표현식은 'r'로 시작  
const rDesc = /.*/;  
  
// 함수  
function getName() {  
    ...  
}  
  
// 이벤트 핸들러 - 이벤트 핸들러는 'on'으로 시작  
const onClick = () => {};  
const onKeyDown = () => {};  
  
// 불린 반환 함수 - 반환 값이 불린인 함수는 'is'로 시작  
let isAvailable = false;
```

Private 변수명은 '\_'로 시작한다

```
let _privateVariableName;
let _privateFunctionName;

// 객체일 경우
const customObjectName = {};
customObjectName.propertyName;
customObjectName._privatePropertyName;
_privateCustomObjectName;
_privateCustomObjectName._privatePropertyName;
```

## 4. 선언과 할당

변수 : 값이 변하지 않는 상수는 const, 값이 변하는 변수는 let 로 선언한다

const 를 let 보다 위에 선언한다 (코드가 정리되어 가독성이 좋아진다)

```
// Bad - 그룹화 없음
let foo;
let i = 0;
const len = this._array.length;
let bar;

// Good
const len = this._array.length;
const len2 = this._array2.length;
let i = 0;
let j = 0;
let foo, bar;
```

const 와 let 은 사용시점에 선언 및 할당을 한다

```
// Bad - 블록 스코프 밖에서 변수 선언
function foo() {
  const len = this._array.length;
  let i = 0;
  let j = 0;
  let len2, item;

  for (; i < len; i += 1) {
    ...
  }

  len2 = this._array2.length;
  for (j = 0, len2 = this._array2.length; j < len2; j += 1) {
    item = this._array2[j];
    ...
  }
}

// Good
function foo() {
  const len = this._array.length;
  for (let i = 0; i < len; i += 1) {
    ...
  }

  // 사용 시점에 선언 및 할당
  const len2 = this._array2.length;
  for (let j = 0; j < len2; j += 1) {
    const item = this._array2[j];
    ...
  }
}
```

선언과 할당의 분리를 허용하는 경우 선언만 하는 변수는 var 을 한 번만 사용하는 방식을 허용한다 (ES5)

```
// Bad - 불필요하게 개행
var foo,
    bar,
    quux;

// Good - 선언만 하는 변수, 한 줄로 연결
var foo, bar, quux;

// Good
var foo;
var bar;
var quux;
```

## 5. 공백

키워드, 연산자와 다른 코드 사이에 공백이 있어야 한다

```
// Bad
var value;
if(typeof str==='string') {
    value=(a+b);
}

// Good
var value;
if (typeof str === 'string') {
    value = (a + b);
}
```