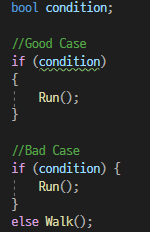
검성 라인하르트 반 아스트레아 팀

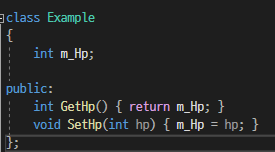
**코딩 가이드라인**(Ver.2021/1/18)가로선

게임인재원 프로그래밍학과 2기 주혜린, 이우진, 강성민 **작성자: 주혜린**

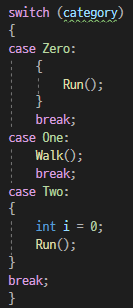
1. **Bracing(중괄호)**
   1. 여는 중괄호와 닫는 중괄호( { , } )는 무조건 라인의 처음에 위치한다.
   2. 문장(Statement)가 단 하나만 있더라도 중괄호를 사용한다.



* 1. 예외 규칙
     1. 변수의 Get/Set 함수 정의 시 한 라인으로 처리한다.



* + 1. Switch/Case문 사용시 break; 를 제외하고 중괄호로 묶는다. 만약 case에 하나의 문장만 있다면 중괄호를 생략 가능하다.



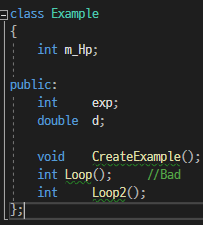
1. **들여쓰기**
   1. 함수의 패러미터 사이에는 띄어쓰기를 한다.



* 1. 비교연산자 전후로 띄어쓰기를 한다.



* 1. 변수 선언 시 지역변수는 타입 뒤에 스페이스바 1번, 멤버 변수와 함수는 탭으로 처리하여 가독성을 높인다. (자동 정렬 시 전부 되돌아가므로 주의)



1. **If 조건문**
   1. If 조건문에서 bool 변수를 사용할 경우, true/false와 명확하게 비교한다.



* 1. 조건문 안에서 다시 재할당을 하지 않는다.



1. **Name(명명 규칙)**
   1. 변수 이름은 Small Camel Case로 짓는다. 포인터 변수의 경우 p를 붙인다.



* 1. 함수 이름은 Pascal Case로 선언하며, 동사를 앞에 쓴다.



* 1. 상수는 Screaming Snake Case로 짓는다.



* 1. 클래스의 멤버 변수는 m\_ Pascal Case를 사용한다.



* 1. 함수의 패러미터를 선언하는 경우 앞에 \_를 붙이는 것을 허용한다.



* 1. Generic 클래스(template) 사용 시 Generic 파라미터는 T로 통일한다(복수 타입 제외).



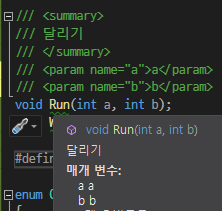
* 1. Bool 타입 변수 선언 시 앞에 is, has 등의 접두어를 붙여 의미를 명확하게 한다.



* 1. 함수 이름을 축약하지 않는다. 자주 사용하는 단어의 경우 축약 가능하다(information -> info)



1. **소스 파일**
   1. 하나의 소스 파일은 하나의 타입(Class, Interface 등)만 가질 수 있다.
   2. 클래스 내부에서만 사용하는 Enum class나 struct의 경우 같은 h 파일 맨 위에 선언 및 정의한다.
2. **주석**
   1. Public 멤버 함수의 경우 정의 부분에 /// 을 사용하여 디폴트 값을 사용하여 Help로 사용한다.



* 1. 내부에서 주석 사용시 // 를 사용한다. 되도록이면 코드 라인 뒤에 적는 것이 아니라 바로 위의 라인에 적는다. 만약, 변수 선언과 같은 간단한 코드는 주석을 뒤에 적을 수 있다.

