**변수**

1. 모든 변수는 소문자로 시작한다.
2. 모든 변수는 명사화 시킨다.
3. 두 개의 단어가 합쳐질 경우 구분을 위해 두 번째 단어의 첫 글자는 대문자로 한다.

Ex)stringhouseName

**함수**

1. 모든 함수는 대문자로 시작한다.
2. 모든 함수는 동사화 시킨다.
3. 두 개의 단어가 합쳐질 경우 구분을 위해 두 번째 단어의 첫 글자는 대문자로 시작한다. Ex)voidboolIsEmpty()
4. 인자의 경우 위의 “변수명” 규칙을 따른다.
5. 인자가 두 개 이상일 경우 “,”는 앞의 인자에 붙여쓰고 한 칸 띄어 두 번째 인자를 쓰도록 한다. Ex)voidabc(inttemp1,inttemp2)
6. 함수의 여는 중괄호는 한 칸 띄어쓰기를 한 후 같은 라인에 쓴다. Ex)voidabc(){
7. 가능하면 하나의 함수는 하나의 기능을 하도록 코드를 작성한다.

**상수화**

1. 의미있는 숫자의 경우 상수화를 시킨다.
2. 상수화 된 변수는 모두 대문자로 쓴다. Ex)constintPI
3. 두 단어가 합쳐질 경우 구분을 위해 단어 사이에 “\_”를 쓴다. Ex)constintMUSIC\_NOTE\_NUMBER

**조건문, 반복문**

1. 조건의 경우 가독성을 위해 한 칸 띄어쓰기를 한다. Ex)if(number>10)
2. 조건문, 반복문 내의 실행문이 한 줄일 경우 중괄호를 쓰지 않는다.
3. 조건문, 반복문 내의 실행문이 두 줄 이상일 경우 중괄호를 쓴다. 이 때, 여는 중괄호는 조건문, 반복문 옆에 한 칸을 띄어 쓰도록 한다. Ex)if(number>10){
4. Else if, else 또한 if와 규칙이 같다.
5. 조건문, 반복문 내의 조건이 복잡할 경우 조건의 함수화를 시킨다. 이 때, 함수명은 어떠한 조건에 대한 것인지 충분히 유추할 수 있는 것이어야 한다.
6. 조건에서 괄호가 연속으로 두 번 이상 나올 경우 가장 바깥 괄호와는 붙여쓰고 연산자를 이용하여 구분 할 경우 연산자 앞 뒤로 한 칸씩 띄워준다. Ex)while((abc==1)&&(def==2)){

**공통**

1. 모든 들여쓰기는 스페이스바가 아닌 탭으로 구분 짓는다.