**1. 변수**

* **선언(Declaration)과 대입(Assignment)**
* 선언: 변수 및 상수의 자료형과 이름을 컴퓨터에게 알려줌
* 대입: 변수 및 상수의 실제 값을 컴퓨터에게 알려줌  
  - 정의(Definition)이라고도 함
* 선언과 대입을 동시에 할 수도 있다.
* 변수는 “변할 수 있는 값”을 의미함  
  - 문자나 문자열까지 변수에 포함함  
  - 상수(const) 또한 변수의 범위에 포함함
* **변수를 사용하는 이유**
* 어딘 가에 저장한 값을 다시 재사용하기 위해서
* 상태(속성, 데이터)의 변화를 기억하기 위해서  
  - 그래서 변수를 ‘mutable’이라고도 부름  
  - 그래서 상수를 ‘Immutable’이라고도 부름
* 매직 넘버에 명확한 이름으로 의미를 부여하여 모호성을 없앰

**2. 변수명에 관한 규칙**

* **변수의 이름은 의미 있게 지어야 함**
* 변수가 무엇을 의미하는지 명확하게 지어야 함
* 변수명에서 데이터가 무엇을 의미하는지 알 수 있어야 함
* **코딩 표준(스탠다드): 변수명은 명확하게 지을 것**
* 변수는 동작이 아닌 어떤 상태를 저장하는 것, 따라서 동사가 아닌 명사로 짓는 것이 보통임
* 명사로 정확하게 어떤 정보를 담는지 알려주는 단어를 사용  
  - 성적이라면 ‘score’  
  - 이름이라면 ‘name’
* 여러 명사가 들어간다면 두 번째 이상의 단어의 첫 글자는 대문자로 사용  
  - 수학 성적이면 ‘math**S**core’  
  - 사과 개수라면 ‘apple**C**ount’
* 즉, 변수의 작명은 Camel Casing 규칙을 따름
* **코딩 표준(스탠다드): 상수(CONST)**
* 상수의 경우, 명사를 사용하여 **대문자**로 작성함  
  ex) SCORE, NUMBER
* 여러 명사가 들어간다면 명사 사이에 언더바(\_) 기호를 사용함  
  ex) MATH\_SCORE, APPLE\_COUNT

**3. 기본 자료형(Primitive Types)**

* **기본 자료형**
* 컴퓨터가 이해할 수 있는 가장 자연스러운 데이터  
  - 전부 다 숫자로 이루어져 있음  
  - 디지털로 이루어져 있으므로 0과 1로 구성됨
* **자료의 크기**
* 최소 단위는 비트(bit)  
  - 1 혹은 0만 담을 수 있음  
  - 비트는 하나의 1 또는 0만 담을 수 있음
* 바이트(byte) = 8비트(bit)  
  - 각 비트는 마찬가지로 하나의 0 또는 1만 가질 수 있음
* 1024Byte = 1KB
* 1024KB = 1MB
* **정수형(Integer Types)**
* 0, 1, 2와 같은 정수를 담을 수 있는 자료형
* byte, short, int, long  
  - byte = 8비트  
  - short = 16비트  
  - int = 32비트  
  - long = 64비트
* 프로그래밍 언어마다 담을 수 있는 자료의 크기 차이는 존재
* **부동소수점형(Floating-Point Types)**
* 3.14와 같은 실수를 담을 수 있는 자료형
* float, double  
  - float = 32비트  
  - double = 64비트
* 프로그래밍 언어마다 담을 수 있는 자료의 크기 차이는 존재
* **자료의 크기**
* 최소 단위는 비트(bit)  
  - 1 혹은 0만 담을 수 있음  
  - 비트는 하나의 1 또는 0만 담을 수 있음

**4. 주석(comment)**

* **코드와 관련된 내용(설명)을 메모할 때 사용**
* **컴파일할 때는 무시되는 문장**
* **오직 사람을 위해서만 사용됨**
* **주석을 지나치게 많이 사용하지 말 것**
* 주석은 코드와 마찬가지로 계속 관리해야 함
* 주석이 너무 많으면 업데이트를 못하는 주석들이 생길 수 있음
* 너무 많으면 오히려 다른 프로그래머들이 코드를 읽는데 불편함
* 반드시, 꼭 필요한 내용만 주석으로 남기는 것이 좋음
* 주석이 없으면 없을수록 좋은 코드임  
  - 코드만으로 다른 사람들이 이해할 수 있다는 것이므로  
  - 코드 자체로 이해가 잘 되는 코드를 작성해야 함
* 소스 코드를 주석 처리하여 남겨놓는 행위는 지양해야 함  
  - ‘혹시라도 필요할지도 모른다’는 생각은 버전 관리 프로그램으로

**5. 상수**

* **상수(constant)**
* 변하지 않는 수(데이터)
* ‘리터럴’이라고도 함, 이 표현이 맞음
* 숫자 뿐만 아니라 문자까지 포함함
* 리터럴, 상수, 매직넘버 등으로 혼용하여 사용하기도 함
* **상수의 별칭 – 매직 넘버(Magic Number)**
* 왜 상수를 매직 넘버라 부르는가?  
  - 마치 마법처럼 (목적이) 무엇인지 전혀 알 수 없기 때문  
  - 따라서 매직 넘버는 되도록이면 사용하지 말 것  
   - 상수를 다른 값으로 바꾸기 위해서 수동으로 직접 그 값을 수정해야 함  
   - 이 경우 변수를 사용하여 관리하는 편이 훨씬 수월하게 작업을 진행할 수 있음