

1주차

절차지향언어

절차지향 프로그래밍에 쓰이는 언어

순서대로 흐름에 따라 코드를 작성

절차지향언어 → C언어

장점

처리속도 빠름

단점

1. 비효율적
2. 유지보수 어려움 (하나가 고장나면 시스템 전체가 고장 → 전체 수리)
3. 엄격하게 순서가 정해져 있어서 비효율적

객체지향 언어

프로그램을 다수의 객체로 만들고 서로 상호작용할 수 있도록 하는 프로그래밍 언어

객체지향언어 → Java C++ Python

특징 (캡슐화, 상속, 다형성, 추상화)

장점

1. 재사용성
2. 생산성 증가 (독립적인 객체로 이루어짐 → 유지 및 보수 용이)
문제가 되는 부분만 고치면 다시 정상 작동가능
3. 자연스러운 모델링

단점

1. 상대적으로 느림
2. 난이도 어려움

자바 프로그래밍 구조

소스파일 - 클래스 - 메소드 - 실행문

1. 소스파일

확장명 .java

public class 이름과 같아야 함

자바는 중괄호 블록 형태

2. 클래스

프로그램을 개발하는 단위

유사한 특징을 지닌 객체들의 속성을 묶어놓은 집합체

클래스 이름은 대문자로 시작하는게 관례

3. 메소드

수행할 작업을 나열한 코드의 모음

자바 어플리케이션은 main으로 실행

3-1 메소드 예시

- 클래스 텔레비전
- 메소드 음량 조절, 채널 변경, 전원 온오프 등 기능적인 것들의 모음

3-2 실행문

작업을 지시하는 변수 선언, 값 저장, 메소드 호출 등

```
System.out.print()  
// 출력문  
System.out.println()  
// 출력하고 줄바꿈  
System.out.printf()  
// 지시자를 사용한 출력문
```

자바 주석

```
// 주석문  
  
/*  
   여러 줄 주석  
*/
```

변수와 상수

1. 변수(Variable)

계속 변하는 값을 저장할 수 있는 공간

```
int a; // 변수 선언  
a = 10; // 변수 할당  
  
int b = 10; // 한 줄에 선언과 초기화
```

2. 상수 (Constant)

변하지 않는 항상 같은 값을 가지는 수

```
final int c; // 상수 선언
```

자료형 (Data Type)

변수를 선언할 때 자료형을 알려주어야 한다

정수형

byte

$-2^7 \sim 2^7-1$ (-128 ~ 127)

short

$-2^{15} \sim 2^{15}-1$ (-32768 ~ 32767)

int

$-2^{31} \sim 2^{31}-1$ (-2147483648 ~ 2147483647)

long

$-2^{63} \sim 2^{63}-1$ (-9223372036854775808 ~ 9223372036854775807)

자료형을 사용할 때 L을 붙여준다

ex)

`long a = 300000L;`

실수형

float

32bits

double

64bits

문자형

char

16bits

논리형

boolean

1bit

true, false

문자열

String

참조 자료형인 `String` 은 변수가 값을 직접 가리키지 않고, 객체를 가리킨다.

`char` 의 참조 자료형이기 때문에 배열으로써 한 글자씩 데이터를 관리할 수 있습니다.