

불린(Boolean)

char

unsigned short

unsigned int

unsigned long

complex

unsigned char의 기억범위

변수

예약어

변수명 작성 규칙

외부 변수(External Variable)

레지스터 변수(Register Variable)

4Byte 단정도 실수 3.141592 초기값

변수 value\_Pi 선언

```
printf("%3.2s", input_Value);
```

```
printf("8.6s", "help me");
```

```
Scanner scan = new Scanner(System.in);
```

```
int a = scan.next();
```

```
int a = scan.nextLine();
```

```
int a = scan.nextInt();
```

```
%o
```

```
%x
```

```
printf("%f",245.2555);
```

```
printf("%e",25.43);
```

printf("%3s","help me");

4

024

0x24

- → << → & → |

5

16진수 d

scan.close();

7

sizeof(int)

8

strlen()

11

find(값)

13

C언어의 대표적인 표준 라이브러리

stdio.h

stdlib.h

Java의 대표적인 표준 라이브러리

java.util

Python의 대표적인 표준 라이브러리

random

14

절차적 프로그래밍 언어

절차적 프로그래밍 언어의 장점

절차적 프로그래밍 언어의 단점

종류

C
ALGOL
COBOL
FORTRAN
15

객체지향 프로그래밍 언어

객체지향 프로그래밍 언어의 장점

객체지향 프로그래밍 언어의 단점

종류
Java
C++
Smalltalk
특징

캡슐화(Encapsulation)

정보 은닉(Informaion Hiding)

추상화(Abstraction)

상속성(Inheritance)

다형성(Polymorphism)

조건의 참, 거짓 여부를 판단하여 저장할 때 사용  
숫자 1과 0만을 이용하여 참과 거짓을 표현  
기본값은 거짓  
1Byte의 크기를 가짐  
Java에서 사용되는 문자형의 데이터 타입  
크기는 2Byte  
한 문자를 저장할 때 사용  
C 또는 C++언어에서 사용하는 자료형  
정수형 자료형에서 음수를 표현하지 않음  
양수의 표현 범위를 2배로 증가  
Python에서 기본적으로 지원  
C, Java 언어에서는 외부 라이브러리를 통해서만 사용할 수 있는 자료형  
크기는 16Byte  
실수와 허수의 합으로 이루어진 숫자 표현을 저장  
0~255

컴퓨터가 명령을 처리하는 도중 발생하는 **값을 저장하기 위한 공간**  
**변할 수 있는 값**

정해진 기능을 수행하도록 이미 용도가 정해져 있는 단어  
변수 이름 등 다른 용도로 활용할 수 없음  
첫글자는 영문자나, \_로 시작해야 하며 숫자는 올 수 없음  
공백이나 \*,+,-,/ 등의 **특수문자를 사용할 수 없음**  
현재 파일이나 다른 파일에서 선언된 변수나 함수를 참조하기 위한 변수  
함수 밖에서 선언되며 함수가 종료된 뒤에도 값이 소멸되지 않음  
다른 파일에서 참조할 경우 초기화 못함  
초기화하지 않으면 자동으로 0으로 초기화 됨  
CPU 내부에 기억영역을 할당받음  
처리 속도가 높음  
변수를 사용하는 함수나 블록이 종료되면 자동으로 소멸하는 기억 클래스

```
float value_Pi = 3.141592f;
```

전체 3자리를 확보한 후 input\_value에 저장된 문자열을 오른쪽부터 2글자만 출력  
help m

Scanner 클래스의 객체 변수 scan을 키보드로 입력받을 수 있도록 생성  
입력값(공백까지의 모든 문자열)을 문자열로 반환  
입력받은 라인 전체(엔터키 전)를 문자열로 반환  
입력값을 정수형으로 반환  
정수형 8진수를 입•출력하기 위해 지정  
정수형 16진수를 입•출력하기 위해 지정  
245.255500  
2.54000e+01

help me

Java, C언어

8진수는 숫자 앞에 0를 붙임

Python 8진수는 숫자 앞에 0o를 붙임

Java, C언어, Python

16진수는 숫자 앞에 0x을 붙임

연산자 우선순위

13

프로그램 종료 전에 close()메소드를 이용하여 사용하던 메모리 영역을 해제

4Byte

널 문자를 제외한 순수한 문자열의 길이만 구함

문자열에서 처음 검색되는 '값'의 위치를 반환

찾지 못한 경우 -1을 반환

모든 문자열 메소드는 대•소문자 구분

데이터 입•출력에 사용되는 기능들을 제공

주요 함수: printf, scanf, fprintf, fscanf, fclose, fopen 등

자료형 변환, 난수 발생, 메모리 할당에 사용되는 기능들을 제공

주요 함수: atoi, atof, srand, rand, malloc, free 등

날짜 처리, 난수 발생, 복잡한 문자열 처리 등에 관련된 기능을 제공

주요 클래스: Date, Calendar, Random, StringTokenizer 등

무작위 선택을 위한 기능을 제공

주요 메소드: choice(), sample(), random(), randrange() 등

일련의 처리 절차를 정해진 문법에 따라 순서대로 기술해 나가는 언어

실행 속도가 빠름

같은 코드를 복사하지 않고 다른 위치에서 호출하여 사용할 수 있음

모듈 구성이 용이함

구조적인 프로그래밍 가능

프로그램 분석 어려움

유지 보수나 코드의 수정이 어려움

1972년 미국 벨 연구소의 데니스 리치에 의해 개발 시스템 소프트웨어를 개발하기 편리하여 시스템 프로그래밍 언어로 널리 사용됨 자료의 주소를 조작할 수 있는 포인터 제공 고급 프로그래밍 언어이면서 저급 프로그램 언어의 특징을 모두 갖추 UNIX의 일부가 이 언어로 구현됨
수치 계산이나 논리 연산을 위한 과학 기술 계산용 언어 PASCAL과 C언어의 모체가 됨
사무 처리용 언어 영어 문장 형식으로 구성되어 있어 이해와 사용이 쉬움 4개의 DIVISION으로 구성
과학 기술 계산용 언어 수학과 공학 분야의 공식이나 수식과 같은 형태로 프로그래밍 가능

**객체들을 조립해서 프로그램을 작성**할 수 있도록 한 프로그래밍 기법

상속을 통한 재사용과 시스템의 확장이 용이  
코드의 재활용성이 높음  
자연적인 모델링에 의해 분석과 설계를 쉽고 효율적으로 할 수 있음  
사용자와 개발자 사이의 이해를 쉽게 해줌  
대형 프로그램의 작성이 용이  
소프트웨어 개발 및 유지보수가 용이  
프로그래밍 구현을 지원해 주는 정형화된 분석 및 설계 방법이 없음  
구현 시 처리 시간이 지연됨

분산 네트워크 환경에 적용이 가능 멀티스레드 기능을 제공하므로 여러 작업을 동시에 처리 가능 운영체제 및 하드웨어에 독립적이며 이식성이 강함 캡슐화가 가능하고 재사용성이 높음
C언어에 객체지향 개념을 적용한 언어 모든 문제를 객체로 모델링하여 표현
1세대 객체지향 프로그래밍 언어 중 하나로 순수한 객체지향 프로그래밍 언어 최초로 GUI를 제공한 언어

데이터(속성)와 데이터를 처리하는 함수를 하나로 묶은 것  
캡슐화된 객체의 세부 내용이 외부에 은폐되어, 변경이 발생할 때 오류의 파급 효과가 적음  
캡슐화된 객체들은 재사용이 용이

캡슐화에서 가장 중요한 개념  
다른 객체에서 자신의 정보를 숨기고 자신의 연산만을 통하여 접근을 허용

불필요한 부분을 생략하고 객체의 속성 중 가장 중요한 것에만 중점을 두어 개략화하는 것(모델화)  
데이터의 공통된 성질을 추출하여 슈퍼 클래스를 선정하는 개념

이미 정의된 상위 클래스의 모든 속성과 연산을 하위 클래스가 물려받는 것

상속성을 이용하면 하위 클래스는 상위 클래스의 모든 속성과 연산을 자신의 클래스 내에서 다시 정의하지 않고서도 즉시 사용가능

메시지에 의해 객체가 연산을 수행하게 될 때 하나의 메시지에 대해 각 객체가 가지고 있는 고유한 방법으로 응답할 수 있는 능력

객체들은 동일한 메소드명을 사용하며 같은 의미의 응답을 함