

1. 객체지향언어

1) 객체지향 언어 정의

(1) 객체

- 우리가 주변에서 볼 수 있는 물체나 논리적으로 생각 가능한 그 어떤 것을 의미함
- 객체지향 프로그래밍이란 객체를 중심으로 하여 프로그램을 작성하는 방법을 말함

(2) 객체의 구성

- 객체의 상태: 객체의 현재 상황과 가능한 모든 상태임(객체 → 볼펜)
- 객체가 할 수 있는 일(볼펜이 할 수 있는 일 → 쓰기)
- 객체의 상태, 할 수 있는 일 (멤버 변수, 속성/멤버 함수, 메소드 등으로 표현함)

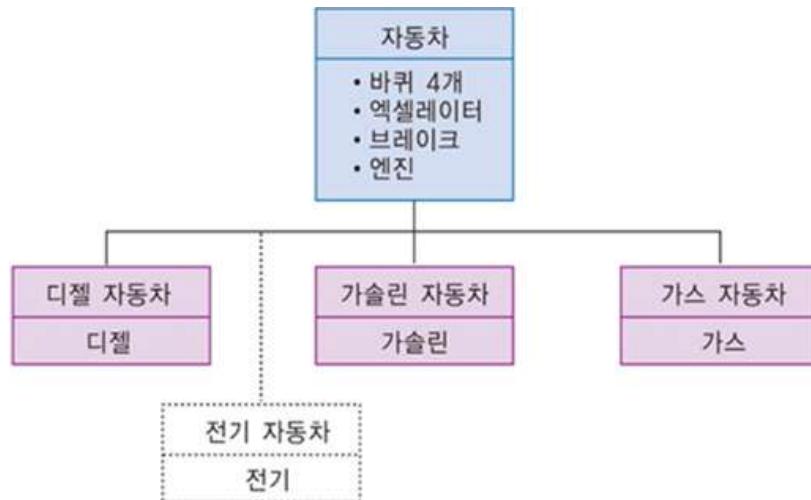
(3) 클래스의 특징

- 캡슐화(Encapsulation): 외부에서 알 필요가 없는 정보는 감추어 하나의 캡슐 상태로 유지하여 외부에서는 함부로 접근하거나 변경 등이 불가능 함
- 다형성(Polymorphism): 외부에서는 하나의 작업처럼 보이지만 내부에서는 각기 다른 방법으로 구현 되는 것임

예: '+'의 경우

숫자 계산	두 수를 더하는 작업
문자 계산	주어진 문자를 결합하는 작업
검색	논리적 AND 의미

- 상속성(Inheritance): 하나의 클래스가 정의 되고, 그 이후에 어떤 클래스라도 이 클래스의 정의 중에서 하나 이상의 정의를 물려 받을 수 있는 성질
→ 같은 기능을 하는 클래스는 다시 작성할 필요 없이 계속 물려 받아 쓸 수 있고, 결과적으로 빠른 프로그램 개발을 보장함



2) Java

- 1994년 선(SUN)사의 가전제품을 제어하기 위한 언어 개발을 시작한 것이 그 시초임
- 가전제품을 목적으로 만들어져 낮은 시스템에서도 운영이 가능하도록 설계됨

- 운영 체제나 중앙처리 장치에 관계없이 모든 플랫폼에서 사용 가능함
- 자바와 플랫폼 사이에 자바 가상 머신이 인터페이스 역할을 수행함

- ▶ Java 언어의 특징
 - 객체지향 언어임
 - C와 C++과 비슷한 모양(문법)을 가져 사용이 쉬움
 - 객체지향의 다형성을 위해 실행 시간에 함수 호출을 결정할 수 있음
 - 동적이고 편리한 메모리 관리를 지원하며 자동으로 메모리를 수거하여(garbage collection) 효율성을 높임

3) 절차지향 언어

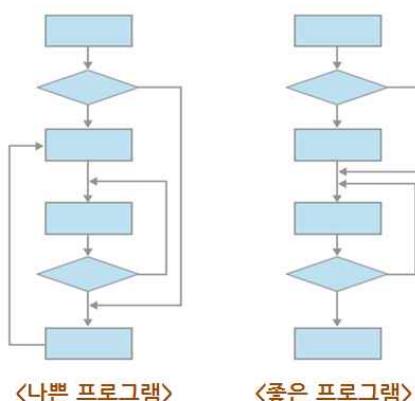
- 프로그램 코드가 순서대로 실행되는 언어
- 파스칼, 코볼, 포트란, 베이직, C언어 등

냉장고에 소고기를 넣는 과정	냉장고에 소고기를 넣는 프로그램
1. 냉장고 문을 연다.	1. open 냉장고
2. 소고기를 넣는다.	2. insert 소고기
3. 냉장고 문을 닫는다.	3. close 냉장고

4) 구조적 프로그래밍

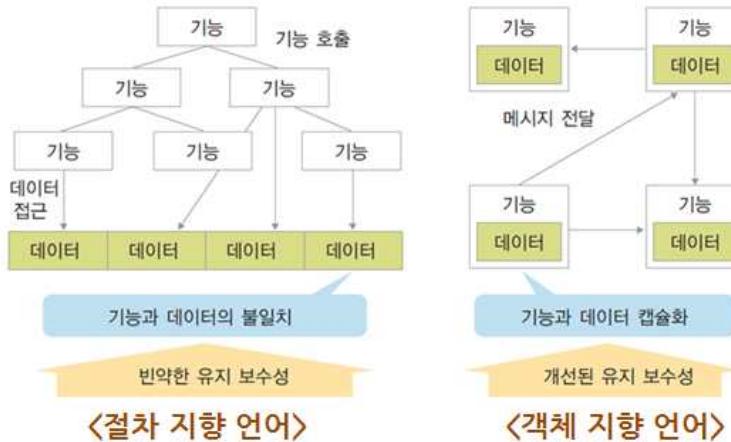
- 절차 지향 언어는 명령을 실행하는 순서가 자주 바뀌거나 복잡해지면 운영 및 유지 비용이 많이 들
- 이러한 문제를 해결하기 위해 등장한 개념이 구조적 프로그래밍임
- 구조적 프로그래밍은 말 그대로 명령어 실행 순서가 뒤바뀌거나 제어 구조가 복잡해지지 않도록 몇 가지 규칙에 따라 프로그램을 작성
- 구조가 단순해질수록 제3자가 쉽게 이해하고 수정할 수 있어 프로그램을 체계적으로 관리할 수 있음
- goto문의 무분별한 분기 구조를 개선하고 모든 명령문의 처리를 블록으로 모듈화시키기 위해 등장

<구조적 프로그래밍의 예>



5) 절차지향 언어와 객체지향 언어의 차이점

절차 지향 언어	데이터와 데이터를 처리하는 기능이 별도로 관리
객체 지향 언어	데이터와 기능을 묶어 캡슐화시킨 후 메시지를 전달하여 일을 처리



2. 유닉스/리눅스 기반 언어

1) UNIX 기반의 언어

ProC	- C언어의 UNIX 확장판임 - C언어를 사용한 UNIX의 데이터베이스 관리에 많이 사용함
C	- UNIX의 가장 중심적 언어 - UNIX 자체를 C 언어로 제작하여 UNIX 프로그래밍에 필수임
Shell 프로그래밍	- 여러 가지 Shell 명령어를 순차적으로 수행함 - Shell : 쉘은 UNIX에서 대화형 사용자 인터페이스를 부르는 용어로, 사용자가 입력하는 명령어를 이해하고, 실행하는 역할을 수행함

2) Linux 기반의 언어

Perl	- 인터넷의 등장과 함께 인기를 누린 스크립트 언어임 - 초보자도 배우기 쉽고 객체지향적인 특징을 지원함
Python	- 가장 최근에 등장한 강력하고 배우기 쉬운 언어임 - 쉬운 문자열 제어와 객체지향적 특성을 제공함 - C언어와 연계성을 제공하며, Windows에서도 사용 가능함
PHP	- 인터넷에서 사용하는 대표적인 스크립트 언어임 - 컴퓨터에서 실행하지 않고 웹 서버에서 실행되는 대표적인 Server Side Script 언어임
Java	- 운영체제에 구애 받지 않는 Java 언어도 지원함
Lisp	- UNIX/Linux의 경우 Emacs라는 에디터의 제어 언어로 사용함
FORTRAN	- UNIX/Linux버전에 따라 F77,F90, F95등 다양한 버전을 지원함
C/C++	- UNIX와 마찬가지로 Linux에서도 지원함
Tcl	- X- Window 프로그래밍에 많이 쓰이는 스크립트 계열의 언어임

3. 웹 및 닷넷 기반 언어

1) 웹 기반 언어

(1) HTML

- Hypertext Markup Language: 하이퍼텍스트 마크업 언어임
- SGML에서 비롯된 하이퍼텍스트를 표기하는 언어임
- 일반적으로 인터넷에서 웹 페이지를 표시하기 위해 쓰임
- 문법은 HTML 태그(tag)로 구성됨

(2) XML

- Extensible Markup Language: 확장가능 마크업 언어임
- 다른 특수 목적의 마크업 언어를 만드는 용도로 권장되는 다목적 마크업 언어임
- 주로 다른 시스템, 특히 인터넷에 연결된 시스템들끼리 데이터를 쉽게 주고받을 수 있게 함

(3) JSP

- Java Server Pages: 자바 서버 페이지임
- HTML 내에 자바 코드를 삽입하여 웹 서버에서 동적으로 웹 페이지를 생성하여 웹 브라우저에 돌려주는 언어임
- 서블릿(Java Servlet)과는 달리 HTML 표준에 따라 작성되므로 웹 디자인하기가 상당히 편리함

(4) ASP

- Active Server Page: 액티브 서버 페이지임
- 마이크로소프트사에서 개발한 서버 측의 웹 스크립트 환경임
- Visual Basic에 바탕을 두고 있어 마이크로소프트 데이터베이스의 접근이 매우 편리함

2) 닷넷 기반 언어

(1) 닷넷 플랫폼과 닷넷 프레임워크

- 최근 들어 C#과 닷넷(.NET)이라는 새로운 패러다임의 프로그램 환경이 전개되고 있음
- 닷넷 플랫폼은 Windows 2000을 포함하는 기존의 윈도우 운영체제 서비스와 새로운 API(Application Programming Interface)를 제공하는 개발 프레임워크임
- 마이크로소프트사는 2000년 7월에 C# 언어와 함께 닷넷 플랫폼을 발표
- 닷넷 플랫폼은 마이크로소프트사의 수많은 신기술을 포함하여 활용하고 있으며, 마이크로소프트사는 닷넷 플랫폼의 개발과 정착을 위해 거액의 연구비를 집중적으로 투입하고 있음

(2) C#

- C#은 닷넷 환경의 응용 프로그램을 개발하기 위한 프로그래밍 언어로서 간결성, 안전성,

현대성, 객체지향, 인터넷 중심이라는 목표에 맞춰 설계됨

- C#은 2000년도에 마이크로소프트사의 엔지니어인 앤더스 헤일스버그와 스콧 월타우스를 중심으로 개발되었는데, C, C++, Java, Visual Basic 등의 언어로부터 많은 영향을 받음
- C#은 C의 빠른 속도, C++의 객체지향 구조, Java의 보안, Visual Basic의 빠른 개발 시간 등 기존 언어의 장점들을 바탕으로 설계되었고, 컴포넌트 기반의 다중 웹 응용 프로그램 개발에 매우 유용한 언어로 여겨지고 있음

▶ C#의 주요 특징 요약

- C#을 이용하면 새로운 클래스와 메소드(method), 프로퍼티(property)를 정의하는 것뿐만 아니라 객체지향 프로그래밍의 근간이라고 할 수 있는 캡슐화, 상속성, 다형성을 구현할 수 있음
- C#에서는 XML 스타일의 문서화 방식을 지원하여 응용 프로그램에 대한 참고문서를 쉽게 작성할 수 있게 도와줌
- C#은 클래스가 지켜야 하는 속성이나 메소드의 세부사항을 정의하는 규약인 인터페이스(interface)를 지원함
- C#은 구조체를 지원하고 컴포넌트 기반의 기술을 지원함
- C#은 C++ 스타일의 포인터(pointer)를 이용하여 직접 메모리에 접근하는 것을 허용함