



# 전공진로탐색 (문화융합지식학부)

개발자 이해하기: 개발과 개발자 용어 알기

동덕여자대학교  
데이터사이언스 전공  
권 범

# 목차

- ❖ 01. 개발자는 대체 뭐하는 사람인가요?
- ❖ 02. 코딩과 프로그래밍, 앱과 프로그램
- ❖ 03. 프로그래밍 언어도 외국어인가요?

# 01. 개발자는 대체 뭐하는 사람인가요?

- 02. 코딩과 프로그래밍, 앱과 프로그램
- 03. 프로그래밍 언어도 외국어인가요?

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ 시작하기 전에 (1/2)

개발자는 어떤 일을 하는 사람이며,  
어떤 종류의 개발자가 있는지,  
개발 직군에 따라 어떤 공부를 해야 하는지 알아보자

### 핵심 키워드

웹 개발자

모바일 개발자

게임 개발자

응용 소프트웨어 개발자

임베디드 개발자

정보 보안 전문가

인공지능 개발자

데이터베이스 개발자

클라우드 개발자

데브옵스 엔지니어

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ 시작하기 전에 (2/2)

- 이미 개발자가 뭔지 알고 '개발자가 돼야겠다'라고 다짐한 사람도 있겠지만, 이 수업을 듣는 대부분의 솜솜이는 개발자가 뭐 하는 직업인지 잘 모르거나 개발자가 하는 일을 알고 싶어 하는 사람일 것임

먼저 개발자는 무엇을 하는 사람인지, 어떤 종류가 있는지,  
그리고 구체적으로 어떤 일을 하는지 알아보자



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ 개발자 vs 작가 (1/4)

- 개발을 어렵게 생각할 필요는 전혀 없음
- 쉽게 생각해 보면 **개발자**가 하는 일은 작가가 하는 일과 비슷함
- 무에서 유를 창조하고 누군가 이해할 수 있도록 무언가를 작성한다는 점도 비슷함
- 다만 개발은 컴퓨터가 이해할 수 있는 언어로,  
작가는 사람이 이해할 수 있는 언어로 작성한다는 점이 다름

개발자	작가
코딩	글 쓰기
프로그래밍	글 짓기
개발	책 쓰기

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ 개발자 vs 작가 (2/4)

- 작가는 소설을 쓰는 소설가, 영화나 드라마를 쓰는 시나리오 작가, 시를 쓰는 시인처럼 다양한 종류가 있음
- 마찬가지로 세상에 존재하는 소프트웨어의 종류도 다양하기에, 개발자도 무엇을 개발하느냐에 따라 여러 직군으로 나뉨

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ 개발자 vs 작가 (3/4)

- 소설가, 시나리오 작가, 시인이 '글을 쓰는 사람'이라는 부분에서는 같지만, 세부 영역 안으로 들어가면 전문적으로 쌓아야 하는 역량이 다른 것처럼, 개발자도 종류마다 사용하는 프로그래밍 언어도 다르고 쌓아야 하는 지식도 다름
- 한 분야만 집중해서 그 분야의 전문성을 갖추는 개발자도 있고, 소설과 시를 둘 다 집필하는 작가처럼 둘 이상의 분야를 병행해서 넓은 스펙트럼으로 개발 실력을 갖추는 개발자도 있음



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ 개발자 vs 작가 (4/4)

- 개발자의 종류를 알고 그 역할이 무엇인지 이해한다면, 개발자가 담당할 수 있는 일과 그렇지 않은 일은 무엇인지 파악하는 데 큰 도움이 됨

더 나아가 개발자를 준비하는 솜솜이도  
자신의 적성에 맞는 분야는 무엇이며,  
이를 위해서 어떤 준비를 하면 좋을지 알 수 있음

# 01. 개발자는 대체 뭐하는 사람인가요?

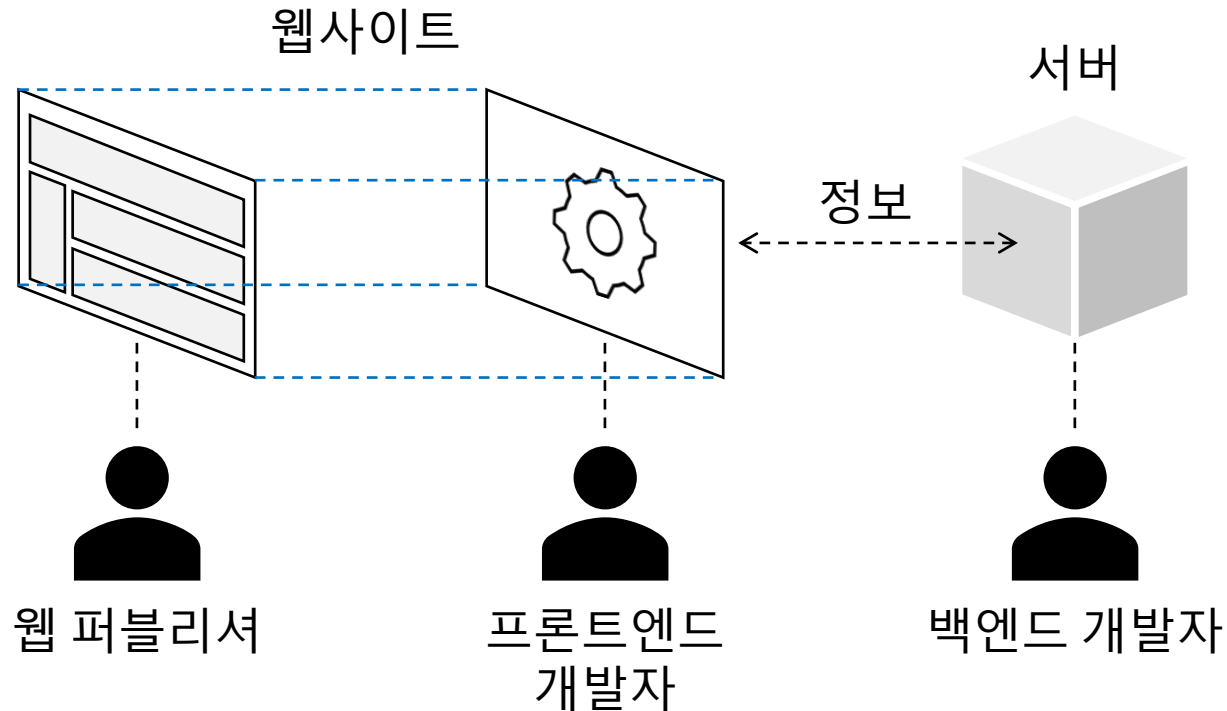
## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자 (1/2)

- 웹 개발자는 말 그대로 웹사이트를 개발하는 개발자임
- 우리가 PC나 스마트폰에서 웹 서핑을 하며 방문하는 각종 웹사이트는 웹 개발자가 만듦
- 일반 프로그램이나 모바일 앱으로 알고 사용하는 소프트웨어 중에서도 내부적으로는 웹사이트를 바탕으로 만들어진 프로그램이 많음

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자 (2/2)

- 물론 웹 개발은 포괄적인 용어임
- 웹사이트는 눈에 보이는 부분과 보이지 않는 부분으로 구성돼 있고, 각각 담당하는 개발자들이 협업해서 만들기 때문임
- 웹 개발자의 범주에 속하는 개발자는 다음과 같음



# 01. 개발자는 대체 뭐하는 사람인가요?

- ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자: 웹 퍼블리셔/UI 개발자
  - 웹 퍼블리셔는 웹사이트에서 보이는 부분을 담당하는 개발자임
  - 국내에서는 웹 퍼블리셔라는 단어를 많이 사용하고,  
해외에서는 보통 UI(User Interface) 개발자라고 부름
  - 주로 사용하는 언어는 HTML과 CSS이며, 자바스크립트를 사용하기도 함

이들은 웹사이트의 기능을 프로그래밍하기보다는  
웹사이트의 외적 요소를 코드로 구현하는 역할을 함

타 직군에 비해 진입 장벽이 낮고 디자인 분야와 밀접한 측면이 있어서  
웹 디자이너를 비롯한 비전공자들이  
코딩을 시작하는 첫 단계로 도전하는 분야이기도 함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자: 프론트엔드 개발자 (1/2)

- **프론트엔드 개발자**는 웹 퍼블리셔처럼 화면에 보이는 요소들을 만들 뿐만 아니라, 자바스크립트를 사용하여 그 요소들에 기능을 부여하는 일까지 담당함
- 이미지 슬라이드나 웹 채팅 창, 온라인 디자인 도구와 같은 다양한 기능이 프론트엔드 개발자의 손에서 만들어짐
- 이러한 프론트엔드 개발을 다른 말로 **클라이언트 개발**이라고 부르기도 함
- 모바일 앱을 개발하는 것도 사용자가 직접 보고 다루는 부분을 만드는 것이므로 클라이언트 개발에 해당함

이들은 사용자에게 보여 줄 웹사이트 화면을 만들고  
사용자의 클릭이나 드래그와 같은 동작에 따라  
웹사이트의 다양한 기능이 실행되도록 프로그래밍함

회사마다 웹 퍼블리셔와 프론트엔드 개발자를 따로 두기도 하고,  
디자이너 또는 프론트엔드 개발자가 웹 퍼블리셔 역할을 겸하기도 함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자: 프론트엔드 개발자 (2/2)

- 프론트엔드 개발자는 HTML과 CSS에 대한 지식도 필요하지만, 무엇보다도 **자바스크립트**를 집중적으로 공부해야 함
- 프론트엔드 분야의 프로그래밍은 기본적으로 자바스크립트를 사용해서 이루어지기 때문임
- 오늘날에는 자바스크립트의 단점을 보완한 언어인 타입 스크립트도 많이 사용되고 있음
- 이들에 대한 지식을 기본으로 갖춘 뒤 리액트 같은 라이브러리를 공부하면 프론트엔드 개발자로 한 걸음 더 나아갈 수 있음

### 라이브러리

- ✓ 라이브러리는 하나 이상의 프로그램에서 공통으로 가져다 쓸 수 있는 기능들을 따로 분리해 둔 것을 말함
- ✓ 기존에 생성한 라이브러리를 활용하면 프로그램을 개발할 때 모든 기능을 처음부터 다시 프로그래밍 할 필요 없어 시간과 노력을 절약할 수 있음

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자: 백엔드 개발자 (1/2)

- **백엔드 개발자**는 웹사이트에서 눈에 보이지 않는 요소를 개발함
- 프론트엔드 개발자가 만든 화면에 표시되는 데이터는 백엔드 개발자가 구현한 서버에서 처리됨
- 그렇기 때문에 이들을 **서버 개발자**라고도 부름
- 웹사이트가 아닌 모바일 앱의 서버를 다루는 개발자도 백엔드 개발자임

데이터베이스에 저장되는 각종 정보에 대한 입출력,  
데이터 분석과 가공, 로그인을 비롯한 보안 요소 관리 등이  
백엔드 개발자를 통해 이뤄짐

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자: 백엔드 개발자 (2/2)

- 백엔드 개발에 사용하는 프로그래밍 언어는 매우 다양함
- 전 세계적으로 자바스크립트, 파이썬, PHP, Java 등이 널리 사용됨
- 각 언어에는 백엔드 프로그램을 편리하게 만들 수 있게 도와주는 제작 도구들이 있는데, 이를 **프레임워크**라 부름
- 국내에서는 Java 언어와 스프링 프레임워크 조합을 많이 사용함
- 오늘날에는 예전에 비해 다양한 언어 프레임워크 조합이 더 다양해지고 있음

좋은 백엔드 개발자가 되기 위해서는 언어와 프레임워크를 다루는 실력 외에도 서버에 네트워크, 데이터베이스 등의 다양한 지식이 필요함



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ① 웹사이트를 만드는 개발자 – 웹 개발자: 풀스택 개발자

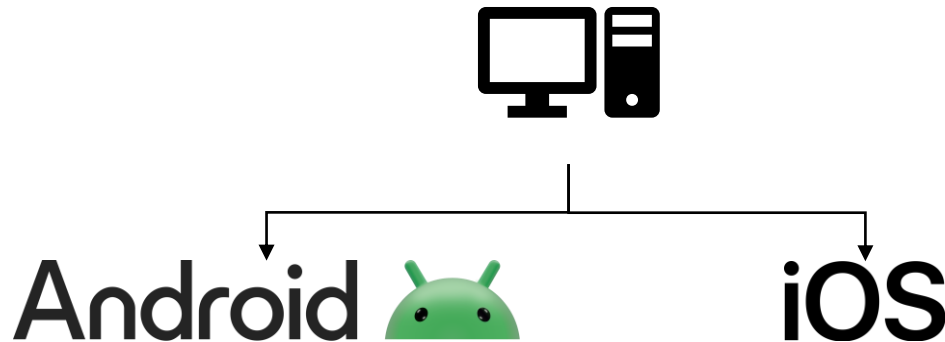
- 프론트엔드와 백엔드 관련 지식을 고루 갖추어 혼자서도 웹 서비스 전체를 개발할 줄 아는 개발자를 **풀스택 개발자**라고 부름
- 풀스택 개발자는 다수의 개발자가 서비스를 세분화해 전문적으로 분담하는 대기업보다는 소수 인원이 제품 전반을 만드는 소기업이나 스타트업에서 필요로 하는 개발자임

이처럼 웹 개발에 대한 폭넓은 역량과 경험을 쌓고  
깊이 있는 지식을 갖춘 풀스택 개발자가 되기 위해서는  
꾸준한 공부가 필요함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ② 손 안의 앱을 만드는 개발자 – 모바일 개발자 (1/4)

- **모바일 개발자**의 역사는 다른 개발자들에 비해 길지 않지만, 스마트폰이 대중화된 오늘날에는 많은 개발자가 이 직군에서 일하고 있음
- 모바일 개발자들은 명칭에서 알 수 있듯 스마트폰에서 동작하는 애플리케이션을 개발하며, **앱 개발자**라고도 함
- 크게 안드로이드 개발자와 iOS 개발자로 나뉘고, 회사나 서비스에 따라서 혼자 이 둘을 모두 담당하기도 함



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ② 손 안의 앱을 만드는 개발자 – 모바일 개발자 (2/4)

- **안드로이드 개발자**는 갤럭시를 비롯한 안드로이드 스마트폰 기반의 모바일 앱을 개발함
- 예전에는 Java를 많이 사용했으나, 2017년부터는 보다 간결한 언어인 코틀린도 많이 사용하게 됨
- 일반적으로는 둘 중 하나의 언어를 선택하여 개발하지만, 폭넓은 개발을 위해서는 양쪽 모두 익혀 두는 것이 유리함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ② 손 안의 앱을 만드는 개발자 – 모바일 개발자 (3/4)

- 안드로이드 개발자와 시장을 양분하는 **iOS 개발자**는 아이폰과 아이패드용 모바일 앱을 개발함
- 예전에 iOS 개발은 오브젝티브-C 라는 언어를 사용했지만, 오브젝티브-C보다 현대적인 언어인 스위프트가 출시되고 나서부터는 스위프트를 주로 사용함
- 안드로이드 진영의 Java가 아직도 널리 쓰이는 것과는 달리 오브젝티브-C를 대체할 목적으로 만들어진 스위프트의 등장 이후 오브젝티브-C는 거의 사용되지 않음

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ② 손 안의 앱을 만드는 개발자 – 모바일 개발자 (4/4)

- 회사에 따라서는 한 명의 앱 개발자가 안드로이드와 iOS 모두를 개발하기도 함
- 안드로이드와 iOS 앱을 각각 다른 언어로 따로따로 개발하기도 하고, 하나의 소스 코드로 양쪽 모두에서 동작하는 앱을 만들 수 있는 **크로스 플랫폼**을 사용하기도 함
- 크로스 플랫폼의 종류로는 닥트라는 언어를 사용하는 플러터, 자바스크립트를 사용하는 리액트 네이티브, C#을 사용하는 자마린 등이 있음

macOS나 iOS용 프로그램 개발은 Mac에서만 가능하기 때문에  
iOS 개발자는 mac이 꼭 필요함

개발 환경 '설정'이 Mac에서만 가능하기 때문임!  
그래서 iOS 개발은 다소 진입 장벽이 높은 편임

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ③ 게임을 만드는 개발자 – 게임 개발자 (1/3)

- **게임 개발자**는 게임의 전체적인 구조를 설계해서 게임이라는 하나의 결과물을 만드는 직군임
- 게임 진행을 위한 복잡한 알고리즘부터 물리 엔진 구현, 3D 그래픽 렌더링, 수많은 접속자가 몰리는 서버의 원활한 관리 등 게임의 모든 구성 요소를 다룰 수 있어야 함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ③ 게임을 만드는 개발자 – 게임 개발자 (2/3)

- 그렇기 때문에 게임을 개발하기 위해서는 프로그래밍과 컴퓨터, 수학 등 여러 분야에 걸친 지식과 높은 숙련도가 필요하며, 대체 각 전문가들이 함께 만듦
- 다만 소규모 인디 게임은 혼자서 또는 소수의 인원으로 개발하기도 함

게임 프로그래밍은 고난도 작업인 만큼, 역할에 따라 컴퓨터 공학을 비롯하여 수학, 물리, 그래픽, 네트워크, 컴퓨터 시스템에 대한 심도 있는 공부가 필요함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ③ 게임을 만드는 개발자 – 게임 개발자 (3/3)

- 고사양 게임 프로그래밍은 기기의 성능을 최대한으로 끌어내는 것이 중요하므로 상대적으로 무거운 고수준 언어 대신, 기계와 밀접하고 가벼운 C나 C++을 많이 사용함
- 유니티 게임 엔진을 사용하는 게임에서는 C#을 사용하고, 웹 게임에서는 자바스크립트를 활용하기도 함
- 서버도 게임의 구성 요소이므로 백엔드 개발자들이 사용하는 언어와 프레임워크도 게임 개발에 활용됨



# 01. 개발자는 대체 뭐하는 사람인가요?

- ❖ ④ 컴퓨터 프로그램을 만드는 개발자 – 응용 소프트웨어 개발자 (1/2)
  - 웹과 모바일이 지금처럼 부상하기 이전에는 일반적으로 '개발자'라고 하면 응용 소프트웨어 개발자를 의미했음
  - 사실 응용 소프트웨어는 모바일 어플리케이션도 포함하는 개념이지만, **응용 소프트웨어 개발자**는 통상적으로 PC용 프로그램을 개발하는 사람을 지칭함
  - 우리가 흔히 컴퓨터에서 사용하는 워드, 포토샵, 압축 프로그램, 동영상 재생 프로그램 등이 응용 소프트웨어이며, 응용 소프트웨어 개발자가 이를 만들어 냄

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ④ 컴퓨터 프로그램을 만드는 개발자 - 응용 소프트웨어 개발자 (2/2)

- 응용 소프트웨어 개발에 사용할 수 있는 언어는 다양함
- C나 Java, 파이썬으로 개발하기도 하고, 웹 개발에 사용하는 언어로 제작하기도 함
- 윈도우용 프로그램 개발에는 C++과 C#을 주로 사용하고,  
Mac용 프로그램 개발에는 스위프트를 사용함
- 프로그램이 동작할 운영체제, 컴퓨터 구조와 소프트웨어 공학에 관한  
지식도 응용 소프트웨어 개발자가 갖춰야 할 중요한 역량임

응용 소프트웨어는 응용 프로그램이라고도 함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑤ 하드웨어를 제어하는 개발자 – 임베디드 개발자 (1/3)

- **임베디드 개발자**는 정해진 목적에 맞게 설계된 하드웨어의 작동을 프로그래밍하는 직군임
- 또한 임베디드 개발자가 다루는 **임베디드 시스템**은 기계 또는 기타 제어가 필요한 시스템을 제어하는 컴퓨터 시스템임
- 개인용 컴퓨터처럼 범용적인 목적을 수행하는 일반 소프트웨어와는 달리 전자 제품 같은 기기 내부에 내장되어 정해진 목적에 맞는 기능을 수행함
- 일상에서 주로 사용하는 냉장고의 온도 조절기나 스마트 TV를 비롯해 자동차, 로봇이나 항공기 같은 기기도 모두 임베디드 시스템임

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑤ 하드웨어를 제어하는 개발자 – 임베디드 개발자 (2/3)

- 그래서 임베디드 시스템 개발자가 하는 일도 크게 두 가지로 나뉨
- 하나는 하드웨어 자체를 개발하는 일이며,  
다른 하나는 이를 작동시킬 소프트웨어를 개발하는 일임
- 대체로 두 가지 일을 모두 하는 사람을 임베디드 개발자,  
소프트웨어만 개발하는 사람을 임베디드 소프트웨어 개발자라고 부름

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑤ 하드웨어를 제어하는 개발자 – 임베디드 개발자 (3/3)

- 임베디드 개발은 기계를 직접적으로 다루는 프로그래밍이므로 실행 속도가 빠르고 가벼운 C 언어를 주로 사용하고, 기계를 세부적으로 다룰 경우에는 C++이나 어셈블리어를 사용함
- 컴퓨터 하드웨어와 운영체제, 네트워크 등과 같은 컴퓨터 공학 지식이 필요하며, 분야에 따라 보드와 회로를 다루는 능력도 요구됨
- IoT 기기의 운영체제로 리눅스를 많이 사용하므로 리눅스 커널을 깊이 있게 공부해 두는 것도 임베디드 개발자를 준비하는데 큰 도움이 됨

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑥ 빈틈없는 정보 파수꾼 - 정보 보안 전문가 (1/3)

- 컴퓨터 보안은 호로 큰 건물 하나를 지키는 것과도 같음
- 언제, 어디로 해커가 숨어들지 모르기 때문에 건물 구조를 구석구석 알고 있어야 하고, 해커가 동원할 수 있는 모든 방법을 예상할 수 있어야 하며, 그에 대한 대책을 마련해야 함
- 다른 모든 곳을 철통같이 지켜도 작은 틈새 하나만 뚫리면 모든 것이 물거품이 되는 분야가 바로 보안임
- 사이버 보안, 정보 기술 보안이라고도 하며, 데이터 도난이나 손상을 차단하고 컴퓨터 시스템을 보호하는 역할을 수행함

### 해커와 크래커

- ✓ 사실 **해커**는 '컴퓨터와 프로그래밍에 대한 전문 지식을 가진 사람'을 뜻하는 단어임
- ✓ 이러한 전문 지식을 범죄에 활용하는 즉, 사람들이 흔히 해커라고 부르는 사람의 정확한 명칭은 **크래커**임
- ✓ 요즘은 보안 업계에 종사하는 해커를 화이트 해커, 크래커는 블랙 해커라고 더 많이 표현함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑥ 빈틈없는 정보 파수꾼 - 정보 보안 전문가 (2/3)

- **정보 보안 전문가**가 되려면 프로그래밍, 네트워크, 컴퓨터 구조, 운영체제 등 컴퓨터 공학 전반에 걸친 지식들이 필요함
- 이에 더하여 어느 세부 분야로 진출하느냐에 따라 웹, 데이터베이스, 포렌식, 암호학, 악성 코드 분석 등을 추가로 공부해야 함
- 해커가 어디로 파고 들어 공격해 올지 모르기 때문에 다양하면서도 깊이 있는 지식과 경험을 쌓아야 하고, 빠르게 발전하는 기술에 뒤처지지 않도록 끊임없이 공부해야 함

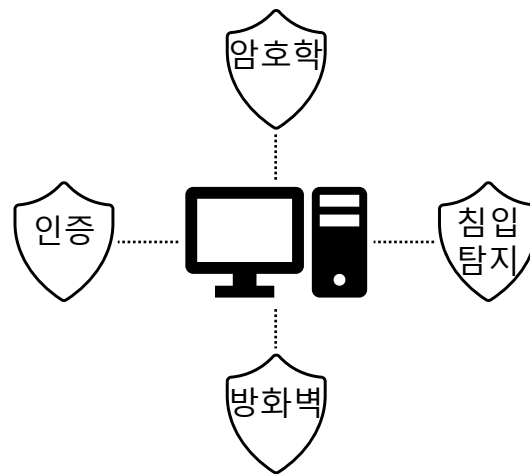
**단순히 이론만 공부하는 것이 아니라 모의 해킹 등을 통한 실습과 테스트를 거치며 경험을 쌓는 것도 중요함**

포렌식은 수사의 핵심 증거를 찾기 위해  
범죄에 사용된 컴퓨터에서 정보를 수집하는 것임

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑥ 빈틈없는 정보 파수꾼 - 정보 보안 전문가 (3/3)

- 정보 보안 전문가로서 근무할 수 있는 곳은 다양함
- 대기업의 보안 부서에서 일할 수도 있고,  
보안 업무를 전문적으로 하는 업체에 취직할 수도 있으며,  
국가정보원이나 경찰청 사이버 수사대 등 국가 기관에 들어갈 수도 있음
- 정보 보안 기술을 연구하는 곳에 연구원으로 들어가기도 함
- 기본적으로는 중요한 정보를 보호하고 보안 사고를 방지하는 일이기 때문에  
이로부터 오는 스트레스를 관리하는 능력과 함께 높은 윤리 의식과 책임감  
또한 요구되는 직무임





# 01. 개발자는 대체 뭐하는 사람인가요?

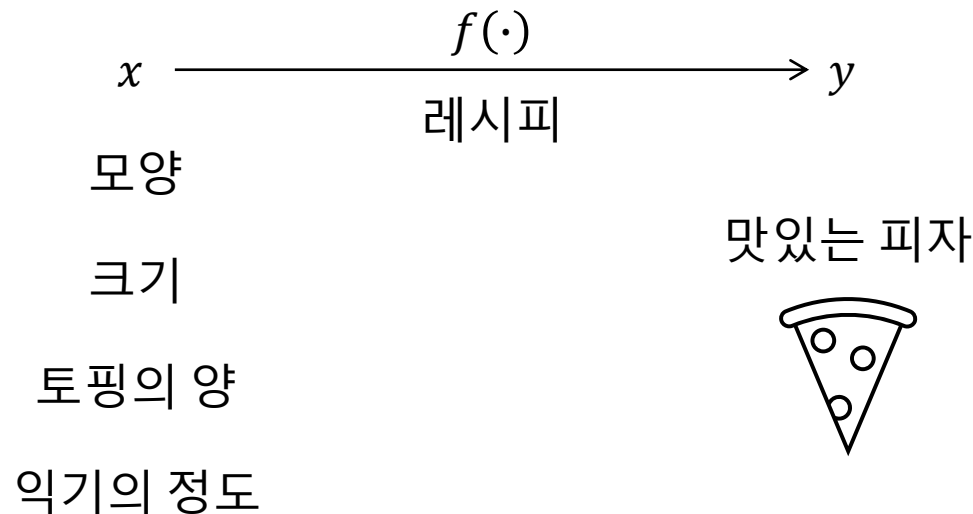
## ❖ ⑦ 기계를 가르치는 개발자 – 인공지능 개발자 (1/4)

- **인공지능**의 발전으로 인해 사람들의 삶이 여러 분야에서 편리해지면서 인공지능 개발자에 대한 수요도 커졌음
- 인공지능 개발은 크게 머신러닝 개발과 딥러닝 개발로 구분할 수 있음
- 인공지능을 포함해 머신러닝과 딥러닝은 기계에게  
'사람이 어디까지 알려 주는 가'에 따라 구분을 달리 함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑦ 기계를 가르치는 개발자 – 인공지능 개발자 (2/4)

- 종업원에게 '맛있는 피자를 만드는 방법'을 알려 주는 것에 비유해 보면, 인공지능은 종업원에게 반죽부터 화덕에 굽기까지의 모든 과정을 친절히 알려 주는 것임
- **머신러닝**은 피자 of 이상적인 모양과 크기, 토핑의 양, 익기의 정도를 알려주거나, 전국의 맛있는 피자들을 맛 보여준 뒤 종업원이 직접 피자를 만들어 보면서 레시피를 알아내도록 하는 것임



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑦ 기계를 가르치는 개발자 – 인공지능 개발자 (3/4)

- **딥러닝**은 그중에서도 고도의 기술로, 사람의 뇌를 흉내 내어 인공 신경망을 만들어 이와 같은 학습을 하도록 하는 것임
- 세부 직종마다 차이는 있지만, 대체로 수학, 통계학, 컴퓨터 공학 등 깊이 있는 지식을 요구하기 때문에 상대적으로 오랜 공부가 필요함

인공지능(Artificial Intelligence, AI)

머신러닝(Machine Learning)

인공신경망(Artificial Neural Networks)

딥러닝(Deep Learning)

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑦ 기계를 가르치는 개발자 – 인공지능 개발자 (4/4)

- 실제 업계에서 인공지능 개발자는 그 범위가 넓어 정확하게 규정하기 어려움
- 인공지능을 학문적으로 깊이 연구하는 연구직,  
인공지능 소프트웨어를 개발하는 프로그래머,  
데이터에서 의미 있는 정보를 찾아내는 분석가 등이  
모두 인공지능 관련 개발자라고 할 수 있음

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑧ 디지털 정보의 사서 – 데이터베이스 개발자 (1/2)

- 백엔드와 관련된 여러 업무 중에서도 데이터베이스에 특화된 업무를 수행하는 사람들이 있음
- IT 서비스를 이용할 때, 우리가 사용하는 모든 정보는 서버의 데이터베이스에 저장됨
- 데이터베이스는 디지털 정보들을 보관하는 대형 도서관에 비유할 수 있음
- **데이터베이스 개발자**는 이 도서관에서 사서와 같은 역할을 맡아 정보를 효과적으로 저장, 조회, 관리할 수 있도록 데이터베이스를 설계함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑧ 디지털 정보의 사서 – 데이터베이스 개발자 (2/2)

- 데이터베이스 개발자는 사서가 도서관에 책장을 배치하고 책들의 위치를 지정하는 것과 비슷하게 데이터베이스를 구축함
- 데이터베이스 구축 방식에 따라 데이터의 저장, 처리 속도와 안정성이 결정됨
- 데이터베이스에 문제가 생기면 사용자들이 불편을 겪을 수 있고, 중요한 정보가 손실될 수도 있으므로 데이터베이스 개발자의 역할은 매우 중요함

**데이터베이스 개발자는 데이터베이스 관리 시스템(DBMS)에 대한 깊은 이해와 SQL과 같은 데이터베이스 질의 언어에 능숙해야 함**

또한, 데이터 구조, 알고리즘, 보안에 대한 지식도 갖춰야 함

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑨ 클라우드 서비스를 제공하는 개발자 – 클라우드 개발자 (1/3)

- 온라인 서비스를 만들고 운영하기 위해서는 이들을 구동할 컴퓨터인 서버가 필요함
- 서버는 24시간 동작하여 사람들이 원활하게 서비스를 이용할 수 있도록 함
- 서버 시스템을 구축하고, 서버가 안정적으로 돌아가도록 유지 보수하며, 사용자 증가, 하드웨어 결함 등 각종 이슈에 대응하려면 고도의 기술적 지식이 필요함
- 이런 작업을 한두 명의 개발자가 처리하기는 매우 어려움

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑨ 클라우드 서비스를 제공하는 개발자 – 클라우드 개발자 (2/3)

- 하지만, 요즘은 여러 기업에서 서버에 관련된 다양한 기능들을 제공하여 기술 부담을 줄여 주고 있음
- 서버 구축을, 건물을 짓는 작업에 비유하면, 대형 건설사가 건설, 증축 및 철거, 경비 업무, 상수도 설치, 공간 대여 등을 도맡아 주는 것과 비슷함
- 아마존의 AWS나 마이크로소프트의 Azure 같은 클라우드 컴퓨팅 서비스가 대표적인 예시이며, 이를 클라우드 서비스라고 부름



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑨ 클라우드 서비스를 제공하는 개발자 – 클라우드 개발자 (3/3)

- 클라우드 서비스를 적절히 활용하여 자사의 서비스를 구축하려면 다양한 지식이 필요함
- 해당 분야에 관한 학위나 자격증, 실무 경험을 갖추면 **클라우드 개발자**로 일하는 것이 더 수월해 질 수 있음

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑩ 개발과 운영을 함께 관리하는 개발자 – 데브옵스 개발자 (1/3)

- 높은 매출을 올리는 음식점을 방문하면 음식의 맛뿐만 아니라 주문 접수, 조리, 홀서빙, 포장, 배달 등 다양한 업무들이 신속하고 유기적으로 이뤄지는 모습을 볼 수 있음
- 이러한 전 과정을 효율적이고 체계적으로 구축하고, 직원 간의 팀워크를 맞춰가는 것은 매장 성공의 필수 요소임

# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑩ 개발과 운영을 함께 관리하는 개발자 – 데브옵스 개발자 (2/3)

- 개발 분야도 마찬가지로
- 소프트웨어는 한 번 개발하고 끝나는 것이 아니라 기능 추가 및 수정, 오류 해결, 성능 개선 등의 업데이트 작업이 서비스 종료 전까지 계속됨
- 마치 생물의 신진대사처럼 프로그래밍, 코드 통합, 테스트, 새 버전 출시, 모니터링, 피드백, 긴급 문제 해결 등의 과정이 지속적으로 이뤄짐
- 이처럼 **개발**과 **운영**의 전 과정들이 빠르고 안정적으로 진행되도록 관리하는 것을 **데브옵스(DevOps)**라고 함



# 01. 개발자는 대체 뭐하는 사람인가요?

## ❖ ⑩ 개발과 운영을 함께 관리하는 개발자 – 데브옵스 개발자 (3/3)

- **데브옵스 개발자**는 이 과정들이 빠르고 연속적으로 진행되도록 자동화하고, 시간과 자원의 낭비가 없도록 최적화를 수행함
- 또한, 성능 문제나 버그가 신속하게 발견되도록 테스트 및 모니터링 시스템을 설계하고 모든 기록들을 수집해서 관리함
- 이처럼 데브옵스 개발자는 빠른 개발과 높은 품질, 효율적인 운영을 통해 자사의 서비스가 경쟁 우위를 점할 수 있도록 도와줌

데브옵스 개발자가 되기 위해서는  
컴퓨터 과학, 정보 기술, 소프트웨어 엔지니어링 등의 배경지식과  
프로그래밍, 네트워킹, 보안 등의 다양한 기술적 지식이 필요함

## ❖ 10가지 키워드로 정리하는 핵심 포인트 (1/2)

- **웹 개발자**는 웹사이트를 만들고 관리하며, 역할에 따라 웹 퍼블리셔/UI 개발자, 프론트엔드 개발자, 백엔드 개발자 등이 있음
- **모바일 개발자**는 스마트폰이나 태블릿, 스마트 워치 등 모바일 기기에서 동작하는 애플리케이션을 개발함
- **게임 개발자**는 게임의 전체적인 구조를 설계하고 개발함
- **응용 소프트웨어 개발자**는 PC용 프로그램을 개발하는 사람들을 지칭함
- **임베디드 개발자**는 하드웨어를 제어하는 소프트웨어를 개발함

## ❖ 10가지 키워드로 정리하는 핵심 포인트 (2/2)

- **정보 보안 전문가**는 각종 보안 위협으로부터 시스템과 서비스를 방어함
- **인공지능 개발자**는 인공지능을 학문적으로 깊이 연구하는 연구직, 프로그래머, 분석가 등이 있음
- **데이터베이스 개발자**는 대량의 정보를 다루고 그 체계를 구축함
- **클라우드 개발자**는 클라우드 형태로 제공되는 서버 관련 서비스를 다룸
- **데브옵스 개발자**는 개발 전반의 과정들을 최적화하고 관리함

## 02. 코딩과 프로그래밍, 앱과 프로그램

- 01. 개발자는 대체 뭐하는 사람인가요?
- 03. 프로그래밍 언어도 외국어인가요?

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 시작하기 전에 (1/3)

개발자가 하는 일과 개발자가 만드는 것은 무엇인지,  
어떤 특성을 가지고 있는지 알아보자.  
그리고 일반적으로 사용하는 용어의 개념을  
확실하게 이해할 수 있도록 좀 더 자세하게 알아보자.

#### 핵심 키워드

코딩

프로그래밍

개발

소프트웨어

프로그램

데이터

라이브러리



## 02. 코딩과 프로그래밍, 앱과 프로그램

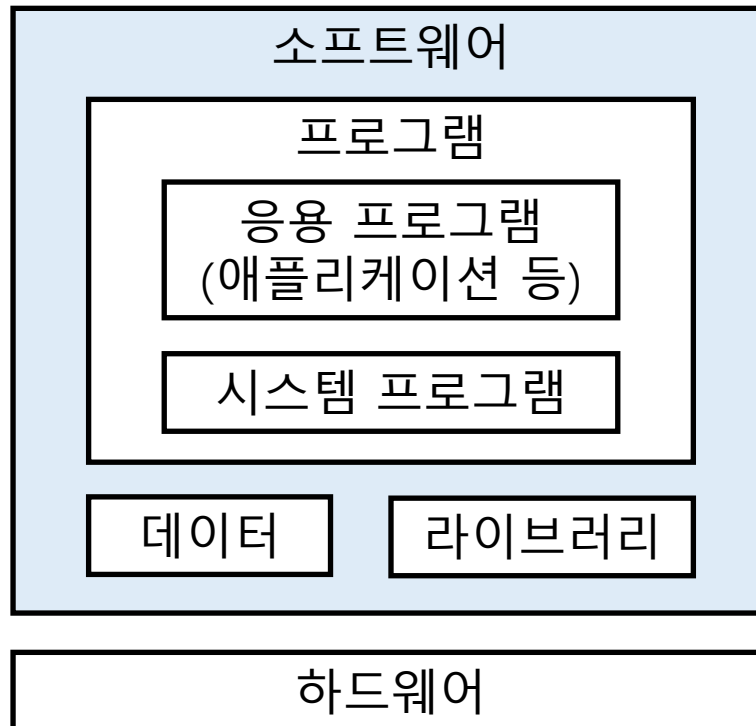
### ❖ 시작하기 전에 (2/3)

- 개발자가 하는 일은 보통 **개발**이라고 부르지만, 프로그래밍 또는 코딩이라고 부르기도 함
- 또한 개발자를 프로그래머라고 부르기도 하지만, 드물지만 코더라고 부르기도 함
- 일반적으로 이런 용어를 구분 없이 혼용하지만, 사실 각 용어마다 의미에 따른 작은 차이가 있음

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 시작하기 전에 (3/3)

- 소프트웨어, 프로그램, 그리고 애플리케이션도 같은 의미로 혼용해서 사용하는 경우가 많음
- 셋 중 가장 포괄적인 의미를 가진 것은 **소프트웨어**임



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 프로그래밍 언어로 코드 작성하기 - 코딩 (1/4)

- 개발자나 해커가 검은 화면 위에 무언가 타이핑하는 것을 영화나 미디어에서 접한 적이 있을 것임
- 이들이 컴퓨터로 작업해 만든 결과물을 소스 코드라고 부름
- **소스 코드**는 컴퓨터에 어떤 명령을 내리기 위해 컴퓨터가 이해할 수 있는 언어로 작성한 정보임

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 프로그래밍 언어로 코드 작성하기 – 코딩 (2/4)

- **코딩**은 소스 코드를 작성하는 작업임
- 작가가 펜을 잡고 글을 쓰는 것처럼 키보드로 타이핑해서 코드를 입력하는 단순 작업을 의미함
- 코딩이란, 부호를 나타내는 'code' 글자 뒤에 'ing'를 붙여 '부호를 작성한다'라는 뜻을 가진 단어임
- 따라서 **코더**는 컴퓨터가 처리할 수 있는 명령어로 코드(부호)를 작성하는 사람을 지칭함

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 프로그래밍 언어로 코드 작성하기 – 코딩 (3/4)

- 소스 코드는 작업을 어떻게 수행할 것인지를 알려 주는 **설계문**, 컴퓨터에 각종 지시를 내리는 **명령문**, 각 코드가 어떤 내용이지 개발자가 쉽게 알아 볼 수 있도록 설명해 주는 **주석** 같은 다양한 정보로 이루어져 있음

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 프로그래밍 언어로 코드 작성하기 - 코딩 (4/4)

- 다음 코드는 프로그래밍 언어 중 하나인 파이썬으로 작성한 코드임
- 이 코드는 [3, 5, 2, 4, 1]과 같이 정렬되지 않은 일련의 숫자를 [1, 2, 3, 4, 5]처럼 오름차순으로 정렬해 주는 작업을 수행함

```
1 arr = [3, 5, 2, 4, 1] # 명령문: 배열을 초기화
2
3 # 주석: 배열의 크기만큼 반복
4 def bubble_sort(arr): # 설계문: 함수를 정의
5     # 주석: 배열의 크기에서 1을 빼 다음 j만큼 반복
6     for j in range(len(arr) - 1, 0, -1): # 명령문: 외부 반복문
7         for k in range(j): # 명령문: 내부 반복문
8             # 주석: 만약 현재 인덱스 값이 다음 인덱스 값보다 클 경우 실행
9             if arr[j] > arr[j + 1]: # 명령문: 조건문
10                 # 주석: 서로 위치를 변환
11                 arr[j], arr[j + 1] = arr[j + 1], arr[j] # 명령문: 스왑
```

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터가 할 일 설계하기 - 프로그래밍 (1/5)

- 프로그래밍은 코딩보다 더 넓은 개념임
- **프로그래밍**은 컴퓨터에서 동작하는 프로그램을 설계하고 만드는 전 과정을 의미함
- 건물을 지을 때 설계도 없이 건물을 지을 수 없듯이,  
프로그램을 만들 때에도 반드시 설계가 필요함

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터가 할 일 설계하기 - 프로그래밍 (2/5)

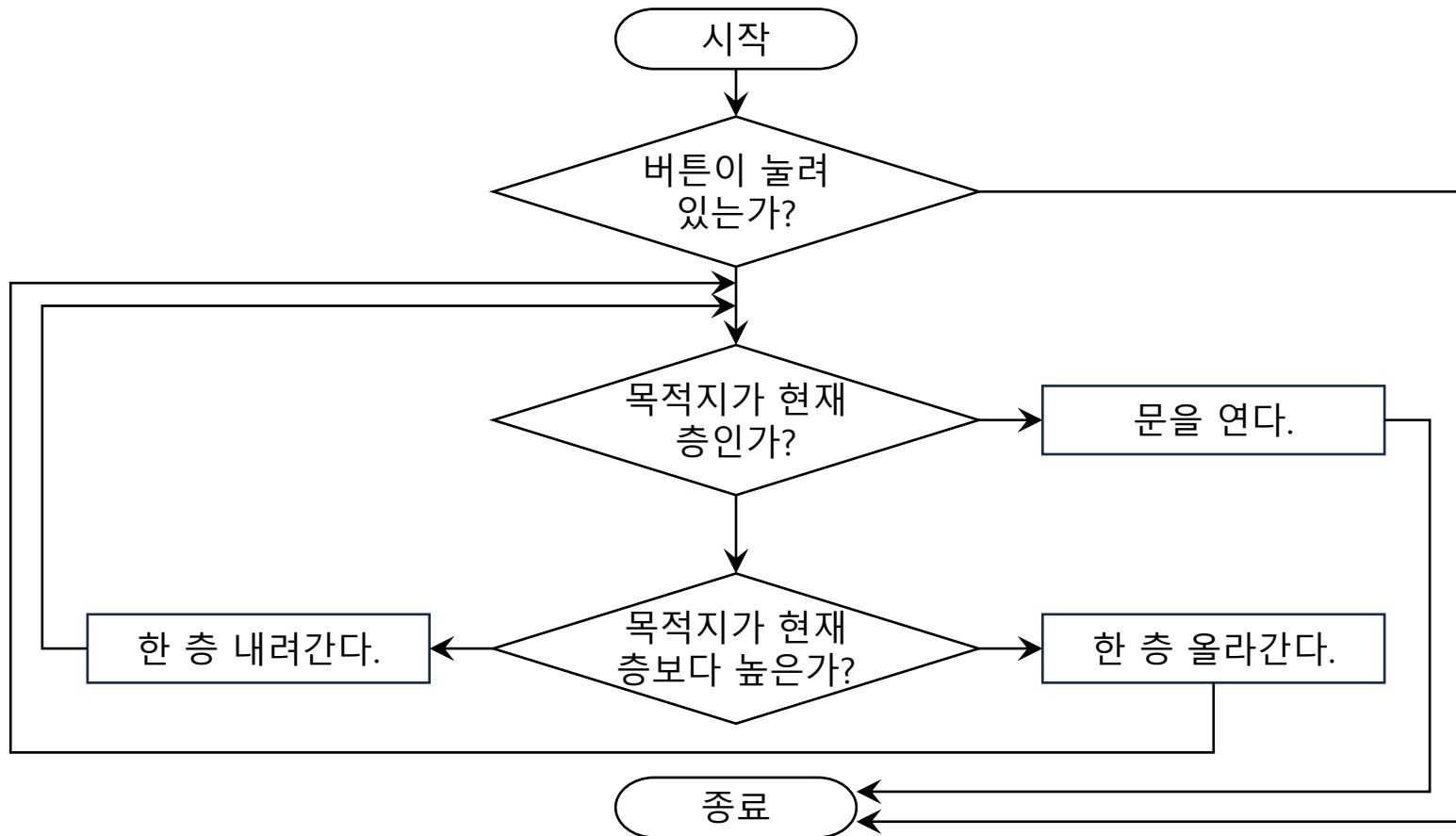
- **프로그래머**는 단순히 코드만 작성하는 코더와는 다르게 개발할 프로그램에 어떤 기능을 추가하고, 어떤 방식으로 구현할지 미리 계획할 수 있어야 함
- 코딩이 프로그래밍 코드를 입력하는 단순 글쓰기라면  
프로그래밍은 목적인 바에 따라 문법이 맞고  
기승전결이 완벽하게 갖춰진 책을 만드는 글짓기라고 할 수 있음



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터가 할 일 설계하기 - 프로그래밍 (3/5)

- 다음은 엘리베이터 작동 방법을 순서도로 표현한 것임
- 순서도란 일을 처리하는 과정을 순서대로 도식화한 것을 말함



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터가 할 일 설계하기 - 프로그래밍 (4/5)

- 컴퓨터가 수행할 절차대로 순서도를 구성하는 것, 즉 어떤 순서로 할지, 발생할 수 있는 상황은 무엇이고 그에 따라 어떤 행동을 할지, 어떤 과정을 몇 번 반복할지 등을 설계하는 것이 **알고리즘**임
- 이것을 프로그래밍 언어로 작성하는 과정을 **프로그래밍**이라 하고, 이때 만들어진 결과물을 **소스 코드**라고 하는 것임

프로그래밍 또한 소스 코드를 작성하는 일이기 때문에  
일반적으로 '코딩'과 '프로그래밍'은  
같은 의미로 사용되고 있음

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터가 할 일 설계하기 - 프로그래밍 (5/5)

- 글을 쓸 때는 각 문장이 문법에 맞아야 하며, 각 문단과 글 전체 내용도 문맥적으로 자연스럽고 논리적으로 결함이 없어야 함
- 프로그래밍도 마찬가지임
- 앞의 알고리즘 설계에 논리적인 오류가 있거나 고려하지 못한 변수가 있다면 엘리베이터 문이 열리지 않거나 목적인 층에 도착하고도 멈추지 않을 수 있음
- 또한 알고리즘이 효율적으로 설계되지 않은 경우에는 목적지로 가는 도중 층마다 멈추는 등 불필요한 일이 발생할 수도 있음

**그러므로 프로그래밍은  
논리적인 사고력과 꼼꼼함 등을 필요로 하는 작업임**

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 소프트웨어를 만들고 운영하는 과정 - 개발 (1/5)

- 소프트웨어 **개발**은 코딩이나 프로그래밍 외에 좀 더 많은 요소가 결합돼 있음
- 실제로 시장에 판매할 수 있는 프로그램을 개발하는 것은 물론, 발생할 수 있는 다양한 이슈를 처리할 수 있는 능력이 필요함

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 소프트웨어를 만들고 운영하는 과정 - 개발 (2/5)

- 기획팀에서 "특정한 기능을 수행하는 스마트폰 앱을 만들어 달라"고 주문하면  
(① **요구 사항 분석**), 개발자는 먼저 요구 사항을 구체화한 뒤(② **시스템 명세**)  
이를 어떻게 구현할지 전반적인 구상을 시작함(③ **구조 설계**)
- 예를 들어, 어떤 종류의 앱을 만들지, 프로그래밍 언어는 무엇을 사용할지,  
데이터는 어떤 방식으로 저장하고 관리할지, 어떤 서버에서 구축할지 등을 결정함



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 소프트웨어를 만들고 운영하는 과정 - 개발 (3/5)

- 앱의 전반적인 구조를 설계하고 나면 본격적으로 제작에 들어가는데, 여기에 프로그래밍과 코딩이 포함됨(④ 구현)
- 이 과정에서 작가가 글을 쓸 때 문장을 수정하고 맞춤법을 고치면서 글을 다듬어 가는 것처럼 프로그래밍도 끊임없는 오류 점검과 기능 테스트, 코드 수정이 이뤄짐(⑤ 테스트)



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 소프트웨어를 만들고 운영하는 과정 - 개발 (4/5)

- 모든 과정을 마치고 제품을 출시한 후에도 각종 버그를 찾아 해결하고 기능을 업데이트하며 앱이 안정적으로 운영되도록 꾸준히 관리하는 작업이 필요함

(⑥ 유지 보수)



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 소프트웨어를 만들고 운영하는 과정 - 개발 (5/5)

- 개발은 코딩이나 프로그래밍보다 더 포괄적인 개념으로 소프트웨어의 기술적 계획부터 제작, 사후 관리까지 모두 포함하는 용어임
- 개발자의 업무는 단순히 프로그램을 코딩하며 소프트웨어를 만들어 내는 데 국한되지 않음
- 서비스가 제대로 작동하는지 모니터링하는 것,  
기존 소프트웨어의 문제점이나 비효율적인 부분들을 찾아 개선하는 것,  
데이터베이스 체계를 구축하는 것,  
개발 과정을 자동화하고 설계를 문서로 정리하는 것 등  
여러 역할이 있으며, 모두 소프트웨어 개발에 있어 중요한 부분임



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소 – 소프트웨어 vs 프로그램 vs 애플리케이션

- 우리는 흔히 대화를 나눌 때 소프트웨어와 프로그램, 애플리케이션을 구분하지 않고 혼용해서 사용함
- 그만큼 IT 용어를 많이 사용하고 있다는 의미이기도 함
- 사실 이 세 가지 용어는 엄밀하게 말하면 각각 다른 뜻을 갖고 있음

이번에는 프로그램과 소프트웨어, 애플리케이션의  
정확한 개념을 알아보자

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소: ① 소프트웨어와 프로그램 (1/5)

- 컴퓨터 본체와 CPU, 메모리 등 눈에 보이고 만져지는 부분이 하드웨어라면 소프트웨어는 보이지 않는 부분이라고 할 수 있음
- 또는 하드웨어가 컴퓨터의 몸이라면, 소프트웨어는 영혼이라고 할 수 있음
- 소프트웨어는 프로그램과 라이브러리, 데이터 등으로 이뤄짐

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소: ① 소프트웨어와 프로그램 (2/5)

- 소프트웨어와 프로그램이라는 단어는 널리 알려져 있고, 일반적으로 혼용해서 사용함
- 컴퓨터에서 더블클릭하여 실행하는 아이콘들을 소프트웨어라고 하기도 하고, 어떨 때는 프로그램이라고도 함

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소: ① 소프트웨어와 프로그램 (3/5)

- 이메일을 보내는 소프트웨어를 만든다고 가정해 보자
- 먼저, 받는 사람의 이름과 주소, 내용 등과 같은 정보가 필요함
- 이런 정보를 **데이터**라고 함
- 이 메일을 보내는 과정에는 내용 쓰기, 서식 지정하기, 이미지 삽입하기, 파일 첨부하기 등과 같은 기능이 필요함
- 이처럼 어떤 일을 수행하기 위해 필요한 기능을 일정 단위로 묶은 것을 **라이브러리**라고 함

## 02. 코딩과 프로그래밍, 앱과 프로그램

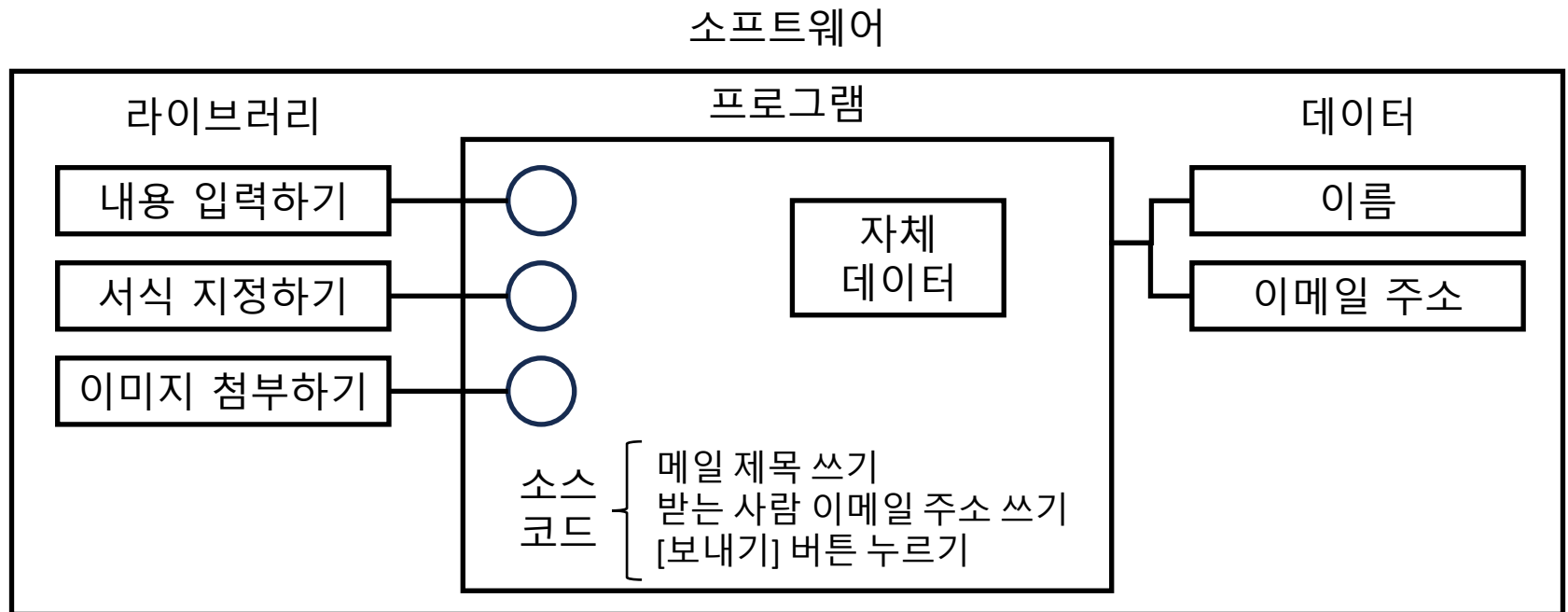
### ❖ 컴퓨터의 보이지 않는 요소: ① 소프트웨어와 프로그램 (4/5)

- 그리고 받는 사람, 메일 제목, 메일 내용을 입력하는 것부터  
마지막에 [보내기] 버튼을 누르기까지의 모든 과정을 수행할 수 있도록  
만든 것이 **프로그램**임
- 프로그램은 특정 목적을 위해 실행할 수 있는 대상으로써  
주어진 일련의 작업들을 수행하기 위해 프로그래밍됨
- 라이브러리들 중 일부는 프로그램의 구성 요소로 들어가기도 함

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소: ① 소프트웨어와 프로그램 (5/5)

- 그리고 이 모든 것을 아우르는 것이 **소프트웨어**임
- 컴퓨터의 물리적이지 않은 모든 요소를 포함하는 용어임
- 앞에서 언급한 프로그램, 데이터, 라이브러리뿐만 아니라 이 모든 과정이 이뤄지는 환경의 기반이 되는 Windows나 macOS, 안드로이드, 리눅스 등과 같은 운영체제 역시 소프트웨어에 해당함



## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소: ② 애플리케이션 (1/2)

- 스마트폰이 대중화된 이후로 **애플리케이션**이라는 단어는 사람들에게 익숙해졌음
- 흔히 줄여서 **어플** 또는 **앱**이라고 부름
- 사실 이는 새로 생긴 단어가 아님
- 윈도우 기반의 PC를 사용해 왔다면 응용 프로그램이라는 단어를 한 번쯤은 접해 봤을 것임
- 이 응용 프로그램이 바로 애플리케이션임

## 02. 코딩과 프로그래밍, 앱과 프로그램

### ❖ 컴퓨터의 보이지 않는 요소: ② 애플리케이션 (2/2)

- 프로그램은 시스템 프로그램과 응용 프로그램으로 나뉨
- 시스템 프로그램이 집을 구성하는 건물과 수도, 전기, 난방시설이라면 응용 프로그램은 집 안의 청소기나 세탁기, 냉장고 등에 비유할 수 있음
- **시스템 프로그램**이 운영체제를 구성하는 소프트웨어로써 사용자가 컴퓨터를 원활히 사용할 수 있는 환경을 제공한다면, **응용 프로그램**은 각각의 목적에 따라 사용자가 직접 손에 쥐고 실행하여 활용할 수 있는 프로그램임
- 우리가 흔히 PC나 스마트폰에서 아이콘을 더블클릭하거나 터치해서 실행하는 워드, 포토샵, 게임, 크롬 등의 프로그램들은 모두 응용 프로그램, 다른 말로 애플리케이션이라고 부름



## ❖ 7가지 키워드로 정리하는 핵심 포인트 (1/2)

- **코딩**은 프로그래밍 언어로 된 코드를 입력하는 작업 자체를 말함
- **프로그래밍**은 컴퓨터가 할 일의 절차와 알고리즘을 설계하는 것을 말함
- **개발**은 소프트웨어를 설계, 구현, 운영, 관리하는 데 필요한 전반적인 기술적 과정들을 통칭함
- 컴퓨터의 하드웨어가 본체와 CPU, 메모리 등 눈에 보이고 만져지는 부분이라면, **소프트웨어**는 보이지 않는 부분을 통칭함

## ❖ 7가지 키워드로 정리하는 핵심 포인트 (2/2)

- **프로그램**은 사용자의 명령에 따라 목적에 맞는 작업을 수행하는 일련의 명령 모음임
- 시스템 프로그램은 운영체제의 일부로써 컴퓨터 이용 환경을 조성하는 프로그램이며, 응용 프로그램은 사용자가 목적에 따라 직접 사용하는 프로그램임
- **데이터**는 프로그램을 실행하는 데 필요한 기초 자료를 말함
- **라이브러리**는 하나 이상의 프로그램에 활용될 수 있는 데이터와 명령어들의 집합임

## 03. 프로그래밍 언어도 외국어인가요?

- 01. 개발자는 대체 뭐하는 사람인가요?
- 02. 코딩과 프로그래밍, 앱과 프로그램

# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 시작하기 전에 (1/3)

컴퓨터와 소통하기 위한 언어에는 어떤 종류가 있는지,  
개발자의 워드 프로세서라고 불리는  
통합 개발 환경(IDE)은 무엇인지 알아보자

### 핵심 키워드

기계어

어셈블리어

저수준 언어

고수준 언어

프로그래밍 언어

컴파일 언어

인터프리터 언어

IDE

# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 시작하기 전에 (2/3)

- 프로그래밍 언어라는 단어를 처음 접했을 때 아래와 같은 생각으로 겁을 먹고, 시작하기도 전에 포기하는 사람이 많음
- 그리고 지구상에 프로그래밍 언어가 9,000개나 존재한다는 사실을 알면 코딩이 더욱더 멀고 어렵게 느껴질 수 있음

“**프로그래밍 언어**라고? 컴퓨터를 다루는 거랑 언어가 무슨 연관이 있지?  
가뜰이나 영어 공부도 어려운데,  
개발자가 되려면 외국어 같은 걸 하나 더 배워야 한다는 건가?”



# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 시작하기 전에 (3/3)

- 하지만 벌써부터 겁먹을 필요는 없음
- 프로그래밍 언어를 배운다는 것은 새로운 스포츠나 게임을 익히는 것만큼이나 흥미롭고 즐거운 일이기 때문임

이번에는 프로그래밍 언어와  
개발 환경에 대해 알아보자

# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 컴퓨터에 명령을 내릴 때 사용하는 언어 – 프로그래밍 언어 (1/4)

- 우리는 매일 기계에 명령을 내림
- 스위치로 전등을 켜다가 끄기도 하고, TV 리모컨으로 채널을 선택하거나 볼륨을 조절하기도 함
- 스마트폰으로는 아이콘이나 버튼을 손가락으로 터치해서 실행하고, 제스처를 사용해서 화면을 넘기거나 사진을 확대해 보기도 함
- 아마도 키보드와 마우스로 컴퓨터를 사용하는 것 정도가 일반인들이 기계에 내리는 가장 복잡한 형태의 명령일 것임

## 03. 프로그래밍 언어도 외국어인가요?

### ❖ 컴퓨터에 명령을 내릴 때 사용하는 언어 – 프로그래밍 언어 (2/4)

- 우리가 이처럼 손쉬운 방법으로 기계를 사용할 수 있는 이유는 개발자를 비롯한 엔지니어들이 기능을 모두 기계에 넣어 두었기 때문임
- 개발자들이 스마트폰 앱, PC용 워드 프로세서, 콘솔용 게임 등에 필요한 기능을 프로그래밍해 두고 사용자가 클릭이나 터치로 편리하게 기능을 실행할 수 있도록 만들어 준 것임



# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 컴퓨터에 명령을 내릴 때 사용하는 언어 – 프로그래밍 언어 (3/4)

- 그러나 개발자는 무에서 유를 창조해야 하는 사람들임
- 이미 다른 개발자가 컴퓨터에 구현해 놓은 것을 실행하는 것이 아니라, 개발자가 직접 목적에 맞는 새로운 기능을 컴퓨터에 추가해야 함
- 숫자 오름차순 정렬 예시를 살펴본 것처럼 사람이 보기에는 매우 단순해 보이는 작업도 컴퓨터에게 알려 주기 위해서는 매우 복잡하고 많은 과정을 명령으로 입력해야 함

# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 컴퓨터에 명령을 내릴 때 사용하는 언어 – 프로그래밍 언어 (4/4)

- 이는 스위치나 리모컨, 마우스의 간단한 조작으로 작성하기에는 너무나 복잡한 작업임
- 마치 전화 상담원이 고객과 상담하는 업무를 후임에게 손짓과 눈짓만으로 가르치는 것만큼이나 불가능에 가까운 일임
- 이 둘 사이에도 정확하게 소통할 수 있는 약속된 언어가 필요한 것처럼 사람과 컴퓨터 사이에도 언어라 불릴 만큼 정교하고 풍성한 소통 수단이 필요함
- 이를 위해 만들어진 것이 **프로그래밍 언어**임

# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (1/8)

- 지구상의 수많은 언어 중에서 대표적인 프로그래밍 언어로  
C, C++, C#, Java, 파이썬, 자바스크립트 등을 꼽을 수 있음
- 만약 코딩을 처음 배운다면 이 많은 언어 중 무엇부터 시작해야 할까?
- 언어를 하나만 익히는 것도 어려운데  
세상에는 왜 이렇게 많은 프로그래밍 언어가 존재할까?

# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (2/8)

- 컴퓨터에 명령을 내린다는 목적으로 만들어지 언어인 만큼  
프로그래밍 언어로 소프트웨어를 개발하다 보면 아쉬운 부분이 생기기 마련임
- 그래서 좀 더 읽기 편한 언어, 짧은 코드로 더 많은 지시를 내릴 수 있는 언어,  
보다 빨리 작동하는 언어, 오류로부터 안전한 언어를 원하는 사람들이 생겼음
- 이처럼 다양한 요청과 수요에 따라 계속해서 새로운 프로그래밍 언어가  
만들어졌고, 그 결과로 오늘날 널리 사용되는 프로그래밍 언어들은  
과거에 비해 편리하면서도 오류 발생이 적고 더 강력한 기능들을 갖추게 됨

## 03. 프로그래밍 언어도 외국어인가요?

### ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (3/8)

- 다음은 숫자 5개를 오름차순 정렬하는 작업을 Java와 파이썬으로 작성한 소스 코드임

```
void bubbleSort(int arr[]) {  
    int n = arr.length;  
    for (int j = 0; j < n - 1; j++) {  
        for (int k = 0; k < n - j - 1; k++) {  
            if (arr[k] > arr[k + 1]) {  
                int temp = arr[k];  
                arr[k] = arr[k + 1];  
                arr[k + 1] = temp  
            }  
        }  
    }  
}
```

Java

```
def bubble_sort(arr):  
    n = len(arr)  
    for j in range(0, n, 1):  
        for k in range(0, n - j - 1, 1):  
            if arr[k] > arr[k + 1]:  
                arr[k], arr[k + 1] = arr[k + 1], arr[k]
```

파이썬

### 03. 프로그래밍 언어도 외국어인가요?

#### ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (4/8)

- 세부적인 내용은 다르지만, 서로 비슷한 부분이 많음을 발견할 수 있음
- 한국어와 영어, 아랍어와 같은 언어는 글자의 형태, 언어 구조 등이 달라서 하나의 언어를 알아도 다른 언어를 이해하기 어렵지만,  
프로그래밍 언어는 문법과 구문을 대다수 공유하며 비슷한 요소로 구성돼 있음
- 그렇기 때문에 프로그래밍 언어 하나를 배우고 나면  
다른 언어로 작성된 코드도 웬만해서 큰 어려움 없이 읽을 수 있음
- 그렇다고 서로 비슷하기만 한 건 아님

언어마다 설계 목적과 철학, 강점이 있어  
각기 다른 매력이 있음

### 03. 프로그래밍 언어도 외국어인가요?

#### ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (5/8)

- 그렇기 때문에 개발자에게 새로운 프로그래밍 언어를 익히는 일은 어렵고 부담되는 일이 아니라 즐겁고 설레는 일임
- 여러 개의 프로그래밍 언어를 사용할 줄 아는 것은 개발자들 사이에서는 흔한 일임

**프로그래밍 언어는 하나만 제대로 배워 두면  
다른 언어들도 쉽게 익힐 수 있음**

## 03. 프로그래밍 언어도 외국어인가요?

- ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (6/8)
  - 앞서 프로그래밍 언어가 9,000개에 달한다고 했지만,  
실제로 사용하는 프로그래밍 언어의 수는 훨씬 적음
  - 언어의 발전에 따라 낡고 오래된 언어는 새로운 언어로 대체되기도 하고,  
꾸준한 업데이트를 거치며 오랫동안 사랑받는 언어도 있음



# 03. 프로그래밍 언어도 외국어인가요?

## ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (7/8)

- 구글 검색엔진에 '프로그래밍 언어 순위'를 검색해 보자
- 오늘날 많이 사용하는 인기 프로그래밍 언어를 확인할 수 있음
- 티오베(TIOBE)에서는 프로그래밍 언어의 인기를 확인할 수 있는 지수를 공개함

### 2025년 1월 프로그래밍 언어 순위

Jan 2025	Jan 2024	Change	Programming Language		Ratings	Change
1	1			Python	23.28%	+9.32%
2	3	▲		C++	10.29%	+0.33%
3	4	▲		Java	10.15%	+2.28%
4	2	▼		C	8.86%	-2.59%
5	5			C#	4.45%	-2.71%
6	6			JavaScript	4.20%	+1.43%
7	11	▲		Go	2.61%	+1.24%
8	9	▲		SQL	2.41%	+0.95%
9	8	▼		Visual Basic	2.37%	+0.77%
10	12	▲		Fortran	2.04%	+0.94%

[자료출처] <https://www.tiobe.com/tiobe-index/>

### 03. 프로그래밍 언어도 외국어인가요?

- ❖ 배우기 쉬운 언어는 없을까? – 프로그래밍 언어가 다양한 이유 (8/8)
  - 이와 같이 각각의 프로그래밍 언어는 주로 사용하는 분야와 사용처가 다름
  - '어떤 개발자가 될 것인가'에 따라 해당 생태계에 맞는 프로그래밍 언어를 공부해야 하지만, 가장 많이 사용하는 언어 중 하나를 골라 천천히 시작해도 괜찮음
  - 하나의 언어를 능숙하게 다룰 수 있으면 다른 언어를 배우는 것은 금방이기 때문임
  - 현재 가장 인기 있는 언어는 파이썬임

- ❖ 01. 개발자는 대체 뭐하는 사람인가요?
- ❖ 02. 코딩과 프로그래밍, 앱과 프로그램
- ❖ 03. 프로그래밍 언어도 외국어인가요?

# THANK YOU!

## Q & A

- Name: 권범
- Office: 동덕여자대학교 인문관 B821호
- Phone: 02-940-4752
- E-mail: [bkwon@dongduk.ac.kr](mailto:bkwon@dongduk.ac.kr)