

# 컴퓨터 AI 공학부 Basic

목차

1. 개발자 종류
2. 프로그래밍 언어
3. IDE(통합 개발 환경)
4. 용어 정리

컴퓨터·AI공학부 AI학술동아리 <MLP>

# 1. 개발자 종류

# 개발자 종류











- 웹 개발자 - 웹사이트를 만들고 관리하며, 역할에 따라 웹 퍼블리셔(UI 개발자), 프론트엔드 개발자, 백엔드 개발자 등이 존재
- 모바일 개발자 - 스마트폰이나 태블릿, 스마트 워치 등 모바일 기기에서 작동하는 애플리케이션을 개발
- 게임 개발자 - 게임의 전체적인 구조를 설계하고 개발
- 응용 소프트웨어 개발자 - PC용 프로그램을 개발하는 사람들을 지칭
- 임베디드 개발자 - 하드웨어를 제어하는 소프트웨어를 개발
- 정보 보안 전문가 - 각종 보안 위협으로부터 시스템과 서비스를 방어
- AI 개발자 - 인간의 사고와 학습같은 지적 능력을 컴퓨터를 통해 구현

# 웹 개발자

- 웹 퍼블리셔(UI 개발자) - 웹사이트에서 보이는 부분을 담당
- **프론트엔드 개발자** - 웹 퍼블리셔 역할 + 자바스크립트로 프로그래밍 요소를 개발
- **백엔드 개발자** - 웹사이트에서 눈에 보이지 않는 요소 개발, (= 서버 개발자)

## 2. 프로그래밍 언어

# 프로그래밍 언어 인기 순위

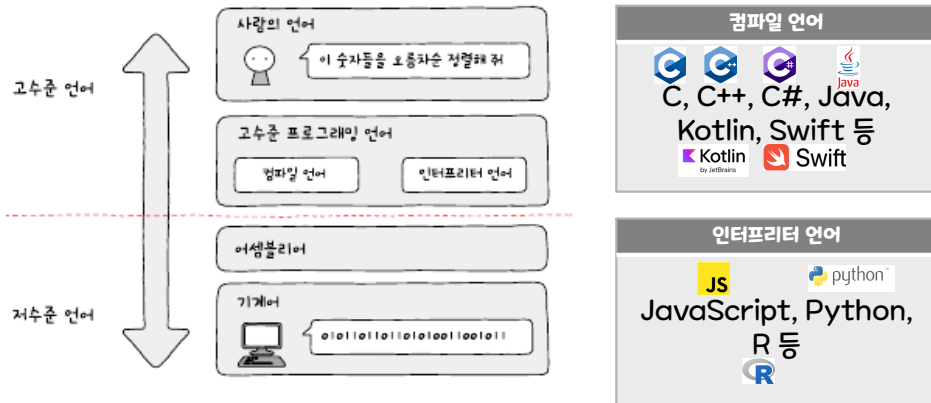
Jan 2023	Jan 2022	Change	Programming Language		Ratings	Change
1	1			Python	16.36%	+2.78%
2	2			C	16.26%	+3.82%
3	4	▲		C++	12.91%	+4.62%
4	3	▼		Java	12.21%	+1.55%
5	5			C#	5.73%	+0.05%
6	6			Visual Basic	4.64%	-0.10%
7	7			JavaScript	2.87%	+0.78%
8	9	▲		SQL	2.50%	+0.70%
9	8	▼		Assembly language	1.60%	-0.25%
10	11	▲		PHP	1.39%	-0.00%

<https://www.tiobe.com/tiobe-index/>

# 고수준 언어, 저수준 언어

- 저수준 언어 - 컴퓨터의 언어(0과 1 = 기계어)에 가까운 언어
  - 어셈블리어 : 기계에 직접 세부적인 명령을 내리기 유용해 임베디드 개발자들이 많이 사용
- 고수준 언어 - 사람의 언어에 가까운 언어
  - 컴파일 언어 : 코드 전체를 미리 기계어로 번역(컴파일)한 다음 실행 파일을 생성해서 실행하는 언어(오류로부터 안전, 프로그램 속도 빠름)
    - 예) C, C++, C#, Java, Kotlin, Swift 등
  - 인터프리터 언어 : 작성된 코드를 한 줄 한 줄 통역하면서 실행되는 언어(신속하고 유연한 개발에 이용)
    - 예) JavaScript, Python, R 등

# 고수준 언어, 저수준 언어





# 프로그래밍 언어

- Java - 플랫폼에 독립적인 언어 -> 플랫폼에 맞는 Java Runtime Environment만 설치되어있다면 다른 OS에서 빌드한 프로그램을 다시 빌드하는 과정없이 그대로 사용 가능
- 빌드 : 소스 코드를 실행할 수 있는 소프트웨어로 만드는 것
- JVM 언어는 모두 플랫폼 독립적 - Kotlin
- 다른 언어들은 플랫폼에 종속적, 다시 빌드 과정 필요

# 앱 개발 언어

- Native App

(안드로이드나 iOS와 같은 각 플랫폼에 특정 프로그래밍 언어를 사용하여 만들어진 앱)

- 안드로이드 앱 – Java, Kotlin

- iOS 앱 – Swift

- Cross-Platform App

(한 가지의 개발 언어와 프레임워크로 만들어진 앱, 안드로이드와 iOS 둘 다 사용 가능)

- React Native(Meta)






- Flutter(Google)

### 3. IDE(통합 개발 환경)



# IDE

- 윈도우 기본 프로그램인 메모장에서도 코딩은 가능
- IDE(Integrated Development Environment)  
(= 통합 개발 환경) - 코딩을 도와주는 프로그램(오류 잡기, 코드 예시 제안 등)
- Logic Error(논리적 오류) : 결과가 예상과 다를 때  
(수정 난이도 어려움)
- IDE는 논리적 오류를 잡아주지 못함

# IDE 종류

- 이클립스 - Java 프로그래밍에 이용(무료)  eclipse
- 인텔리제이 - 자바 프로그래밍에 이용(무료 / 유료)  IntelliJ IDEA  
by JetBrains
  - » 학교 계정 - 유료로 무료로 사용가능
- 파이참 - 파이썬 프로그래밍에 이용(무료 / 유료)  PyCharm  
by JetBrains
  - » 학교 계정 - 유료로 무료로 사용가능
- 안드로이드 스튜디오 - 안드로이드 개발에 이용(무료)  Android Studio
- 엑스코드 - 아이폰 앱, 맥용 프로그램 개발에 이용(무료) 
  - » App스토어에 앱을 올리려면 유료 개발자 등록 필요

# IDE 종류

- **비주얼 스튜디오** - C/C++/C# 프로그래밍에 이용(무료) 
- **비주얼 스튜디오 코드** - 프로그램 자체는 가벼운 코드 편집 위주의 기능만 포함(정확히는 코드 에디터)  
원하는 플러그인 추가 시 IDE로 구성 가능  
여러 언어로 프로그래밍 가능  
React Native, Flutter 개발에 이용   
**Python** 프로그래밍에 이용(무료)

## 4. 용어 정리

# 용어 정리

- 디버그 : 주로 오류 수정 프로그램과 그 작업을 통칭하는 단어
- 디버깅 : 오류를 찾아 제거하는 행동(개발 - 코딩 : 20%, 디버깅 80%)
- 빌드 : 소스 코드를 실행할 수 있는 소프트웨어로 만드는 것
- 배포 : 소프트웨어를 사용자들에게 전달하는 것
- 유지 보수 : 소프트웨어 제품 출시 이후 계속되는 문제 해결 및 각종 업데이트 작업
- 리팩토링 : 코드의 결과는 바꾸지 않으면서 코드를 리모델링하는 것
  - 알고리즘 개선, 코딩 스타일 통일 등(유지 보수와도 연결 됨)
- 코드 리뷰 : 서로의 코드를 확인하고 피드백을 주고받는 과정
- 문서화 : 소프트웨어와 그 소스 코드를 쉽게 파악할 수 있도록 문서를 작성해 두는 것 (소스 코드 주석 + 문서)



# 프레임 워크? 라이브러리?

- **프레임워크** - 특정 프로그램을 개발하기 위한 **여러 요소들을** 제공하는 **기본 틀**  
Ralph Johnson - a reusable design of all or part of a system that is represented by a set of abstract classes and the way their instances interact.
- **라이브러리** - 자주 사용되는 로직을 재사용하기 편리하도록 잘 정리한 일련의 **코드들의 집합**
- 무언가를 **기반**으로 그 위에 프로그램을 **‘만들면’** 프레임워크
- 개발자가 프로그램을 만들기 위해 **‘가져다 쓰면’** 라이브러리
- 둘의 차이는 무엇인가?
  - 프레임워크는 프레임워크가 전체적인 흐름을 제어 (제어의 역전, inversion of Control)
  - 라이브러리는 개발자가 전체적인 흐름을 제어