

---

# Hello World

---

iOS School 5기 수업에 오신 여러분 환영합니다.

주영민

# 여러분을 소개해주세요.

---

- 제가 아닌 옆에서 소개해주세요.
- 옆에 분이 어떤분인지 물어보세요.
- 왜 이 수업을 듣기로 했는지, 이 수업이 끝난 후의 계획은?

# 이 수업은요?

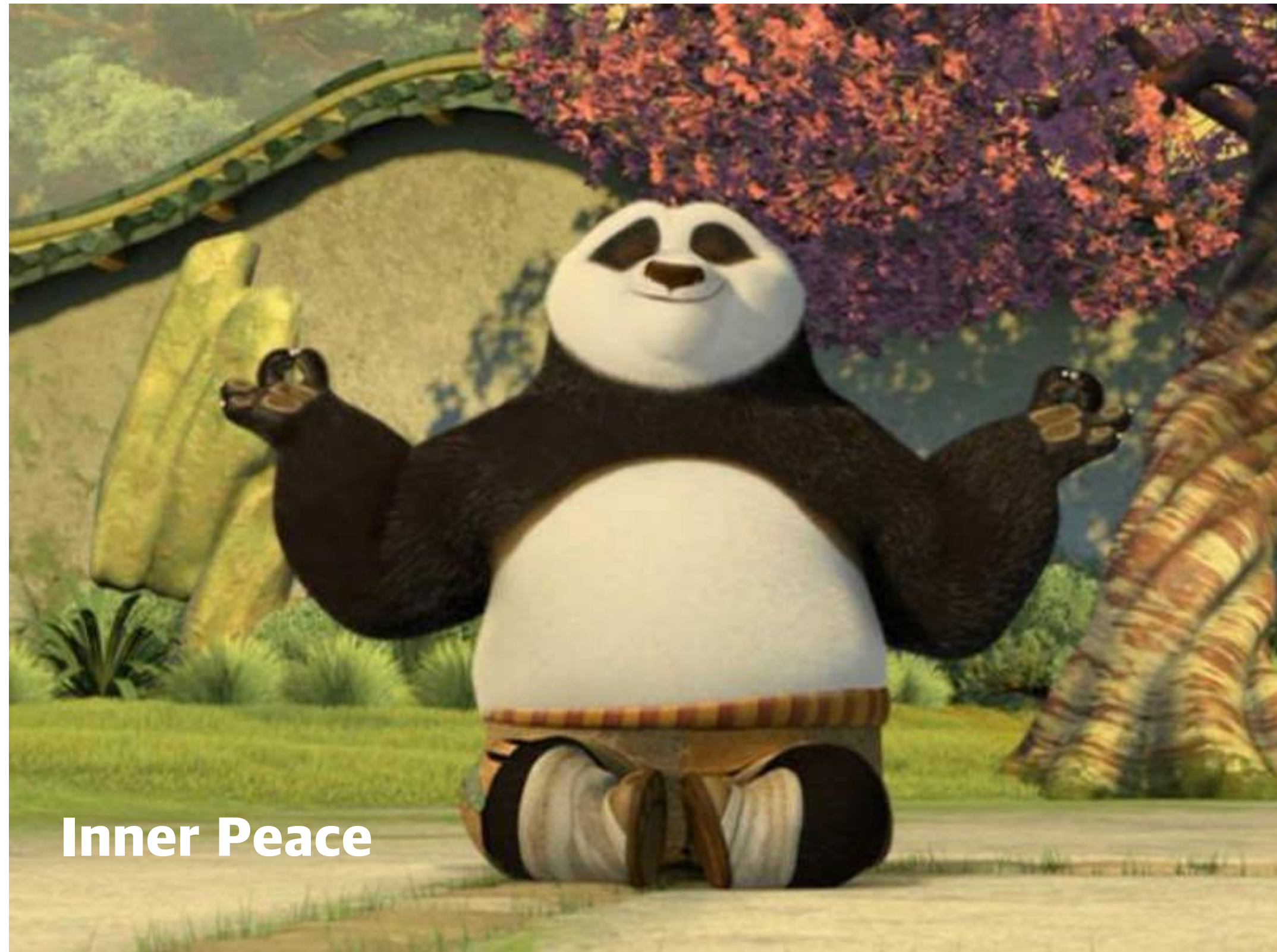
---

- 비전공자 대상수업으로 기초부터 시작합니다.
- 조금이라도 모르는 것은 알때까지 의심하고, 질문하세요.
- 저는 등대일 뿐, 각 배의 선장은 여러분이에요.
- 포기하지 마세요! 이제 시작입니다.

# 준비 되셨나요?

---

- 마지막으로 하고 싶으신 말 모두 들어드려요~!



**Inner Peace**

---

# 컴퓨터 이해하기

---

# 컴퓨터 이해하기

---

- 프로그램이란 무엇 일까요?
- 컴퓨터는 어떻게 동작할까요?



# 컴퓨터란?

---

컴퓨터란 수치, 문자, 그림, 소리등의 데이터를 처리하여 사용자가 원하는 정보를 제공해 주는 장치



# 컴퓨터 시스템

---

컴퓨터는 크게 하드웨어와 소프트웨어로 두가지로 구분 된다.

## 하드웨어

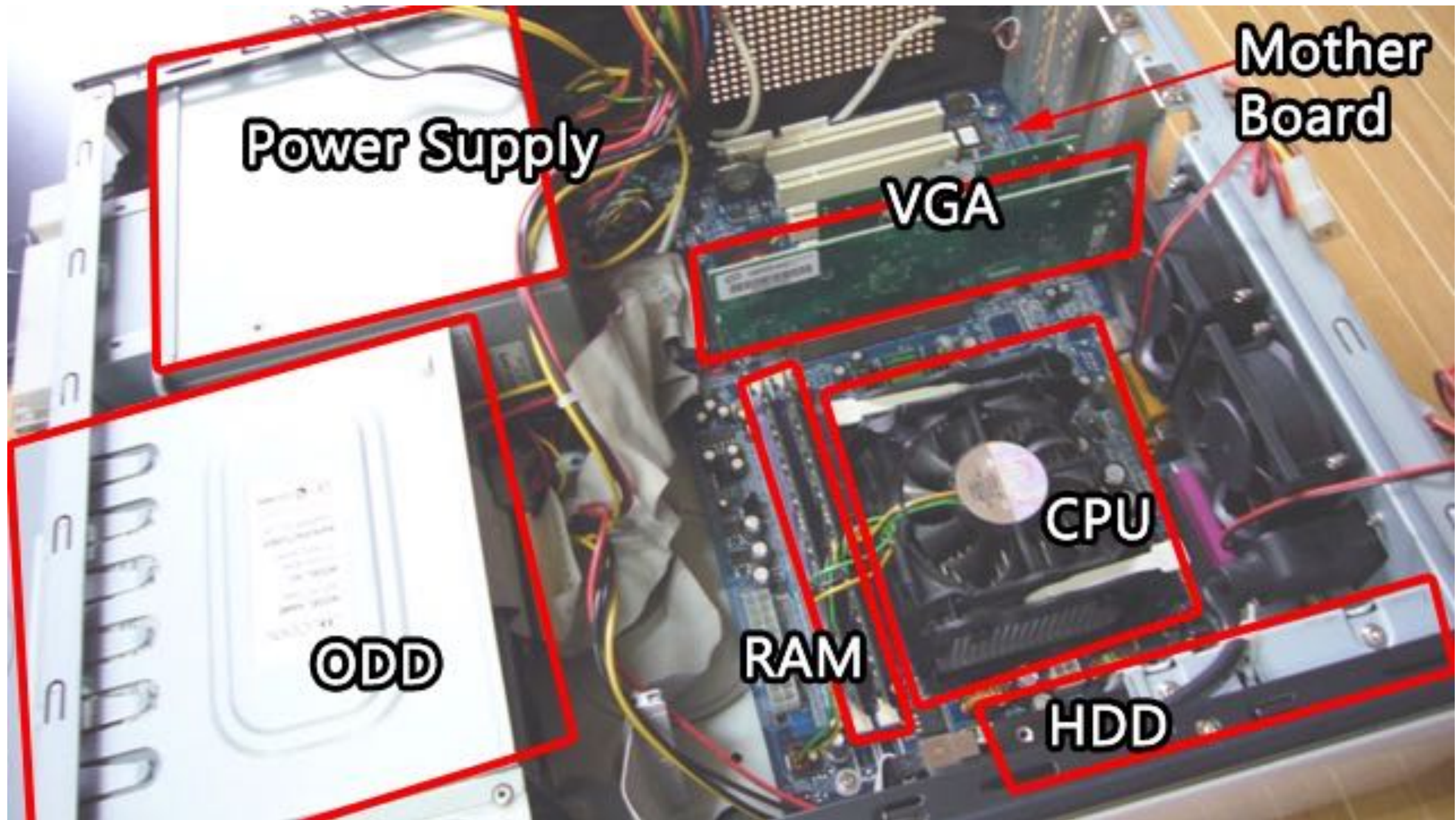
- 중앙처리장치
- 기억장치
- 입력장치
- 출력장치

## 소프트웨어

- 시스템 소프트웨어
- 응용 소프트웨어

# 하드웨어

---

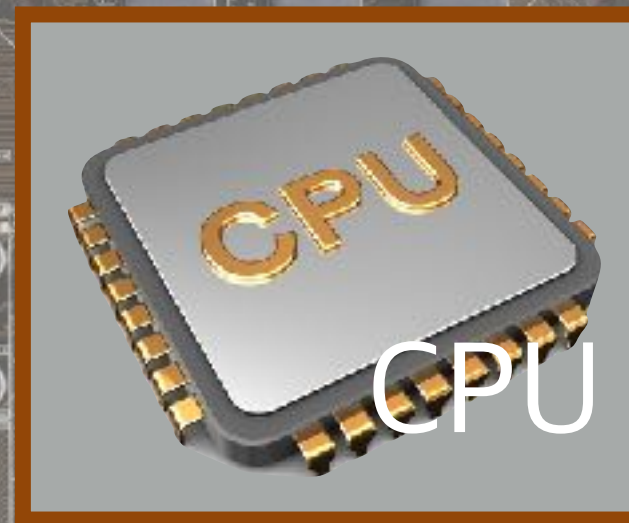




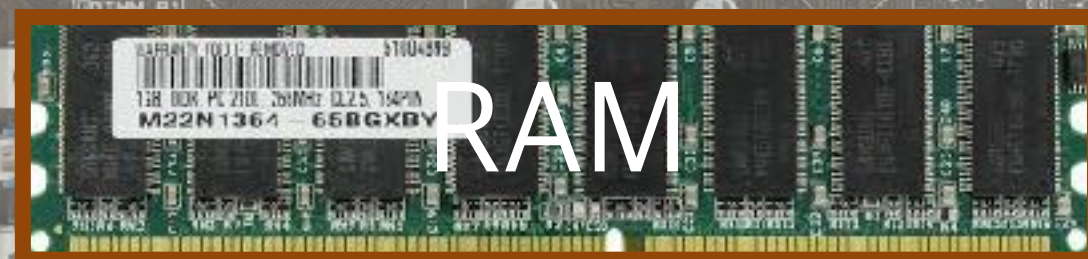
# Mainboard



HDD



CPU



RAM



# 운영체제(OS)

---

운영 체제(運營體制, 문화어: 조작체계) 또는 오퍼레이팅 시스템(영어: Operating System, OS)은 시스템 하드웨어를 관리할뿐 아니라 응용 소프트웨어를 실행하기 위하여 하드웨어 추상화 플랫폼과 공통 시스템 서비스를 제공하는 시스템 소프트웨어이다.



# 컴퓨터 프로그램

---

컴퓨터 프로그램 : 컴퓨터에 의해 실행되는 지시사항의 모음인 **컴퓨터 소프트웨어**의 한 예이다. 대부분의 프로그램은 실행 중(즉, 명령어를 '불러들일' 때)에 사용자의 입력에 반응하도록 구현된 명령어의 집합으로 구성되어 있다. 대부분의 프로그램들은 **하드디스크 등의 매체**에 바이너리 형식의 파일로 저장되어 있다가 사용자가 실행시키면 **메모리로 적재되어 실행된다**....

-위키백과-

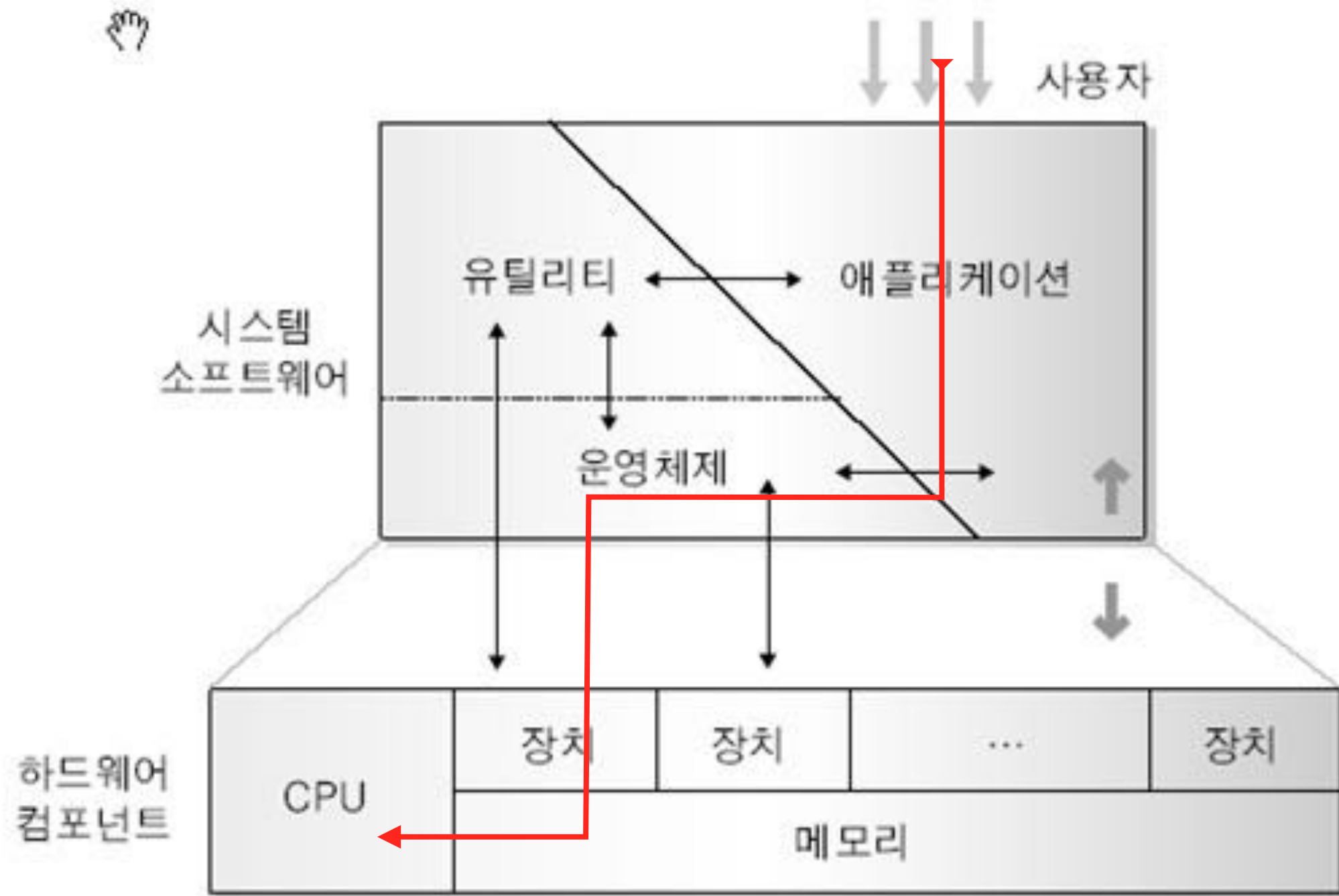
프로그램? vs 프로세스?

# 응용프로그램

---

응용 소프트웨어(영어: application software)는 넓은 의미에서는 운영 체제에서 실행되는 모든 소프트웨어를 뜻한다.

# 컴퓨터 작동 원리





---

# 프로그래밍

---

# 천조국 아빠의 교육법

---

<https://www.youtube.com/watch?v=GrsqZSDMKO4>

# 컴퓨터 프로그램

---

컴퓨터 프로그램 : 컴퓨터에 의해 실행되는 지시사항의 모음인 컴퓨터 소프트웨어의 한 예이다. 대부분의 프로그램은 실행 중(즉, 명령어를 '불러들일' 때)에 **사용자의 입력에 반응하도록 구현된 명령어의 집합**으로 구성되어 있다. 대부분의 프로그램들은 하드디스크 등의 매체에 바이너리 형식의 파일로 저장되어 있다가 사용자가 실행시키면 메모리로 적재되어 실행된다....

-위키백과-

프로그램을 만드는 행위 = 프로그래밍

# 프로그래밍 언어

---

- 프로그래밍 언어란 컴퓨터에서 일을 수행하기 위한 프로그램을 작성하기 위해 사용하는 기호체계를 의미한다.

# 저급언어

---

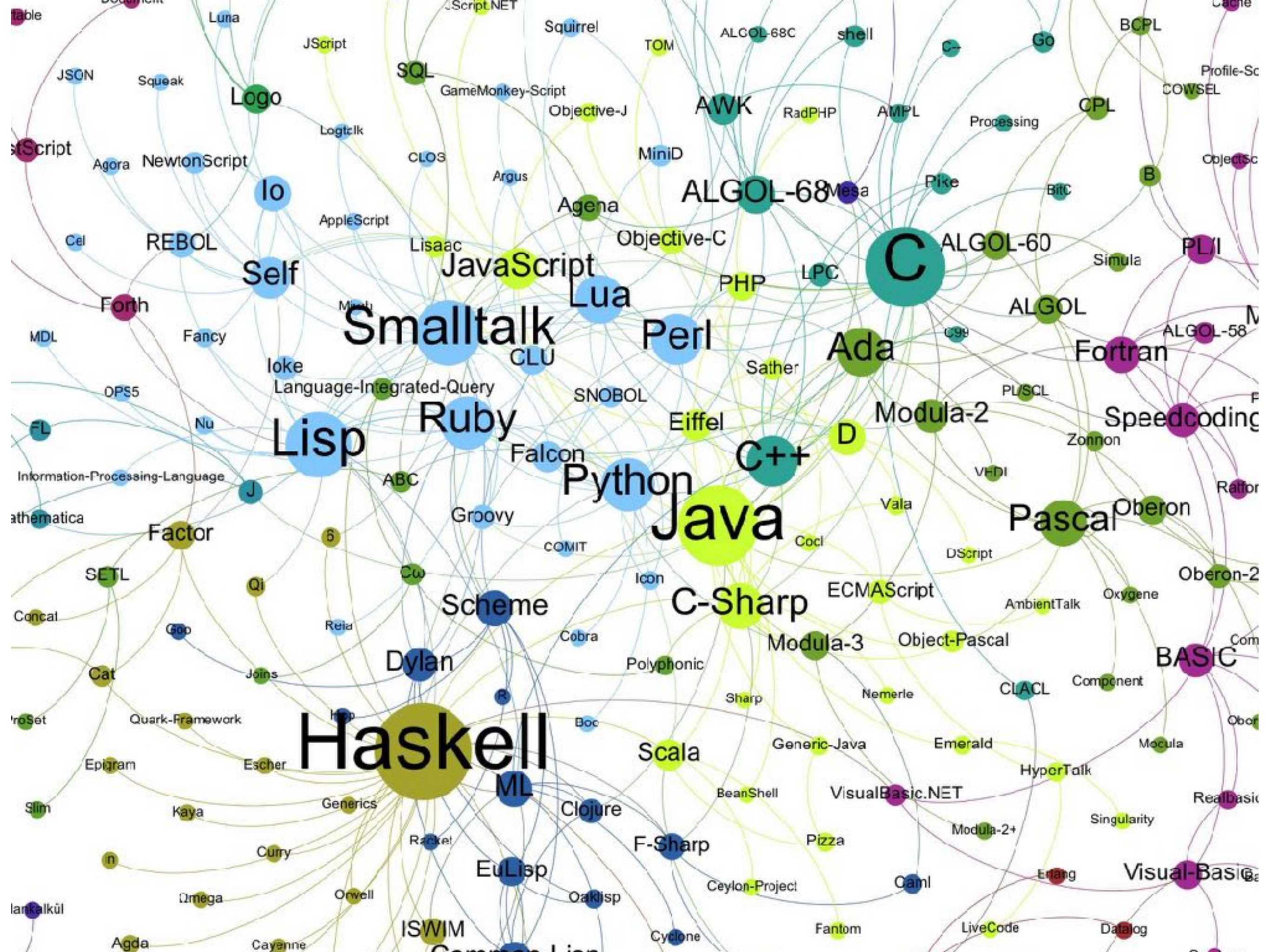
- 기계어 : 0과 1로 된 이진수 형태의 언어, 컴퓨터가 직접 이해할 수 있는 유일한 언어이다.
- 어셈블리어 : 기호로 나타낸 언어. 어셈블러라는 번역기를 통해 기계어로 번역된다. 기계어보다 쉽게 작성할수 있다.
- 공통점 : 저급언어는 하드웨어에 직접 작용하기 때문에 속도가 빠르나 컴퓨터 시스템에 따라 이해하는 내용이 달라 동작하지 않을수 있고, 프로그래밍 하기가 어렵다.

# 고급언어

---

- 기계적인 특성을 제거한 자동화 프로그래밍 언어로 기계어 대신 개발자가 쉽게 개발할수 있는 기호로 작성할수 있게 도와준다. 컴파일러를 통해 기계어로 변환된다.
- 현재 우리들이 사용하는 대다수의 언어는 고급언어로 작성되어 있다.







# 컴파일 언어

## < 컴파일 언어 vs 스크립트 언어 >

- 컴파일 : 사람이 이해할수 있는 언어를 기계언어로 변환시키는 작업

고급언어 → 기계어

```
main( ) { printf("hello, world\n");}
```

```
377 H 203 304 \b 303 \0 \0 001 \0 002 \0 \0 \0
\0 \0 \0 \0 \0 \0 \0 \0 \0 \0 h e l l o ,
w o r l d \0 \0 \0 \0 001 033 003 ; $ \0 \0
\0 -003 \0 \0 \0 334 376 377 377 @ \0 \0 \0 37....
```

- 종류 : C, java, objective-C, Swift
- 특징 : 실행속도가 빠르다.

미리 컴파일러를 통해 컴파일을 해야한다.  
컴파일 과정에서 버그를 발견할수 있다.



# 스크립트 언어

---

## < 컴파일 언어 vs 스크립트 언어 >

- 인터프리터 : 명령어들을 한 줄씩 읽어들이어서 실행하는 프로그램.
- 종류 : html, javascript, php...
- 특징 : 실행속도가 느리다  
파일로 저장되고 전달된다.  
컴파일 과정이 없어 버그를 찾아내기가 어렵다.

# Swift

---

스위프트(Swift)는 애플의 iOS와 OS X를 위한 프로그래밍 언어로 2014년 6월 2일 애플 세계 개발자 회의(WWDC)에서 처음 소개되었다. 기존의 애플 운영체제용 언어인 오브젝티브-C와 함께 공존할 목적으로 만들어졌다. 오브젝티브-C와 마찬가지로 LLVM으로 빌드되고 같은 런타임을 공유한다. 클로저, 다중 리턴 타입, 네임스페이스, 제네릭스, 타입 유추 등 오브젝티브-C에는 없었던 현대 프로그래밍 언어가 갖고 있는 기능을 많이 포함시켰으며 코드 내부에서 C나 오브젝티브-C 코드를 섞어서 프로그래밍하거나 스크립트 언어처럼 실시간으로 상호작용하며 프로그래밍 할 수도 있다....

# 프로그래밍 환경

---

**통합 개발 환경**(統合開発環境, Integrated Development Environment, IDE)은 코딩, 디버그, 컴파일, 배포 등 프로그램 개발에 관련된 모든 작업을 하나의 프로그램 안에서 처리하는 환경을 제공하는 소프트웨어이다. 종래의 소프트웨어 개발에서는 컴파일러, 텍스트 편집기, 디버거 등을 따로 사용했다. 이러한 프로그램들을 하나로 묶어 대화형 인터페이스를 제공한 것이 통합 개발 환경이다. 최근의 통합 개발 환경(IDE)은 그래픽 사용자 인터페이스(GUI) 응용 프로그램 개발용 고속 개발 도구가 많다.

\*Swift로 개발은 xCode에서 실행