

(산대특) 스마트 팩토리 혁신을 위한 AI 솔루션 개발자 양성과정

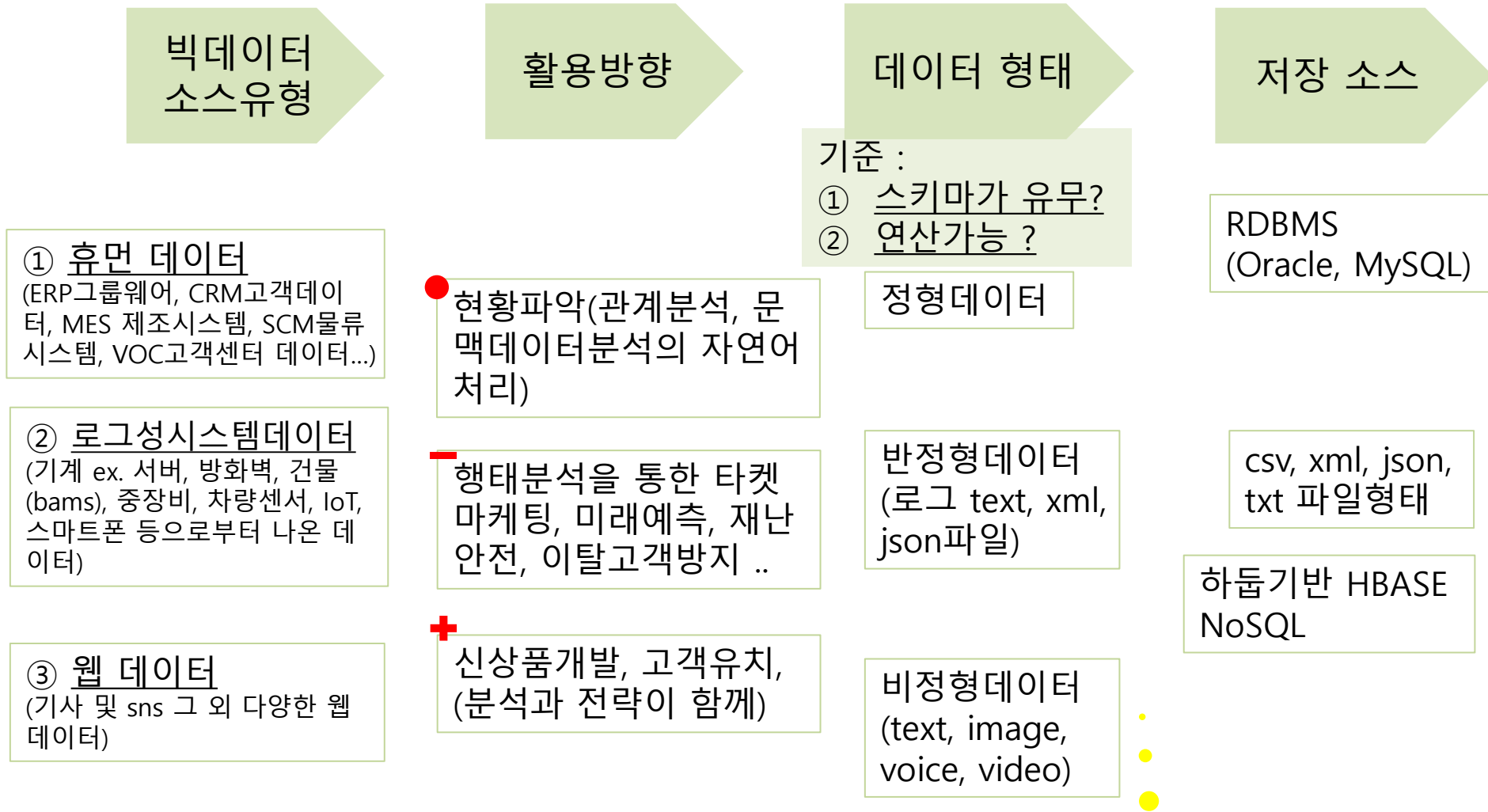
- 이 소 영 강사

yisy0703@naver.com

<https://github.com/yisy0703/ai>

WWW192.168.0.94

AI 빅데이터의 속성 정리



비정형데이터는 정형화 어플리케이션의 성능이 중요(java, R, Python)

단계별 과정 개요

단계별 과정 개요

1단계 실무 기초



프로그래밍 언어 기본/활용, 데이터 탐색/전처리/시각화

- 분산 객체지향 **프로그래밍 언어의 기본**부터 **활용**과 **고급**에 이르기까지 **코딩 기초 역량 확보**
- 빅데이터 탐색/수정/병합/집계/시각화 등의 작업을 통해 **데이터탐색**과 **전처리 능력**을 배양하고 프로젝트를 통해 실무 역량을 갖추게 함
- 데이터의 수집과 분석에 대한 애플리케이션 개발과 저장, 분석, 시각화에 이르는 빅데이터 분석 기본 역량 확보



2단계 실무 심화



머신러닝, 딥러닝, 인공지능망, 시각인지시스템

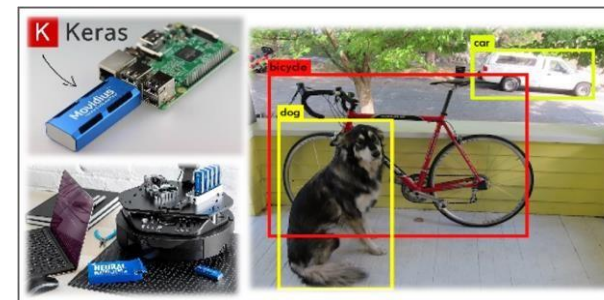
- 빅데이터 분산 플랫폼과 **머신러닝 기법**을 통해 빅데이터를 분석할 수 있는 역량 확보
- 딥러닝 개발환경 구축 및 **인공신경망** 알고리즘의 이해 및 적용 역량 확보
- OpenCV를 활용한 **시각인지 알고리즘** 구현
- 심층 신경망 구현, 합성곱 신경망 구현, 신경망 최적화 역량 확보



3단계 프로젝트

AI기반 머신러닝과 딥러닝을 이용한 실시간 빅데이터 분석 및 인지 시스템 개발 프로젝트

- 프로젝트 전반의 관리·지도와 빅데이터 분석가 및 서비스 개발자의 멘토링이 결합된 **실질적인 결과물 중심의 프로젝트** 수행
- 빅데이터 분야의 아이디어에 대한 검증과 모델링, 서비스 구현 과정을 통해 전문 서비스 개발자로서의 역량 확보
- 인공지능 딥러닝을 활용한 **실시간 시각인지 프로그램 개발** 역량 확보





RDBMS

Sqoop

HDFS

Spark
EDA(Java, Python, SQL)

IoT기기
로그데이터

NoSQL

Pig

SQL로 접근

HIVE

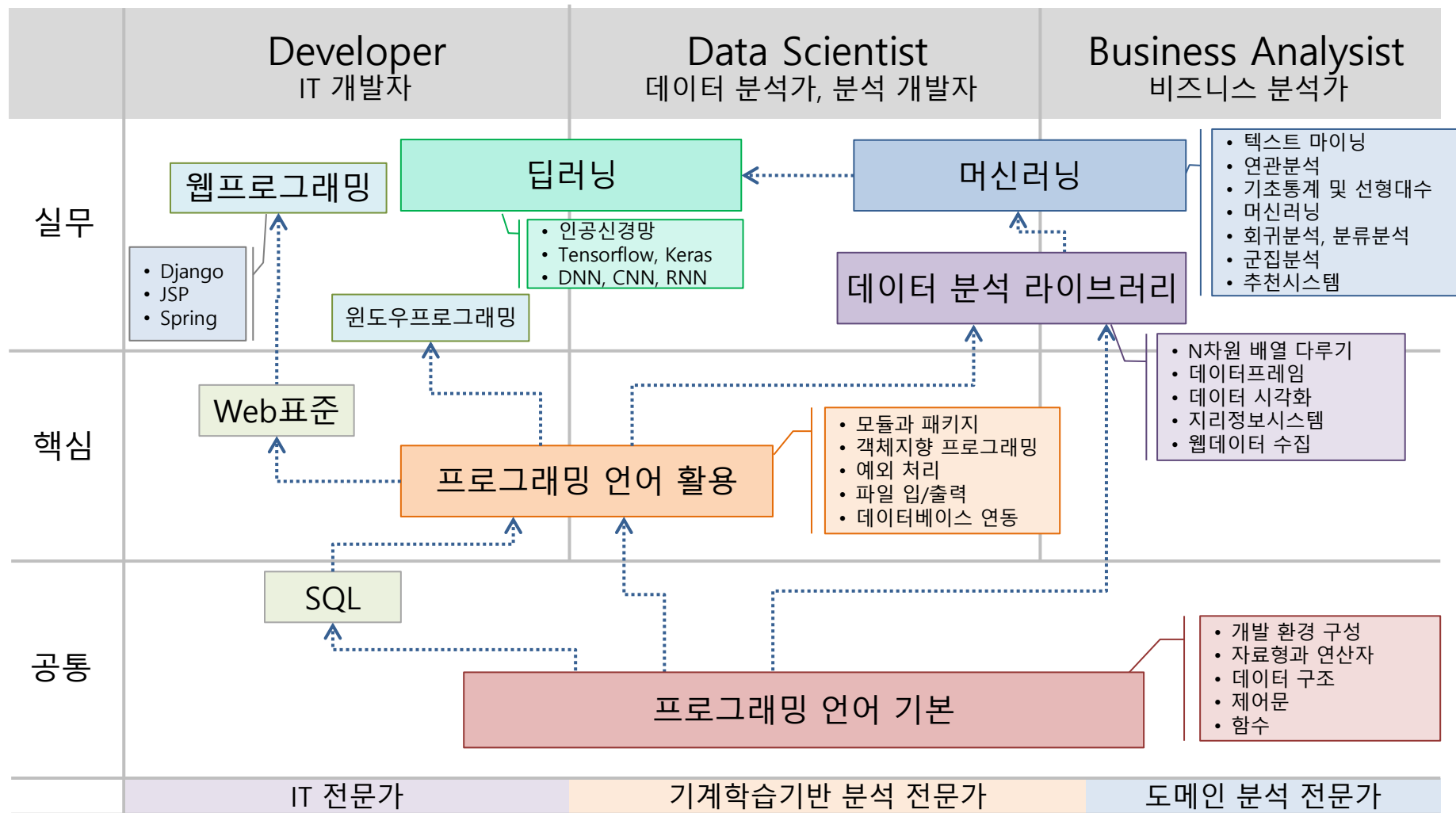
공공데이터
(data.go.kr.)

Web데이터
(기사, SNS...)

크롤러
(Python, R, Java)

Python
(머신러닝 - 통계기반,
딥러닝-지도학습과 비지도학습)

로드맵



일정

- ✓ JAVA 및 JDBC, JSP 4주
- ✓ DB (oracle, MySQL) 2주
- ✓ 웹표준(HTML, CSS, JavaScript, jQuery) 3주
- ✓ Python 3주
- ✓ 머신러닝 1주
- ✓ 딥러닝, 자연어처리 각 1주
- ✓ OpenCV 및 YOLO 1주
- ✓ Flask 및 django 2주
- ✓ 프로젝트(개별프로젝트 2주, 팀프로젝트 3주) 5주
- ✓ 취업지원팀 : 이력서 특강, 취업특강 등

이 과정을 마치면...

기본 라이브러리 불러오기

```
In [1]: 1 import tensorflow as tf
2 print(tf.__version__)
3
4 from tensorflow import keras
5
6 import numpy as np
7 import pandas as pd
8 import matplotlib.pyplot as plt
```

1.14.0

데이터셋 불러오고 학습 데이터와 검증 데이터로 나누기

```
In [2]: 1 redwine = pd.read_csv("winequality-red.csv", sep=";")
2 redwine_X = redwine.iloc[:, :-1]
3 redwine_y = redwine.iloc[:, -1] - 3
4
5 from sklearn.model_selection import train_test_split
6 train_X, test_X, train_y, test_y = train_test_split(redwine_X,
7                                                    redwine_y,
8                                                    test_size=0.2)
```

💡 딥러닝 라이브러리 Keras를 이용한 분류 예

경량 모델 정의하기

```
In [3]: 1 model = keras.Sequential([
2     keras.layers.Flatten(input_shape=(11,)),
3     keras.layers.Dense(100, activation='relu'),
4     keras.layers.Dropout(0.2),
5     keras.layers.Dense(200, activation='relu'),
6     keras.layers.Dropout(0.4),
7     keras.layers.Dense(50, activation='tanh'),
8     keras.layers.Dropout(0.1),
9     keras.layers.Dense(6, activation='softmax')
10 ])
11
12 model.compile(optimizer='adam',
13               loss='sparse_categorical_crossentropy',
14               metrics=['accuracy'])
```

손실함수와 옵티마이저 정의하기

```
In [4]: 1 history = model.fit(train_X, train_y.to_numpy(), epochs=300)
```

Epoch 1/300
1119/1119 [=====] - 0s 161us/sample - loss: 1.2863 - acc: 0.4343

학습시키기

평가하기

```
In [5]: 1 test_loss, test_acc = model.evaluate(test_X, test_y, verbose=1)
2 print('\n테스트 정확도:', test_acc)
```

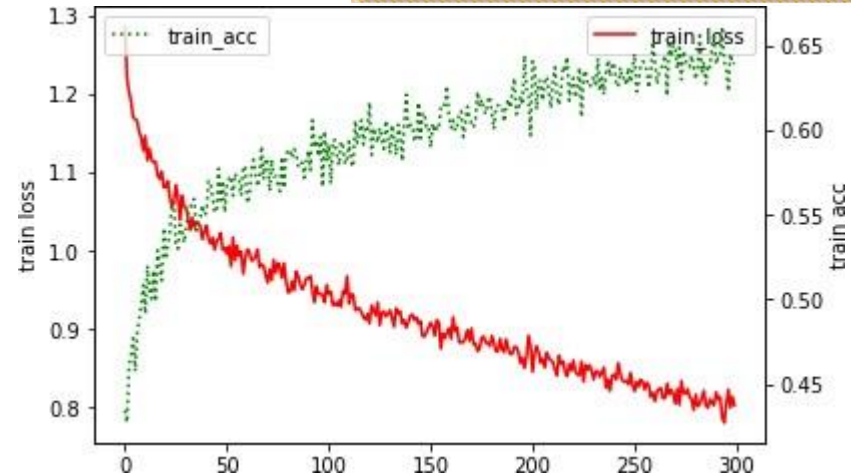
480/480 [=====] - 0s 94us/sample - loss: 1.0884 - acc: 0.6000

테스트 정확도: 0.6

학습 모델 시각화하기

```
In [6]: 1 import matplotlib.pyplot as plt
2 %matplotlib inline
3 fig, (ax1, ax2) = plt.subplots(figsize=(6,4))
4 ax1.plot(history.history['loss'], 'r-', label="train_loss")
5 ax1.set_ylabel("train loss")
6 ax1.legend()
7 ax2 = ax1.twinx()
8 ax2.plot(history.history['acc'], 'g:', label="train_acc")
9 ax2.set_ylabel("train acc")
10 ax2.legend()
11 plt.show()
```

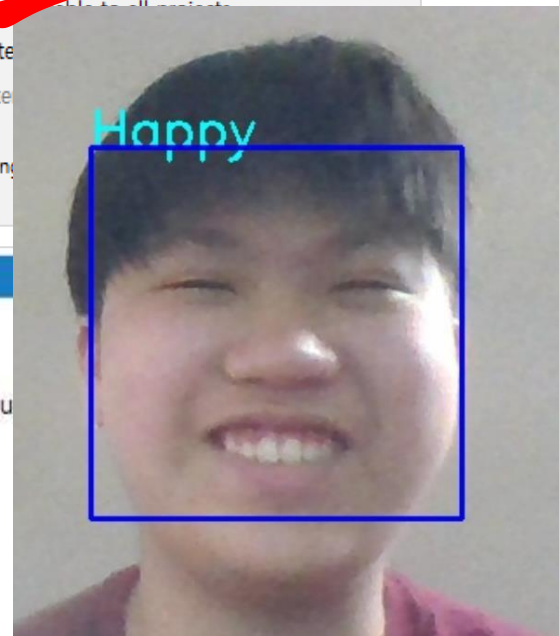
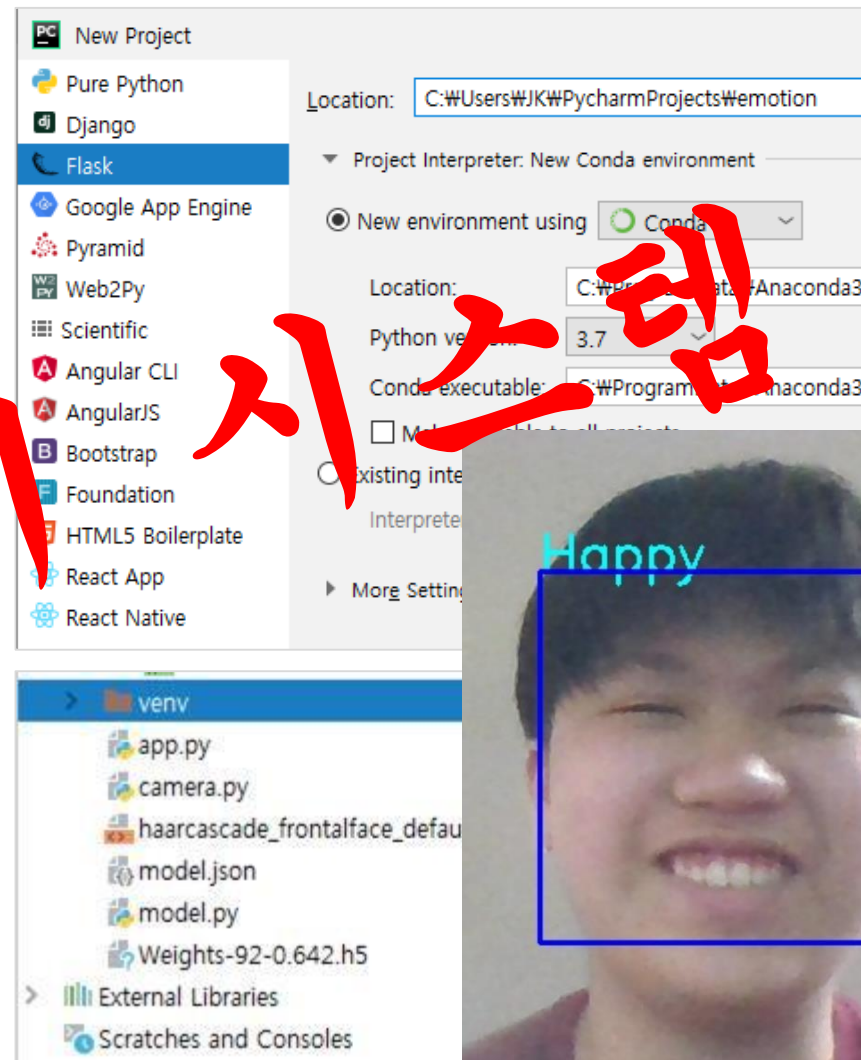
학습 시 손실과 정확도를 그래프로 표현



이 과정을 마치면...



시스템
시각인식



이 과정을 마치면...



컴퓨터 프로그래밍이란?



컴퓨터



C
C++
Java
Python
JavaScript
...
언어



언어를
이용해서
프로그램을
만드는
행위

컴퓨터 프로그래밍

```
System.out.println(2);  
System.out.println('Hello');  
...
```

컴퓨터 프로그래밍 소스

컴파일
단계



컴파일러 결과
(프로그래밍)

```
0101 0010 1010 1000  
0000 0101 1011 1010
```

...
컴퓨터가 이해할 수 있는 형태

어떤 언어를 해야 할까?

- CPU 종속적인 언어

Intel 사 CPU



어셈블리 PG A

AMD사 CPU



어셈블리 PG A (탑재불가)

어떤 언어를 해야 할까?

CPU 독립적인 언어 = OS 종속적인 언어

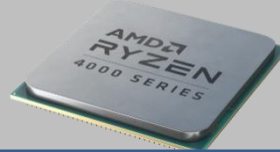
Intel 사 CPU



OS (Windows)

a.c -> a.exe

AMD사 CPU



OS (Windows)

a.c -> a.exe

Intel 사 CPU



OS (MacOS)

a.c -> a.exe 탑재불가

어떤 언어를 해야 할까?

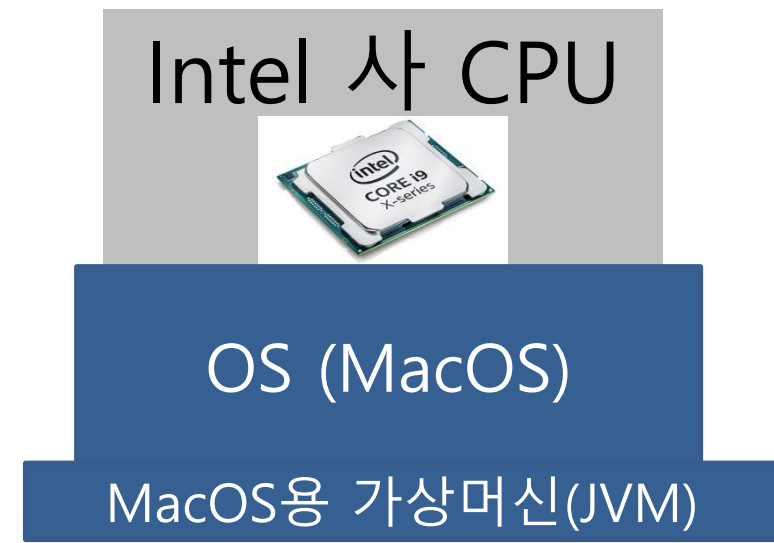
OS 독립적인 언어(=플랫폼 독립적인 언어)



a.java -> a.class
a.py



a.java -> a.class
a.py

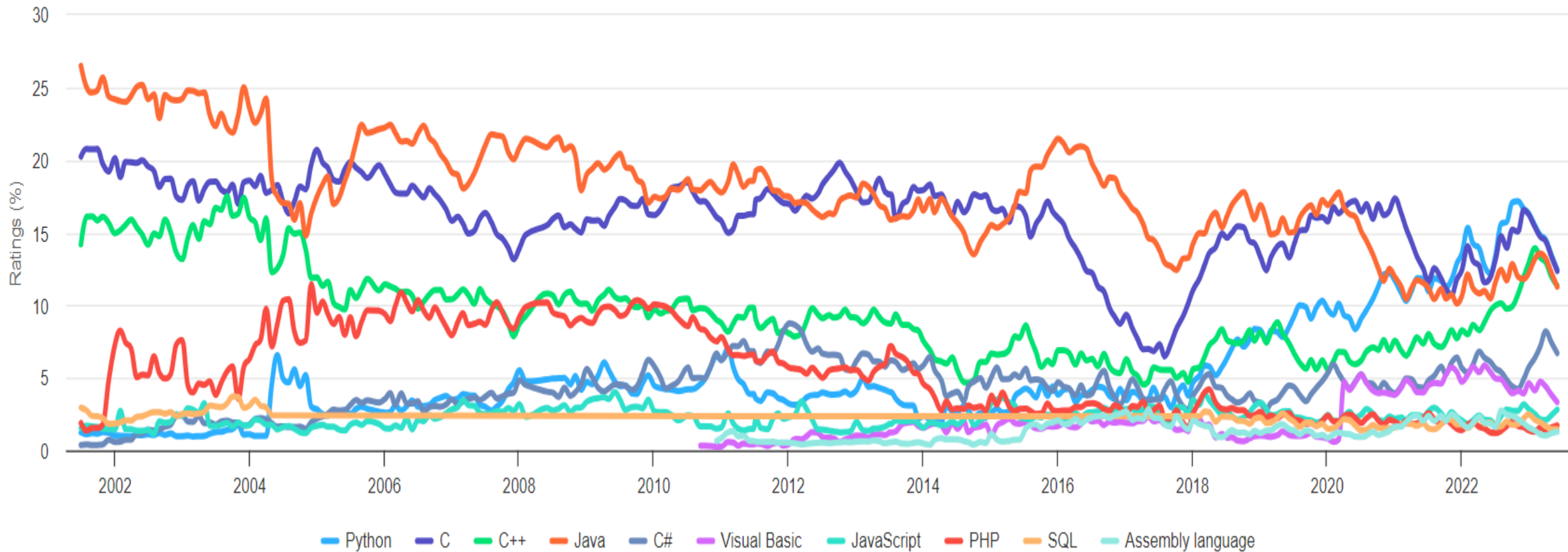


a.java -> a.class
a.py

Java 웹 개발자 양성과정

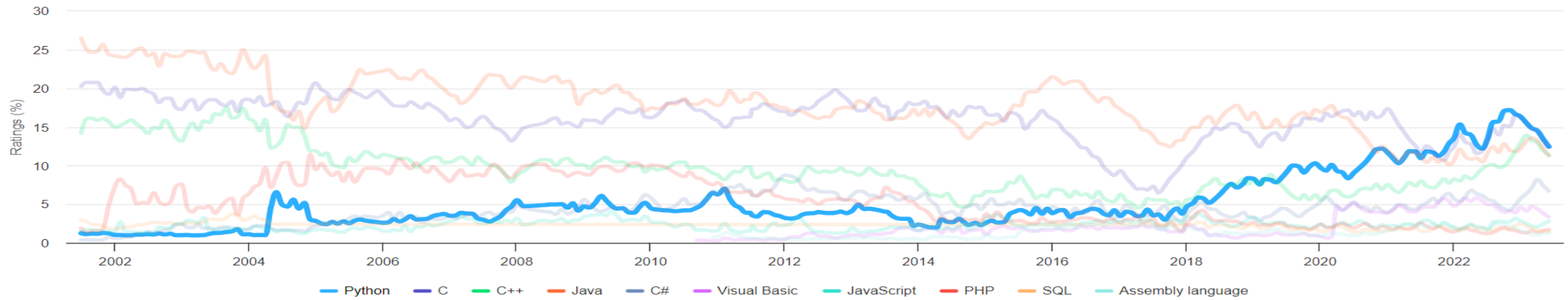
TIOBE Programming Community Index

Source: www.tiobe.com

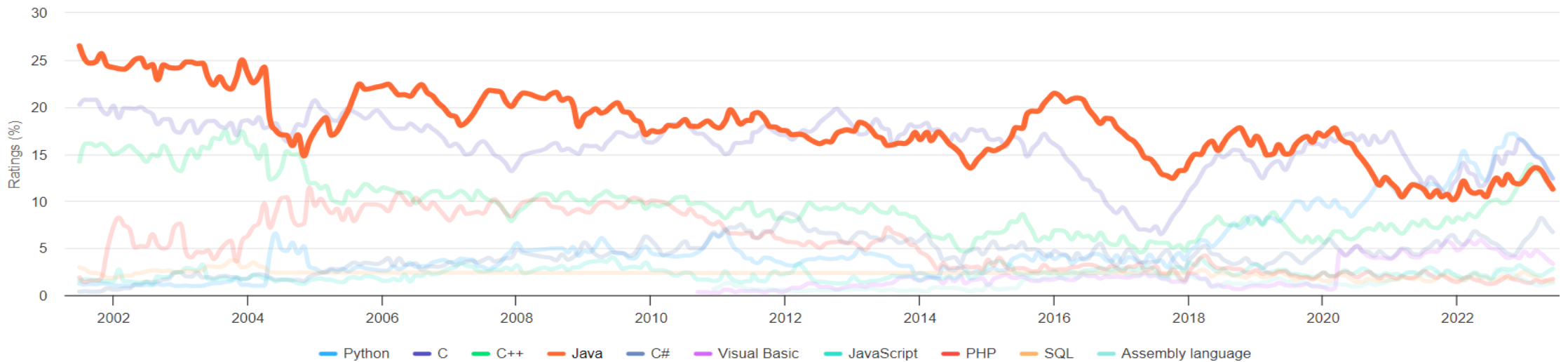


source : <https://tiobe.com/tiobe-index>

Source: www.tiobe.com



Source: www.tiobe.com



초보 개발자는 토끼와 거북이 중 거북이가 되자

시스템 폴더 관리

ai₩DOWNLOAD : 개발환경을 위한 다운로드 파일

ai₩IDE : 통합개발환경 폴더

ai₩source : 수업시간 개발 소스

ai₩letNote : 교안 노트

ai₩note : 그날 그날 수업한 내용을 10~20줄로 요약한 파일

ai₩취업 : 취업에 관련된 정보 및 이력서 포트폴리오

시스템 설치

1. Chrome 브라우저 <https://www.google.com/chrome/browser>

2. 자바(jdk17) : <https://www.oracle.com/>

3. 이클립스 IDE(jdk 버전과 맞는 버전의 eclipse) :

<https://www.eclipse.org/downloads/packages/release/2023-06/r>

(참조) <https://soif0180.tistory.com/entry/Eclipse-이클립스-버전별-정보-및-JDK-최소-요구-버전>)

4. Git <https://git-scm.com>

1. 내 소스 코드 저장
2. 버전 관리 도구
3. 소스 코드 공유
4. 협업 공간

깃허브의 레파지터리(원격저장소) 생성하고 올리기

1. <http://www.github.com>에 가입하기(sign up) 후 코드가 포함된 메일 확인
2. Sign in
3. Create repository(올릴 공간명)
4. 소스를 내컴퓨터에서 github 인터넷으로 올리고 git 명령어를 사용하기 위해 git-scm을 다운받고 install (<https://git-scm.com/downloads>)
5. Git Bash에서 git 환경 설정
 - > git config --global user.name "comlec0703"
 - > git config --global user.email "comlec0703@gmail.com"
 - > git config --list (확인)

깃허브의 레파지터리(원격저장소) 생성하고 올리기

6. 내 pc의 repository 폴더에 다음의 파일 추가

① README.md (git 소개)

② .gitignore (git 공간에 올리지 않을 파일과 폴더 리스트)

(d:/webPro/.gitignore d:/webPro/source/1_Java/.gitignore)

7. git 공간에 자료 올리기

> git init (초기화)

> git add . (공유 추가할 파일 더하기)

> git status (상태 확인 – 선택)

> git commit -m "first commit" (히스토리 버전 이름)

> git branch -M main (branch 생성 – branch 생성을 하지 않을 경우 master로)

> git remote add origin 본인깃주소 (연결고리)

> git remote -v (확인 – 선택)

> git push -u origin main (branch에 올리기 branch를 생성하지 않았다면 master)

깃허브의 레파지터리 생성

8. 내 pc와 git이 동기화된 거 확인 후, 추가 파일 만든 후 추가 올리기

- > git add .

- > git commit -m "second"

- > git push -u origin main

9. 추가한 파일 수정

- > git add .

- > git commit -m "third"

- > git push -u origin main

깃허브의 레파지터리 생성

10. 협업 (웹에서 수정하거나 다른 장소에서 수정 후 pull 필수)
- > git clone 주소 폴더이름 (다른 pc에 git 내용 그대로 받기)
 - git pull origin main (git 저장소와 내 PC간 동기화)

수정 후

- > git add .
- > git commit -m "last"
- > git push origin main