

STEP. 01

파이썬 시작

파이썬 기본 다지기



01 Python Programming 의미

- Python
 - 컴퓨터 시스템을 구동시키는 소프트웨어를 작성하기 위한 형식 언어
 - 인간이 컴퓨터에 명령을 내릴 때 필요한 프로그래밍 언어
- Programming
 - 인간이 생각하는 것을 컴퓨터에 명령하는 것
 - 프로그램을 만드는 모든 작업이며 개발이라고 칭하기도 함
- Python Programming
 - Python 언어를 이용하여 컴퓨터에 명령을 내리는 행위












02 Python 소개

- 가장 활용도가 높고 쉬운 프로그래밍 언어
- MATLAB, R과 같은 도메인 특화 언어와 Java, C 같은 범용 언어의 장점을 고루 갖춘
- 통계, 머신러닝, 자연어, 이미지, 시각화 등을 포함한 풍부한 라이브러리를 지님
- 인공지능 개발, 웹과 앱 그리고 게임 제작, 핀테크 및 블록체인 구현으로 활용
- 브라우저 기반의 인터랙티브 프로그래밍 환경인 Jupyter Notebook으로 쉽게 구현



02 Python 소개

- TIOBE(프로그래밍 언어 인기 지표)에서 선정한 프로그래밍 언어 인기도 순위
 - Python 1위 차지

Jun 2022	Jun 2021	Change	Programming Language		Ratings	Change
1	2	▲		Python	12.20%	+0.35%
2	1	▼		C	11.91%	-0.64%
3	3			Java	10.47%	-1.07%
4	4			C++	9.63%	+2.26%
5	5			C#	6.12%	+1.79%
6	6			Visual Basic	5.42%	+1.40%
7	7			JavaScript	2.09%	-0.24%
8	10	▲		SQL	1.94%	+0.06%
9	9			Assembly language	1.85%	-0.21%



03 Python 장점

1. 간결하고 직관적인 문법
 - 인간의 사고와 비슷하여 다른 언어에 비해 빠르게 배워 활용 가능
 - 같은 결과를 요구할 때 Java로 10줄, Python으로 3줄!
2. 높은 확장성과 이식성
 - 다른 언어나 라이브러리에 쉽게 연동 가능
3. 다양한 라이브러리 존재
 - 파이썬을 배우고 필요한 라이브러리만 익힌다면 불가능한 일은 없다!



04 Python 활용 분야

- 인공지능 제작
 - 기계 학습에 도움이 되는 다양한 라이브러리로 쉽게 인공지능 개발 가능
- 웹과 애플리케이션 제작
 - Python으로 제작된 구글, 인스타그램, 넷플릭스, 드롭박스처럼 개발 가능
- 게임 제작
 - 2D 게임부터 3D 게임까지 개발 가능
- 핀테크 및 블록체인 구현
 - 핀테크의 다양한 분야에서 응용되며 블록체인 구현 가능



05 Python Editor 종류

- Editor 란?
 - 소스 코드가 들어 있는 파일을 편집할 수 있는 프로그래밍 툴
 - Python은 편의성을 위해 별도의 에디터 프로그램을 설치하여 사용
- Editor 종류
 - Pycharm – 개발자들이 가장 많이 사용하는 에디터
 - Jupyter – 웹 브라우저에서 파이썬을 작성하고 실행하는 에디터
 - Visual Studio Code – 마이크로 소프트가 개발한 에디터
 - IDLE – 파이썬 설치시 내장되어 있는 기본 파이썬 에디터이며 심화적인 기능은 부족한 편



06 Python Jupyter Notebook

- 오픈 소스 기반의 웹 애플리케이션
- 파이썬으로 작성한 여러 개의 코드와 실행 결과를 하나의 문서처럼 관리 가능
- 즉, 프로그램 코드 + 결과 + 문서를 위한 대화식 개발 환경
- 기존의 파이썬 IDLE을 사용하는 것과 비교했을 때, 일부 코드만 실행하여 결과 확인 가능



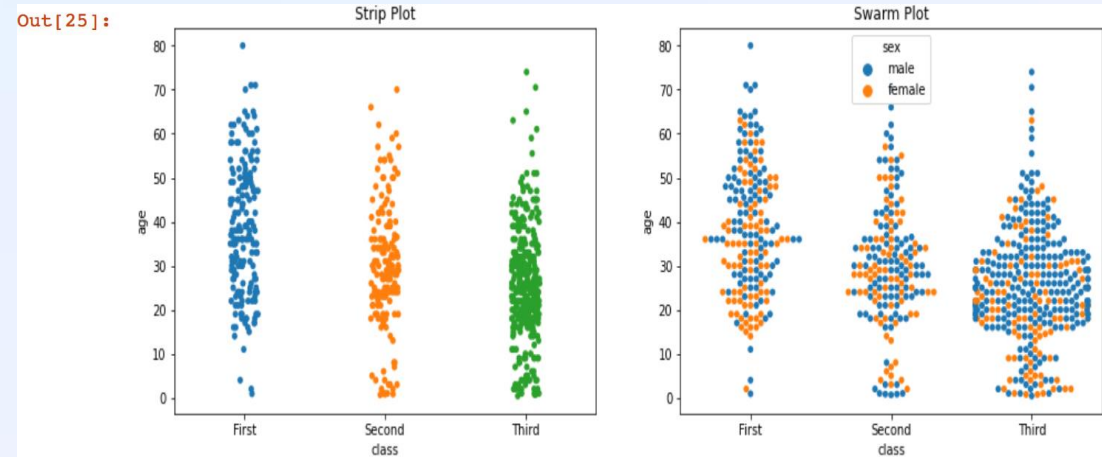
```
In [25]: # 그래프 객체 생성 (figure에 2개의 서브 플롯을 생성)
fig = plt.figure(figsize=(15, 5))
ax1 = fig.add_subplot(1, 2, 1)
ax2 = fig.add_subplot(1, 2, 2)

# 이산형 변수의 분포 - 데이터 분산 미고려
# x축 변수, y축 변수, 데이터 셋, axe 객체(1번째 그래프)
sns.stripplot(x='class', y='age', data=titanic, ax=ax1)

# 이산형 변수의 분포 - 데이터 분산 고려 (중복 X)
# x축 변수, y축 변수, 데이터 셋, axe 객체(2번째 그래프), 성별로 색상 구분
sns.swarmplot(x='class', y='age', data=titanic, ax=ax2, hue='sex')

# 차트 제목 표시
ax1.set_title('Strip Plot')
ax2.set_title('Swarm Plot')

plt.show()
```



Google Colaboratory (Colab)

- Google Colaboratory = Google Drive + Jupyter Notebook
- 구글 코랩은 주피터 노트북 기반의 오픈소스 프로젝트
- 구글 코랩의 장점
 - 대부분의 파이썬 패키지들이 설치 되어 있음
 - 구글 아이디와 인터넷만 있다면 하드웨어와 장소에 구애받지 않고 코딩 가능
 - 구글 드라이브와 연동하여 파일 불러오기
 - 무료로 GPU도 사용할 수 있어서 고성능 딥러닝 프로젝트 가능
 - 주피터 노트북 문서를 여러 사람이 동시에 열어서 함께 편집 가능

colab

