

2016 SWIFT SCHOOL

비전공자를 위한

DATA ANALYTICS 2탄 (PYTHON)

강사 소개

▶ 류 원 경

▶ Mail: me@lyuwonkyung.com

▶ 1~4주차: 파이썬 기초를 강의할 예정

▶ 경제학 전공

▶ 비전공자들이 프로그래밍을 어떻게 배워야 할지에 대한 감각 (나름) 우수!

▶ 미래창조과학부 SW Maestro

▶ 다양한 실무 경력 및 프로젝트 경험 보유

▶ 현재 스타트업 '주식회사 에이비일팔공'의 Co-founder (<https://airbridge.io/>)



강사 소개

▶ 정 헌 재

▶ Mail: hunjaege@gmail.com

▶ 5~8주차: 실전 데이터 분석을 강의할 예정

▶ 수학 전공

▶ 프로그래밍 지식 뿐만 아니라 데이터 분석에 꼭 필요한 수학적/통계적 지식 다량 보유!

▶ 미래창조과학부 SW Maestro

▶ 다양한 프로젝트 경험 보유

▶ 현재 스타트업 '주식회사 에이비일팔공'의 Co-founder (<https://airbridge.io/>)



커리큘럼

- ▶ 1주차: Python을 소개합니다! - 코딩을 위한 준비 (류원경)
 - ▶ 바로 오늘! 시작이 반!
 - ▶ Python 시작 준비: 설치
 - ▶ “Hello world” 프로그램 만들어보기
 - ▶ 변수와 자료형에 대해 알아보기

커리큘럼

- ▶ 2주차: Python 기초 프로그래밍 (류원경)
 - ▶ 이제 슬슬 제대로 해볼까요?: 전용 에디터 준비
 - ▶ 프로그래밍의 기본 중 기본 '제어문'
 - ▶ 나와 상호작용하는 프로그램을 만들기 위한 '입출력'
 - ▶ 간단 프로젝트: 구구단 프로그램을 만들어보자!

커리큘럼

- ▶ 3주차: Python 프로그래밍 응용 (류원경)
 - ▶ 복습: 덧셈 계산기 만들어보기
 - ▶ 프로그래밍의 꽃 '함수': 사칙연산 계산기 만들어보기
 - ▶ 데이터 분석을 위한 '문자열 처리'

커리큘럼

- ▶ 4주차: 외부 패키지를 활용한 파이썬 프로그래밍 (류원경)
 - ▶ 파이썬이 사랑 받는 이유 풍부한 '외부 패키지'
 - ▶ 마법같은 외부패키지 사용법
 - ▶ 실전: 외부패키지를 이용해 데이터 수집해보기
 - ▶ 분석을 하려면 데이터가 있어야 하니까!

커리큘럼

- ▶ 5주차: 데이터 분석 1 (정헌재)
 - ▶ 이제 기초는 끝났다! 파이썬으로 데이터를 만져보자!
 - ▶ 데이터 분석 외부 패키지(numpy, pandas) 사용법 익히기
 - ▶ 샘플 데이터로 직접 데이터 다뤄보기 (데이터는 내 생각만큼 깨끗하지 않아요. 데이터를 활용가능한 'Workable format'으로 만듭시다.)

커리큘럼

- ▶ 6주차: 데이터 분석 2 (정헌재)
 - ▶ 음.. 지금 데이터의 트렌드는 어떨지?
 - ▶ 데이터 간의 상관관계(Corelation)를 찾아봅시다!
 - ▶ Scikit learn 패키지를 활용한 'Linear regression'

커리큘럼

- ▶ 7주차: 데이터 시각화 (정헌재)
 - ▶ 데이터 분석의 꽃 '데이터 시각화'
 - ▶ 파이썬 시각화 라이브러리를 통해, 데이터를 '이야기'하는 방법에 대해 알아보시다.
 - ▶ 당신의 발견을 공유하는 것은 인지상정! 시각화된 데이터 Export 하기

커리큘럼

- ▶ 8주차: 실전 데이터 프로젝트 (정헌재)
 - ▶ 짹짹! 드디어 마지막! 여기까지 잘 따라 오셨다면 당신은 이미 Python을 이용한 데이터 분석 고수!
 - ▶ Titanic 데이터 분석 프로젝트 해보면서 타이타닉호의 생존자를 예측해봅시다!

SESSION 1

PYTHON을 소개합니다!

(코딩을 위한 준비)

프로그래밍, PYTHON이란 무엇일까요?

- ▶ 컴퓨터 프로그래밍?
 - ▶ 말 그대로 '컴퓨터 프로그램'을 만드는 것. (ing!)
- ▶ 컴퓨터 프로그램?
 - ▶ 컴퓨터가 무슨 일을 하도록 '명령'들을 모아놓은 것.
- ▶ 명령?
 - ▶ 어떻게 컴퓨터에게 명령을 할 수 있을까?

프로그래밍, PYTHON이란 무엇일까요?

- ▶ 사람의 언어, 컴퓨터의 언어
 - ▶ 명령을 하려면 서로 의사소통이 되어야한다.
 - ▶ 사람들도 한국어, 영어, 중국어, 일본어.. 다양한 언어로 소통을 하듯이, 컴퓨터에도 다양한 언어들이 존재한다.
 - ▶ 그 중 우리가 배우는 것이 바로 'Python'이라는 '프로그래밍 언어'



프로그래밍, PYTHON이란 무엇일까요?

- ▶ 한글은 조선시대에 세종대왕님이 만듬.
- ▶ Python은 1991년 '귀도 반 로섬'이라는 네덜란드 양반이 만듬.
(지금은 구글에서 나와 드롭박스에서 일하고 있다고 함.)
- ▶ 파이썬은?
 - ▶ 쉽다: 복잡한 컴퓨터 공학적 사전 지식이 (거의) 필요하지 않다. 문법이 간단하다. 뭐든지 빨리 만들어낼 수 있다.
 - ▶ 덕분에 많은 사람들이 사랑해준다. -> 좋은 도구와 선례들이 많이 나와 있음
-> 더 많은 사람들이 배우고 사랑해준다. -> 대세가 되었다. (반복!)



PYTHON을 시작하기 위한 준비: 설치

- ▶ 드디어 무언가를 해봅시다!: 내 컴퓨터를 개발머신으로 만들어봅시다.
 - ▶ 듣기만 해도 신나죠? 이제 개발자가 되는겁니다!
 - ▶ 더 이상 내 컴퓨터는 그냥 '게임용', '서핑용', '업무용' 이 아닌거죠! '개발용' 머신이 되는겁니다! 무엇이든 창조해 낼 수 있죠.
- ▶ 윈도우를 사용하시는 분?
- ▶ 맥을 사용하시는 분?

PYTHON을 시작하기 위한 준비: 설치

▶ 윈도우 쓰시는 분!

- ▶ 주소창에 아래 주소를 또박또박 안틀리게 잘 따라 치세요.

<http://bit.ly/1ZM0YxK>

- ▶ 다운로드 받으시고, 설치하세요. (다음 버튼을 잘 누르면 됩니다.)

▶ 맥 쓰시는 분!

- ▶ 이미 여러분도 모르게 깔려 있습니다.. ㅎㅎㅎ...

프로그래밍 세계에 오신것을 환영합니다: HELLO WORLD!

- ▶ 각자 환경에 따라 Python을 실행해봅시다.
 - ▶ 윈도우: Windows+R -> 실행창에 'cmd' 입력 -> 'python' 이라고 입력 후 엔터
 - ▶ 맥: Alt+SpaceBar -> '터미널' 실행 -> 'python' 이라고 입력 후 엔터
- ▶ 아래의 깜빡거리는 화면이 나오면 성공입니다.

```
~ ➤ python
Python 2.7.10 (default, Jul 14 2015, 19:46:27)
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.39)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

PYTHON을 시작하기 위한 준비: 설치

- ▶ (간혹) 윈도우에서 설치는 잘 했는데 실행이 안되어요!
 - ▶ 큰일이군요.
 - ▶ 아까 본 검은창에 아래를 토씨하나 틀리지 말고 잘 입력해보세요!
(도와주세요 정선생님!)

setx PATH

"C:\Python27;C:\Python27\Scripts
;C:\Python27\Lib\site-packages"

프로그래밍 세계에 오신것을 환영합니다: HELLO WORLD!

- ▶ 그냥 따라 쳐 보세요. 그리고 엔터를 쳐 보세요.

```
print "Hello World!"
```

- ▶ 환영합니다. 여러분은 지금 막 Python이라는 컴퓨터 '언어'를 통해 컴퓨터에게 첫 '명령'을 내렸습니다.
- ▶ 프로그래머(Programmer)가 되셨군요!

프로그래밍 세계에 오신것을 환영합니다: HELLO WORLD!

- ▶ 아직 뭐가 뭔지 하나도 모르겠죠?
- ▶ 그냥 한 번 연습해봅시다.
 - ▶ Hello World 말고
 - ▶ “Python Rules” 출력해보기.
 - ▶ “나는 데이터 분석의 왕이 될꺼야!” 출력해보기.
 - ▶ 새해 소원 하나 출력해보기.

프로그래밍 세계에 오신것을 환영합니다: HELLO WORLD!

- ▶ 이렇게 잘 나왔으면 성공입니다!

```
>>> print "Hello World!"  
Hello World!  
>>> print "Python Rules!"  
Python Rules!  
>>> print "나는 데이터 분석의 왕이 될꺼야!"  
나는 데이터 분석의 왕이 될꺼야!  
>>> print "새해에는 100억을 벌어야지!"  
새해에는 100억을 벌어야지!  
>>> 
```

변수와 자료형에 대해 알아보기

- ▶ 뭔가 단원 제목이 어려워진다..
- ▶ 뭔진 모르겠지만 일단 한 번 해보자. 아래 내용을 파이썬 검은 화면(앞으로는 멋지게 'Python 쉘' 이라고 부를 예정)에 따라 쳐봅시다!

```
banana = 3000
print banana
coke = 1500
total = banana + coke
print total
```

```
>>> banana = 3000
>>> print banana
3000
>>> coke = 1500
>>> total = banana + coke
>>> print total
4500
```

변수와 자료형에 대해 알아보기

- ▶ 바나나가 3,000원.. 콜라가 1,500원..
- ▶ 이렇게 가격(혹은 그 무엇이랄지도)을 저장한 뒤, 나중에 꺼내 쓸 수 있도록 가르키는 것을 '변수'라고 합니다. 'print banana' 라고 했더니, 저장된 값이 튀어나왔죠!
- ▶ 우리가 한 일은, 바바나의 가격과 콜라의 가격을 더해서 (+) 'total' 이라는 변수에 값을 대입(수학시간 같죠?) 하고, 'print'를 통해 출력했죠.

변수와 자료형에 대해 알아보기

- ▶ 아직 헛갈리시나요? 아래와 같이 입력해보세요.

```
banana = 6000
```

```
print banana                # 6000
```

```
banana = 9000
```

```
print banana                # 9000
```

```
total = banana + coke
```

```
print total                 # 10500
```

- ▶ 변수의 값은 이렇게 바뀔 수 있습니다. 신기하죠?

변수와 자료형에 대해 알아보기

- ▶ 변수에는 숫자만 들어갈 수 있냐? 아닙니다!
- ▶ 이번에는 좀 재밌는걸 해보죠.

```
name = "아이유"  
stupid = "바보"  
truth = name + stupid  
print truth
```

- ▶ 당신은 아이유를 바보로 만들었군요. (이렇게 문자열도 저장 가능하답니다.)

변수와 자료형에 대해 알아보기

- ▶ 한 번 해봅시다!
- ▶ 'name'을 본인 이름으로 바꾸고 다시 (+) 연산을 해서 print 해보세
요.
- ▶ 바보..

변수와 자료형에 대해 알아보기

- ▶ 숫자와 문자열을 넣었던 것 처럼, 변수에는 다양한 종류의 값을 넣을 수 있습니다.
 - ▶ 숫자형 (Number): 당연히도 소수(0.1)나 음수(-1)도 넣을 수 있지요.
 - ▶ 문자열 (String): 쌍따옴표나 따옴표로 감싸야해요.
 - ▶ 리스트 (List)
 - ▶ 딕셔너리 (Dictionary)
 - ▶ 참과 거짓 (Boolean)
 - ▶ 그리고 일단 지금은 몰라도 되는 튜플(Tuple), 집합(Sets) 등등..

변수와 자료형에 대해 알아보기

▶ 리스트 (List)

- ▶ 리스트가 한글로 뭘까요? 목록 같은겁니다.

- ▶ 리스트는 대괄호'[]' 를 이용해 선언합니다.

```
liquors = ["소주", "맥주", "막걸리", "데킬라", "보드카"]
```

- ▶ 그리고 다시 불러올 수 있죠.

```
print liquors[0]      # 소주
```

```
print liquors[2]      # 막걸리
```

- ▶ 순서는 0부터 시작합니다. 처음엔 좀 헷갈릴꺼예요!

변수와 자료형에 대해 알아보기

- ▶ `liquors[0]`에 들어있는 "소주"는 그럼 무슨 자료형일까요?
- ▶ 다시 생각해보면 아래의 구조를 가지고 있는 셈이죠
`liquors = ["소주", "맥주", "막걸리", "데킬라", "보드카"]`

0	소주
1	맥주
2	막걸리
3	데킬라
4	보드카

변수와 자료형에 대해 알아보기

- ▶ 한 번 해봅시다!
- ▶ 'fruits' 리스트에 "사과", "딸기", "수박", "참외", "멜론" 다섯개의 과일들을 순서대로 넣으세요.
- ▶ 'eat' 라는 변수에 "먹고싶다!" 라는 문자열을 넣으세요.
- ▶ 위에서 만들어진 변수들을 이용해 "딸기먹고싶다!", "멜론먹고싶다!" 라는 문장을 출력해보세요!

변수와 자료형에 대해 알아보기

▶ 정답!

```
>>> fruits = ["사과", "딸기", "수박", "참외", "멜론"]
>>> eat = "먹고 싶다!"
>>> print fruits[1] + eat
딸기먹고 싶다!
>>> print fruits[4] + eat
멜론먹고 싶다!
```


변수와 자료형에 대해 알아보기

▶ 딕셔너리 (Dictionary)

▶ 딕셔너리는 '사전'

▶ 딕셔너리는 중괄호'{'}'를 이용해 선언합니다.

```
person = {  
    "name": "아이유",  
    "age": 23,  
    "job": "가수"  
}
```

▶ 다시 불러올 수도 있죠.

```
print person["age"]          # 23
```

변수와 자료형에 대해 알아보기

- ▶ 리스트가 단순 순서로 이루어져 있다면 (0, 1, 2, 3...)
- ▶ 딕셔너리는 내부 요소에 이름을 붙일 수 있는 거라구 이해하면 좋아요.
(name, age, job...)

변수와 자료형에 대해 알아보기

- ▶ 한 번 해봅시다!
- ▶ americano 라는 딕셔너리 변수를 만들어서,
"name"에 "아메리카노", "price"에 3000, "taste"에 "쓰다" 를 넣어
서 만들어보세요.
- ▶ 엔터를 쳐서 여러줄로 써도 되어요.
- ▶ 다 되면, print를 이용해 이름, 가격, 맛을 순서대로 출력해보세요.

변수와 자료형에 대해 알아보기

▶ 정답!

```
>>> americano = {  
...     "name": "아 메 리 카 노",  
...     "price": 3000,  
...     "taste": "쓰 다 "  
... }
```

```
>>> print americano["name"]  
아 메 리 카 노  
>>> print americano["taste"]  
쓰 다  
>>> print americano["price"]  
3000
```

변수와 자료형에 대해 알아보기

- ▶ 자 이제 오늘의 최종 보스! 리스트와 딕셔너리를 합친걸 한 번 해볼까요?
- ▶ 다음의 표를 리스트와 딕셔너리를 이용해서 "coffee_menu" 라는 변수로 만들어보세요.

	0	1	2
이름	아메리카노	카페라떼	카페모카
가격	3000	4000	4500
맛	쓰다	덜쓰다	달다

- ▶ 힌트: americano, caffe_latte, caffe_mocha 딕셔너리를 미리 만들고, 이를 리스트에 넣으면 되겠죠. `coffee_menu = [...]`

오늘 배운 내용을
꼭 집에서 복습해보세요.

내 것으로 만드는 것이 중요합니다.

고생

많으셨습니다!