

Python 수업 OT

1. 소개
2. 수업 세부 일정
3. Python이란
4. Pycharm 설치

1. 소개

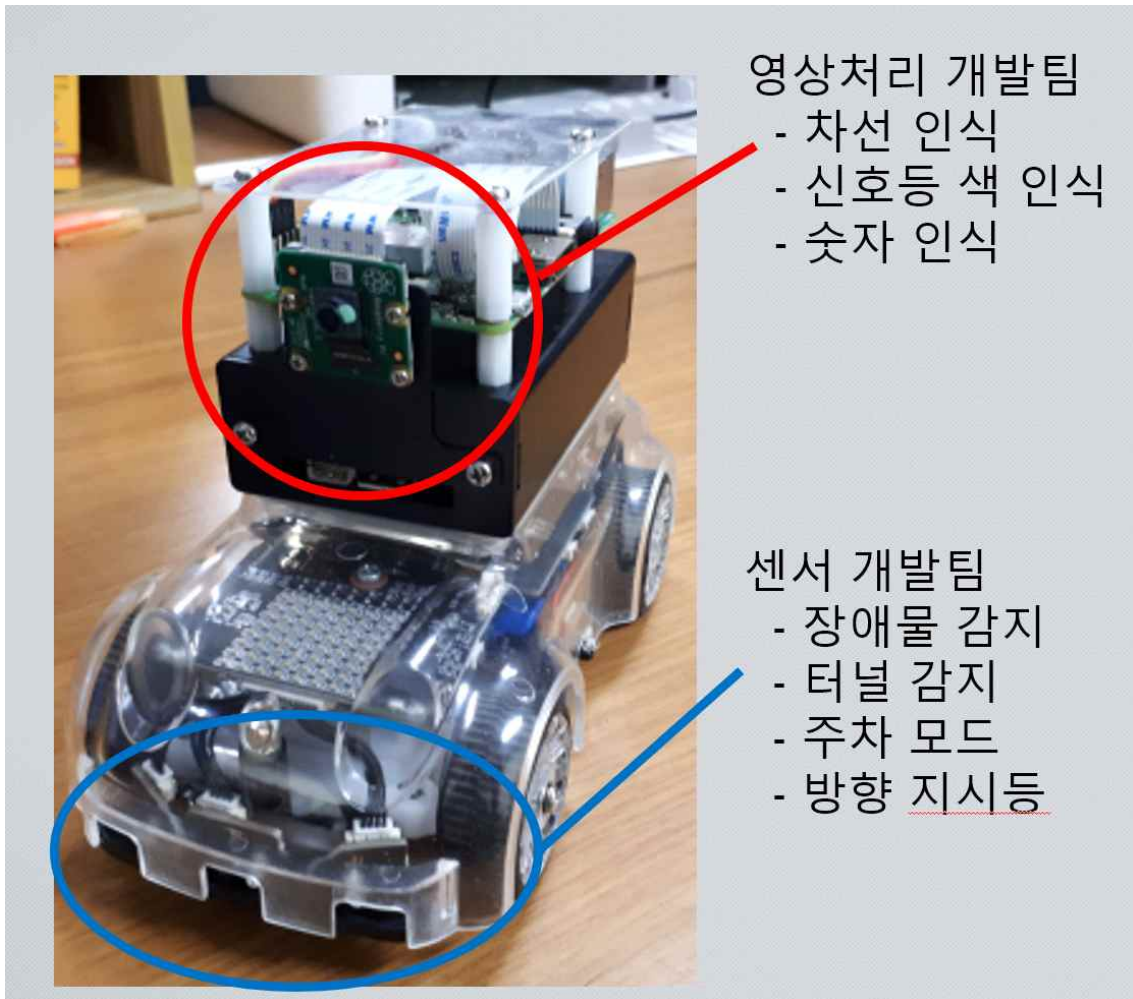
반갑습니다!~ 제 소개를 먼저 하도록 할게요. 저는 현재 고려대학교 대학원 컴퓨터공학전공 인공지능, IoT 관련 연구실에 다니고 있는 손원준이라고 합니다. 제가 프로그래밍을 처음 접하게 된 것은 1학년 여름방학이었어요. 1학년 2학기 때 C언어를 배운다는 얘기를 듣고서 시간이 많이 남으니 독학 한번 해보자 라는 생각으로 시작을 했습니다. C언어를 공부하기 시작한 첫날부터 정말 밥도 안 먹고 밤새 공부를 했었습니다. 지금 생각해보면 어떻게 그렇게까지 열심히 했나 싶을 정도로... 사실 전 끈기도 별로 없고, 머리도 좋은 편도 아니고, 부지런한 편도 아닌데, 평생 못 느꼈던 배움에 대한 희열, 성취, 갈증을 느끼니 그렇게 저절로 몸이 움직여지더라구요. 그 후로 C언어를 정말 깊고 진하게 공부해서 두 달동안 마스터를 하고 테트리스 게임을 스스로 만들 정도가 됐습니다.



요게 제가 그 당시 만들었던 테트리스입니다. 이 외에도



전공과목이었던 전자회로 실험 수업 레포트 쓸 때 필요했던 저항띠 계산기 같은 것도 만들고.. 그 후 많은 프로젝트 중에 가장 크게 했던 것이



자율 주행 모형 자동차 만들기였어요. 이 외에도 탁구 자동 점수계산기, 미로 탈출 로봇, 거북목 자동 교정기 등등 다수의 프로젝트와 프로그램 개발 외주도 맡아서 하게 됐었습니다.

다시 한번 말하지만 저는 머리가 좋지 않아요ㅠㅠ 그 말은 프로그래밍이라는 분야는 충분히 독학으로 정복을 할 수 있다는 말이에요. 다만 가장 중요한 것이 “흥미”를 느끼는 것! 제 수업 목표도 바로 그거예요. 제가 느꼈던 그 흥미, 성취, 희열들을 여러분들에게 전달해주는 것.

물론 프로그래밍에 흥미를 느끼지 못하는 분들이 있을 수 있어요. 하지만 프로그래밍에는 인간이라면 누구나 매력을 느낄 수 있는 요소들이 많습니다. 평소 상상만 하던 것을 내 손으로 직접 만들고 그 것을 사람들이 기분 좋게 사용해준다면 그 만한 희열이 또 있을까요. 프로그래밍을 배우다보면 코딩을 하는 것이 왠지 나만의 소설을 쓰는 것 같은 느낌을 주기도하고 컴퓨터랑 대화하고 있다는 느낌을 많이 받습니다. 저는 영어를 잘 못하지만, 영어를 정말 잘하면 원어민과 자유롭게 대화를 할 수 있게 되면서 성취감이 생기잖아요. 프로그래밍 언어 또한 프리토킹이 될 정도로 (프로그래밍 언어는 프리토킹이 아니라 free writing이라 해야하나요) 공부하다보면 원어민과 대화하는 듯한 성취감을 느낄 수 있습니다.

저는 이 성취감을 전공자들만 느낀다는게 조금 안타까웠습니다. 코딩하는거 정말 별거 아닌데..(정말로요 ㅠㅠ) 비전공자들도 엑셀처럼 쉽게 배워서 다양한 곳에 써먹을 수 있는데.. 라는 생각이 떠나질

않았습니다. 뭐 사회에서는 4차산업혁명시대다. 프로그래밍 스킬은 누구나 갖고 있어야한다. 라고 얘기를 하는데 정작 어떻게 시작해야 될지 알려주는 사람도 없고, 시작하기 겁도 나고, 학원을 찾아 보면 한달 백만원이 넘는 돈을 요구하고.. 따라서 저는 길지 않은 시간이지만 비전공자라도 개발에 충분히 뛰어 들 수 있다는 것, 그리고 코딩이 생각보다 쉽고 대단하다는 것을 알려드리기로 마음먹었죠. 그것도 외부학원보다 훨씬 저렴한 가격으로요! 고작 12시간만 투자하신다면, 일 잘하는 사원, 팀프로젝트의 에이스가 될 수 있는 기회입니다~

용기 내서 제 수업 신청해주셔서 감사합니다! 정말 칭찬드리고 싶어요~ 앞으로 잘 부탁드립니다!



2. 수업 세부 일정

수업 일정		
1주차	<파이썬 기초> - 프로그래밍이란? - 파이썬이란? - 기초 문법 실습	@ 실습 : UP/Down Game, 전화번호부 관리 프로그램
2주차	<함수> - 사용자 정의 함수 만들기 - 내장/외장 함수 <데이터 수집 기초> - 웹 스크레이핑/크롤링이란? - HTML 언어 기초 - 데이터 수집 실습	@ 실습 : 지도 불러오기, 그림 그리는 거북이, 인공지능 공룡 @ 실습 : 일기예보, 환율 정보, 인기검색어 순위, 뉴스 기사
3주차	<데이터 수집 심화> - 파이썬으로 웹에 로그인하기 - 웹브라우저 원격 제어 - API에서 데이터 받아오기	@ 실습 : 내 마이리지 점수, 내가 고른 맛집, 쇼핑 목록, 번역기 API
4주차	[선택1. 머신러닝 입문] - 학습된 컴퓨터에게 미래 또는 과거의 사건을 예측시키는 프로그램을 실습할 예정입니다. [선택2. 업무자동화 입문] - 실제 업무 또는 실생활에서 편하게 사용할 수 있는 자동화 프로그램을 실습할 예정입니다.	@선택1. 머신러닝 실습 : 꽃 종류 예측, 손글씨 인식, 타이타닉 생존자 예측, 온라인 게임 'LoL' 승부 예측 @선택2. 업무자동화 실습 : 인스타 자동 좋아요, 카페 자동 등업, 메일 자동 화신, 중고나라 자동 검색

수업 때 실습을 같이 진행할 겁니다. (위에 나와 있는 실습 내용들은 변경될 수도 있다는 점을 미리 알려드립니다.) 따라서 꼭 노트북을 가져오셔야해요~ 제가 알려드린 내용들을 바탕으로 매주 과제를 내드릴 겁니다! 과제라니.. 과제라는 단어만 봐도 토나오져.. 그치만 공부는 제가 하는 것이 아니라 여러분들이 하는 것이니 당연히! 시간 투자를 하셔야합니다. 하지만 여러분들이 시간이 없다는 것을 잘 알기에 절대 부담이 될 만큼 많이 드리지 않아요~ 절대절대 부담 갖지 마시고 편하게 잘 따라 오시면 됩니다. 수업이 다 끝났을 때쯤엔 남들에게 자랑할만한 프로그램 하나가 완성돼 있을 수도..?

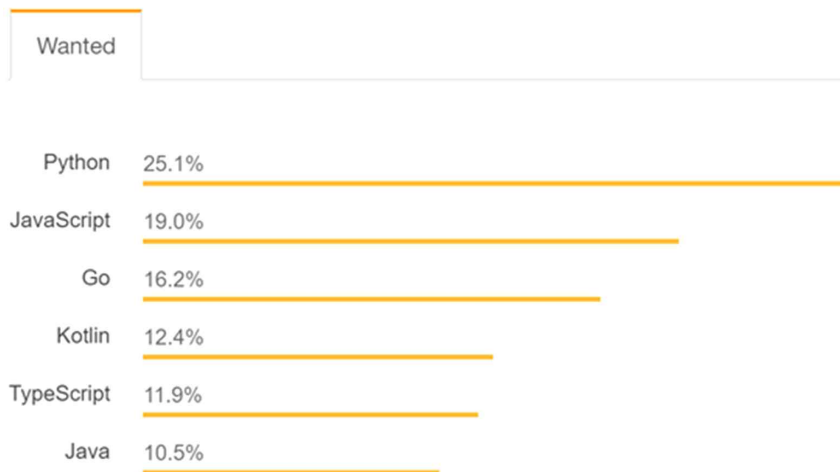
아직 저 표에 있는 단어들이 뭔지 잘 모르고 어려워 보이겠지만, 마지막 주차가 됐을 땐 저 용어들만 봐도 알고 있는 지식들이 머리속에 식씩 지나갈 정도가 돼있을 겁니다~ 혹시 커리큘럼에 대해서 궁금한 점이 있으시면 연락주세요!

3. Python이란

지금 이걸 보고 계신 분들은 파이썬을 선택하신 거잖아요~ 최근 빅데이터, 인공지능 분야가 뜨면서 파이썬이라는 언어가 굉장히 핫해요. 파이썬이란 C언어 같이 프로그래밍 언어의 한 종류데, C언어보다 쉽다는 이유로 많은 사람들이 파이썬으로 입문하길 원하죠. C언어 공부하다가 파이썬을 공부하니 진짜 놀라울 정도로 쉽더라고요. 파이썬이 쉬우니 입문용 언어로 적합하다! 라는 얘기를 들으신 분들도 있을거예요. 그치만 파이썬은 다른 언어보다 '상대적으로' 쉬운 것 뿐이지 입문자들이 보기에는 파이썬도 생각보다 쉽지 않을 겁니다.ㅠㅠ 따라서 각오하셔야 해요!



겁을 주려는 건 아니에요 ㅠㅠ 다만 쉽다는 얘기만 듣고서 접근했는데 막상 어려움을 느끼면 금방 포기할까봐요.. 파이썬을 선택하신 여러분들은 정말 좋은 선택을 하셨다고 말씀 드리고 싶어요! 파이썬은 현재 어떠한 분야에서든 가장 핫하게 떠오르고 있는 프로그래밍 언어이고, 현직 개발자들도 모두 파이썬을 배우길 갈망하고 있으니까요! 파이썬을 익혀놓으면 분명히 시간이 지나면서 더 빛이 나는 무기가 될 거라고 확신합니다~



(↑ 세계최대의 개발자 커뮤니티 사이트 StackOverflow에서 진행한 “가장 배우고 싶은 언어” 설문 조사 결과. 우리의 파이썬이 당당히 1위)

지금부터는 조금 지루한 설명이 있을거예요. 첫 수업이라 생각하고 시간내서 읽어주세요~

자 이제 파이썬에 대한 간단한 설명을 해드릴게요. 파이썬은 프로그래밍 언어의 한 종류예요. “프로그래밍 언어”라는 단어에서도 보듯이 이건 언어입니다. 언어의 사전적인 정의를 살펴보면

언어' 言語

★

+ 단어장 저장

파생어 : 언어적
발음녹음 ?

관련 어휘

✓ 예문집

T ▾

📄

🖨

명사

생각, 느낌 따위를 나타내거나 전달하는 데에 쓰는 음성, 문자 따위의 수단. 또는 그 음성이나 문자 따위의 사회 관습적인 체계.

간단히 말해 본인의 생각을 음성이나 문자로 전달하는 것이 언어입니다. 그럼 프로그래밍의 정의를 볼까요.

프로그래밍 programming

+ 단어장 저장

발음녹음 ?

T ▾📄🖨

명사

<컴퓨터> 컴퓨터 프로그램을 작성하는 일. 일반적으로는 프로그램 작성 방법의 결정, 코딩(coding), 예러 수정 따위의 작업을 이르지만 특수하게 코딩만을 이룰 때도 있다.

프로그래밍은 컴퓨터 프로그램을 만드는 일을 말해요. 따라서 “프로그래밍” + “언어”란 본인이 생각하는 프로그램을 만들 수 있게 도와주는 수단을 말합니다. 자신의 생각을 전달하는 거죠 컴퓨터한테! 프로그래밍 언어도 정말 영어 같이 하나의 언어라고 할 수 있죠. 영어는 reading, writing, speaking, listening 을 모두 잘해야하지만, 프로그래밍 언어는 writing만 잘하면 되는 아주 큰 장점이 있죠 ㅋㅋ

프로그래밍 언어에는 300가지가 넘는 종류가 있어요. 매년 새로운 언어가 생기고 있고, 소멸되기도 합니다.



(↑ 다양한 프로그래밍 언어들)

그 중 사람들이 많이 쓰는 언어가 살아남고 그 전통을 이어오고 있어요. 파이썬은 1980년부터 네덜란드의 “귀도 반 로섬”에 의해 만들어지기 시작하여 현재까지 잘 생존해오고 있으며 최근 데이터 과학 분야가 뜨면서 주목 받고 있습니다.

잠깐 여기서 딴 소리 하나만..

파이썬은 네덜란드의 “귀도 반 로섬”이 1989년 크리스마스에 연구실이 갇혀있어 할게 없어서 심심한 김에 만들었다고 하네요. 리스펙.. 당시 귀도 반 로섬은 영국 BBC에서 방영한 “몬티 파이썬의 플라잉 서커스” 라는 개그 프로의 팬이었다고 합니다. 따라서 그 이름을 딴 Python을 이름으로 지정했다고 하네요.

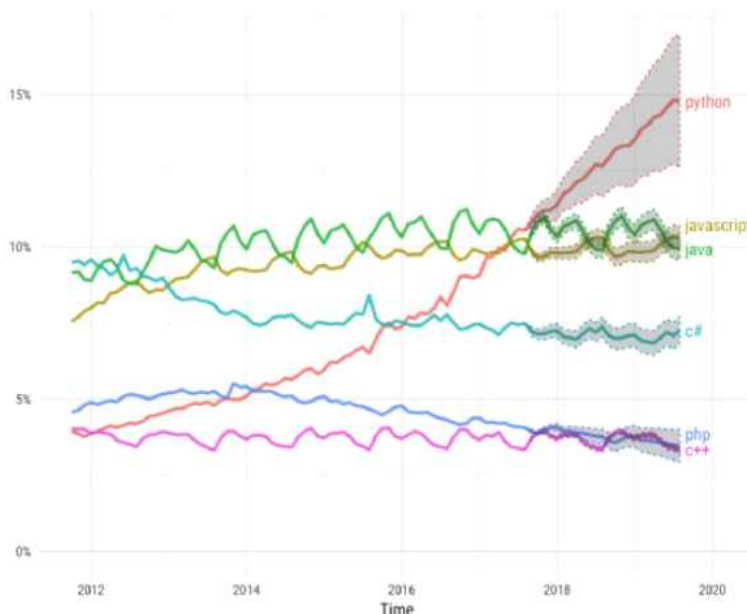


참고로, python을 사전에서 찾아보면 구렁이, 뱀, 비단뱀 등으로 뜻을 갖습니다. 따라서 python 언어의 로고도 뱀모양을 하고 있습니다.

저희가 배울 파이썬의 언어 점유율 순위를 볼까요

TIOBE에서 선정한 프로그래밍 언어 월간 점유율 순위 (2018년 6월 기준)				
1. Java	2. C	3. C++	4. Python	5. C#
16.380%	14.000%	7.668%	5.192%	4.402%
1.74% 증가	7.00% 증가	2.92% 증가	1.64% 증가	0.95% 증가
6. VB.NET	7. PHP	8. JavaScript	9. SQL	10. Ruby
4.124%	3.321%	2.923%	1.987%	1.182%
0.73% 증가	0.63% 증가	0.15% 감소	1.99% 증가	1.25% 감소
전년 동월 대비 변화 수치이다. 자세한 내용은 이곳에서 확인할 수 있으며, 기준은 다음과 같다.				

Java와 C언어는 매월 1,2위를 다투는 부동의 원투펀치입니다. 그렇지만 그 뒤를 따라 4위에 파이썬이 쫓아오는게 보입니다. 파이썬이 쫓아오는 속도는 무섭습니다.



연도에 따른 개발자들의 파이썬 선호도 상승 예상입니다. 파이썬이 이렇게 인기가 많아지는 이유가 아마 높은 생산성에 있지 않나 싶어요. 생산성이 높다는 것은 파이썬이 워낙에 쉽다보니 같은 시간 동안에 더 많은 코딩을 할 수 있는 것을 의미합니다. 몇 년동안 다른 언어로 개발을 해오던 개발자들이 파이썬을 만나면 모두 놀라자빠지고, 파이썬으로 갈아타고는 합니다. 언어가 쉽다보니 다른 언어로 코딩을하면 10줄을 써야하는 것을 파이썬은 1~2줄만으로 해결할 수 있으니 같은 시간을 개발하더라도 파이썬은 더 많은 양의 코딩을 할 수 있게 되는 겁니다.

“문명”, “심즈”, “월드 오브 탱크” 등의 게임이 파이썬으로 제작되었습니다. 저희가 잘 알고 있는 “카카오톡”, “드롭박스”, “인스타그램” 도 파이썬으로 만들어져 있습니다. 야후, 구글 카카오, 심지어 나사에서 까지 파이썬을 이용하고 있습니다.

프로그래밍 언어는 고급 언어, 저급 언어로 분류를 하고 있습니다. 고급, 저급을 나누는 기준은 진짜 ‘급’을 따지는 것이 아니구요. 저급에 가까울 수록 기계어에 가깝고, 고급에 가까울 수록 사람의 언어에 가까운 것입니다. 파이썬은 고급언어에 속하구요! 기계어란,

```

1011111111000101111111100011010101010
101101111111100010111111100011010101
0101010101111111100010111111100011010
101010101011111111000101111111000111
010101010101011111111000101111111000
111010101010101011111110001011111111
000111010101010101011111110001011111
111000111010101010101011111110001011
111111000111010101010101011101101110

```

요런 애들이예요. 컴퓨터는 사실 '0'과 '1'밖에 모르는 바보예요. 하.. 2학년 때 저희 과에서 저급언어를 배운적이 있었는데, 어셈블러 라는 언어였어요.

```

extern printf

section .data
prompt_int:    db    '%d', 10, 00
prompt_chr:    db    '%c', 10, 00

section .bss
sum:          resd   1

section .text
global  main

main:
    mov     dword [sum], 10
    add     dword [sum], 20

    push    dword [sum]
    push    prompt_int
    call    printf

```

어셈블러로 짠 코드인데 무려.. 그냥 10과 20을 더하는 프로그램입니다.. 저급언어 공부하는 사람들은 진짜 상 줘야돼요. 확실히 뭔가 영어는 영어를 쓰긴하는데 많이 어려워 보이는 느낌이 나죠. 반면 고급언어인 파이썬으로 똑같은 프로그램을 짜볼게요.

```

a=10
b=20
print(a+b)

```

정말 간단하죠. 프로그래밍을 안 배워도 전부 이해할 것만 같은 그런 간결함이에요. 음 다른 언어의 경우를 한번 봐볼까요. 아래는 C언어로 10+20을 해본 프로그램이에요.

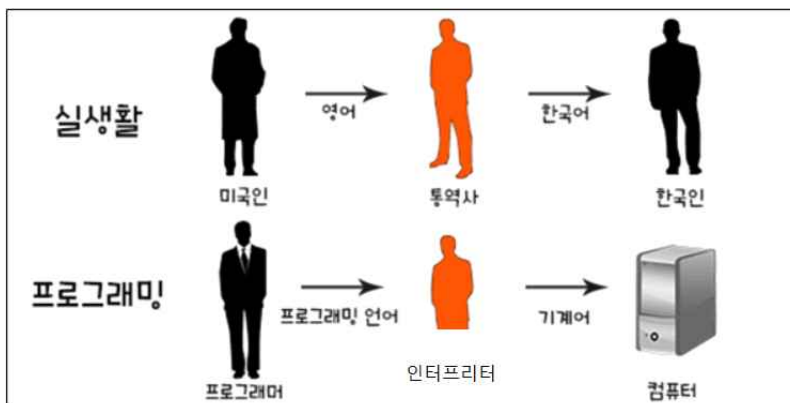
```
#include <stdio.h>

int main(void)
{
    int a, b, sum;
    a = 10;
    b = 20;
    sum = a + b;
}
```

뭔지 알 것 같으면서도 이상한 문자들이 중간 중간 끼여있죠. 파이썬이 확실히 간결하고 직관적이고 쉽다는 것으로 다시 한번 알려드리고 싶었어요ㅎㅎ

그럼 왜 저급언어는 소멸되지 않고 살아있냐 물어보실 수 있는데 저급언어는 기계어에 가깝기 때문에 속도가 굉장히 빠릅니다. 당연히 컴퓨터가 얼른 알아보기 쉬우니까 실행속도가 빠르겠죠? 반면 고급언어인 파이썬은 속도가 느립니다. 그게 대표적이 단점이에요. 따라서 어떤 빠른 연산을 필요로 하는 분야에는 부적절한 언어예요. 예를 들어 드론! 드론이 한쪽으로 기울게 되면 얼른 그 기울기를 계산해서 다시 반대방향으로 기울도록 처리를 해야하는데 연산속도와 프로그램 실행 속도가 느리다면 드론은 이미 뒤집혀져 땅에 떨어졌을거예요. 그래서 드론, 로봇 등 같은 빠른 속도를 필요로 하는 분야에는 C언어를 많이 사용합니다. C언어도 고급언어긴 하지만 고급언어 중에서도 속도가 빠르고, 익히기 어려운 편에 속해 중급언어로 구분되기도 한답니다.

저렇게 인간의 언어와 비슷하게 코딩(프로그램 짜는 것을 말합니다.)을 했는데도 기계가 알아먹는게 신기하죠. 사실 파이썬도 결국엔 '0'과 '1'뿐인 기계어로 변환됩니다. 파이썬을 기계어로 변환시켜주는 것을 인터프리터라고 해요. 쉽게 말씀드리면 인터프리터는 파이썬을 기계어로 통역시켜주는 통역사 같은 역할이에요.



프로그래밍 언어에는 대표적으로 두 종류의 통역사가 있는데, 하나는 인터프리터이고, 다른 하나는 컴파일러입니다. 인터프리터를 사용하는 언어에는 대표적으로 저희가 배울 파이썬, 컴파일러를 사용하는 언어에는 대표적으로 C언어가 있습니다. 인터프리터 같은 경우에는 내가 짠 코드를 A4용지에 써서 가져다 준다면 한줄 한줄 씩, 한줄 단위로만 번역을 합니다. 그래서 만약 중간에 내가 틀리게 쓴 부분이 있어서 오류를 내뱉더라도 그 윗줄까지는 모두 번역을 한 상태이기 때문에, 어느 정도 실행결과가 나오죠. 반대로 컴파일러는 A4용지 하나를 가져다주면, A4용지 하나를 그냥 모두 통째로 번역합니다. 그래서 중간에 내가 틀린 부분이 있으면, 오류를 내뱉고서 아예 번역 시작도 안합니다.

프로그램을 실행할 때, 인터프리터보다 컴파일러가 번역하는 것이 실행속도가 빠릅니다.

그렇다면 굳이 실행속도가 느린 인터프리터를 사용하는 이유는 뭘까요? 그 이유는 코드의 수정이 쉽다는 장점이 있습니다. 내가 틀린 부분이 있다면, 그 부분만 살짝 지워서 다시 인터프리터에게 가져다 주면 딱 그 부분부터 번역을 재개 합니다. 하지만 컴파일러는 틀린 부분을 지워서 다시 가져다주면 또, A4용지의 첫 부분부터 검토합니다. 저는 이 인터프리터의 장점이 되게 좋더라고요. 코드의 수정이 많은 프로그램을 짤 때는 파이썬이 1초라도 시간을 아끼게 해줍니다.

기본적인 지루했던 얘기는 요까지만 딱 할게요! 윗 내용들 몰라도 사실 프로그래밍 할 수 있어요. 그치만 파이썬 마스터했다고 자랑하고 다니는데 친구가 “왜 다른 언어도 아니고 파이썬 배웠어? 고급언어가 뭐야? 인터프리터가 뭐야?” 이런거 물어봤을 때 잘 대답하면 멋있잖아요 ㅎㅎ. 마지막으로 해드리고 싶은 얘기는 정말 열심히 하셔야해요! 프로그래밍 언어를 공부하는 것은 무빙워크와 같아요. 무빙워크에 가만히 서있으면 시간이 지남에 따라 계속 뒤로 가잖아요. 마찬가지로 프로그래밍 언어 공부에 손을 놓으면 계속 뒤쳐져요. 반대 방향으로 슬슬 천천히 걸어가면 제자리에 머물러 있습니다. 빠르게 걸어가거나 뛰어야지만 거슬러 올라갈 수 있겠죠. 정말 열심히 빠짝하셔야 실력이 늡니다. 그렇게 해서 끝에 도달하게 된다면 이제는 손을 놓아도 평생 그 지식들을 유지할 수 있어요. 혹시나 욕심이 생겨서 파이썬을 배우면서 다른 언어도 하고 싶어! 라는 분이 계시다면 정말 급하지 않는 이상 비추입니다. 예를 들어 프랑스어를 배우면서 독일어를 한다고 생각하시면 돼요. 비슷하지만 다른 문법, 단어들이 많기 때문에 동시에 공부하면 이게 프랑스언지 독일언지 헷갈립니다. 따라서 한 언어만 빠짝 열심히 하시고, 그런 다음 다른 언어로 넘어가시는 순서로 공부해주세요~ 아래에는 파이썬 언어와 개발툴 설치 과정입니다. 꼭 수업시간 전에 미리 설치해와주세요! 혹시나 “아나콘다”나 “쥬피터 노트북”이 설치돼있으신 분들이라면 그 프로그램들을 모두 “삭제”하고 아래 설치과정을 따라 해주세요. (아나콘다와 Pycharm이 같이 설치되어 있으면 충돌이 일어납니다.)

6. 파이썬 설치

- 컴퓨터에 파이썬 언어를 설치 해줍니다.(정확히는 파이썬 번역가인 인터프리터를 설치해주는 과정입니다.)

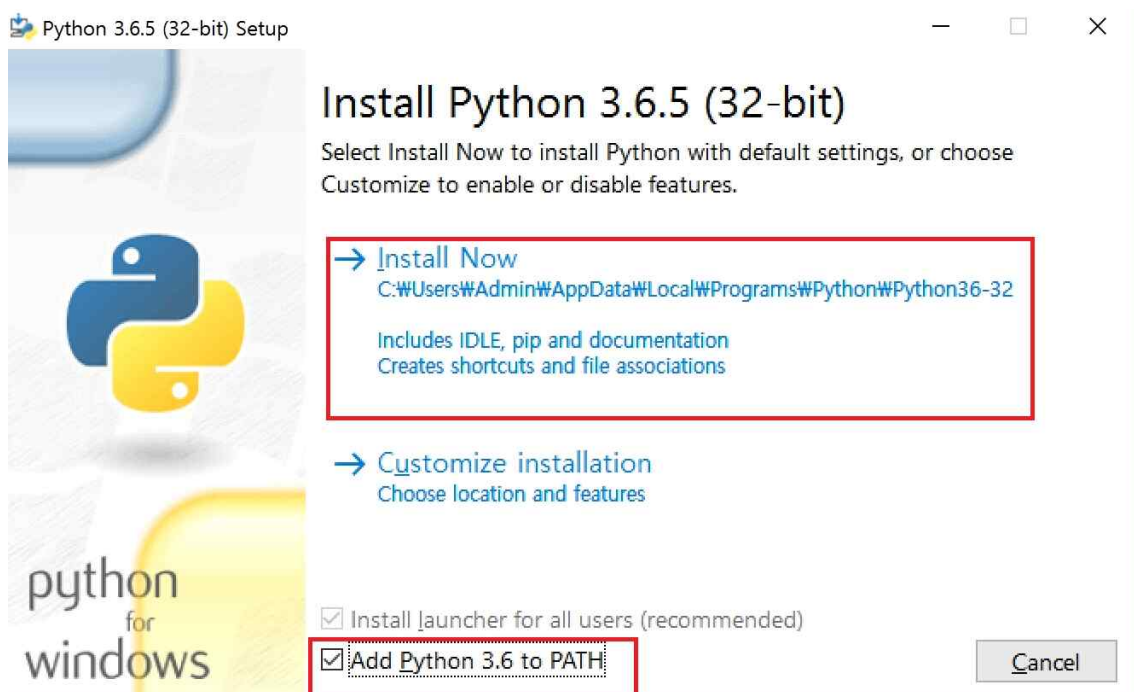
1. <https://www.python.org/downloads/windows/>

위 사이트로 들어가서 아래 **빨간 네모**를 눌러서 설치를 시작해주세요
(혹시 본인의 운영 체제가 32bit 인 경우, **파란 네모**를 설치 해주세요.)

Python Releases for Windows

- [Latest Python 3 Release - Python 3.6.5](#)
- [Latest Python 2 Release - Python 2.7.15](#)
- [Python 3.6.6rc1 - 2018-06-12](#)
 - Download [Windows x86 web-based installer](#)
 - Download [Windows x86 executable installer](#)
 - Download [Windows x86 embeddable zip file](#)
 - Download [Windows x86-64 web-based installer](#)
 - Download [Windows x86-64 executable installer](#)
 - Download [Windows x86-64 embeddable zip file](#)
 - Download [Windows help file](#)

2. (2번 과정이 제일!!!! 중요합니다!!!! 잘 따라해주세요.) "Add Python 3.6 to PATH" 체크 하신다음에 Install Now 누르시고 쪽쪽 next 누르면서 설치 완료하시면됩니다. (저걸 체크안하시면 나중에 어떤 설정들을 다시 해줘야해서 고생합니다 ㅠㅠ)

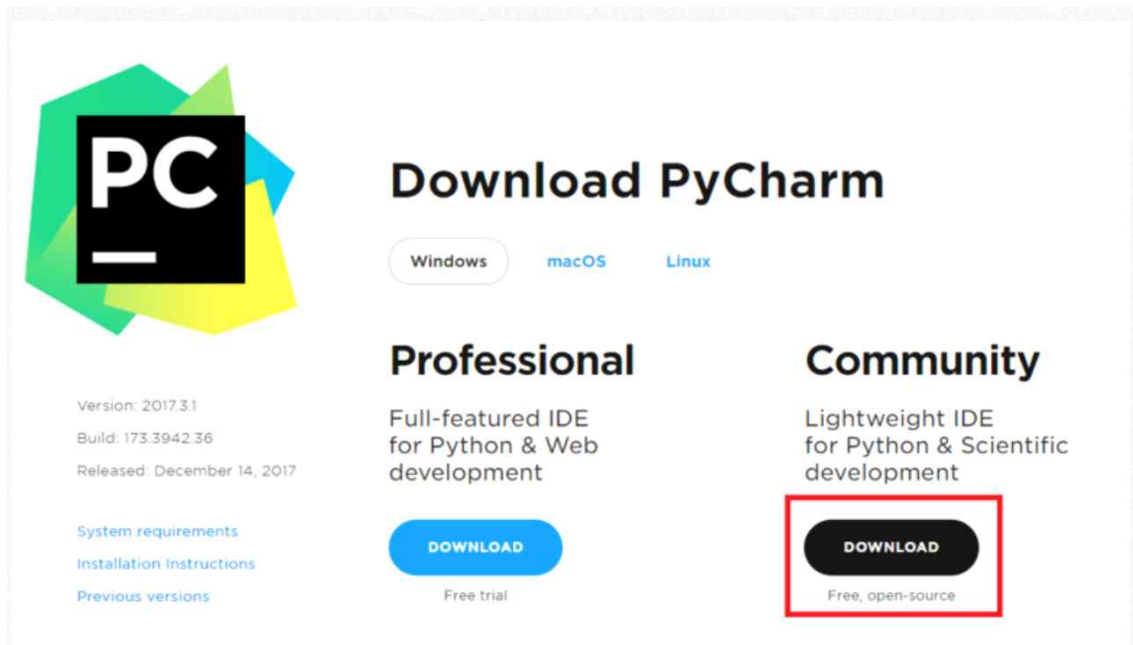


7. PyCharm 설치

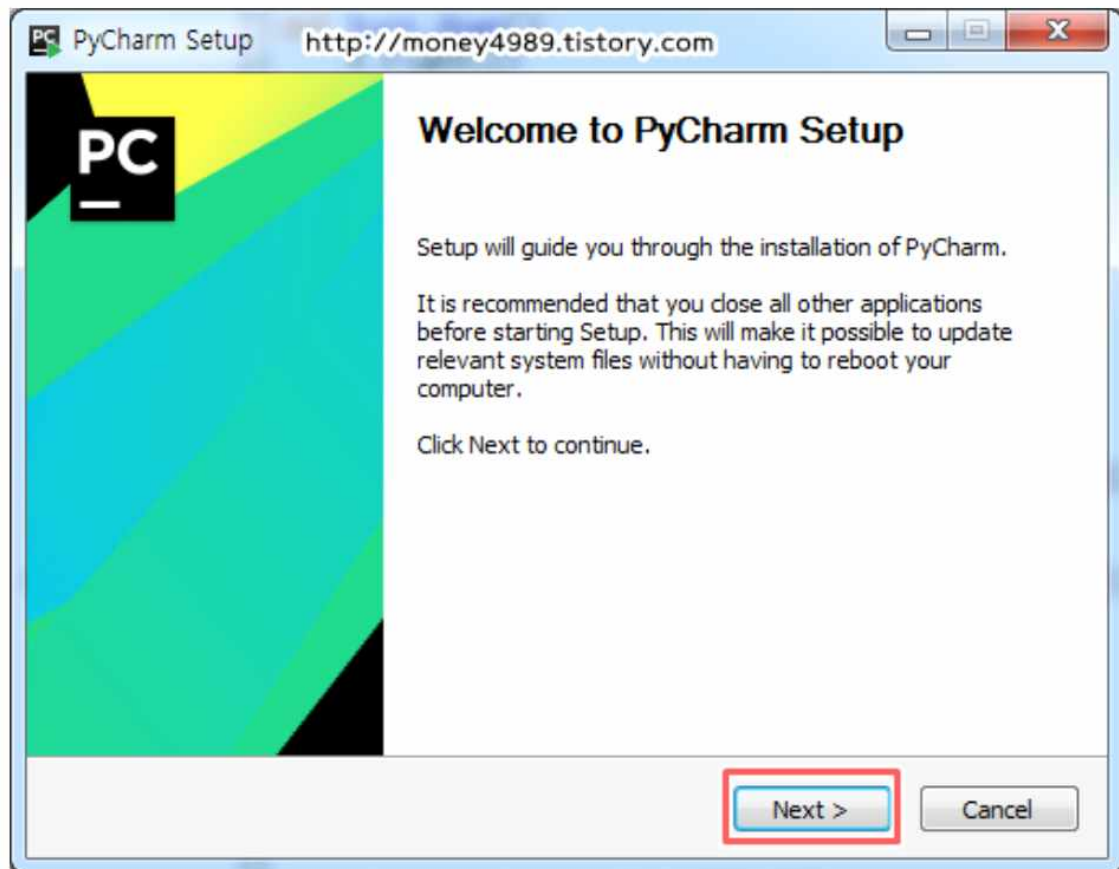
- PyCharm이란 파이썬을 이용해 개발을 할 수 있도록 도와주는 개발 환경이에요. 이제 아래 순서를 따라서 설치를 진행해주세요.(쉽게 말해 코딩을 쉽게 해주도록 도와줍니다. 예를들어 자동완성기능이라던지.. 자동 들여쓰기 등등등..)

1. <https://www.jetbrains.com/pycharm/download/#section=windows>

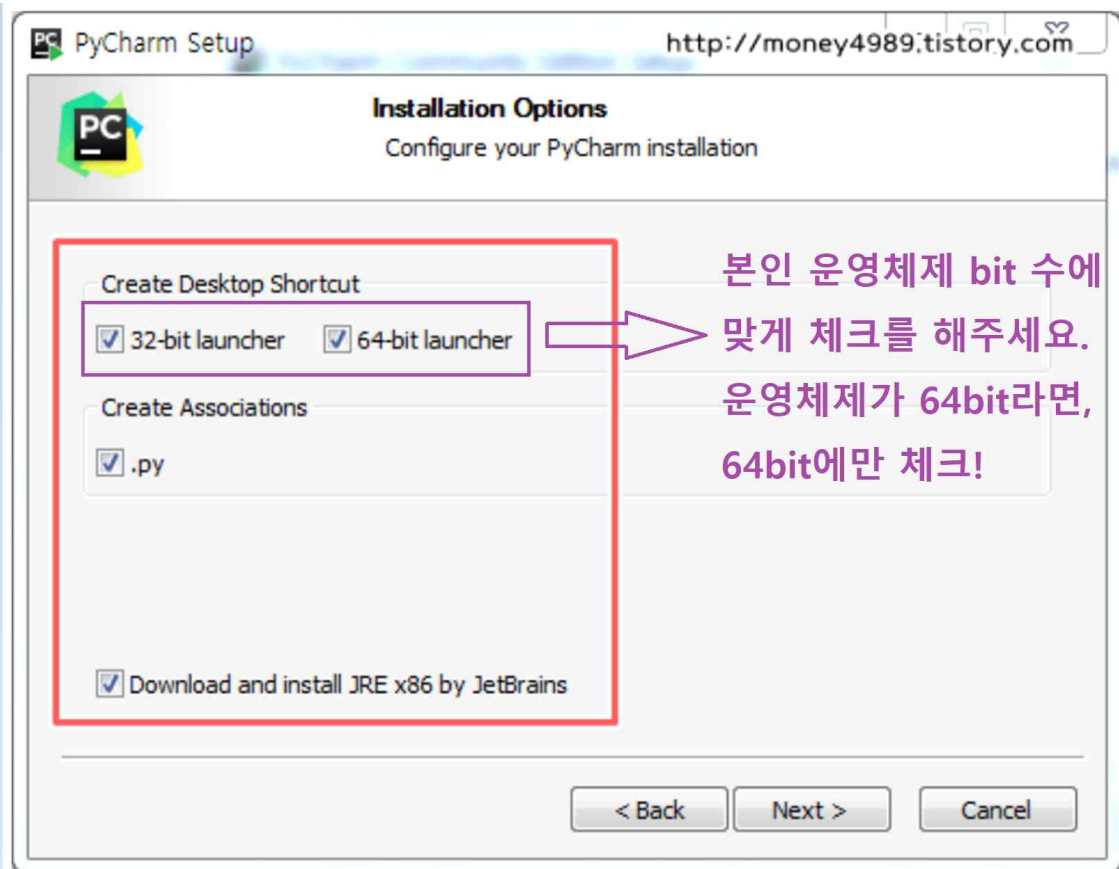
위 사이트 들어가셔서 무료버전인 Community 를 다운로드 받아주세요.



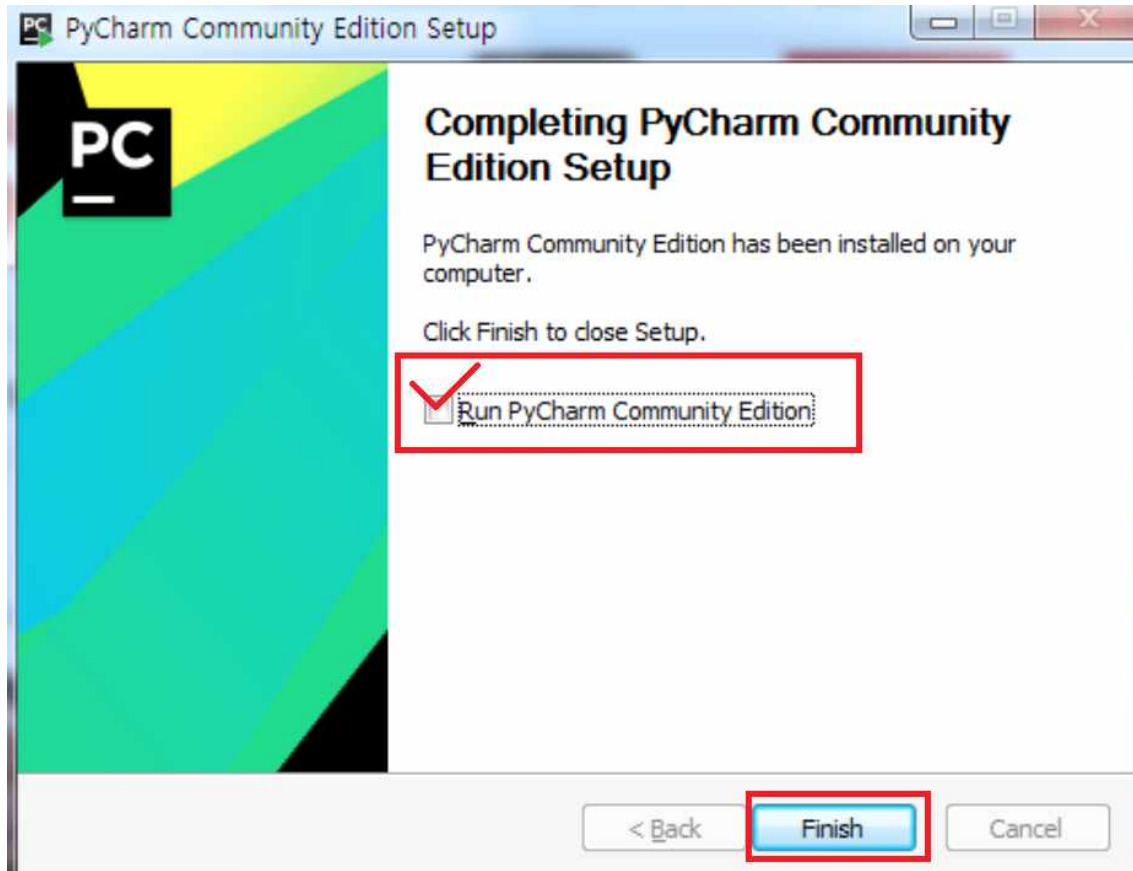
2. 설치를 쭉쭉 진행합니다.



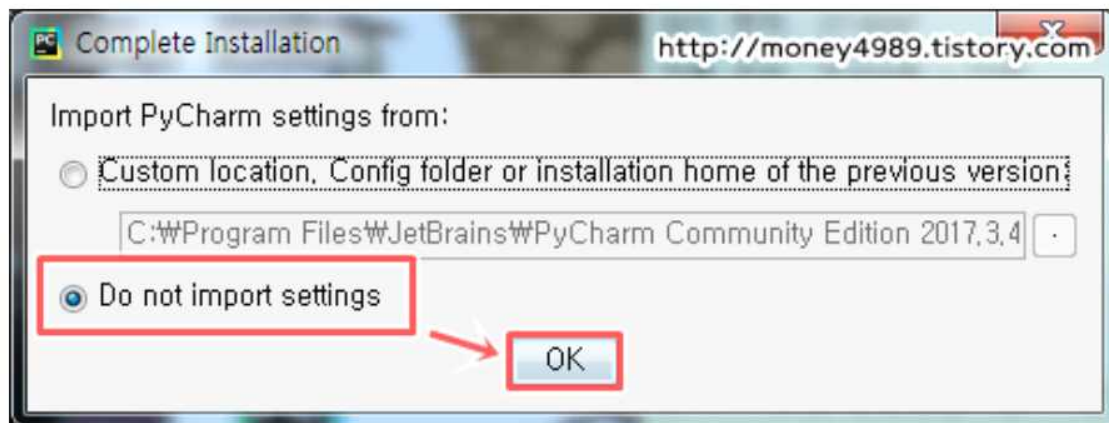
3. 모두 체크 해주시고, 쪽쪽 넘어가서 설치를 해주세요.



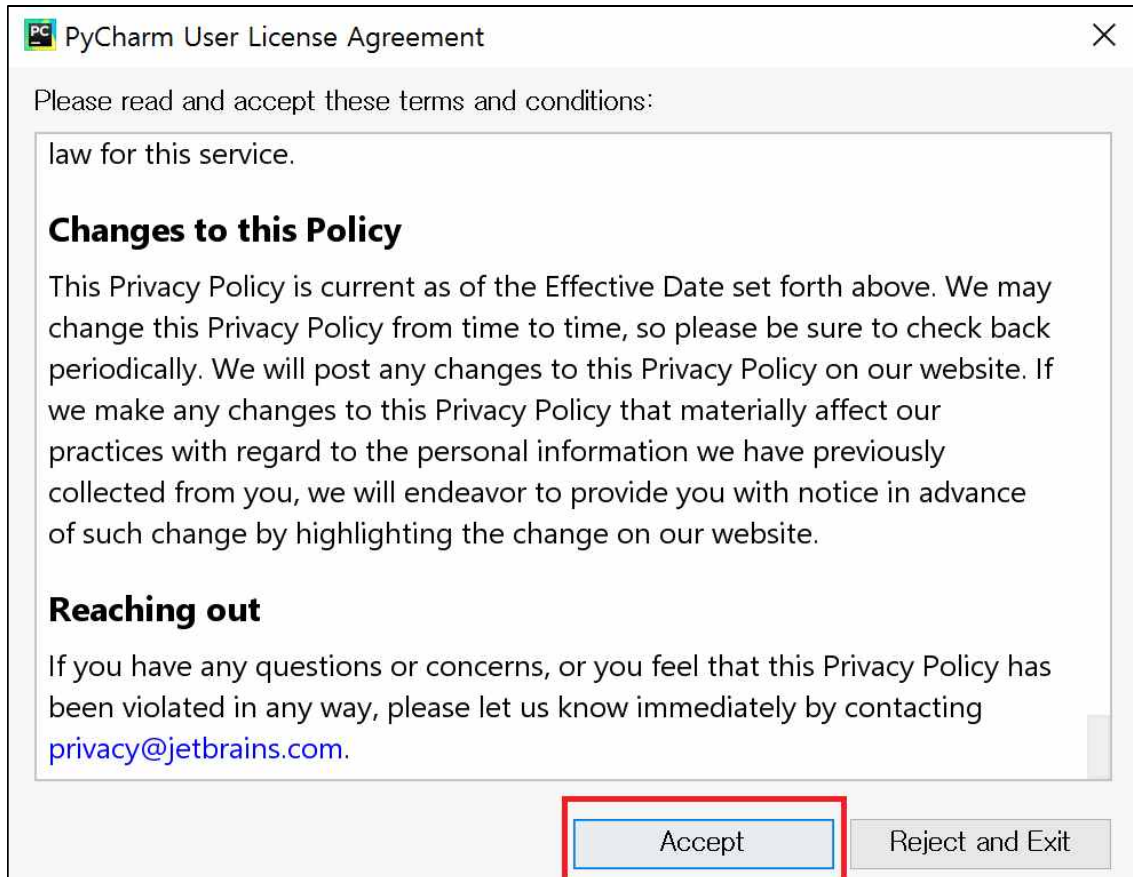
4. 설치가 완료되면 요기를 체크해주시고 “Finish”를 눌러주세요.



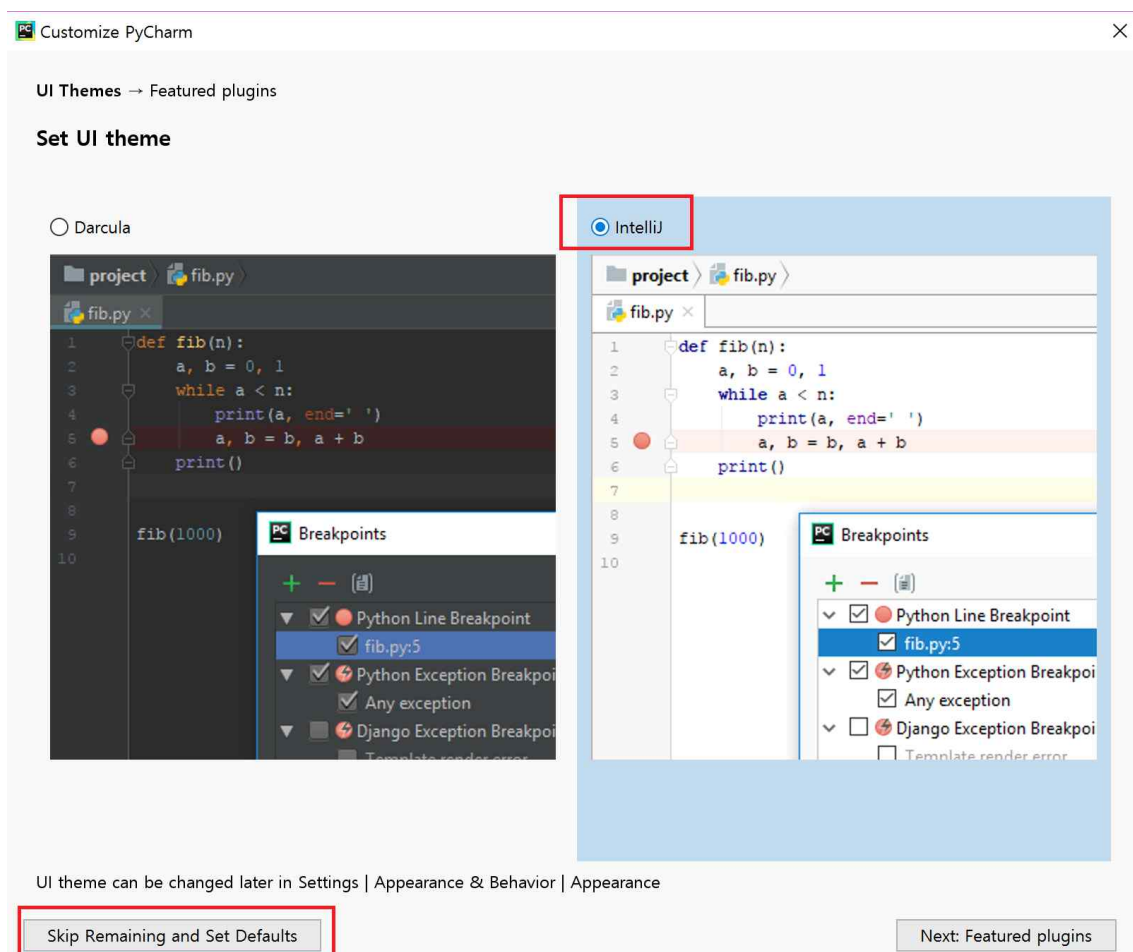
5. 그 후 창이 뜨면 “Do not import settings”를 선택하시고 확인버튼을 눌러주세요.



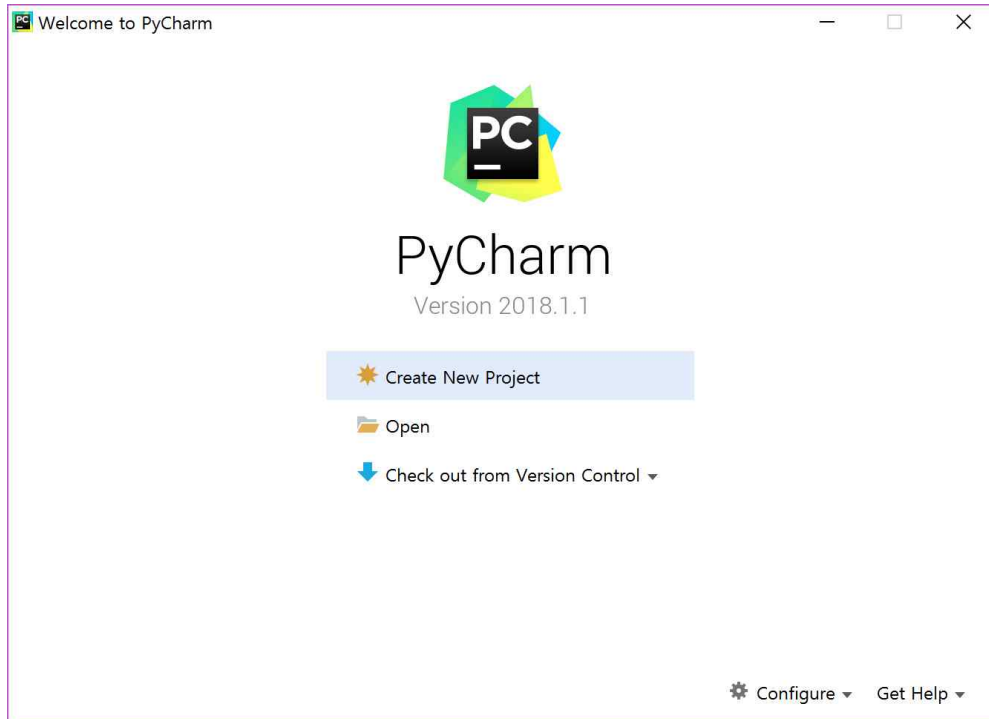
6. 요게 뜨면 맨 밑까지 스크롤 내려주시고 “Accept” 눌러주세요



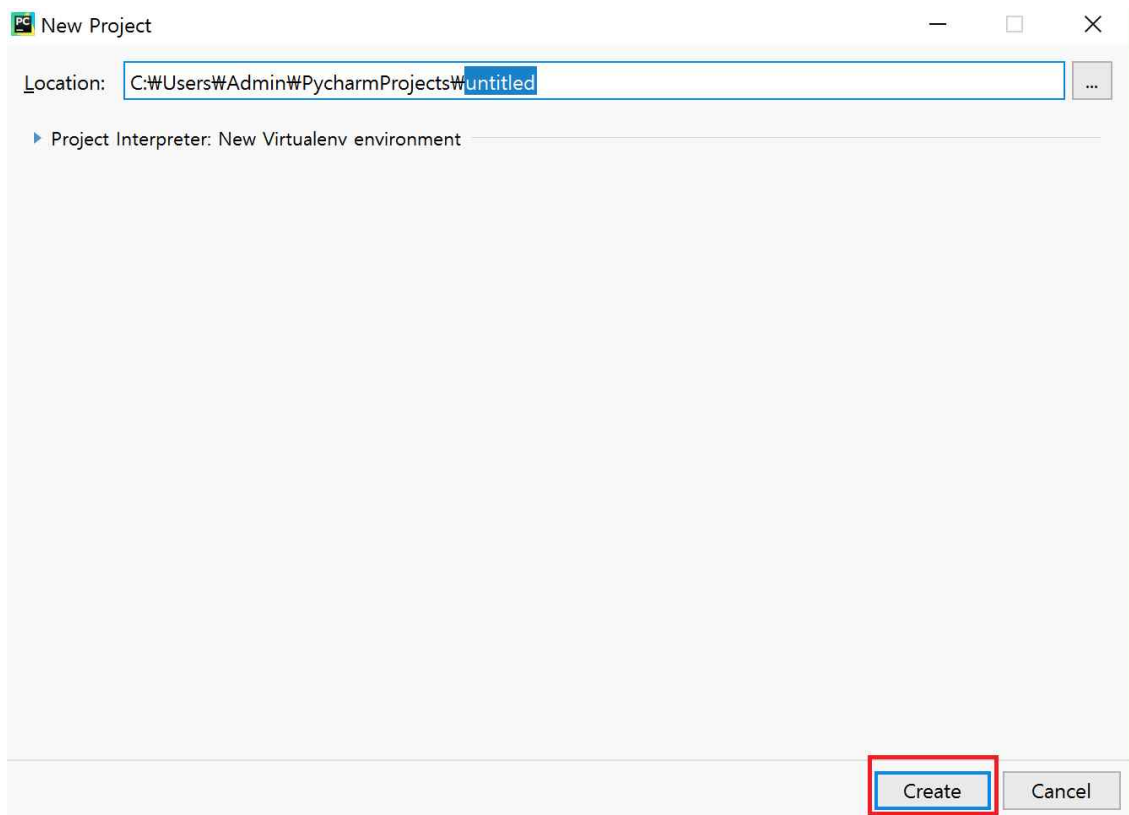
7. PyCharm의 테마를 설정하는 부분인데요. 저는 하얀색을 선택했습니다. 검은색 테마는 눈이 아파서...ㅠㅠ 테마는 마음대로 고르시고
그 다음 스킵을 눌러주세요!



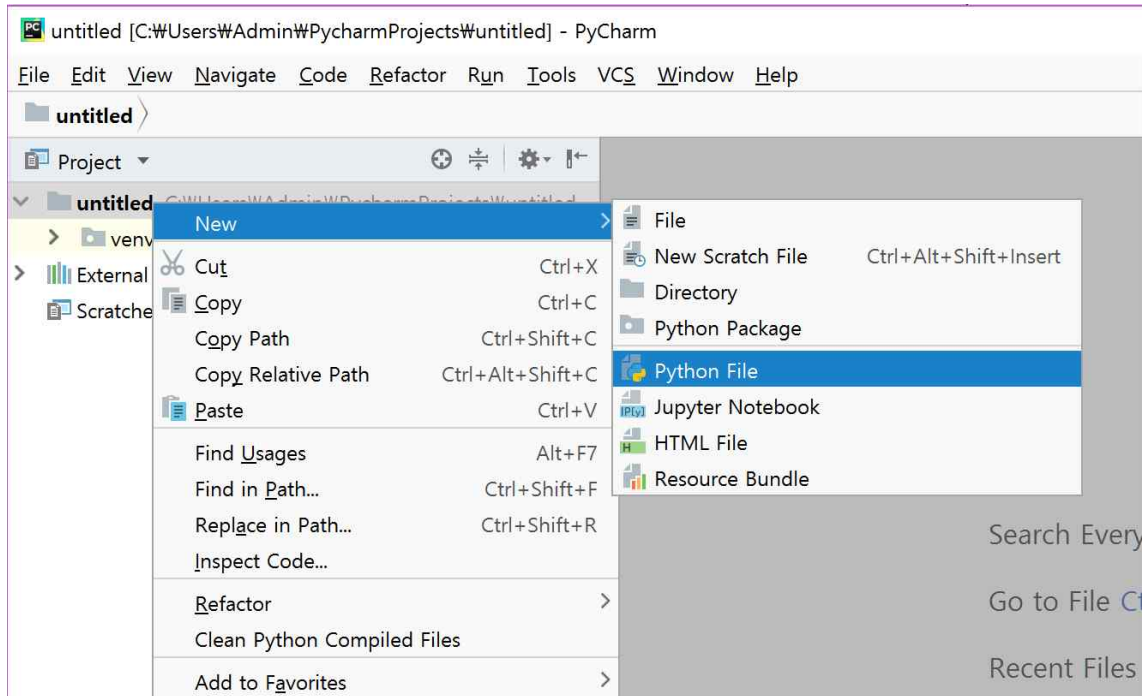
8. 설치가 완료 됐으니, 간단한 프로그래밍을 해볼게요! 파이참이 실행되면 새 프로젝트를 만들어주세요.



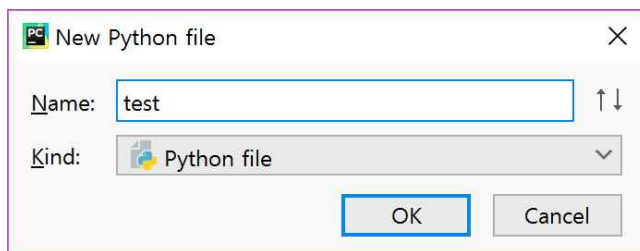
9. create 눌러주세요.



10. “untitled”에 우클릭을 해서 “New-Python File”을 눌러주세요.(새로운 파이썬 파일을 만드는 과정입니다. 쉽게 말해 프로그램을 작성할 새하얀 빈 도화지를 하나 만드는 겁니다. 앞으로 수도 없이 하게 됩니다. 기억해주세요!)

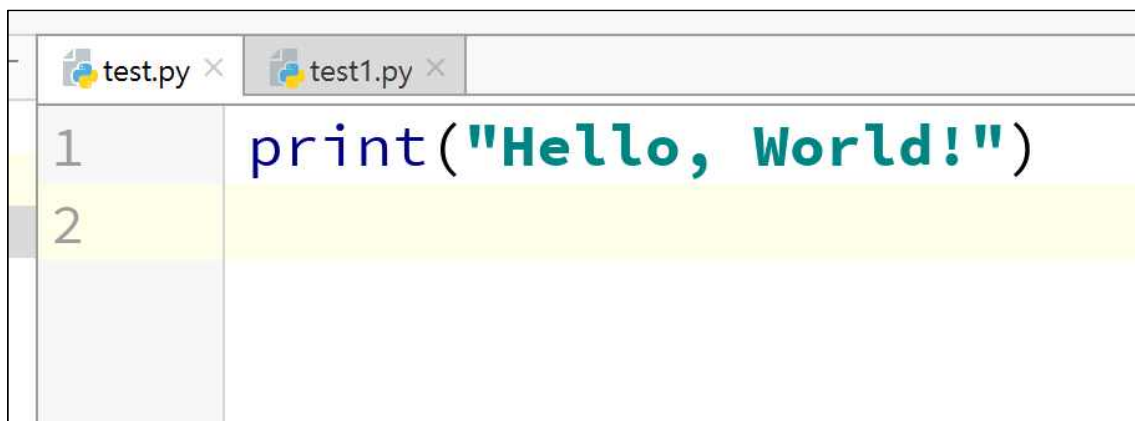


11. 아무 이름이나 쓰시고, OK!

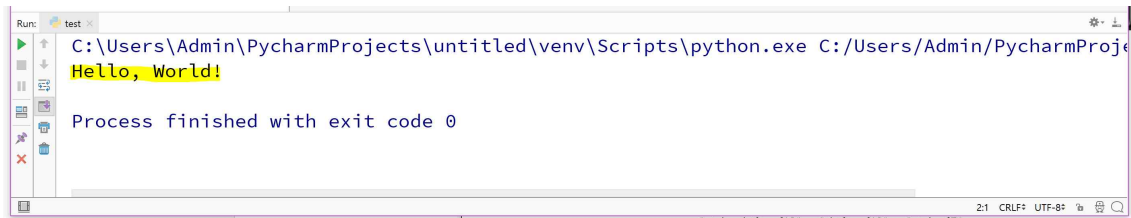


12. 그 다음, 하얀 도화지 위에 다음과 같이 똑같이 써주세요! 한글자라도 틀리면 에러가 납니다.

“Hello, World”라는 문자열을 출력하는 프로그램입니다.



13. 똑같이 쓰신 후, Ctrl+Shift+F10을 누르면 창 아래에 다음과 같은 실행결과를 볼 수 있습니다. Hello, World! 라고 출력이 된걸 확인할 수 있습니다.



여기까지 따라오시느라 고생하셨습니다~
인생 첫 프로그래밍에 성공하셨습니다! ㅎㅎ
수업날 배요~