

\이름

소속

왜 이 수업을 듣게 되었는지

자신의 독특한 취향

팀 이름

팀원 이름

10:10까지 휴식

비밀번호

anse2020

WIFI : 407-202/sw407202

13:35까지 휴식 시간

open.gnu.ac.kr

ALT+CTRL+T

linux 에서 파이썬 설치 되었는지 확인하는 방법

1. alt+ctrl+t
2. python --version 또는 python3 --version
3. 가능하면 python3 권장
4. 없으면 설치: sudo apt-get install python3

윈도우에서 파이썬 설치

링크를 따라 설치

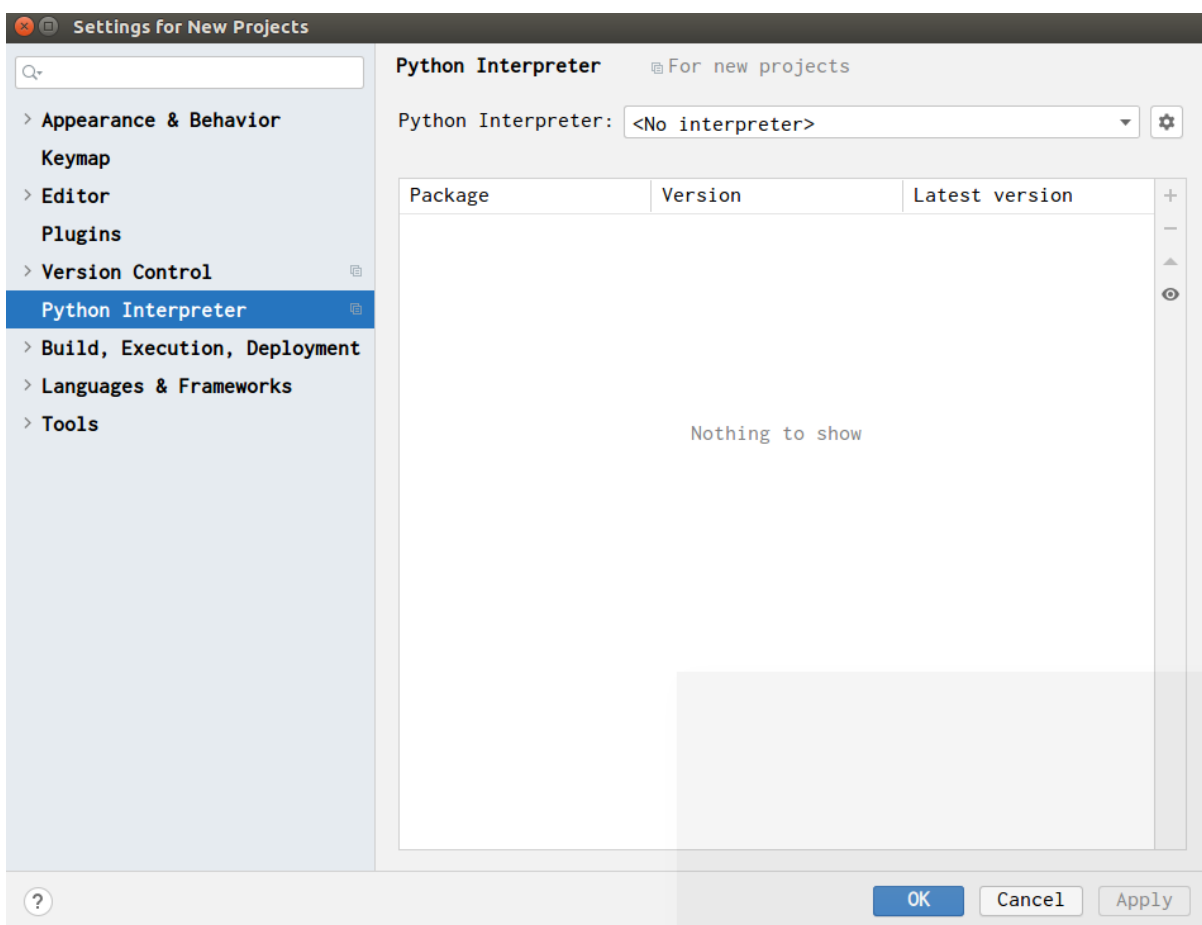
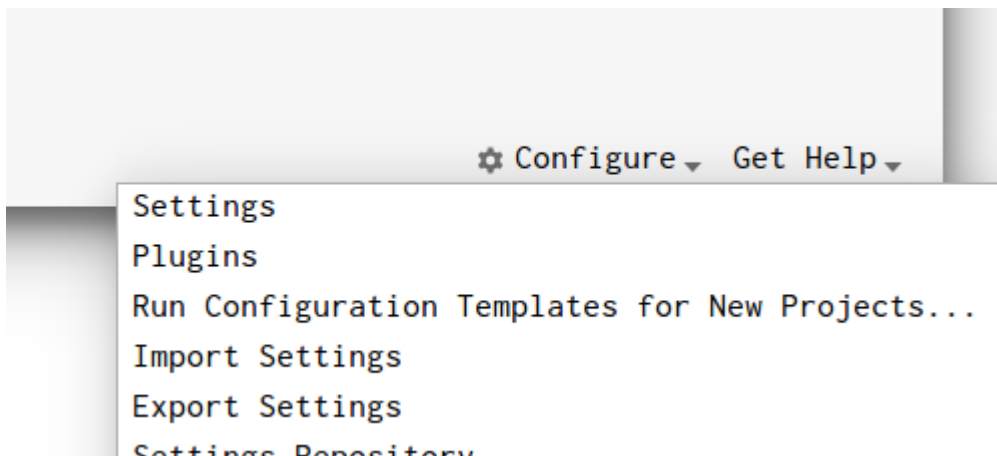
파이참 다운받기

다운로드 한 파일 압축 풀기

1. 터미널에서 (alt+ctrl+t)
2. cd 다운로드 (엔터)
3. cd pycharm-2020-community.2.3 (엔터)
4. cd bin (엔터)
5. ./pycharm.sh (엔터)

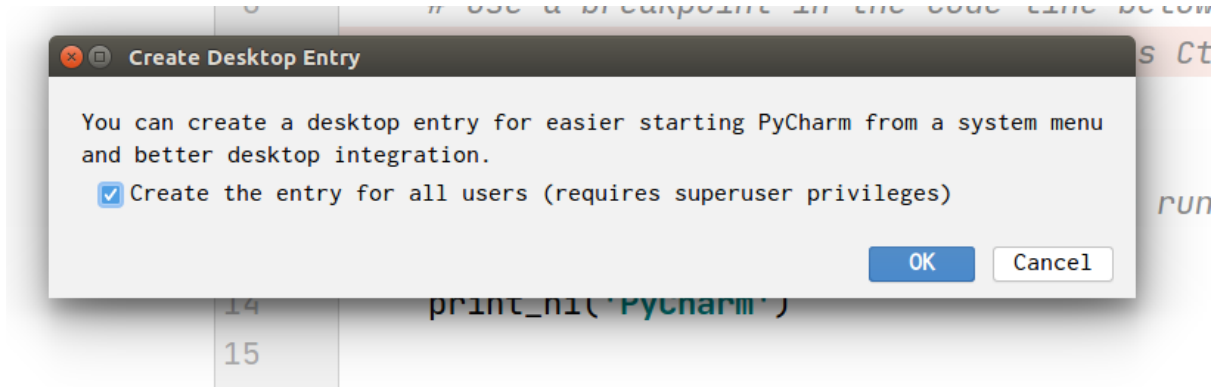
pycharm 설정하는법 : <https://url.kr/p7IEHF>

=[| ,

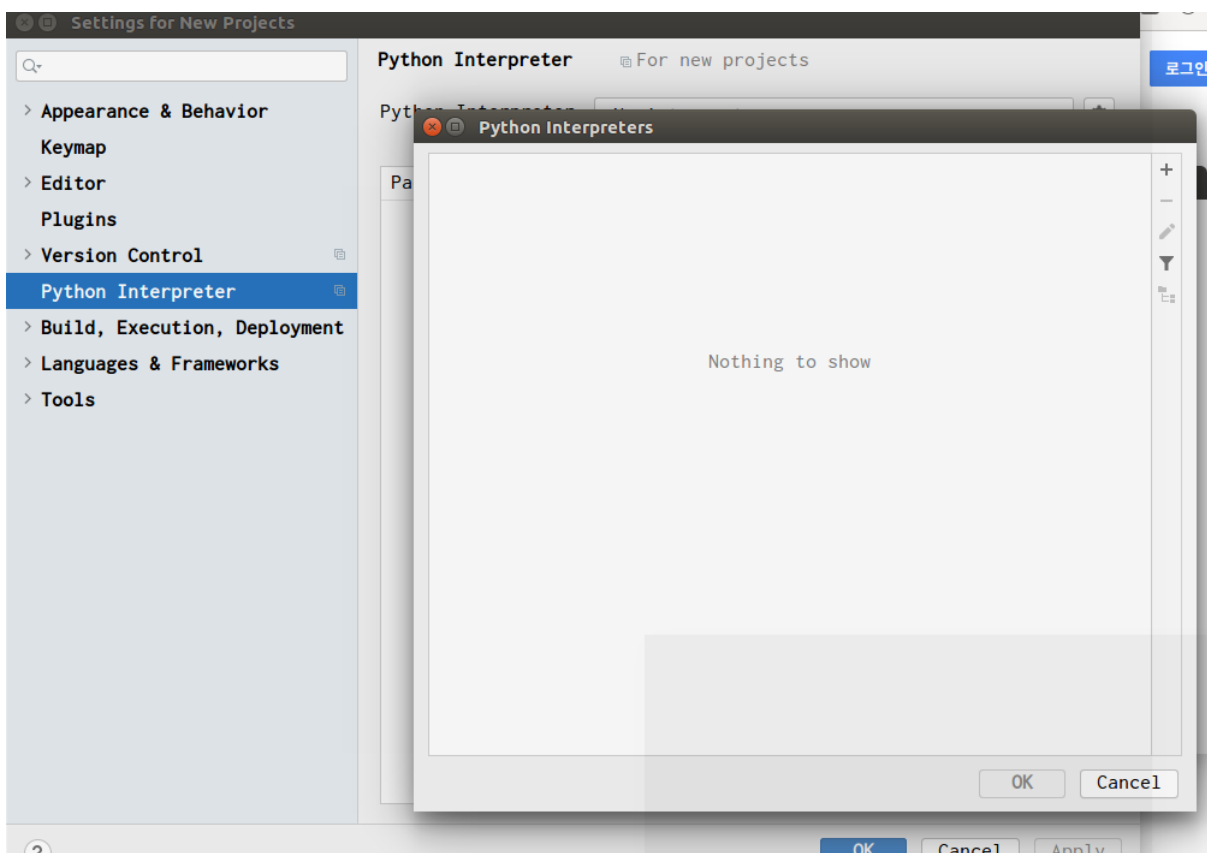


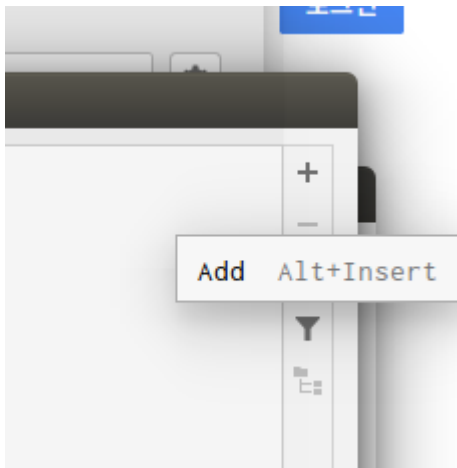
단축키 만드는법

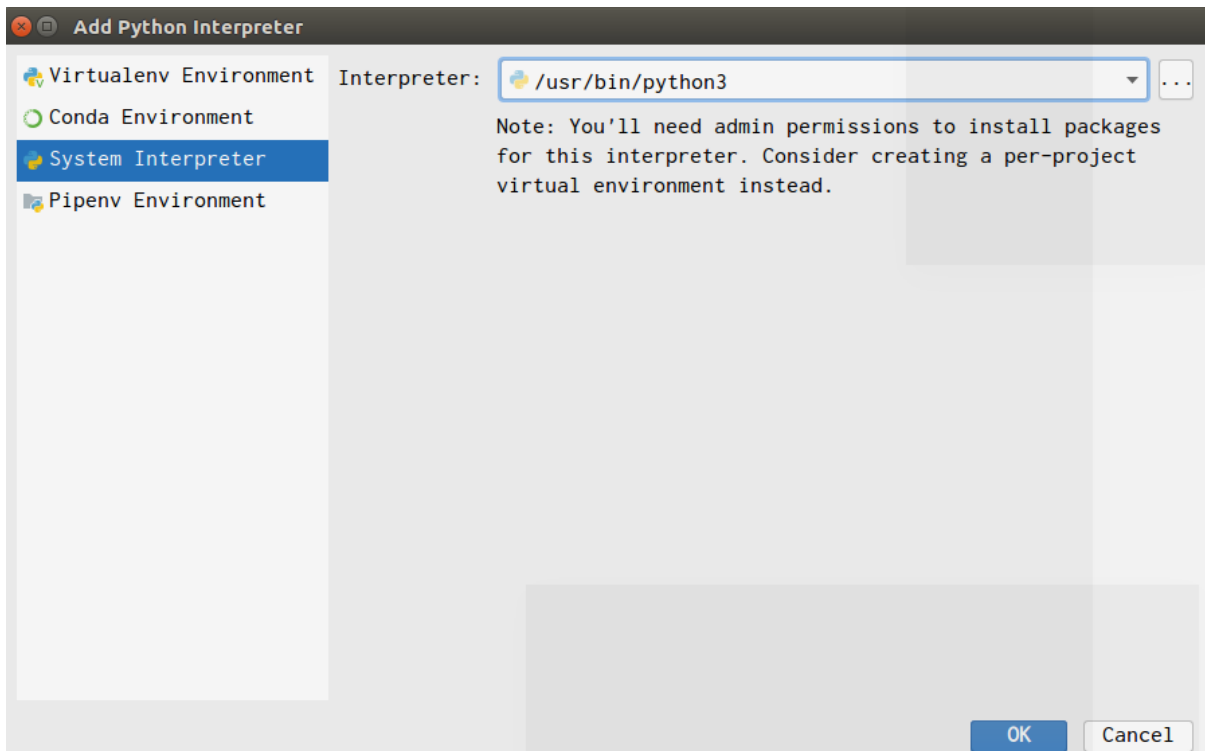
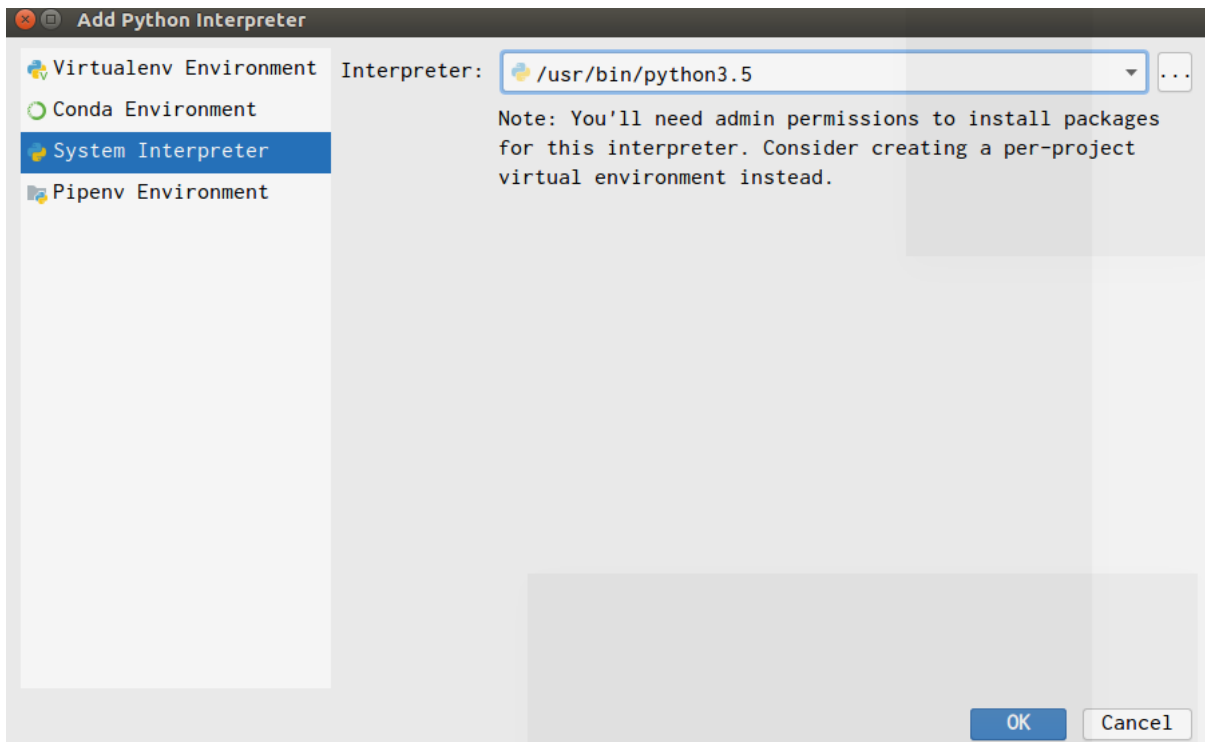
상단 창에서 Tools -> Create Desktop Entry 클릭

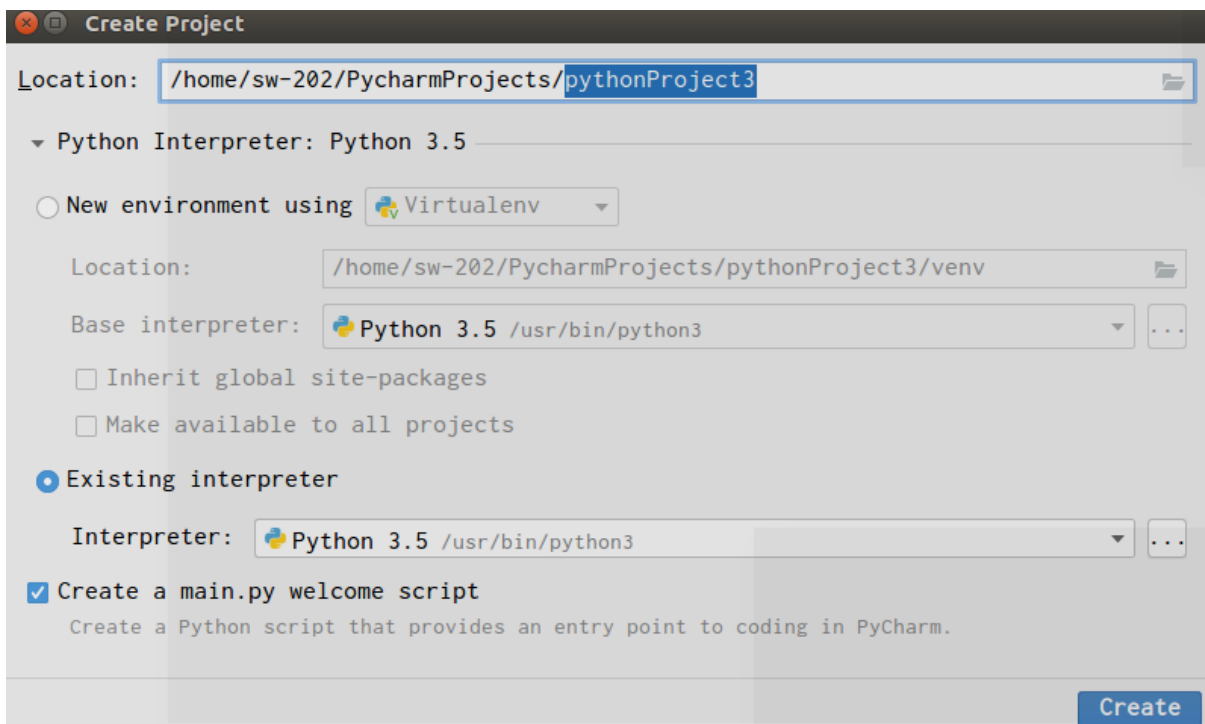
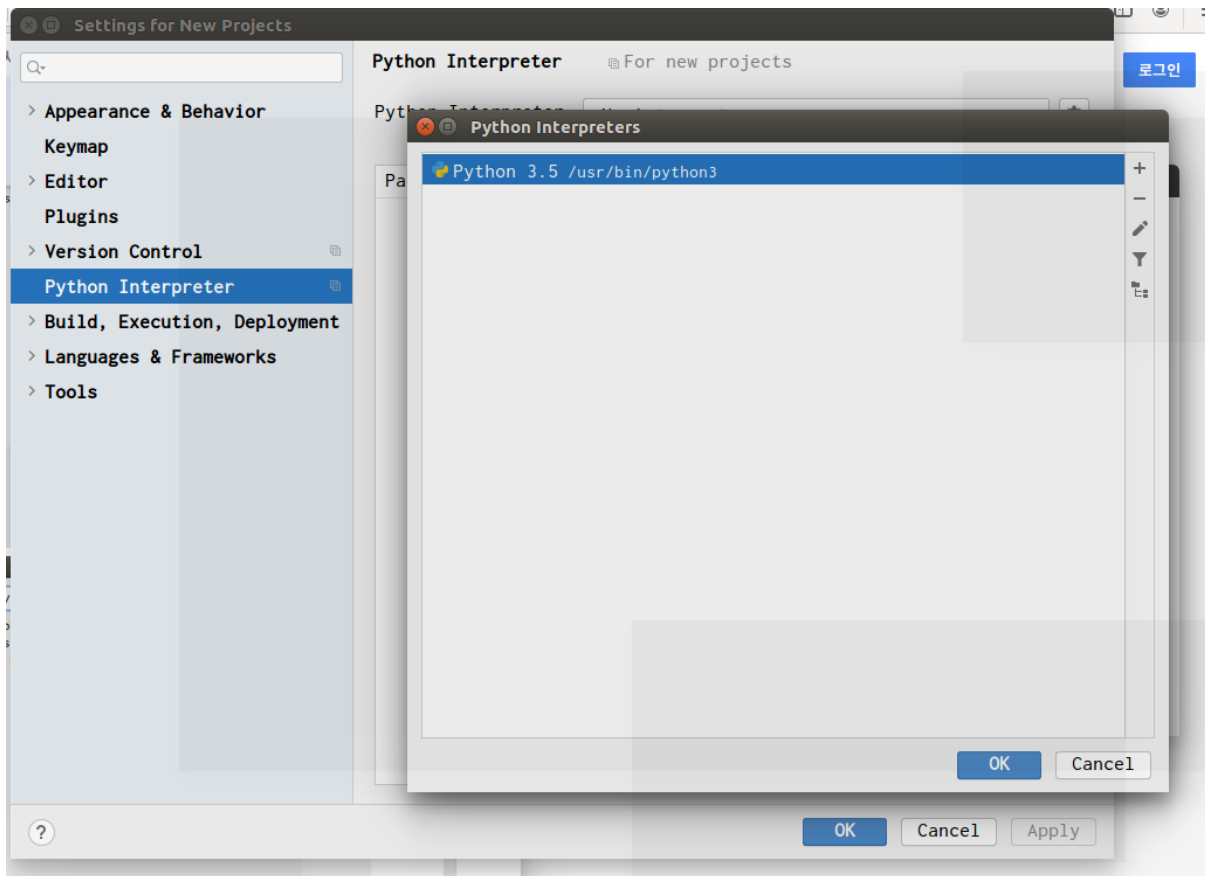


체크박스 클릭 OK!









Existing interpreter !! Not **New environment using**

리눅스에서 할 것
로그인 했던 패스워드 사용

import turtle

오류가 생기는 경우

터미널에서
파이썬 패키지 매니저 설치

sudo apt-get install python3-pip

없다고 하면

sudo apt-get update <- update를 먼저 해주세요

그리고

sudo apt-get install python3-pip

패키지 매니저로 모듈설치 + **apt python3-tk** 설치

pip3 install tk

sudo apt-get install python3-tk

import turtle as t

거북이실행 창이 종료 안되게 하려면 코드 마지막 줄에 다음 명령
삽입

t.mainloop()

질문

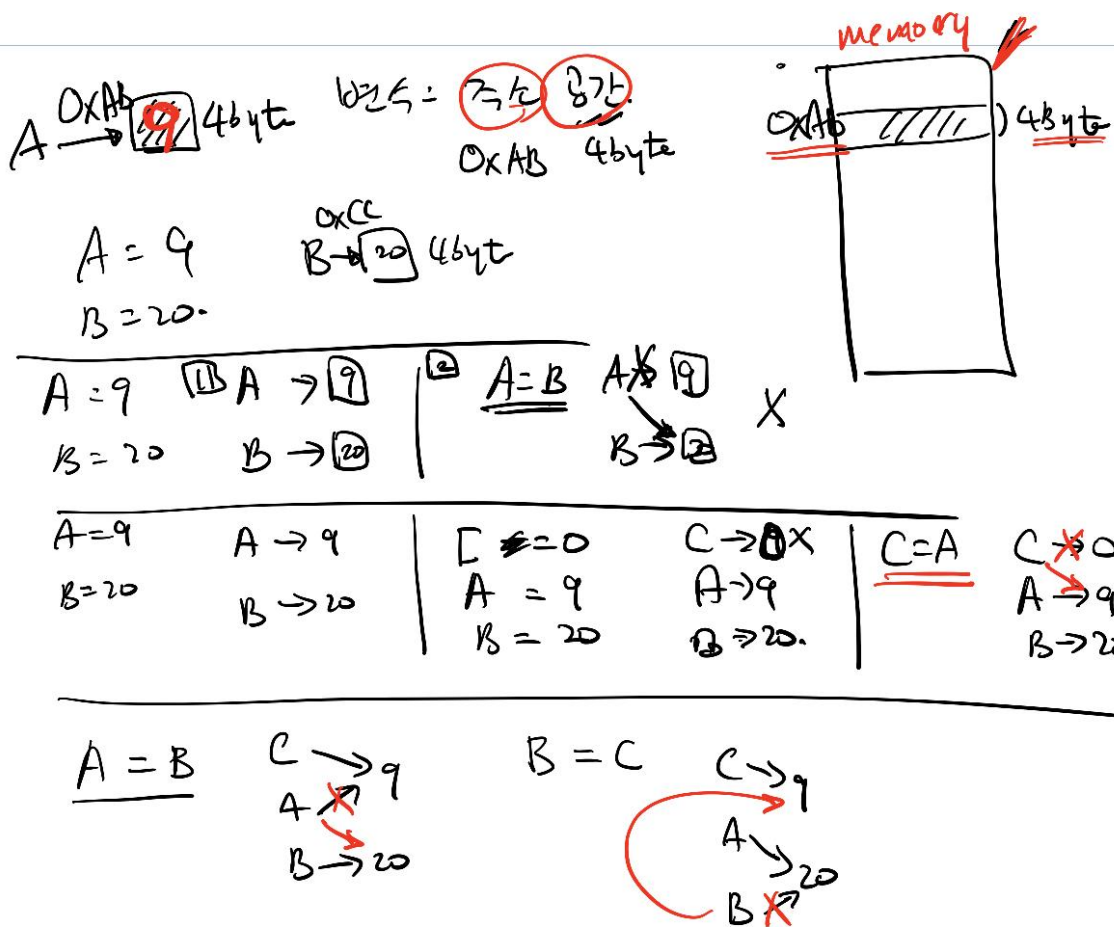
1. 두 변수에 저장된 정보 서로 맞 교환하기

$$A = 9 \quad B = 20$$

$$\left[\begin{array}{l} A = A + B \\ B = B - A \\ A = A + B \\ B = -B \end{array} \right.$$

$\Rightarrow$ 다중 $\frac{2}{2}$ X

$$\begin{array}{l} C = A \\ A = B \\ B = C \end{array}$$



2. turtle을 사용하여 별모양 그리기

3. 생각으로 문제를 풀어 보기

5*32//2 % 5**

$5 * 3^{**2} // 2 \% 5$
 num
 ①
 2
 3
 4

$* \div + -$
 Binary 2m...
 Unary 1m...
 A -, + (X)

②

2번 문제 답

2015011446 하영우

```
import turtle as t

d=300
while True:
    t.forward(d)
    t.left(360/5*2)
    t.forward(d)
    t.left(360/5)

t.mainloop()
```

2015012004 박성호

```
import turtle as t
t.left(36)
for x in range(5):
    t.forward(300)
    t.left(144)

t.mainloop()
```

2015011459 강성욱

```
import turtle as t

d = 300

t.left(-108)
t.forward(d)
t.left(144)
t.forward(d)
t.left(144)
t.forward(d)
t.left(144)
t.forward(d)
t.left(144)
t.forward(d)

t.mainloop()
```

2015011447 허영준

```
import turtle as t
d=100
t.left(36)
t.forward(d)
for x in range(4):
    t.left(144)
```

```
t.forward(d)
t.mainloop()
```

2016011602 안호균

```
import turtle as t
d=240

t.left(36)
t.forward(d)
t.left(144)
t.forward(d)
t.left(144)
t.forward(d)
t.left(144)
t.forward(d)
t.left(144)
t.forward(d)
t.setheading(90)
t.mainloop()
```

2018010680 진유남

```
import turtle as t

for i in range(5):
    t.forward(200)
    t.left(144)

t.mainloop()
```

2020080167 김영재

```
import turtle as t
```

```
d = 300
t.speed(1)
t.right(36)
t.forward(d)
for i in range(4):
    t.left(144)
    t.forward(d)
```

2015012009 배상훈

```
import turtle as t
```

```
for x in range(0,5):  
    t.left(144)  
    t.forward(200)
```

2015011954 고성웅

```
import turtle as t  
d=100  
t.forward(d)  
t.left(144)  
t.forward(d)  
t.left(144)  
t.forward(d)  
t.left(144)  
t.forward(d)  
t.left(144)  
t.forward(d)  
t.left(144)  
t.forward(d)  
t.mainloop()
```

#2014010639 남상민

```
import turtle as tt
```

```
size=100
```

```
for c in range(5)  
    tt.forward(d)  
    tt.left(144)
```

2015012065 전대협

```
import turtle as t  
  
t.forward(100)  
t.left(144)  
t.forward(100)  
t.left(144)  
t.forward(100)  
t.left(144)  
t.forward(100)  
t.left(144)  
t.forward(100)  
t.left(144)  
t.forward(100)  
t.left(144)  
  
t.mainloop()
```

2015011963 김경환

```
import turtle as t
```

```
d=100  
e=144
```

```
t.forward(d)  
t.right(e)  
t.forward(d)  
t.right(e)  
t.forward(d)
```

```
t.right(e)
t.forward(d)
t.right(e)
t.forward(d)
t.right(e)
```

```
t.mainloop()
```

2018011389 백은선

```
import turtle as t
```

```
d=100
```

```
for x in range(5):
    t.forward(d)
    t.right(144)
```

```
t.mainloop()
```

```
import turtle as t
```

```
for i in range(5):
    t.forward(300)
    t.right(144)
```

```
t.mainloop()
```

지정진

2015012063 장선도

```
import turtle as t
```

```
d = 100
```

```
a = 144
```

```
for x in range(5):
```

```
    t.forward(d)
```

```
    t.left(a)
```

```
t.mainloop()
```

김정규

```
import turtle as t
```

2017011393 임채헌

```
import turtle as t
d=100
e=144
t.forward(d)
t.left(e)
t.forward(d)
t.left(e)
t.forward(d)
t.left(e)
t.forward(d)
t.left(e)
t.forward(d)
t.left(e)
t.mainloop()
```

2019080127 주효정

```
import turtle as t
d=100
e=144
t.forward(d)
t.left(e)
t.forward(d)
t.left(e)
t.forward(d)
t.left(e)
```

```
t.forward(d)
t.left(e)
t.forward(d)
t.mainloop()
```

2016011493 김성진

```
import turtle as t

t.shape("turtle")
t.color("purple")
d=300
for i in range(5):
    t.forward(d)
    t.left(180 - 36)

t.mainloop()
```

2015011336 김경훈

```
import turtle as t

t.shape('turtle')
t.speed(0)
for i in range(5):
    t.fd(100)
    t.lt(144)
t.mainloop()
```

2016011301 민운기

```
import turtle as t
for x in range (5):
    t.forward(100)
    t.left(144)
```

2014011990 현동훈

```
import turtle as t

t.forward(100)
t.left(144)
t.forward(100)
t.left(144)
```

```
t.forward(100)
t.left(144)
t.forward(100)
t.left(144)
t.forward(100)
t.left(144)
t.mainloop()
```

2019011539 최민영

```
import turtle as t
for x in range(5):
    t.forward(100)
    t.right(144)

t.mainloop()
```

김광재

```
import turtle as t

t.shape('turtle')
t.speed(1.0)
for i in range(5):
    t.forward(120)
    t.right(145)
```

2016011673 강만길

```
import turtle as t
```

```
d = 100
t.forward(d)
t.left(36)
t.forward(-d)
t.left(36)
t.forward(d)
t.left(36)
t.forward(-d)
t.left(36)
t.forward(d)
t.mainloop()
```

for 변수이름 **in** 반복횟수:

----- 반복할 문장

여기는 반복 안됨

range(5)

0, 1, 2, 3, 4

시작은 항상 **0**

끝이 지정된 숫자 **-1 : 5-1 = 4**

range(2, 5)

2, 3, 4

시작을 재지정 **2**

끝이 지정된 숫자 **-1 : 5-1 = 4**

range(2, 5, 2)

2, 4

시작을 재지정 **2**

끝이 지정된 숫자 **-1 : 5-1 = 4**

증가 값: **+2** 씩 증가, **2, 4**

질문

1. 구구단 출력
2. 다음의 모양 출력하기

```
*
**
***
****
```

3. 다른 모양 (옵션)
-는 출력 안함

```
----*
---**
--***
-****
*****
*
*
*
```

1번 문제 이름 쓰기	2번문제
2015012004 박성호 <pre>for i in range(2, 10): for j in range(2, 10): print(f'{i} * {j} = {i*j}')</pre>	<pre>for i in range(1, 5): print('*' * i)</pre> 트리 만들기 <pre>size = 5 #사이즈 조절 가능 for i in range(size): print(' ' * (size-i), '*' * i) for i in range(size-3): print(' '*(size-1), '*')</pre>
#2015011336 김경훈 <pre>n = int(input("몇 단? : ")) for i in range(9) : print(n, '*', i + 1, '=', n * (i + 1))</pre>	<pre>for i in range(4) : print('*' * (i+1))</pre> <pre>a = "*" for i in range(4) : print(" " * (4 - i), "*", a * i)</pre>

2020080167 김영재 (구구단 출력) <pre>for i in range(2, 10): for j in range(1, 10): print("{0} * {1} = {2}".format(i, j, i * j)) print("\n")</pre>	2020080167 김영재 (문양 넣기) <pre>for j in range(1, 5): print("*" * j)</pre>
2018010680 진유남 <pre>for i in range(1,10): print(i,"단") for j in range(1,10): print(i,"*",j,"=",i*j)</pre>	(문양) <pre>s = "*" for k in range(1,5): print(s*k)</pre> (트리양) <pre>space = " " star = "*" for i in range(4,0,-1): print(space*i+(star+space)*(5-i)) for k in range(1,3): print(space*5+star)</pre>
2019011925 여명준 <pre>for i in range(1, 10): for j in range(1, 10): print(i, " * ", j, " = ", i * j)</pre>	<pre>for i in range(1, 5): print(" * " * i)</pre> <pre>for i in range(1, 7): if i >= 5: print(" * ") else: print(" " * (5 - i), " * " * i)</pre>
2019011539 최민영 <pre>for a in range(1,10): for b in range(1,10): print(a,"*",b,"=",a*b)</pre>	<pre>for x in range(1,5): print("*"*x)</pre>
2018011389 백은선 <pre>for x in range (1,10): print(x,"단") for y in range (1,10): print(x,"*",y,"=",x*y) print("-----")</pre>	<pre>for x in range (1,5): print("*"*x)</pre> (트리만들기) <pre>for x in range (1,5): y = 5 - x print(" "*y,"*"*x)</pre>
강만길 2016011673 <pre>for x in range(9): for y in range(9):</pre>	<pre>for x in range(4): print("*"*(x+1))</pre> 추가문제

<pre>print(x+1,"*",y+1,"=", (y+1)*(x+1))</pre>	<pre>for x in range(1,5): print(" "* (5-x), " "*x) for x in range(1,3): print(" ", " ")</pre>
2015011447 허영준 <pre>for x in range(2,10): print(x, '단') for y in range(1,10): print(x, '*', y, '=', x*y)</pre>	<pre>for x in range(1,5) print(" "*x) 추가 문제 for x in range(5,9): print(' '* (9-x), ' '* (x-4)) print(' *') print(' *')</pre>
2015012063 장선도 <pre>for x in range(1,10): for y in range(1,10): print(x, '*', y, '=', x*y)</pre>	<pre>for x in range(5): print() for y in range(x+1): print('*',end='')</pre>
2015011963 김경환 <pre>a=1 b=1 x=1 for x in range(1,10): for b in range(1, 10): print(a * b) a=a+1</pre>	<pre>for x in range(1,5): print(' '*x)</pre>
2015012065 전대협 <pre>for i in range(2,10) : for j in range(1,10) : print(i,"*",j,"=",i*j)</pre>	<pre>for i in range(1,5) : print(" "*i) 트리 j=4 for i in range(5) : print(' '*j,"*", " "*i) j=j-1</pre>
임채현 <pre>for x in range(1,10): for y in range(1,10): q=x*y print(x, "*", y, "=", q)</pre>	<pre>for x in range(1,10): print("----", x, "단입니다----") for y in range(1,10): q=x*y print(x, "*", y, "=", q)</pre>
김정규 <pre>print("★ 구구단\n") for x in range(2, 10): print("----- [" +</pre>	<pre>for x in range(1,5): print(" "*x)</pre>

<pre> str(x) + "단] -----") for y in range(1, 10): print(x, "X", y, "=", x*y) print("----- -") </pre>	-트리 <pre> n=int(input('number: ')) for i in range(1, n+1): print(" "*(n-i), "***"(2*i-1)) </pre>
배상훈 구구단 <pre> for a in range(1,10): for b in range(1,10): print(a, "*", b, "=", a*b,end=" ") print("\n") </pre>	배상훈 별 <pre> for x in range(1,5): for y in range(0,x): print(" ",end=" ") print("\n") </pre>
주효정 <pre> for x in range(1,10): for i in range(1,10): print(x,"*",i,"=",x*i) </pre>	<pre> for x in range(1,5): print(" "*x) </pre>
김성진 <pre> for x in range(1,10): for y in range(1,10): print(x, "*", y, "=", x*y,end=" ") print("\n") </pre>	<pre> for z in range(5,0,-1): print(" " *z+"* "*(5-z)) </pre>
강성욱 <pre> for i in range(2, 10): for j in range(1, 10): x = i * j print(i, "*", j, "=", x) print("\n") </pre>	<pre> for x in range(1,10): print(" "*x) </pre>
남상민 <pre> for c in range(9): print(str(c+1)+'단') for d in range(9): print(str(c+1)+'x'+str(d+1)+'='+str((c+1)*(d+1))) print("") </pre>	<pre> for c in range(5): for d in range(c): print(" ",end=" ") print("") </pre>
고성웅 <pre> for a in range(1,10): for b in range(1,10): print(a, '곱하기', b, '는', a*b) </pre>	<pre> for c in range(1,5): print(' '*c) </pre>

2016011602 안호균 <pre>for x in range(1,10): a = (list(range(x, (x*10), x))) print(x,"단",a)</pre>	

18p에 설명

foo= int(input("지시어 설명 문"))
int 정수 -100-- 100
bar=float(input("실수형 값입력"))
float 실수 3.14
char 글자형

if (조건a):

참일 때 실행

elif (조건b):

b조건이 참일 때 실행

elif (조건c):

c조건이 참일 때 실행

else:

위의 조건이 모두 거짓일 때 실행

if (foo > bar):

print(foo, "is larger")

else:

printf(bar, "is larger")

질문

1. 숫자를 입력 받고 임의의 숫자를 찾는 게임

```
import random
```

```
foo = random.randint(0, 100)
```

2. 입력받은 세 숫자n의 크기 비교 (가장 큰 값 찾기)

참고: 조건 여러 개 한 번에 묶기

(A > B) and (A > C)

질문의 답

1번 이름쓰기	2번썬
1번 2015012004 박성호 <pre>import random as r foo = r.randint(0, 100) while True: num = int(input("(0~100) 임의의 숫자를 입력해주세요: ")) if num == foo: print("정답입니다!") break elif num > foo: print("숫자가 너무 큼니다.") elif num < foo: print("숫자가 너무 작습니다.")</pre>	2번 <pre>a = int(input("정수를 입력해 주세요: ")) b = int(input("정수를 입력해 주세요: ")) c = int(input("정수를 입력해 주세요: ")) if (a > b) & (a > c): max = a elif b > c: max = b else: max = c print(f'{a}, {b}, {c} 중 가장 큰 수는 {max}')</pre>
장선도 <pre>import random rand1 = random.randint(0,100) while 1: select = int(input("0~100 중, 임의의 수를 입력하시오")) if(rand1 == select): print("일치") break else: print("불일치") if (rand1 > select):</pre>	<pre>a = int(input("0~100 중, 임의의 수를 입력하시오")) b = int(input("0~100 중, 임의의 수를 입력하시오")) c = int(input("0~100 중, 임의의 수를 입력하시오")) if(a>=b) and (a>=c): print("입력한 값 중 가장 큰 값은", a, '입니다.') elif (b >= a) and (b >= c): print("입력한 값 중 가장 큰 값은", b, '입니다.') else:</pre>

<pre> print('입력값보다 큼니다.') else: print('입력값보다 작습니다.')</pre>	<pre> print("입력한 값 중 가장 큰 값은", c, '입니다.')</pre>
2015011336 김경훈 <pre> import random foo = random.randint(1,100) ans = int(input("몇 일까요?")) while ans != foo : if ans > 99 or ans < 1: print("1이상 99이하의 숫자입니다.") ans = int(input("몇 일까요?")) elif ans > foo : print("그 것보단 작습니다.") ans = int(input("몇 일까요?")) elif ans < foo : print("그 것보단 큼니다.") ans = int(input("몇 일까요?")) print("정답입니다!")</pre>	<pre> num_list = [] for i in range(3) : num = int(input("숫자를 입력해주세요. ")) num_list.append(num) if num_list[0] > num_list[1] and num_list[0] > num_list[2] : print(num_list[0], "이 가장 크네요.") elif num_list[1] > num_list[0] and num_list[1] > num_list[2] : print(num_list[1], "이 가장 크네요.") elif num_list[2] > num_list[0] and num_list[2] > num_list[1] : print(num_list[2], "이 가장 크네요.")</pre>
2016011594 김진섭 <pre> import random foo = random.randint(1,100) a=int(input("숫자를 입력하세요")) while a != foo: if a > foo : print("더 작은 수입니다") a = int(input("숫자를입력하세요")) elif a < foo: print("더큰수입니다") a = int(input("숫자를입력하세요")) print("정답입니다")</pre>	<pre> a = int(input("숫자를 입력하세요")) b = int(input("숫자를 입력하세요")) c = int(input("숫자를 입력하세요")) if (a>=b) & (a>=c) : print("입력한 값 중 가장 큰 값은", a, "입니다.") elif (b >= a) & (b >= c) : print("입력한 값 중 가장 큰 값은", b, "입니다.") else: print("입력한 값 중 가장 큰 값은", c, "입니다.")</pre>
2019011925 여명준 <pre> import random foo = random.randint(0, 100) while True: N = int(input("--->")) if foo == N: print("Yes") break elif foo > N: print("Low") else:</pre>	#Q2 <pre> a = int(input("Enter a: ")) b = int(input("Enter b: ")) c = int(input("Enter c: ")) if ((a >= b) and (a >= c)): print(a) elif ((b >= a) and (b >= c)): print(b) elif ((c >= a) and (c >= b)): print(c) print("Answer: ", a)</pre>

<pre>print("High")</pre>	
2015011447 허영준 <pre>import random y=random.randint(0,100) while True: x = int(input("숫자 :")) if y==x: break if y > x: print("UP!") else: print("DOWN!") print('정답')</pre>	<pre>x,y,z=int(input('x값 :')),int(input('y값 :')),int(input('z값 :')) if (x>y) and (x>z): print("x가 가장큰 :",x) elif y>z: print("y가 가장큰 :",y) else: print("z가 가장큰 :",z)</pre>
2017011393 임채현 <pre>import random as r foo = r.randint(0, 100) while True: num = int(input("0~100CHOOSE:")) if num == foo: print("YOU WIN!") break if num > foo: print("BIG") if num < foo: print("SMALL")</pre>	<pre>a = int(input("숫자를 입력하세요")) b = int(input("숫자를 입력하세요")) c = int(input("숫자를 입력하세요")) if (a>b) and (a>c) print(a) elif (b>c) print(b) else: print(c)</pre>
2014010927지정진 <pre>import random foo = random.randint(0, 100) while(1): a = int(input("Input number")) if(a == foo): print("True") break elif(a<foo): print("bigger than foo") else: print("smaller than foo")</pre>	<pre>a = int(input("input a val")) b = int(input("input b val")) c = int(input("input c val")) if(a>=b) and (a>=c): print(a) elif(b>=c): print(b) else: print("Max val = ", c)</pre>
2020080167 김영재 랜덤 숫자 찾기 <pre>import random foo = random.randint(0, 10)</pre>	2020080167 김영재 가장 큰 숫자찾기 <pre>num1 = int(input("첫 번째 숫자 입력 : \n")) num2 = int(input("두 번째 숫자 입력 : \n"))</pre>

<pre>while True: num = int(input("input number :")) if (num == foo): print("You finally get it!\n") break else: print("Oops!\n")</pre>	<pre>num3 = int(input("세 번째 숫자 입력 : \n")) if (num1 >= num2) and (num1 >= num3): ans = num1 elif (num2 >= num1) and (num2 >= num3): ans = num2 else: ans = num3 print(ans, "is largest number\n")</pre>
20	
2016011673 강만길 <pre>import random foo = random.randint(0,10) while True: bar = int(input("number plz")) if foo > bar: print("값이 더 큼니다") elif foo < bar: print("값이 더 작습니다") else: print("정답입니다") break</pre>	<pre>a=float(input("값을 입력해주세요")) b=float(input("값을 입력해주세요")) c=float(input("값을 입력해주세요")) if (a>b) and (a>=c): print(a) elif (b>=a) and (b>=c): print(b) else: print(c)</pre>
2019011539 최민영 <pre>import random foo=random.randint(0,100) while True: x=int(input("0~100 중 숫자를 입력하시오.:")) if (x==foo): print("정답!") break elif (x > foo): print("DOWN!") else: print("UP!")</pre>	<pre>a=int(input("정수를 입력하시오. ")) b=int(input("정수를 입력하시오. ")) c=int(input("정수를 입력하시오. ")) if (a>=b and a>=c): print(a) elif (a>=b and a<=c): print(c) elif (a<b and b>=c): print(b) elif (a<b and b<=c): print(c)</pre>
김정규 <pre>import random answer = random.randint(1, 100) print("정답은 " + str(answer) + "입니다.") while True: user = int(input("1~100 사이의 값을 입력하세요 >>> "))</pre>	<pre>import random a=random.randint(0, 100) print(a) b=random.randint(0, 100) print(b) c=random.randint(0, 100) print(c) if (a >= b) and (a >= c): largest=a elif (b >= a) and (b >= c): largest=b</pre>

<pre> print("입력값은 " + str(user) + "입니다.") if user > answer: print("입력값이 정답보다 큽니다.") if user < answer: print("입력값이 정답보다 작습니다.") if user == answer: print("정답입니다!") break </pre>	<pre> else: largest=c print(largest, "is the greatest of all") </pre>
2019080127 주효정 <pre> import random foo = random.randint(0,100) while True: x=int(input("임의의 숫자를 입력")) if (foo==x): print("오 정답") break if (foo>x): print("작아요!") if (foo<x): print("커요!") </pre>	<pre> a= int(input("a 정수를 입력")) b= int(input("b 정수를 입력")) c= int(input("c 정수를 입력")) if (a>b) and (a>c): print("a") if (b>c) and (b>a): print("b") if (c>a) and (c>b): print("c") </pre>
2018011389 백은선 <pre> import random foo = random.randint(0, 100) while True: num = int(input("1. 숫자를 입력하세요. ")) if foo == num : break if foo > num : print("foo가", num, "보다 더 큽니다.") if foo < num : print("foo가", num, "보다 더 작습니다.") print(foo,"를 찾았습니다.") </pre>	<pre> num1 = int(input("숫자1을 입력해주세요. ")) num2 = int(input("숫자2를 입력해주세요. ")) num3 = int(input("숫자3을 입력해주세요. ")) maxnum = 0 if (num1 > num2) and (num1 > num3) : maxnum = num1 if (num2 > num1) and (num2 > num3) : maxnum = num2 if (num3 > num1) and (num3 > num2) : maxnum = num3 print("가장 큰 숫자는",maxnum,"입니다.") </pre>
2015011963 김경환 <pre> import random as r f=r.randint(0,100) print(f) while True: u=int(input("숫자를 입력하세요: ")) if u<f: print('정답보다 작습니다') </pre>	<pre> a= int(input('첫번째 숫자를입력하십시오: ')) b= int(input('두번째 숫자를입력하십시오')) c= int(input('세번째 숫자를입력하십시오')) if a>=b and a>=c: </pre>

```
elif u>f:
    print('정답보다 큼니다')
else:
    print('정답입니다')
    break
```

```
print('가장큰숫자는',a)
if b>=c and b>=a:
    print('가장큰숫자는',b)
if c>=a and c>=b:
    print('가장큰숫자는',c)
```

2015012065 전대협

```
import random
f = random.randint(0,100)

while True:
    a = int(input("숫자를 입력하세요"))
    if a == f :
        print("정답입니다.")
        break
    elif a > f :
        print("숫자가 큼니다.")
    elif a < f :
        print("숫자가 작습니다.")
```

```
A = int(input("숫자 A를 입력하세요"))
B = int(input("숫자 B를 입력하세요"))
C = int(input("숫자 C를 입력하세요"))

if A >= B and A >= C :
    print("가장 큰 값은",A)
elif B >= A and B >= C :
    print("가장 큰 값은", B)
elif C >= A and C >= B :
    print("가장 큰 값은", C)
```

2015011459 강성욱

```
import random
foo = random.randint(0, 100)
print(foo)

while True:
    n = int(input("숫자를
    입력해주세요."))
    if n == foo:
        print("정답입니다.")
        break
    elif n > foo:
        print("입력값이 정답보다 큼니다.")
    elif n < foo:
        print("입력값이 정답보다
        작습니다.")
```

```
import random
foo1 = random.randint(0, 100)
foo2 = random.randint(0, 100)
foo3 = random.randint(0, 100)
print(foo1)
print(foo2)
print(foo3)

if foo1 > foo2 and foo3:
    print(foo1, "이 가장 크다.")
elif foo2 > foo1 and foo3:
    print(foo2, "이 가장 크다.")
else:
    print(foo3, "이 가장 크다.")
```

2016011602 안호균

```
import random
foo = random.randint(0,100)

number = int(input("Insert Number"))

while number != foo:
    if foo > number :
        print("bigger")
        number = int(input("Insert
        Number"))
    elif foo < number:
        print("smaller")
        number = int(input("Insert
        Number"))
    else:
        print("right")
```

```
number1 = int(input("Input a
number"))
number2 = int(input("Input a
number"))
number3 = int(input("Input a
number"))

if (number1 > number2) and (number1 >
number3):
    print(number1, "is the biggest
number")
elif (number2 > number3) and (number2
> number3):
    print(number2, "is the biggest
number")
else :
    print(nubmer3, "is the biggest
```

	<pre>number")</pre>
2015011446 하영우 <pre>import random foo = random.randint(0, 100) while True: bar = int(input("1부터 100까지의 숫자 중 임의 숫자를 맞춰보시오")) if (foo == bar): print("무작위의 숫자와 동일합니다. 프로그램을 종료합니다.") break elif(0 <= bar < foo): print("무작위의 수보다 작습니다.") elif (foo < bar <= 100): print("무작위의 수보다 큼니다.") else: print("범위를 벗어났습니다. 1~100 사이의 수를 입력하십시오.")</pre>	<pre>a = int(input("a의 값을 입력하십시오.)) b = int(input("b의 값을 입력하십시오.)) c = int(input("c의 값을 입력하십시오.)) if a>b & a>c: print("a의 값이 제일 크고, 그 값은",a,"입니다.") elif b>a & b>c: print("b의 값이 제일 크고, 그 값은", b, "입니다.") else: print("c의 값이 제일 크고, 그 값은", c, "입니다.")</pre>
김성진 <pre>import random x=random.randint(0,10) while True: y=int(input("숫자를 입력하세요.)) if x==y: print("게임끝") break else: print("계속도전")</pre>	<pre>import random x=random.randint(0,10) y=random.randint(0,10) z=random.randint(0,10) if x>=y and x>=z: print("x가 가장 큰 숫자 입니다.") elif y>=z and y>=x: print("y가 가장 큰 숫자 입니다.") else: print("z가 가장 큰 숫자 입니다.")</pre>
2014011990 현동훈 1번문제 <pre>import random foo=random.randint(0,100) while True: bar = int(input("숫자를 입력하세요")) if(foo > bar): print(bar,"is smaller") elif(foo < bar): print(bar,"is larger") else: print("일치합니다.") break</pre> 2번문제 <pre>a = int(input("숫자를 입력하세요")) b = int(input("숫자를 입력하세요")) c = int(input("숫자를 입력하세요"))</pre>	

<pre> if (a>b) and (a>c): print(a,"는 가장 큰 수입니다.") elif (b>a) and (b>c): print(b,"는 가장 큰 수입니다.") else: print(c,"는 가장 큰수입니다.") </pre>	
2015011954 고성웅 <pre> import random r = random.randint(0,100) while True: a = int(input("숫자를 입력하세요")) if(a==r): print("맞췄습니다!") break elif(a>r): print("입력하신 숫자가 큼니다. 다시해보세요.") elif(a<r): print("입력하신 숫자가 작습니다. 다시해보세요.") </pre>	<pre> a=input('처음 숫자를 입력해주세요.') b=input('다음 숫자를 입력해주세요.') c=input('마지막 숫자를 입력해주세요.') if((a>=b) and (b>=c)): print('가장 큰 값은', a, '이고', '가장 작은 값은', c, '입니다.') elif((a>=c) and (c>=b)): print('가장 큰 값은', a, '이고', '가장 작은 값은', b, '입니다.') elif((b>=c) and (c>=a)): print('가장 큰 값은', b, '이고', '가장 작은 값은', a, '입니다.') elif((b>=a) and (a>=c)): print('가장 큰 값은', b, '이고', '가장 작은 값은', c, '입니다.') elif((c>=a) and (a>=b)): print('가장 큰 값은', c, '이고', '가장 작은 값은', b, '입니다.') elif((c>=b) and (b>=a)): print('가장 큰 값은', c, '이고', '가장 작은 값은', a, '입니다.') </pre>

20201108

어제 한 것 복습 문제 풀이

1. 컴퓨터와 가위 바위 보 게임을 하여 **5판 3승** 하면 이기도록 만들어보자

- while
- if else
- import random; random.randint(시작, 끝) ; 끝 값 포함
- 또는 random.choices(["가위", "바위", "보"]) 이나 random.choice(["3가위", "바위", "보"])을 사용할 수 있음. 단, 비교를 "가위", "바위", "보"와 해야 하고 입력도 "가위", "바위", "

- 보"로 받아야 함(귀찮음)
- 숫자와 가위 바위 보 매핑 해서 해결
- 이긴 횟수를 저장할 변수
- 게임 반복
- 흔히 하는 실수: =, == 구분 안하기

이름쓰고, 코드 쓰기

2015011446 하영우

```
import random as r

gcount = 0
wincount = 0

while True:
    gbb_com = r.randint(1, 3)
    gbb = str(input("가위, 바위, 보를 입력하세요"))
    if (gbb == "가위" and gbb_com == 1) or (gbb == "바위" and gbb_com == 2)
or (gbb == "보" and gbb_com == 3):
        print("비겼습니다.")
        gcount += 1
    elif (gbb == "가위" and gbb_com == 2) or (gbb == "바위" and gbb_com ==
3) or (gbb == "보" and gbb_com == 1):
        print("졌습니다.")
        gcount += 1
    elif (gbb == "가위" and gbb_com == 3) or (gbb == "바위" and gbb_com ==
1) or (gbb == "보" and gbb_com == 2):
        print("이겼습니다.")
        wincount += 1
        gcount += 1
    else:
        print("오류! 정확한 값을 입력하세요.")

    if gcount >= 5:
        break

print("경기가 끝났습니다. 결과는 5판", wincount, "승입니다.")
```

2015011336 김경훈

```
import random

userScore = 0
comScore = 0
while userScore < 3 and comScore < 3 :
    randnum = random.randrange(3)
```

```

user = input("무엇을 내시겠습니까? (가위, 바위, 보) : ")

if randnum == 0 :
    randnum = "가위"
elif randnum == 1 :
    randnum = "바위"
elif randnum == 2 :
    randnum = "보"
# if (user == randnum )
    if (user == "가위" and randnum == "가위") or (user == "바위" and randnum
== "바위") or (user == "보" and randnum == "보") :
        print("비겼습니다!\n")
    elif (user == "가위" and randnum == "바위") or (user == "바위" and
randnum == "보") or (user == "보" and randnum == "가위") :
        print("졌습니다!\n")
        comScore = comScore + 1
    elif (user == "가위" and randnum == "보") or (user == "바위" and randnum
== "가위") or (user == "보" and randnum == "바위") :
        print("이겼습니다!!\n")
        userScore = userScore + 1
    else :
        print("잘못 입력하셨습니다.")
print("현재 스코어 \nuser : %s승, com : %s승\n" %(userScore, comScore))

if comScore == 3 :
    print("최종 패배하였습니다.")
elif userScore == 3:
    print("최종 승리하였습니다.")

```

2015012004 박성호

```

import random as r

def worl(user, com, i, win):
    if user == com:
        print(f'무승부 {i}전 {win}승')
    elif ((user == 1)&(com == 3))|((user == 2)&(com == 1))|((user ==
3)&(com == 2)):
        win += 1
        print(f'이겼습니다! {i}전 {win}승')
    else:
        print(f'졌습니다 {i}전 {win}승')
    return win

def userNum(user):
    if user == '가위':
        user = 1
    elif user == '바위':
        user = 2

```

```

        elif user == '보':
            user = 3
        else:
            print("정확한 단어를 입력해주세요.")
            user = 0
        return user

i = 1
win = 0
while True:
    user = input("가위, 바위, 보 중에 입력 해주세요")
    com = r.randint(1,3)    #1 = 가위, 2 = 바위, 3 = 보

    user = userNum(user)
    win = worl(user, com, i, win)

    print()
    if win >= 3:
        break
    if (i == 5):
        print(f'패배 {win}승')
        i = 0
        win = 0
    i += 1

print(f'승리! {i}전 {win}승')

```

2020080167 김영재(가위 바위 보 게임)

```
import random as R
```

```
RSP_alphago = R.randint(0, 2) # 가위 바위 보를 내주는 변수
# 사람과 똑같이 가위 = 0 바위 = 1 보 = 2
```

```
count_human = 0 # 이긴 횟수를 카운트하는 변수
count_alphago = 0 # 컴퓨터가 이긴 횟수를 카운트
```

```
while count_human < 3 or count_alphago < 3:
```

```
    if count_human == 3 :
        print("#" * 5, " 인간 승리ㅎㅎㅎㅎ ", "#" * 5)
        break
```

```
    elif count_alphago == 3 :
        print("#" * 5, " 컴퓨터 승리TTTTTT ", "#" * 5)
        break
```

```
    RSP_human = int(input("가위 = 0 / 바위 = 1 / 보 = 2 중 하나를
입력하세요(숫자로만 입력) :"))
```

```
    if RSP_human == 0 :
        if RSP_alphago == 0 :
```

```

        print("사람 입력 : 가위\t 컴퓨터 입력 : 가위\n비겼습니다.\n")
        continue
    elif RSP_alphago == 1 :
        print("사람 입력 : 가위\t 컴퓨터 입력 : 바위\n졌습니다.\n")
        count_alphago += 1
        continue
    else :
        print("사람 입력 : 가위\t 컴퓨터 입력 : 보\n이겼습니다.\n")
        count_human += 1
        continue
    elif RSP_human == 1 :
        if RSP_alphago == 0 :
            print("사람 입력 : 바위\t 컴퓨터 입력 : 가위\n이겼습니다.\n")
            count_human += 1
            continue
        elif RSP_alphago == 1 :
            print("사람 입력 : 바위\t 컴퓨터 입력 : 바위\n비겼습니다.\n")
            continue
        else :
            print("사람 입력 : 바위\t 컴퓨터 입력 : 보\n졌습니다.\n")
            count_alphago += 1
            continue
    elif RSP_human == 2 :
        if RSP_alphago == 0 :
            print("사람 입력 : 보\t 컴퓨터 입력 : 가위\n졌습니다.\n")
            count_alphago += 1
            continue
        elif RSP_alphago == 1 :
            print("사람 입력 : 보\t 컴퓨터 입력 : 바위\n이겼습니다.\n")
            count_human += 1
            continue
        else :
            print("사람 입력 : 보\t 컴퓨터 입력 : 보\n비겼습니다.\n")
            continue
    elif RSP_human > 2 :
        print("$$$ 잘못 입력하셨습니다. $$$\n가위 = 0, 바위 = 1, 보 = 2 중
하나만 입력해주세요.\n")
        continue

```

2015012009 배상훈

```

import random #랜덤 함수 import
win = 0 #변수 선언
lose = 0
while True: #반복문
    a = random.randint(0,2) # 0 : 목 , 1 : 찌 , 2: 빠
    b = int(input("0 : 목 , 1 : 찌 , 2: 빠 // your choice is :"))
    if((a==0 and b==2) or (a ==1 and b==0) or (a==2 and b == 1)): #유저가
이길 경우
        print("you win")
        win = win + 1
        print("win :",win," , lose :",lose)
        if(win == 3): #3승 했을 때
            print("you win three times!! Congratuation!!!")
            break #반복문 종료
    elif((a==0 and b==1) or (a ==1 and b==2) or (a==2 and b == 0)):
#유저가 질 경우
        print("you lose")
        lose = lose + 1
        print("win :",win," , lose :", lose)
        if (lose == 3): #3패 했을 때

```

```

        print("you lose three times!! Cheer up!!")
        break #반복문 종료
    else : #비길 경우
        print("draw")

```

김성진

```

num = 0
win = 0
while (num>=0):
    import random;
    foo = random.randint(1, 4)
    if foo == 2:
        com = "바위"
    elif foo == 3:
        com = "가위"
    else:
        com = "보"
    play=str(input("가위 바위 보 중 하나를 내세요"))
    if play=="가위":
        num = num+1
        if com == "보":
            win = win + 1
            print("승")
        elif play== "바위":
            num = num+1
            if com == "가위":
                win = win + 1
                print("승")
        elif play == "보":
            num = num+1
            if com == "바위":
                win = win + 1
                print("승")
        else:
            print("올바르게 입력해주세요")
    if win == 3:
        print("게임 우승")
        break
    elif num==5:
        print("게임 패배")
        break

```

김정규

```

import random

com = random.choice(['가위', '바위', '보'])
user = 0

user = input("가위, 바위, 보 중 하나를 입력 (끝내려면 '종료' ):")

com = random.choice(["가위", "바위", "보"])

if user == '종료':
    print('프로그램 종료\n')

```

```

elif (user == '가위' and com == '보') \
    or (user == '바위' and com == '가위') \
    or (user == '보' and com == '바위'):
    print('사용자 : ' + user + ', 컴퓨터 : ' + com + ', 사용자
승\n')
elif (user == '가위' and com == '가위') \
    or (user == '바위' and com == '바위') \
    or (user == '보' and com == '보'):
    print('사용자 : ' + user + ', 컴퓨터 : ' + com + ', 무승부\n')
elif (user == '가위' and com == '바위') \
    or (user == '바위' and com == '보') \
    or (user == '보' and com == '가위'):
    print('사용자 : ' + user + ', 컴퓨터 : ' + com + ', 컴퓨터
승\n')
else:
    print('다시 입력해 주세요.\n')

```

2016011493김성진

```

import random
account=0
bcount=0
while True:
    user=input("가위바위보 중 하나를 입력하세요:")
    if user=='가위':
        if random.choice(["가위","바위","보"])=="가위":
            print("무승부")
            account +=1
        elif random.choice(["가위","바위","보"])=="바위":
            print("패")
            account +=1
        else:
            print("승")
            bcount +=1
    if user=='바위':
        if random.choice(["가위","바위","보"])=="가위":
            print("승")
            bcount +=1
        elif random.choice(["가위","바위","보"])=="바위":
            print("무승부")
            account +=1
        else:
            print("패")
            account +=1
    if user=='보':
        if random.choice(["가위","바위","보"])=="가위":
            print("패")
            account +=1
        elif random.choice(["가위","바위","보"])=="바위":
            print("승")
            bcount +=1
        else:
            print("무승부")
            account +=1

    if account+bcount==5 and bcount>=3:
        print("승리하였다!")
        break
    elif account+bcount==5 and bcount<3:
        print("패배하였다!")

```

break

2015011954 고성웅

```
import random
x=0;y=0;z=0
import random; computer=random.choice(["가위", "바위", "보"])
while True:
    player = input("가위?바위?보?")
    computer = random.choice(["가위", "바위", "보"])
    if player=='가위':
        z = z + 1
        if computer=="보":
            x=x+1
            print("이겼습니다!",x,"승입니다.")
        elif computer=="바위":
            y=y+1
            print("졌습니다.",x,"승입니다.")
        elif computer=="가위":
            print("비겼습니다.",x,"승입니다.")
    elif player=="바위":
        z = z + 1
        if computer=="가위":
            x=x+1
            print("이겼습니다.",x,"승입니다.")
        elif computer=="바위":
            print("비겼습니다.",x,"승입니다.")
        elif computer=="보":
            y = y + 1
            print("졌습니다.", x, "승입니다.")
    elif player=="보":
        z = z + 1
        if computer=="바위":
            x=x+1
            print("이겼습니다.",x,"승입니다.")
        elif computer=="보":
            print("비겼습니다.",x,"승입니다.")
        elif computer=="가위":
            y = y + 1
            print("졌습니다.", x, "승입니다.")
    if z==5:
        break
    if x==3:
        break
    if y==3:
        break
if z==5:
    if x>y:
        print('당신은 이겼습니다')
    elif x<y:
        print('당신은 졌습니다.')
if x==3:
    print('당신은 이겼습니다.')
elif y==3:
    print('당신은 졌습니다.')
if z==5:
    if x>y:
        print('당신은 이겼습니다')
    elif x<y:
        print('당신은 졌습니다.')
```

2018010680 진유남

```
import random

win = 0

game = 0

while True:

    x = input("가위, 바위, 보를 입력하세요: ")

    y = random.choice(["가위", "바위", "보"])

    if x=="가위":

        game = game+1

        if y=="가위":

            print("비겼습니다.")

        elif y=="바위":

            print("졌습니다.")

        else:

            print("이겼습니다.")

        win = win+1

    elif x=="바위":

        game = game+1

        if y=="바위":

            print("비겼습니다.")

        elif y=="보":

            print("졌습니다.")

        else:

            print("이겼습니다.")

        win = win+1

    else:
```

```

game = game+1

if y=="보":

    print("비겼습니다.")

elif y=="가위":

    print("졌습니다.")

else:

    print("이겼습니다.")

win = win+1

if game==5 and win>=3:

    print("이김")

    break

elif game==5 and win<3:

    print("lose :( ")

    break

```

2018011389 백은선

```

import random
num = 0
t = 1

```

```

while (t < 6):
    my = int(input("가위 (1) 바위 (2) 보 (3) !"))
    com = random.randint(1, 3)

```

```

    if (my == 1):
        t = t + 1
        if (com == 1): print("비겼습니다.")
        elif (com == 2): print("졌습니다.")
        else:
            num = num + 1
            print("이겼습니다.")

```

```

    elif (my == 2):
        t = t + 1
        if (com == 1):
            num = num + 1
            print("이겼습니다.")
        elif (com == 2): print("비겼습니다.")
        else: print("졌습니다.")

```

```

elif (my == 3):
    t = t + 1
    if (com == 1): print("졌습니다.")
    elif (com == 2):
        num = num + 1
        print("이겼습니다.")
    else: print("비겼습니다.")

else : print("숫자를 잘못 입력하셨습니다.")

if num >= 3:
    print("5판", num, "승으로 이겼습니다.")
else :
    print("5판", num, "승으로 졌습니다.")

```

김채진

```

import random

# i = 판수
i = 0

# score = 이긴횟수
score = 0

# 5판 내로
while i < 6:
    a = (int(input("Enter (가위: 1, 바위: 2, 보: 3)")))
    ans = random.randint(1, 3)

    print("Answer is ", ans)

    if a == 1:
        i = i+1
        if ans == 1:
            print("draw")
        elif ans == 2:
            print("lose")
        else:
            print("win")
            score = score + 1

    elif a == 2:
        i = i + 1
        if ans == 1:
            print("win")
            score = score + 1
        elif ans == 2:
            print("draw")
        else:
            print("lose")

    elif a == 3:
        i = i + 1
        if ans == 1:
            print("lose")
        elif ans == 2:

```

```

        print("win")
        score = score + 1
    else:
        print("draw")

```

```

if score >= 3:
    print("You Win")
else:
    print("You lose")

```

임채현

```

import random as r
Q = r.randint(1,3)
w=0
while w<5:
    me = int(input("가위=1, 바위=2, 보=3, 입력:"))
    if Q==me:
        print("비겼습니다.")
    elif (Q==1) and (me==2):
        print("당신은 이겼습니다.")
        w=w+1
    elif (Q==1) and (me==3):
        print("당신은 졌습니다.")
    elif (Q==2) and (me==1):
        print("당신은 졌습니다.")
    elif (Q==2) and (me==3):
        print("당신은 이겼습니다.")
        w=w+1
    elif (Q==3) and (me==1):
        print("당신은 이겼습니다.")
        w=w+1
    elif (Q==3) and (me==2):
        print("당신은 졌습니다.")
    if (w==3):
        print("게임종료")
        break

```

2019011925 여명준

```

import random

i = 0
score = 0

while i < 5:
    N = (int(input("Enter (가위: 1, 바위: 2, 보: 3)")))
    ans = random.randint(1, 3)

    print("Answer is ", ans)

    if ((N == 1) and (ans == 3)):
        print("Win")
        score = score + 1

    elif ((N == 2) and (ans == 1)):

```

```
print("Win")
score = score + 1
```

```
elif ((N == 3) and (ans == 2)):
    print("Win")
    score = score + 1
```

```
elif N == ans:
    print("Draw")
```

```
else:
    print("Lose")
```

```
print("Score: ", score)
```

```
if score == 3:
    print("You Win")
    break
```

```
i = i + 1
print("Count =", i)
```

2016011594 김진섭

```
import random
```

```
cwin = 0
pwin = 0
```

```
while cwin<3 and pwin<3:
    x = random.randint(1, 3)
    y = str(input("가위, 바위, 보 중 하나를 입력하세요"))
    if (y=="가위" and x==1) or (y=="바위" and x==3) or (y=="보" and x==2):
        print("플레이어 승리")
        pwin+=1
    elif (y=="가위" and x==2) or (y=="바위" and x==3) or (y=="보" and x==1):
        print("컴퓨터 승리")
        cwin+=1
    else:
        print("비겼습니다")
```

```
print ("플레이어점수:",pwin)
print ("컴퓨터점수:",cwin)
```

2019011539 최민영

```
import random
```

```
win=0
lose=0
```

```
while True:
    x=random.randint(1,3)
    y=int(input("'가위=1, 바위=2, 보=3' 로 입력하시오. "))
    if (x==y):
        print("비겼습니다.")
    elif ((x==1 and y==3) or (x==2 and y==1) or (x==3 and y==2)):
```

```

        print("졌습니다.")
        lose=lose+1
        if (lose==3):
            print("YOU LOSE! TRY AGAIN.")
            break
    elif ((x==1 and y==2) or (x==2 and y==3) or (x==3 and y==1)):
        print("이겼습니다.")
        win=win+1
        if (win==3):
            print("YOU WIN! CONGRATULATION")
            break

```

2015011447허영준

```

import random
i,j,z=0,0,5
while z>0:
    y=random.randint(0,2)
    x=input(str("가위 바위 보 :"))
    if x == '가위':
        x=0
    elif x == '바위':
        x=1
    elif x == '보':
        x = 2
    else:
        print('입력오류')
        x=50
    if x==50:
        print('반칙패')
        j = j + 1
    elif (y == 0 and x == 2):
        print('패')
        j = j + 1
    elif (x==0 and y==2):
        print('승')
        i = i + 1
    elif x==y:
        print('무승부')
    elif x < y:
        print('패')
        j = j + 1
    elif x>y:
        print('승')
        i=i+1
    z=z-1
if i>j:
    print('승리', i,'승',j,'패')
elif i<j:

```

```

        print('패배', i, '승', j, '패')
else:
    print('무승부', i, '승', j, '패')

```

김경환

```
import random as r
```

```

b=0
cc=0
uscore=0
cscore=0
print('5판3승 가위바위보 게임입니다. (단 비길경우 재경기)')
while b<=4:
    cc = r.randint(1, 3)
    user=input('가위,바위,보 중 택 1: ')
    if user=='가위':
        if cc==2:
            print('컴퓨터:바위 vs 당신:가위 [패배]')
            b = b + 1
            cscore = cscore+1
            print('현재 스코어:', '당신', uscore, 'vs', '컴퓨터', cscore)
        elif cc==3:
            print('컴퓨터:보 vs 당신:가위 [승리]')
            b = b + 1
            uscore = uscore + 1
            print('현재 스코어:', '당신', uscore, 'vs', '컴퓨터', cscore)
        else:
            print('컴퓨터:가위 vs 당신:가위 [무승부]')

```

```

    elif user=='바위':
        if cc==1:
            print('컴퓨터:가위 vs 당신:바위 [승리]')
            b = b + 1
            uscore = uscore + 1
            print('현재 스코어:', '당신', uscore, 'vs', '컴퓨터', cscore)
        elif cc==3:
            print('컴퓨터:보 vs 당신:바위 [패배]')
            b = b + 1
            cscore = cscore + 1
            print('현재 스코어:', '당신', uscore, 'vs', '컴퓨터', cscore)
        else:
            print('컴퓨터:바위 vs 당신:바위 [무승부]')

```

```

    elif user=='보':
        if cc==1:
            print('컴퓨터:가위 vs 당신:보 [패배]')
            b = b + 1
            cscore = cscore + 1
            print('현재 스코어:', '당신', uscore, 'vs', '컴퓨터', cscore)
        elif cc==2:
            print('컴퓨터:바위 vs 당신:보 [승리]')
            b = b + 1
            uscore = uscore + 1
            print('현재 스코어:', '당신', uscore, 'vs', '컴퓨터', cscore)
        else:

```

```
print('컴퓨터:보 vs 당신:보 [무승부]')
```

```
else:
```

```
    print('다시입력하세요')
```

```
print('가위바위보 결과' )
```

```
print('컴퓨터:',cscore )
```

```
print('당신:',uscore)
```

```
if cscore>uscore:
```

```
    print('컴퓨터의 승리!')
```

```
else:
```

```
    print('당신의 승리!')
```

전대협

```
import random
```

```
W = 0
```

```
C = 0
```

```
while C < 5 :
```

```
    f = random.randint(1,3)
```

```
    A = str(input("가위, 바위, 보 중 하나를 입력하세요 : "))
```

```
    print("당신 : ",A)
```

```
    if f == 1 :
```

```
        print("컴퓨터 : 가위")
```

```
    elif f == 2 :
```

```
        print("컴퓨터 : 바위")
```

```
    elif f == 3 :
```

```
        print("컴퓨터 : 보")
```

```
    if A == '가위' and f == 3 :
```

```
        print("이겼습니다.")
```

```
        W = W + 1
```

```
    elif A == '바위' and f == 1 :
```

```
        print("이겼습니다.")
```

```
        W = W + 1
```

```
    elif A == '보' and f == 2 :
```

```
        print("이겼습니다.")
```

```
        W = W + 1
```

```
    print('')
```

```
    C = C + 1
```

```
    if W == 3 :
```

```
        print("컴퓨터와 가위바위보에서 3승을 했습니다.")
```

```
        break
```

남상민

```
import random as rd
```

```
winCount=0
```

```
loseCount=0
```

```
boolean = True
```

```
while boolean:
```

```
    print('가위 : 1, 바위 : 2, 보 : 3')
```

```
    hand = int(input('가위 바위 보! : '))
```

```
    foo=rd.randint(1,3)
```

```
    if(hand>3 | hand<1):
```

```

print('다시 입력해 주세요')
elif(hand == foo):
    print('비겼습니다. 다시 입력해 주세요')
# 지는 경우 가위 바위, 바위 보, 보 가위
elif(((hand == 1) & (foo==2)) | ((hand==2) & (foo ==3)) | ((hand == 3) & (foo ==1))):
    loseCount = loseCount+1
elif(((hand == 2) & (foo==1)) | ((hand==3) & (foo ==2)) | ((hand == 1) & (foo ==3))):
    winCount = winCount+1

print('현재 ' + str(winCount) + '승 ' + str(loseCount) + '패')
if(winCount >=3):
    print('이겼습니다!!')
    boolean = False
elif(loseCount >=3):
    print('졌습니다 ㅋㅋ')
    boolean = False
print('\n')

```

2015012063 장선도

```

import random

count = 0
win = 0
user_num = 0

while(count < 5):

    rand = random.randint(0, 2)
    # 0:가위, 1: 바위, 2: 보
    rand_str = ''

    if (rand == 0):
        rand_str = '가위'
    elif (rand == 1):
        rand_str = '바위'
    elif (rand == 2):
        rand_str = '보'

    while 1:
        user = (input('가위, 바위, 보 중에서 하나를 입력하시오. '))
        if (user == '가위'):
            user_num = 0
            break
        elif (user == '바위'):
            user_num = 1
            break
        elif (user == '보'):
            user_num = 2
            break
        else:
            print('다시 입력해 주세요.')

#비교한다.
if (rand == user_num):
    print('컴퓨터 :', rand_str, ', ', '당신 : ', user)
    print('무승부!')
elif (rand == 0 and user_num == 1) or (rand == 1 and user_num == 2) or

```

```

(rand == 2 and user_num == 0):
    print('컴퓨터:', rand_str, ', ', '당신 : ', user)
    print('승리')
    win += 1
else:
    print('컴퓨터 :', rand_str, ', ', '당신 : ', user)
    print('패배')
    count += 1
    if (win >= 3):
        break

if (win >= 3):
    print('최종 승리하셨습니다.')
else:
    print('최종 패배하셨습니다.')

```

지정진

```

import random

usr_point = 0
comp_point = 0
while(usr_point<3) and (comp_point<3):
    #컴퓨터 1:가위 2:바위 3:보
    computer = random.randint(1, 3)
    user = str(input("무엇을 내시겠습니까?"))
    if(computer == 1) and (user == "바위"):
        usr_point = usr_point+1
    elif(computer == 1) and (user == "보"):
        comp_point = comp_point+1
    elif(computer == 1) and (user == "가위"):
        print("무승부")
    elif(computer == 2) and (user == "가위"):
        comp_point = comp_point+1
    elif(computer == 2) and (user == "바위"):
        print("무승부")
    elif(computer == 2) and (user == "보"):
        usr_point = usr_point +1
    elif(computer == 3) and (user == "가위"):
        usr_point = usr_point +1
    elif(computer == 3) and (user == "바위"):
        comp_point = comp_point +1
    else:
        print("무승부")

    for i in range(10):
        print("")

    print("컴퓨터 : ", computer, "유저 : ", user)
    print("컴퓨터의 점수 : ", comp_point)
    print("유저의 점수 : ", usr_point)

    if(usr_point == 3):
        print("당신이 이겼습니다!")
    else:
        print("컴퓨터가 이겼습니다!")

```

2016011602 안호균

```

print("This is rock scissors paper game.")
import random

gcount=0
wincount=0

while gcount <= 4:

    a = random.randint(1,3)
    game1 = str(input("player, input rock, scissors or paper"))
    print(game1)

    if (a == 1 and game1 == "scissors") or (a == 2 and game1 == "rock") or
(a == 3 and game1 == "rock"):
        print("computer wins")
        gcount += 1
    elif (a == 2 and game1 == "rock") or (a ==3 and game1 == "scissors")
or (a == 1 and game1 == ("paper")):
        print("player wins")
        gcount += 1
        wincount +=1
    elif (a == 2 and game1 == "scissors") or (a ==3 and game1 == "paper")
or (a == 1 and game1 == "rock"):
        print("It's draw.")
        gcount += 1
    else:
        print("Input correct word.")
print("End the gmae, you five games",wincount,"wins")

```

2014011990 현동훈

```

while True:
    import random
    gbb_com = random.choice(["가위" , "바위" , "보"])
    gbb = str(input("가위바위보 중 하나를 입력하세요"))
    wincount=0
    wincount_con=0
    if(gbb_com == gbb):
        print("비겼습니다.")
    elif((gbb_com == "가위" and gbb=="바위") or (gbb_com=="바위" and
gbb=="보") or (gbb_com=="보" and gbb=="가위")):
        print("이겼습니다.")
        wincount+=1
    elif((gbb_com=="가위" and gbb=="보") or (gbb_com=="바위" and
gbb=="가위") or (gbb_com=="보" and gbb=="바위")):
        print("졌습니다.")
    elif((wincount==3) or (wincount_com==3)):
        break
    else:
        print("정확히 입력하세요")

```

오늘 할 것
배열 또는 리스트의 이해
함수의 이해

리스트, 배열.

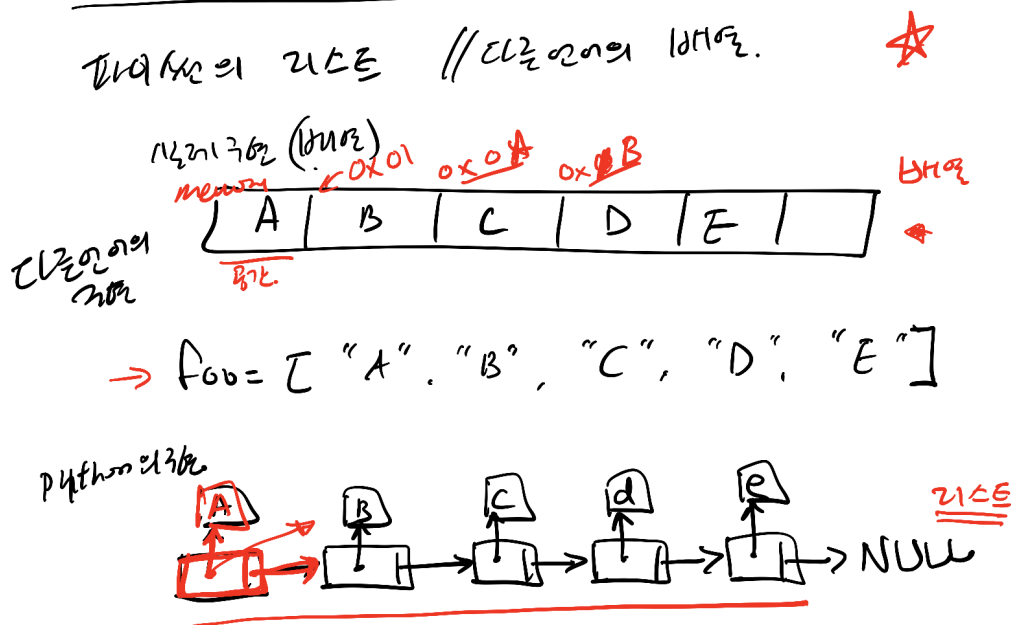
```
com = random.choice(['가위', '바위', '보'])
```

n 개의 길이의 리스트

foo = [인덱스 0, 인덱스 1, 인덱스 2, ... , 인덱스 n-1]

foo[2] → 인덱스 2

```
0 1 2 3 4
foo = ["a", "b", "c", "d", "e"]
for x in range(0, 3): # 0, 1, 2
    print(foo[x])
```



install

Day 01 의 코드에서 **ant.py** 문제 해결

http://open.gnu.ac.kr/mediawiki/index.php?title=Ai_2020-02

터미널에서 다음 명령어 먼저 실행

pip3 install freegames

def 함수이름 (인자1, 인자2):

----- 내용

return 돌려줄값 # 마지막 줄은 옵션

def larger(foo, bar):

if (bar > foo) :

return bar

else

return foo

.호출을 위해서

result = larger(3, 4)

print(result)

39p 참고하여, 가위바위 보 게임을 함수를 활용하여 짧게 수정

이름, 코드
이성진, import random # s = 1, r = 2, p = 3 def check_tie(user, comp): if (user == comp): return True else :

return False

```
def check_win(user, comp):  
    if (user == 1 and comp == 3) \  
        or (user == 2 and comp == 1) \  
        or (user == 3 and comp == 2):  
        return True  
    else:  
        return False
```

```
def game(user, comp):  
    wc = 0  
    if (check_tie(user, comp) == True):  
        print("Tie game")  
    elif (check_win(user, comp) == True):  
        wc = 1  
        print("You won")  
    else:  
        print("You loose")  
    return wc
```

```
def play(gamecount, wincount):  
    gc = 0  
    wc = 0  
    while (gc < gamecount) and (wc < wincount):  
        user = int(input("Scissor = 1, Rock = 2, Paper = 3: "))  
        comp = random.randint(1, 3)  
        wc += game(user, comp)  
        gc += 1  
        print("game count: ", gc, "win count: ", wc)
```

```
gamecount = 5  
wincount = 3  
play(gamecount, wincount)
```

지정진

```
import random
```

```
def versus(user, computer):  
    if (user-computer==-2):  
        return 1  
    elif (user-computer==-1):  
        return 2  
    elif (user-computer==0):  
        print("무승부")  
    elif (user-computer==1):  
        return 1  
    elif (user-computer==2):  
        return 2
```

```
def changeToStr(x):  
    if (x == 1):  
        return "가위"  
    elif (x == 2):  
        return "바위"  
    elif (x == 3):  
        return "보"
```

```
def changeToNum(x):  
    if (x == "가위"):  
        return 1  
    elif (x == "바위"):  
        return 2  
    elif (x == "보"):  
        return 3
```

```
user_score = 0  
computer_score = 0
```

```
while (user_score != 3) | (computer_score != 3):  
    user = changeToNum(input("무엇을 내시겠습니까?"))  
    computer = random.randint(1, 3)
```

```
    print("컴퓨터가 낸 것 : ", changeToStr(computer), "유저가 낸 것 : ",  
changeToStr(user))
```

```
    if (versus(user, computer)==1):  
        print("유저가 이겼습니다")  
        user_score = user_score+1  
    elif (versus(user, computer)==2):  
        computer_score = computer_score+1  
        print("컴퓨터가 이겼습니다")
```

```
if (user_score == 3):  
    print("유저가 최종 승자입니다")  
else:  
    print("컴퓨터가 최종 승자입니다")
```

개미

```
from random import *  
from turtle import *  
from freegames import vector
```

```

ant = vector(0, 0)
aim = vector(2, 0)

def wrap(value):
    if (value >= 200) or (value <= -200):
        value = -value
    return value # TODO

def draw():
    "Move ant and draw screen."
    ant.move(aim)
    ant.x = wrap(ant.x)
    ant.y = wrap(ant.y)

    aim.move(random() - 0.5)
    aim.rotate(random() * 10 - 5)

    #clear()
    goto(ant.x, ant.y)

    colormode(255)

    r=0; g=0; b=0;
    if (ant.x < 0):
        b = -int(ant.x)
    else:
        b = int(ant.x)
    if (ant.y < 0):
        g = -int(ant.y)
    else:
        g = int(ant.y)
    if ((ant.x+ant.y) < 0):
        r = -int((ant.x+ant.y)%255)
    else:
        r = int(ant.x+ant.y)%255

    color(r, g, b)
    dot(4)

    if running:
        ontimer(draw, 100)

setup(420, 420, 370, 0)
hideturtle()
tracer(False)
up()
running = True
draw()
done()

```

2018011389 백은선

`import random`

```

def game(my, win):
    com = random.randint(1, 3)
    if (my == 1 and com == 1) \
        or (my == 2 and com == 2) \
        or (my == 3 and com == 3):
        print("비겼습니다.")
    elif (my == 1 and com == 2) \
        or (my == 2 and com == 3) \
        or (my == 3 and com == 1):
        print("졌습니다.")
    else:
        print("이겼습니다.")
        win = win + 1
    return win

win = 0
count = 1

while count < 6:
    my = int(input("가위 (1) 바위 (2) 보 (3) !"))
    win = game(my, win)
    count = count + 1

if win >= 3:
    print("5판", win, "승으로 이겼습니다.")
else:
    print("5판", win, "승으로 졌습니다.")

```

남상민

```
import random as rd
```

```

def result(hand, com):
    if (hand > 3 | hand < 1):
        return 1
    elif (hand == com):
        return 2
    # 지는 경우 가위 바위, 바위 보, 보 가위
    elif (((hand == 1) & (com == 2)) | ((hand == 2) & (com == 3)) | ((hand == 3) &
(com == 1))):
        return 3
    elif (((hand == 2) & (com == 1)) | ((hand == 3) & (com == 2)) | ((hand == 1) &
(com == 3))):
        return 4

```

```

winCount=0
loseCount=0
boolean = True

```

```

while boolean:
    print('가위 : 1, 바위 : 2, 보 : 3')

```

```

r= result(int(input('가위 바위 보! : ')),rd.randint(1,3))
if(r ==1):
    print('다시 입력해 주세요')
elif(r==2):
    print('비겼습니다.')
elif(r==3):
    loseCount+=1
else:
    winCount+=1

print('현재 ' + str(winCount) + '승 ' + str(loseCount) + '패')
if(winCount >=3):
    print('이겼습니다!!!')
    boolean = False
elif(loseCount >=3):
    print('졌습니다 ㅠㅠ')
    boolean = False
print('\n')

```

개미

"""Ant, 개미 행동 모사하기

할 일

1. 개미가 모서리를 벗어나면 반대편으로 등장하게 만들기
2. 개미가 지나간 자리에 흔적을 남기기
3. 개미가 지나간 흔적의 색을 영역에 따라 바꾸기

Hint: colormode(255); color(0, 100, 200)

"""

```

from random import *
from turtle import *
from freegames import vector

```

```

ant = vector(0, 0)
aim = vector(2, 0)

```

```

def wrap(value):
    "Wrap value around -200 and 200."

```

return value # TODO

```
def draw():  
    "Move ant and draw screen."  
    ant.move(aim)  
    ant.x = wrap(ant.x)  
    ant.y = wrap(ant.y)  
    print(str(ant.x)+' , '+str(ant.y))  
  
    if((ant.x>200)|(ant.x<-200)):  
        ant.x*=-1  
    if((ant.y>200)|(ant.y<-200)):  
        ant.y*=-1  
    aim.move(random()-0.5)  
    aim.rotate(random() * 10 - 5)  
  
    #clear()  
    #colormode(255); color(255,255,255)  
    r=int((ant.x+200)/2)+4  
    g=int((ant.y+200)/2)+4  
    b=int((ant.x+ant.y+400)/4)+8  
    colormode(255); color(r,g,b)  
    goto(ant.x, ant.y)  
    dot(7)  
  
    if running:  
        ontimer(draw, 100)  
  
setup(420, 420, 370, 0)  
hideturtle()  
tracer(False)  
up()  
running = True  
draw()  
done()
```

```

import random as r

def worl(user, com, i, win):
    if user == com:
        print(f'무승부 {i}전 {win}승')
    elif ((user == 1)&(com == 3))|((user == 2)&(com == 1))|((user == 3)&(com == 2)):
        win += 1
        print(f'이겼습니다! {i}전 {win}승')
    else:
        print(f'졌습니다 {i}전 {win}승')
    return win

def userNum(user):
    if user == '가위':
        user = 1
    elif user == '바위':
        user = 2
    elif user == '보':
        user = 3
    else:
        print("정확한 단어를 입력해주세요.")
        user = 0
    return user

i = 1
win = 0
while True:
    user = input("가위, 바위, 보 중에 입력 해주세요")
    com = r.randint(1,3)    #1 = 가위, 2 = 바위, 3 = 보

    user = userNum(user)
    win = worl(user, com, i, win)

    print()
    if win >= 3:
        break
    if (i == 5):
        print(f'패배 {win}승')
        i = 0
        win = 0
    i += 1

```

```
print(f'승리! {i}전 {win}승')
```

2019011925 여명준

import random

```
555def rock(score):
    if ((N == 1) and (ans == 3)) or ((N == 2) and (ans == 1)) or ((N == 3)
and (ans == 2)):
        print("Win")
        score = score + 1

    elif N == ans:
        print("Draw")

    else:
        print("Lose")

    return score

#Program Start

i = 0
score = 0

while i < 5:
    N = (int(input("Enter (가위: 1, 바위: 2, 보: 3)")))
    ans = random.randint(1, 3)

    print("Answer is ", ans)

    score = rock(score)

    print("Score: ", score)

    if score == 3:
        print("You Win")
        break

    i = i + 1
    print("Count =", i)
```

김광재

```
def game(u,g):
    victor = 0
    if (u != g):
        if (u=="rock" and g=="sissors") or (u=="sissors" and g=="paper") or
(u=="paper" and g=="rock"):
            print("승리하셨습니다")
            victor = victor +1
            return victor
        else:
            print("패배하셨습니다.")
```

```

else:
    print("비겼습니다")
    return

#.호출을 위해서
win = 0
count = 0
while True:
    import random
    gbb = random.choice(["rock", "paper", "sissors"])
    user = str(input("rock paper sissors 를 입력해주세요 "))
    result = game(user, gbb)
    count = count + 1
    if (count == 4):
        print("게임이 끝났습니다")
        break
    win=result
    if (win>=3):
        print("승리하셨습니다")
    else:
        print("패배하셨습니다")

```

```

2015011336 김경훈
def tieGame(user, com) :
    if user == com :
        return True
    else :
        return False

def noTieGame(user, com) :
    if (user == "가위" and com == "바위") or (user == "바위" and com == "보")
or (user == "보" and com == "가위") :
        return True
    else :
        return False

def game(user, com) :
    if tieGame(user, com) == True :
        print("비겼습니다.")
    else :
        if noTieGame(user, com) == True :
            print("졌습니다.")
            global comScore
            comScore += 1
        else :
            print("이겼습니다.")
            global userScore
            userScore += 1

import random
userScore = 0
comScore = 0
while (userScore < 3 and comScore < 3) :
    user = input("무엇을 내시겠습니까? (가위/바위/보) : ")
    if user == "가위" or user == "바위" or user == "보" :
        game(user, random.choice(["가위", "바위", "보"]))
        print("User Score : %s, Com Score : %s" %(userScore, comScore))
    else :

```

```
print("잘못 입력하셨습니다.")
```

```
if userScore == 3 :  
    print("User Win")  
elif comScore == 3 :  
    print("Com Win")
```

```
print("User Win")  
elif comScore == 3 :  
    print("Com Win")
```

2016011493 김성진

```
import random  
def game ( my , win ) :  
    com = random.randint(1,3)  
    if (my == 1 and com == 1) or (my == 2 and com == 2) or (my == 3 and com  
== 3):  
        print("draw")  
        return win  
    elif (my == 2 and com == 1) or (my == 3 and com == 2) or (my == 1 and  
com == 3):  
        print("victory")  
        win=win+1  
        return win  
    else :  
        print ("lose")  
        return win
```

```
win = 0  
count = 1  
while count < 6 :  
    my = int(input("가위 바위 보 중에 하나를 내시오 (1:가위, 2:바위, 3:보)"))  
    win = game(my , win)  
    count = count + 1
```

```
if win >= 3 :  
    print(win,"승으로 이겼습니다.")  
else:  
    print(win,"승으로 아쉽게 졌습니다.")
```

2015011954 고성웅

```
import random  
def win(player,com):  
    if (player=='가위' and com=='보') or (player=='바위' and com=='가위') or  
(player=='보' and com=='바위'):  
        return True  
    else:  
        return False  
def noo(player,com):  
    if (player==com):  
        return True  
    else:  
        return False  
def game(player,com):  
    winc=0  
    print("상대는",com,"을 냈습니다.")  
    if (win(player,com)==True):  
        print("이겼습니다!")
```

```

        winc=1
    elif (noo(player,com)==True):
        print("비겼습니다.")
    else:
        print("졌습니다...")
    return winc
def play(gamecount,wincount):
    gamec=0
    winc=0
    while (gamec<gamecount) and (winc<wincount):
        player=input("가위?바위?보?")
        com=random.choice(["가위", "바위", "보"])
        winc += game(player,com)
        gamec+=1
    print("현재게임수:",gamec,"이긴횟수",winc)
play(5,3)

```

게임

```

from random import *
from turtle import *
from freegames import vector

ant=vector(0,0)
aim=vector(2,0)

def wrap(value):
    return value
def draw():
    ant.move(aim)
    ant.x=wrap(ant.x)
    ant.y=wrap(ant.y)

    aim.move(random()-0.5)
    aim.rotate(random()*10-5)

    goto(ant.x, ant.y)
    dot(6)
    colormode(255)
    if (ant.x>-200 and ant.x<0) and (ant.y<-200 and ant.y>0):
        color(0, 100, 200)
    elif (ant.x>-200 and ant.x<0) and (ant.y<200 and ant.y>0):
        color(100,243,234)
    elif (ant.x>0 and ant.x<200) and (ant.y<0 and ant.y>-200):
        color(222,153,111)
    elif (ant.x>0 and ant.x<200) and (ant.y<200 and ant.y>0):
        color(234,54,64)

    if running:
        ontimer(draw,100)

    if ant.x>200:
        ant.x=-200
    elif ant.x<-200:
        ant.x=200
    elif ant.y>200:
        ant.y=-200
    elif ant.y<-200:
        ant.y=200

```

```
setup(400,400,370,0)
hideturtle()
tracer(False)
up()
running=True
draw()
done()
```

2016011602 안호균

```
import random
#Rock = 1, Scissors = 2, Paper = 3
def Draw(computer, player):
    if(computer == player):
        return True
    else:
        return False

def Win(computer, player):
    if(computer == 1 and player == 3) or (computer == 2 and player == 1) or
(computer == 3 and player == 2):
        return True
    else:
        return False

def game(computer, player):
    wc = 0
    if(Draw(computer, player) == True):
        print("Draw!")
    elif(Win(computer, player) == True):
        wc = 1
        print("player wins")
    else:
        print("Computer wins")
    return wc

def play(wincount,playcount):
    wc = 0
    pc = 0
    while(wc < wincount) and (pc < playcount):
        player = int(input("Rock, Scissors, Paper"))
        computer=random.randint(1,3)
        wc += game(computer, player)
        pc +=1
        print("play count", pc, "win count:" ,wc)

play(3,5)
```

2018010680 진유남

```
import random

game = 0
win = 0
```

```

def play(user, wincount):
    com=random.choice(["가위","바위","보"])

    if (user==com):
        print("비겼습니다.")

    elif (user=="가위" and com=="바위")or(user=="바위" and com=="보") or (user=="보" and
com=="가위"):
        print("졌습니다.")

    else:
        print("이겼습니다.")
        wincount = wincount+1

    return win

while True:
    x = input("가위, 바위, 보를 입력하세요: ")

    game = game+1
    win = play(x, win)

    if game==5 and win>=3:
        print("승리!")
        break

    elif game==5 and win<3:
        print("패배!")
        break

```

김채진

```

import random

def rock(score):
    if ((N == 1) and (ans == 3)) or ((N == 2) and (ans == 1)) or ((N == 3)
and (ans == 2)):
        print("Win")
        score = score + 1

    elif N == ans:
        print("Draw")

    else:
        print("Lose")

    return score

#Program Start

i = 0
score = 0

while i < 10:

```

```
N = int(input("가위: 1, 바위: 2, 보: 3 "))
ans = random.randint(1, 3)
```

```
print("ans is", ans)
score = rock(score)
print("score is:", score)
```

```
if score == 3:
    print("You win")
    break
```

```
i = i + 1
print("count = ", i)
```

2015012009 배상훈

```
import random #랜덤 함수 import
```

```
def win_lose(comp, user):
    if ((comp == 0 and user == 2) or (comp == 1 and user == 0) or (comp == 2
and user == 1)): # 유저가 이길 경우
        print("you win")
        return True
```

```
    elif ((comp == 0 and user == 1) or (comp == 1 and user == 2) or (comp == 2
and user == 0)): # 유저가 질 경우
        print("you lose")
        return False
    else :
        print("draw game")
```

```
def draw(comp,user):
    if ((comp == 0 and user == 0) or (comp == 1 and user == 1) or (comp == 2
and user == 2)):
        return True
    else :
        return False
```

```
def play(i):
    win = 0
    lose = 0
    while True:
        comp = random.randint(0, 2) # 0 : 묵 , 1 : 찌 , 2: ✎
        user = int(input("0 : 묵 , 1 : 찌 , 2: ✎ // your choice is :"))
        if(win_lose(comp, user) == True):
            win+=1
            if(win==i):
                print("SUCCESS")
                break
        else:
            if(draw(comp,user) == True):
                continue
            else:
                lose+=1
                if(lose==i):
                    print("FAILED")
                    break
```

```
play(3)
```

2019011539 최민영

```
import random
```

```
def rsp(game, win):  
    x = random.randint(1, 3)  
    y = int(input("'가위=1, 바위=2, 보=3' 로 입력하시오. "))  
    if ((x==1 and y==1) or (x==2 and y==2) or (x==3 and y==3)):  
        print("비겼습니다.")  
        return win  
    elif ((x==1 and y==3) or (x==2 and y==1) or (x==3 and y==2)):  
        print("졌습니다.")  
        return win  
    elif ((x==1 and y==2) or (x==2 and y==3) or (x==3 and y==1)):  
        print("이겼습니다.")  
        win=win+1  
        return win
```

```
win=0
```

```
game=1
```

```
while game<6:  
    win=rsp(game, win)  
    game=game+1  
if (win>=3):  
    print("YOU WIN! CONGRATULATION")  
else:  
    print("YOU LOSE! TRY AGAIN")
```

Day 02 크롤러의 이해와 만들어 보기

오늘의 주제

1. 지난 시간 복습
 - a. 자기 코드를 팀원들에게 설명
 - b. 팀원들과 함께 새로 깨달은 내용을 정리
 - c. 발표
2. 빅데이터: 4 V's in big data
 - a. Volume
 - b. Variety
 - c. Velocity
 - d. Value

주제

1. 빅데이터를 활용한 기업의 사례 **2팀** : 진진남선 , 고인물
2. 빅데이터 플랫폼 (하둡, 스파크, 레디스 등) **2팀**: **4K**(스파크), 퍼펙트론(하둡)
3. 자유주제 **3팀**: 젊음, **4A2B**, 러닝머신 **ooo**
3. **What would you want to gather? What information do you want to make out of it?0**
4. **Data crawling for Big Data and AI**

10:15분까지 정리 **10**분 휴식 후 설명

팀 별 새로 깨달은 내용 정리

퍼셉트론

지정진님의 코드를 보고 진리표를 이용하여 조건문을 간단하게 작성할 수 있다는 것을 알게 되었다

지정진님의 코드를 보고,

여명준님의 가위바위보코드에서 이기는경우를 **if**문에 묶어줌으로써 좀 더 간략하게 구성되는 것을 알 수 있었다.

조건문 대하여 어느정도 감을 잡을 수 있게 되었다.
교수님이 만드신코드와 다르게 함수를 구현하여 똑같은 프로그램을 만들수 있다는 것을 알았다.

여명준님의 코드를 보고, **if**문 하나에 조건식 여러개를 한번에 묶어 간략하게 작성할 수 있다는 것을 알았다.

여명준님의 코드를 보고, 메인문의 시작 부분에 주석을 달아서 구분을 쉽게 하는것이 좋다고 생각하였다.

러닝머신

박성호님의 코드를 보고 중괄호를 사용하여 매개변수를 이용할 수 있다는 것을 알게 되었다. 그리고 중괄호로 매개변수 출력시 문자열 앞에 **f**를 붙여줘야한다

print(f"문자열 {value}")

남상민님의 코드를 보고 **boolean**을 이용하여 반복문을 이용할 수 있다는 것을 알게 되었다

boolean = True

while boolean:

김채진님의 코드에서 **while**문 탈출조건으로 **break**를 이용하지 않고 비교문을 바로이용하여 코드를 줄일 수 있다.

지정진님의 개미 코드의 **wrap**함수에서 **value** 값에 **-**를 붙여서 쉽고 간결하게 표현 가능 하다는 것을 알게 되었다.

and 대신 (**&**), **or** 대신 (**|**)파이프 가 가능하다는 것을 알게 되었다.

점심

https://docs.google.com/presentation/d/1238Eqki3QTvRf3MNMYY7M-Ma2KW02y_FSud6KSRwXEVA/edit#slide=id.ga17124f88e_0_190

while True와 **if**로 5판 3선에서 이길 때 까지 게임을 반복 할 수 있는 것을 알게 되었다.

함수의 **return** 값이 **True** 나 **False**일 경우 **if**문에서 함수 이름으로만 사용이 가능하다

ex) "if 함수명==True" 대신에 **"if 함수명:"**

"shift"+"\"="|" => **or**을 대신해서 사용할 수 있다.

함수 안에서 사용하는 변수와 밖에서 사용하는 변수는 이름이 같아도 서로 다른 값을 가진다. 이때문에 **return**을 사용

4A2B

큰 규모의 함수 코드를 짤 때 먼저 작은 규모의 함수코드를 짜고 그 작은 규모의 함수들을 큰 규모의 함수코드에 포함 해 큰 규모의 함수를 완성하는 것이 효율이 좋다. - 고성웅 가위바위보

while문을 사용해서 조건을 설정하여 긴 코드를 간단하게 줄일 수 있다. - 안호균 가위바위보

not이나 **or** 같은 코드를 **!**와 **|** 같은 간단한 문자로 표현할 수 있다. - 지정진 가위바위보

코드 옆에 주석을 적어 처음 코드를 보는 사람들도 코드를 짤 목적과 어떻게 코드가 작동하는지 알 수 있다. - 배상훈 가위바위보

4K

while문을 사용하여 긴 코드를 짧게 줄일 수 있다.(가위바위보)
남상민님의 첫부분에 **as**로써 다른 간단한 이름으로 호출할 수 있는 법을 알았다.

진유남님 코드와 안호균님 코드를 보다가 입력변수를 문자로 받을지와 숫자로 받을지를 보았고 입력시 오타를 줄이려면 숫자로 입력하는 게 더 괜찮지않을까 생각해보게 되었다.

남상민님의 코드에서 **boolean**변수를 이용하여 **break**와 **continue**를 대신할 수 있다는 것을 알게 되었다

<<분산처리단계>>

1. 클러스터에서 데이터 읽기
2. 애널리틱스 운영 수행 및 결과값 클러스터 입력 등의 전 과정
동시 진행

<https://www.oss.kr/news/show/8dcc413f-940d-4edc-8ea8-ee28f3eff7ae>

<https://futurecreator.github.io/2018/08/14/apache-spark-basic/>

<https://blog.naver.com/shakey7/221608433990>

<https://blog.naver.com/ridesafe/222071596486>

<https://aws.amazon.com/ko/emr/features/spark/>

1. 빅데이터의 정의
2. 빅데이터 플랫폼의 필요성
3. 스파크의 정의
4. 스파크의 기능
5. 스파크의 구조
6. 하둡과 비교하였을 때 스파크의 장점
7. 하둡과 비교하였을 때 스파크의 단점
8. 빅데이터 플랫폼의 동향

진진남선

가위바위보 - 함수의 변수 명과 **while**문의 변수 명이 같아도 된다.

함수를 사용해서 숫자와 문자를 서로 맞대응 시킬 수 있다.

가위바위보- (!= 함수로 나 혹은 컴퓨터가 3승이 될때까지 반복하는 while 함수로 5판 3선승 함수를 만들 수 있다.)

개미- (x,y 범위에 따라 색 조건문을 규정하여 색칠 함수를 만들 수 있다.)

clear()를 이용하면 turtle의 흔적을 지울 수 있다.

개미 코딩중 def 함수 밑에 if문을 사용하여 연필이 내가 지정한 x,y좌표를 벗어나면 ant.x=ant.x*(-1)를 사용하여 다시 화면 안으로 들어오게 할 수 있음.

|는 or이다.w

고인물

while문에서 조건을 활용할 수 있다 - 지정진_가위바위보

From random import*코드가 의미하는 바는 랜덤이라는 외부코드를 들고온다는 의미고 *을 붙이면 그 외부코드의 모든 것을 들고 온다는 걸 의미한다. 이는 import random과 동일한 의미 => random코드 전부를 가져온다.

from freegames import vector 의미: 외부코드 freegames 에서 vector(= 함수 또는 클래스)를 불러온다. - 교수님_개미 기본코드 일부 클래스만 가져오면 충돌 가능성이 줄어든다.

class는 기능의 단위로서 방법(=함수)들과 변수들의 집합이다.

코딩을 편하게 할려면 다양한 외부코드에 대해 알고 있으면 도움이 된다.

return 값을 반환해서 그 값을 활용가능하다서
주석도 코드다. 주석 읽느라 컴퓨터가 느려 지진 않는다.

Poster

1. 하둡이란?

대용량 빅데이터 소프트웨어인 하둡은
아래의 소프트웨어 계층이라는 구조를

소프트웨어 프로젝트를 운영하는 다양한 형태의
하둡에 있다. 저층은 아키텍처 설계, 이차
프로그램 "고수"이 개발되었고, 이차
프로그램의 아키텍처 계층으로 구축되었다.

하둡은 (High-Availability Distributed Object-
oriented Platform)의 약자이다. 마스프는
코기인어, 더그 키임이 아키텍처 기고 논문
고기리 장본인을 보고 하둡이라고 한다.

2. 하둡의 구성요소

하둡은 크게 **주요 구성**

- sub Hadoop Common
 - 하둡의 다른 모듈을 사용해야 하는 공통 컴포넌트 모듈
- Hadoop HDFS
 - 분산저장체 계층의 모듈
 - 여러개의 서버에서 분산저장된 데이터를
- Hadoop MapReduce (명령어)
 - 분산되어 저장된 데이터를 분산처리 할 수 있게 하는
분산 처리 모듈
- sub Hadoop YARN
 - 분산처리를 위한 클러스터 자원관리 및 스케줄 담당
- sub Zookeeper
 - 분산 처리 시스템, 다중데이터 집합을 묶어 관리하는 시스템
(분산코디네이터)

3. 하둡의 장·단점

장점 - 오픈소스 라이선스에 따른 비용절감이 가능

- 시스템의 유연성 높고, 정해진 구조가 없다

- 입출력 장애에 강하다. 실패하면 전체시스템이
영향을 받음

- 오래전부터 데이터 처리에 적합하다

단점 - HDFS에 저장된 데이터를 변경할 수 없다

- 실시간 데이터 분석이 어려움. 실시간 처리에
부적합

- 데이터 많은 비정규 데이터 처리

- 운영이 어려움

4. 스파크란?

스파크는 아파치 소프트웨어 재단의 오픈소스

소프트웨어 프로젝트인데, 하둡의 시나리오

위해 개발되었다고 한다. 빠른 성능을 위해

인메모리 캐싱과 최적화된 실행을 사용하여

하둡보다 4~10배 빠른 데이터를 처리하기 위해

사용됩니다. 원래 맵리듀스와 맵리듀스 비클리의

AMPLab에서 개발된 소프트웨어지만, 코드가 복잡

이어서 최근에 개발되었으며, 그 이후 계속

유지보수 하고 있습니다.

빅데이터 플랫폼 (하둡, 스파크)

퍼펙트론

지정진, 임채현, 여영준, 주현정

5. 스파크의 구성요소

- 프로세스 관리 시스템

프로세스를 분산시킴으로써 성능을 향상시킨다.

마스터 노드에서 드라이버 프로세스를 생성하고,

드라이버 프로세스는 실행 프로그램을 컨트롤한다.

실행 프로세스는 여러개의 워커 노드로 작업을 분산시키며,

분산된 실행 프로그램들을 통해 스케줄링과 테스트 실행을

최적화할 수 있다.

- RDD

RDD는 자바 가상머신 객체들로 이루어져 있으며,

이 객체들을 프로세스 실행 속도를 향상시킨다.

RDD는 메모리상에 캐싱되고 저장되고 계산되므로,

하둡에 비해 10배 이상 더 빠르게 동작할 수 있다.

6. 스파크의 활용

RDD가 메모리상에 저장되기 때문에

원본데이터에 굉장히 유리하고, 실시간 데이터

처리에 용이하다. 그렇기 때문에

머신러닝을 데이터베이스와 빠른 처리속도를

요하는 실시간 데이터 처리 기술에 활용될 수 있다.

7. 하둡과 스파크의 비교

	하둡	스파크
역할	대량의 데이터를 여러 개의 정제된 분산시켜서 처리	대량의 데이터를 메모리로 처리해야 할 때 사용
성능	단일 데이터 처리 방식 (데이터 read → 데이터 write → 결과 저장) → 데이터 read → 실행결과를 클러스터에 입력	전체 데이터 set을 한번에 처리 (데이터 read → 데이터 write → 실행결과 입력) → 실행결과를 하둡에 비해 100배 정도 빠름
필요 메모리	많은 데이터를 안전하게 저장해야 할 경우	분산처리도 지속적으로 변화하는 스트리밍 데이터 세트에 적용됨 (ex: 실시간 마케팅 캠페인)

8. 스파크의 전망

대다수의 사람들은 하둡을 사용하기 어려워하고,

인공지능과 머신러닝에 적합하지 않다고 생각함

→ 강제로 하둡을 대체할 수 있는 존재로 부상하고 있음

스파크의 실시간 성능은 데이터 분석을 위한 광범위한 개편과 계층

→ 스케줄링 및 자원관리 계층들이 공통된 기능을 가진다

(비 몽고DB 계층 전체가 동일)

머신러닝 분야에서 스파크로 활용하는 사례가 늘고 있음

(비 쿼리베이스 계층 계층 관리 계층과)

개인정보 무단수집

- 정보제공 동의없는
사용자의 정보수집
ex) 틱톡, 캠스캐너 등
- 목적을 속인 오프라인 설문조사

보안 문제 : 개인정보 유출

데이터 처리업체 해킹 피해
(서비스 개선을 목적으로 이용정보 및
개인정보 수집)

사례) 연락처, 아이디, 이름, 연령,
지역 등의 유출로 인한
보이스피싱, 스팸 메일 등 수신
이미지, 음성, 위치 데이터
유출로 인한
사생활 침해

딥페이크 (Deep fake)

영상 편집물
↑
"합성"
인공지능기술

⇒ 가짜 뉴스
스캔들 유포

빅 데이터의 문제점

러닝머신 (남상민, 배상훈,
민문기, 최민영)

지배력 강화

- 거대 세력의 빅브라더 탄생
- 데이터의 권력화
ex) 중국의 '사회신용시스템'

데이터 조작

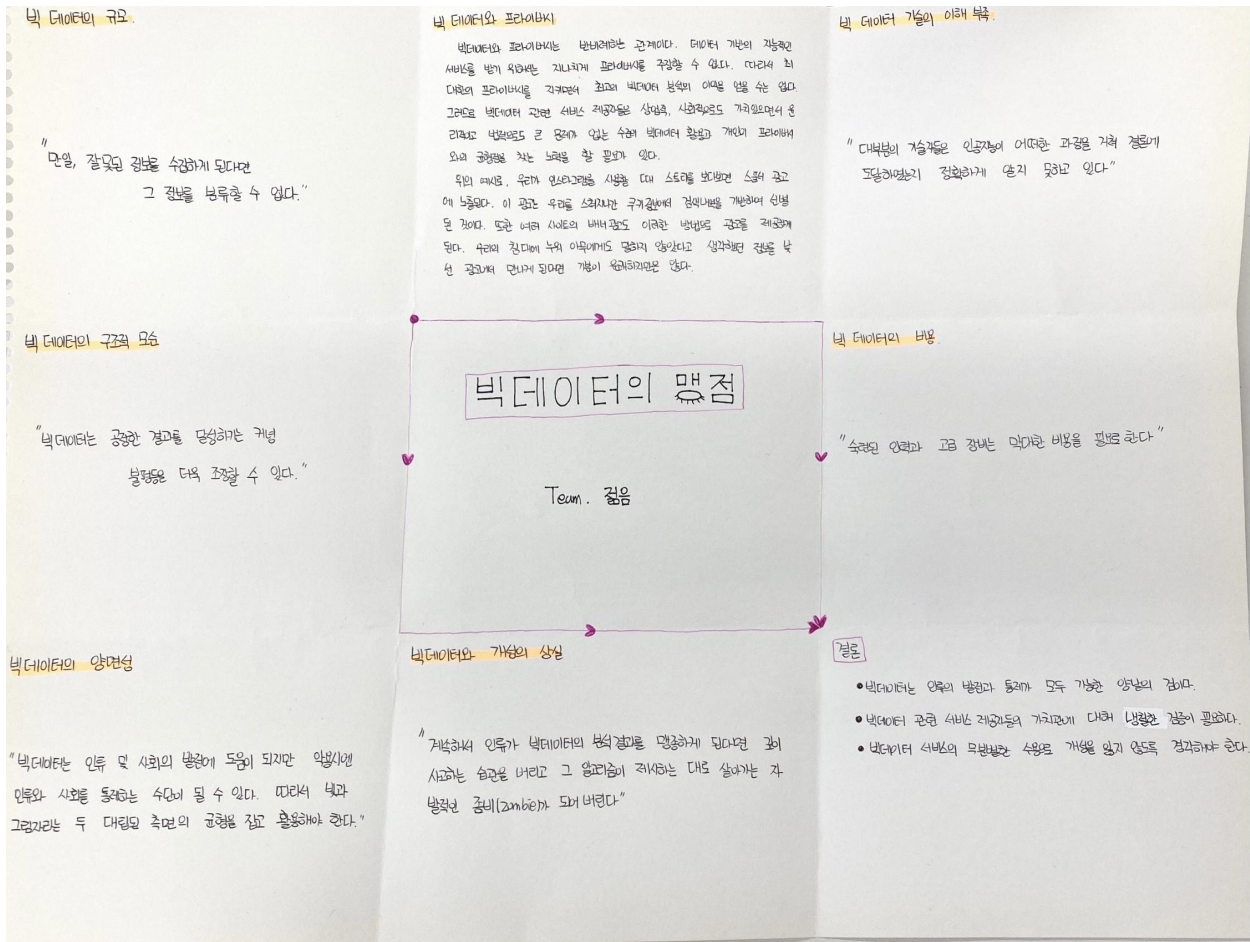
- 임의로 변조 및 조작할 경우
시스템의 오작동
→ 기업에 피해
시스템 오작동으로 인한 사고

과도한 정보 저장

- 정크 데이터의 양 증가
→ 삭제 되지 않는 데이터들
→ 유지비용 증가 및 신뢰성 저하

과도한 정보 처리

- 인간의 손을 거쳐야 하는
가공과정 중 과도한 시간 및
비용 소요
- 처리를 위한 발전된 기술의 필요
→ 비용 및 환경문제 초래



bh

1. 빅 데이터의 정의.

Big Data

일반 데이터가 코다는 못 - 이세상에 존재하는 모든 정보를 의미.

필요한 이유: 사용자의 수많은 경험, 정보는 어떤 것을
원하고 어떻게 이용하는 지에 대한 것을 조사하여
이익을 창출하기 위해
이용 활용하여

2. 빅데이터 플랫폼의 필요성

가까운 데이터를 예측할 수 있고, 기업은 이를 의사결정에
활용할 수 있다. 가까운 데이터를 쉽게 검색에 관한
수준이지만, 점차 의사결정구조에서 자동화 될 것이다.
또 더하면 이 과정이 민간기업에서도 이뤄질 수 있으며
스마트 기기에 결합되면 자동 의사결정 가능 할것임
보통 기업 창고에서 '부정방지 (Anti-Fraud)' 관련
작동해 의심스러운 가입자를 사전에 차단함
기업 및 조직이 데이터로부터 막대한 가치를 창출하려면
데이터에 대한 이해와 분석이 필요하며 이를 활용하는
새로운 데이터 플랫폼이 필요하다.

3. 아파치 스파크 (Apache spark)란?

2013년 UC버클리 연구실에서 처음 개발된 '인 메모리
분석 프레임 워크'로 하둡상에서 모든 연산을 클러스터로
분산하여, 반복작업과 알고리즘 구현에 용이해
데이터 사이언스와 머신러닝 (기계학습) 분야의
관심을 바꾸고 있다. 기존의 하둡을 통해 풀이하는
데이터들은 시간 소모가 크기 때문에 실시간으로
분석해야 하는 업무에 적합되었다. 언어는 Scala,
Java, Python, R 등 다양한 API를
사용 가능하다.

4 스파크의 기능

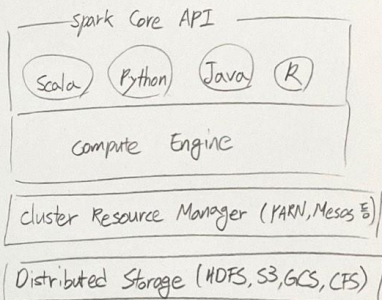
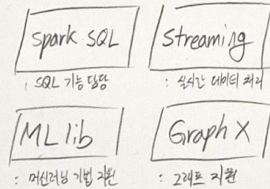
1. Scala, JAVA, Python 등 다양한 언어간
의 고성능의 API를 사용함
2. 맵리듀스와 유사한 일괄처리 기능
(Map-Reduce)
3. 실시간 데이터 처리가능
4. SQL과 유사한 정형 데이터 처리가능
5. 그래프 알고리즘
6. 머신러닝 알고리즘

4 K

빅데이터 플랫폼에 대한 조사

<<스파크>>

5. 스파크의 구조



6. 하둡과 비교하였을 때 스파크의 장점

- 1) 빠른 처리 속도
 - 메모리 효율은 높여 하둡의 맵리듀스보다 10~100배
빠르게 실행
 - 필요한 데이터를 디스크에서 매번 가져오는 대신,
데이터를 메모리에 캐싱해 저장하는 인-메모리 실행
모델로 비약적인 성능 향상
- 2) 개발과 친화적인 API
 - 사용자가 클러스터를 다룬다는 사실을 인지한
필요가 없도록 설계된 런타임 기반의 API
 - 처리한 데이터가 저장된 위치를 기억할 필요 X
 - 파이썬, R과 같은 데이터 분석용으로 인기있는
언어와 기업 친화적인 자바, 스칼라에 대한 지원 제공

7. 하둡과 비교하였을 때 스파크의 단점

- 클러스터링으로 구성된 메모리만으로 모두
운전할 수 없는 수 테라바이트 이상의
데이터 처리에는 적합하지 않은 편
- 즉, 처리할 수 있는 데이터의 양에 한계 존재

넷플릭스 N

3) "3천만 개의 서로 다른 버전의 넷플릭스"

↳ 많은 사람, 언어, 문화(국가, 언어, 문화(국문번호)), 어떤 기기, 형질, 검색한다. 스킴 패턴, 크레딧 정보...

캐드(모바일)

↳ 사람들은 고품질 영화를 보게 한 다음 그 영화에 포함된 영화를 테그 만들도록 했다.
고객이 한 번 본 영화에 붙은 테그와 비슷한 영화를 개인 영화를 추천할 수 있게 되었다.

유저

↳ 영국에서 발행된 "가드 오브 더스"의 미국 버전 라이선스를 얻기 위해 HBO, AMC 등 큰 회사를 제각각보다 더 비싼 가격을 물었다.
[근거] '가드 오브 더스' 감독인 데이비드 핀치의 포디움 앉아 "소셜 네트워크"는 수 많은 사람들이 컴퓨터를 통해 생방송 @영국 권각기 인기가 그걸 본 사람들은 데이비드 핀치가 감독하고 케빈 스미스가 나온 영화를 봄.

기존의 서비스

- 기존: 포자를 대여 서비스 사용
- 한 번 빌려온 대여업체 더 복잡한 인터페이스 기반 분석할 수 있는 NOSQL (데이터베이스), 카산드라 데이터베이스
- 데이터 엔지니어: 웹, 모바일 피그 같은 빅데이터 기술과 테라데이터나 마인스프링클 데이터 같은 전통적인 빅데이터 인텔리전스 도구의 장점을 융합하여 있음.

<페이스북>

이메일 인식 (덱페이스)

- 기존: 눈과 눈, 눈과 코 사이의 거리 파악하는 기술
- 덱페이스: 복귀적인 데이터를 이해하는 인공지능 프로그램
- 물체의 움직임과 방향을 파악해 만든 인공지능망을 활용
- 해당 영국의 이미지를 유추하여 만든 가상 영구로
회전시켜 네로 데이터 결과 연마됨

자유로운 정보 파악

- 어떠한 개인정보 수집 없이 '중대한' 이용 경험 만으로 사용자의 기본정보를 추적해볼 수 있다.
- 실제로 평균 80.1%, 확률 55.12%, 연령은 30.7. 정확하게 얻었음. 이는 즉각 반응을 분석이 있어 특정정보를 쉽게 발견하는 수 있는 유용하다.

<웨이모> : 자율 연구 중인 자동차 기업

기술 /

1. GPS 사용: 움직이지 않아서 도로 정보를 확보하여 충돌 방지
2. 레이더, 카메라, 라이다, 라이다 사용: 도로 정보를 확보하여 충돌 방지
3. 15만 마일에 해당하는 장거리 주행: 7만 마일에 해당하는 레이더 사용, 64개의 레이더가 주시 → 3D로 인식

테스트 /

50마일 코리올, 카탈라, 라살로 개조하여 속한인 도라이나로 인제니어가 설계하여 캘리포니아에서 1만 마일 이상 주행, 17마일 정사리 공로이 심한 것으로 유출한 실험관사로 공로하 거리로 아무 문제 없다 운영

<테슬라> : 전기차와 에너지 저장 기업

기술 /

1. 현재는 자율주행 2단계 (배타적주행)에 속한다.
2. 테슬라는 자율주행 5단계인 완전자율 자동차 구현을 위하여 ADAS (고급 운전자 보조 시스템) 구축함.

테스트 /

- 1도타일링 테슬라 모델: 16마일
- 자율주행 주행: 200마일
- 주행거리: 500마일 이상
- 1도타 차량 호출: 100마일
- 자율 비상 브레이크: 전 테슬라 차량 중 하나 속성-수동으로 거동

진 진 남 선

빅 데이터 활용

기업 사례

<카카오>

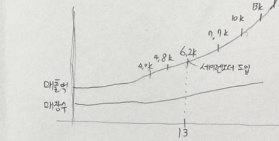
선물하기

- 선물 수신자의 선물 피드백은 바탕으로 선물이 가장 대량
선물 강도감은 제공하여 유저 참여함 수 시간
선물 트렌드를 제안한다

스타벅스 "언제 어디서나 개인화된 서비스, 간편결제, 회원권"

세이레코더
· 전세계에서 한국이 처음 시도한 모바일 원격 주문 APP

· Data 기반: 최근 구매 이력 / 매장 정보 / 주문 시간대, 날씨 등을 분석해 다양한 상황에서 맞춤형 추천서비스 시행

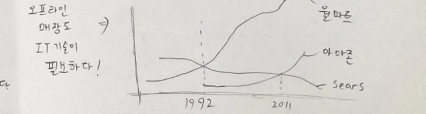


올바른 "소비자들이 회원권의 정보와, 회원권의 시간으로 인해 소용됨 증명"

· 쇼피켓: 우리 사람들 명의로 구매하는 현상 분석하는 서비스

· 플라이: 소비자들의 검색 경험 분석

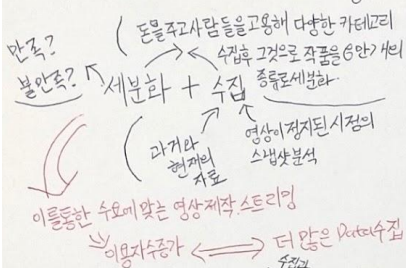
· Data 기반: 기상, 경제, SNS, 웹 검색 기록 등 2백대개 기본으로
드래그, 상환, 맞춤 추천서비스 시행



Netflix

스트리밍 서비스 및 영상제작 기업

고객의 영상에 대한 수요증가를 목적으로 사용



넷플릭스는 위의 과정처럼 다양한 데이터에 따른 세분화
를 통해 수요에 맞는 영상을 공급하고 이를 통해 고객을 늘리고 더
많은 데이터를 확보해 더 높은 정밀도로 다시 고객을 늘리는
선순환 구조를 만들었다.

Walmart

세계에서 가장 큰 유통업체

쇼핑 경험 최적화하고 고객에게 만족감 주기 위해 5가지

방법으로 노력하고 있음

1. 약국, 호텔, 레스토랑에 인접한 위치에 있는 슈퍼마켓을 확대
국, 각본들이 최적적으로 스토어 관리한다.
2. 스노우 데이 아웃, 수로 예측을 통해 적절한 가격에
판매한다.
3. 공급체인 관리, 배송까지 24시간 내로 배송 가능한 배송 시스템을
만들었다.
4. 제품 분류 최적화, 고객 선호도를 분석하여 선반 배치 관리
5. 고객들의 구매 패턴에 따라 상품 배치하고 할인 정보 전달
수 많은 데이터를 처리하여 최적적인 시스템을 만들어 냈다.

페이스북과 링크드인

두 업체 모두 사용자가 입력한 정보, 활동으로부터
데이터를 수집한다. 이 데이터들은 기업에게
제공하고, 광고로 수익을 창출한다.

사용자의 정보를 이용하는 경우, 어떤 정보를 수집하는지,
어떻게 사용하는지 투명하게 보여주는 것이 중요하다.

Microsoft

클라우드 서비스: MS BI, MS Office, 애저, DAAS
→ 전염병을 보는 생산성 및 편의성 개선용 빅데이터
클라우드 서비스 및 광고 정보 제공.

수집 데이터
→ MS SW 사용량, 웹서핑 시간 및 SNS 등
생각, 음악, 영상은 얼마나 오래 사용했는지,
어떤 HW, SW를 사용하는지, 사용자 음성,
컴퓨터 사용량 및 사용량 등.

Big Data 활용 관련 MS의 문제점.
→ MS 고객 스마트폰 플랫폼이 없음.
→ 윈도우를 통해 PC, 태블릿, 스마트폰 등 다양한 기기
를 연결하며

문제점
→ 편의성과 같이 AI의 균형점이 모호

Big Data 활용 기업

team: 고인물

클스 로이스

자동차도 유명하지만 사실은 엔진을 만드는 회사.
자동차도 유명하지만 사실은 엔진을 만드는 회사.

자동차도 유명하지만 사실은 엔진을 만드는 회사.
자동차도 유명하지만 사실은 엔진을 만드는 회사.

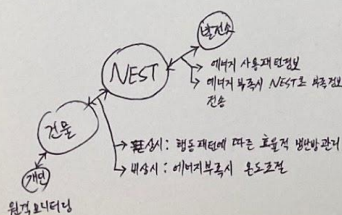
제품에 적용해서 품질 향상
생산선사각화들 통한 모듈도 문제점 피드백.
클스 로이스는 본인이 만드는 모든 제품에 부착시킨
센서. 공장에서 생산시 발생하는 각 부품에 대한
데이터. 정비사들의 피드백 데이터를 통해 제품의
결과 생산공정의 효율화, 기계화를 이뤄가고 있다.

NEST

저렴한 IoT 시스템 제공

집, 차량의 생활 패턴 파악으로 최적의 난방 및 에너지 공급

- 설치 장소에 대한 정보, 온도, 습도, 빛의 밝기, 행동, 입출근로 등 정보
- 타자와 연동 가능
- 개인 정보 인식 → 초기에도 공유치 않겠다고 함.
- point: 거주자를 이해하는 사생활은 점,



로터스 F1 팀: 모더스포르딩

실시간 차량 데이터를 피드백으로 전송.

한 바퀴마다 2천개의 데이터 전송.

이러한 데이터로 파손된 차량도

나쁜 주행이 가능하도록 조정 가능.

아픽시온: 렌스케어 플랫폼

환자에게 대한 많은 정보는 비정형 데이터.

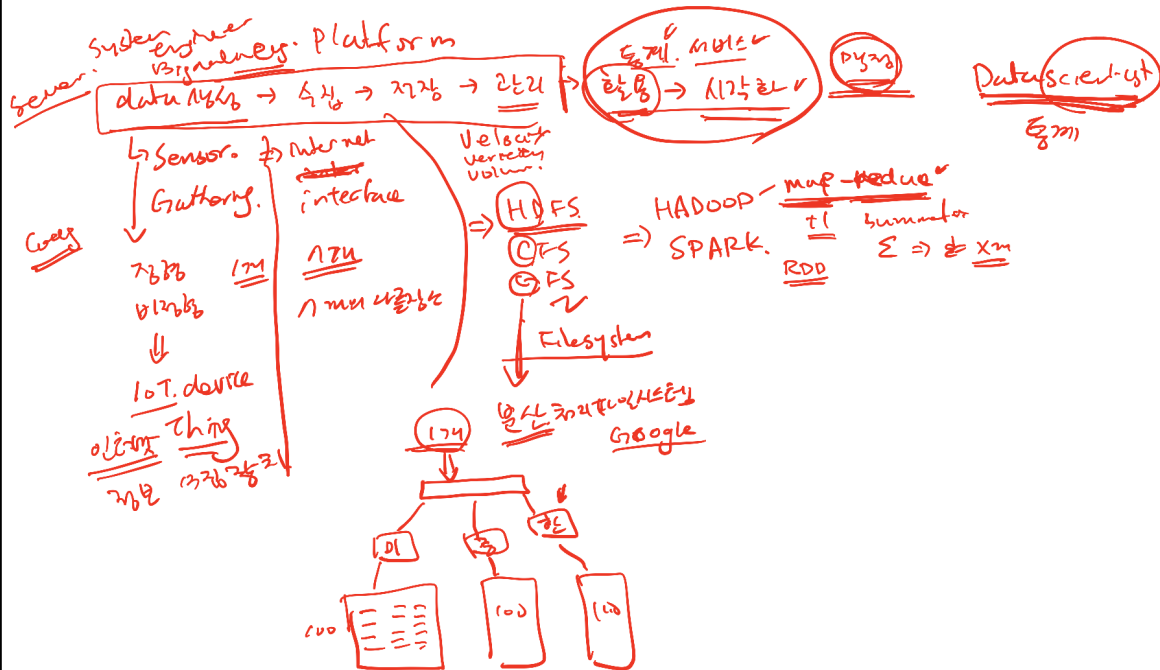
↓ 광학식문자판독기

컴퓨터가 인식 가능한 데이터

↓ 수집

비슷한 증상을 가진 환자들을 그룹화, 관리.

• 또한, 최근 진료로 하지 않은 환자도 기록을
통해 만성질환을 찾아냄.



남상민

```
import requests
import bs4
import csv
from tqdm import tqdm
```

```
headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = base_url + '?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 1000
```

```
#웹사이트를 텍스트로 로딩
raw = requests.get(url=naver_news_search.format('러닝머신'), headers=headers)
```

```
#기사가 max_count개만큼 모일때까지 1000번 찾기
```

```
for itr in tqdm(range(1000)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser') #로딩된 텍스트를 html형식으로 파싱
    news_titles = soup.select('.group_news .list_news li div a[title]') #형식을 만족하는 뉴스기사들 파싱
```

```
#정보기록할 csv파일 만들기
```

```
with open(file_path, 'a', encoding='UTF-8') as f:
    csv_file_w = csv.writer(f)
```

```

#형식에 맞는 뉴스기사정보들 하나씩 불러오기
for i, title in enumerate(news_titles):
    if count>max_count: exit(-1)
    data = [title['title'],title['href']] #각 기사정보에서 기사제목과 링크 가져오기
    csv_file_w.writerow(data) #가져온 기사이름과 링크를 파일에쓰기
    count+=1 #기사하나찾았으니깐 1증가
#현재페이지에서 기사 다찾았을때 다음페이지로 이동
next_link = soup.find(class_='btn_next') #다음페이지로 이동하기위한 주소가 기록된 클래스 찾기
next_link = next_link.attrs['href'] #클래스안의 링크 가져오기
raw = requests.get(url=base_url + next_link, headers=headers) #다음페이지 가져오기

```

박성호 (원래코드에서 수정된 부분을 노란색으로 표시)

```

import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = 'https://search.naver.com/search.naver?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 100000

with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

raw = requests.get(url=naver_news_search.format(20), headers=headers)

for itr in tqdm(range(1000)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('ul.list_news li.bx div.news_wrap div.news_area a[title]')

    with open(file_path, 'a', encoding='UTF-8') as f:
        csv_file_w = csv.writer(f)

        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(' ', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1

        next_link = soup.find(class_='btn_next')
        next_link = next_link.attrs['href']

        raw = requests.get(url=base_url + next_link, headers=headers)

```

2015011594 고성웅

```

import requests
import bs4
import csv
from tqdm import tqdm
headers={'User-Agent': 'Mozilla/5.0/'}
base_url='https://search.naver.com/search.naver'
naver_news_search =
'https://search.naver.com/search.naver?where=news&sm=tab_jum&query={}'

```

```

file_path='naver_news.csv'
count = 1
max_count = 100000
with open(file_path, 'w') as f:
    csv_file_w=csv.writer(f)
    csv_file_w.writerow(['title','url'])
raw=requests.get(url=naver_news_search.format('일본'), headers=headers)
for itr in tqdm(range(1000)):
    soup=bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles=soup.select('.api_subject_bx .group_news .list_news li a[title] ')#원래 코드에서 바꾼부분
    with open(file_path, 'a') as f:
        csv_file_w=csv.writer(f)

        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(',',' '), title['href']]
            csv_file_w.writerow(data)
            count +=1
            next_link=soup.find(class_='btn_next')#원래 코드에서 바꾼부분
            next_link=next_link.attrs['href']

            raw=requests.get(url=base_url+next_link, headers=headers)

```

김채진

```

import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = base_url + '?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 1000

#웹사이트를 텍스트로 로딩
raw = requests.get(url=naver_news_search.format('러닝머신'), headers=headers)

#기사가 max_count개만큼 모일때까지 1000번 찾기
for itr in tqdm(range(1000)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser') #로딩된 텍스트를 html형식으로 파싱
    news_titles = soup.select('.api_subject_bx .group_news li div a[title]')#형식을 만족하는 뉴스기사들 파싱

    #정보기록할 csv파일 만들기
    with open(file_path, 'a', encoding='UTF-8') as f:
        csv_file_w = csv.writer(f)

        #형식에 맞는 뉴스기사정보들 하나씩 불러오기
        for i, title in enumerate(news_titles):

            if count>max_count: exit(-1)

            data = [title['title'], title['href']] #각 기사정보에서 기사제목과 링크 가져오기
            csv_file_w.writerow(data) #가져온 기사이름과 링크를 파일에쓰기

```

```
count+=1 #기사하나찾았으니까 1증가
```

```
#현재페이지에서 기사 다찾았을때 다음페이지로 이동
```

```
next_link = soup.find(class_='btn_next') #다음페이지로 이동하기위한 주소가 기록된 클래스 찾기
```

```
next_link = next_link.attrs['href'] #클래스안의 링크 가져오기
```

```
raw = requests.get(url=base_url + next_link, headers=headers) #다음페이지 가져오기
```

여명준

```
import requests
import bs4
import csv
from tqdm import tqdm
```

```
headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = 'https://search.naver.com/search.naver?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 1000
```

```
#웹사이트를 텍스트로 로딩
```

```
raw = requests.get(url=naver_news_search.format('러닝머신'), headers=headers)
#headers: 에이전트 정보를 심어서 request를 알리게 됨
#raw: HTML 형식으로 뉴스 정보가 저장됨
```

```
#기사가 max_count개만큼 모일때까지 1000번 찾기
```

```
for itr in tqdm(range(1000)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser') #로딩된 텍스트를 html형식으로 파싱
    #raw.text: raw에서 글자들만 가져옴
    #html.parser을 사용하여 태그정보를 익힘
    #만들어진 정보를 soup 변수에 저장
```

```
news_titles = soup.select('.api_subject_bx .group_news li div a[title]') #형식을
만족하는 뉴스기사들 파싱
#select: news, list news, li, div 순으로 찾음 -> title이란 속성을 가져오고 싶음
```

```
#정보기록할 csv파일 만들기
```

```
with open(file_path, 'a', encoding='UTF-8') as f:
    csv_file_w = csv.writer(f)
```

```
#형식에 맞는 뉴스기사정보들 enumerate를 이용하여 하나씩 불러오기
```

```
for i, title in enumerate(news_titles):
    if count > max_count: exit(-1)
    data = [title['title'], title['href']] #각 기사정보에서 기사제목과 링크 가져오기
    #replace: 쉼표 공백처리
    csv_file_w.writerow(data) #가져온 기사이름과 링크를 파일에 쓰기
    count+=1 #기사하나찾았으니까 1증가
```

```
#현재페이지에서 기사 다찾았을때 다음페이지로 이동
```

```
next_link = soup.find(class_='btn_next') #다음페이지로 이동하기위한 주소가 기록된 클래스 찾기
```

```
next_link = next_link.attrs['href'] #클래스안의 링크 가져오기
```

```
raw = requests.get(url=base_url + next_link, headers=headers) #다음페이지 가져오기
#headers: 사람처럼 인식
```

2015011459 강성욱

```
import requests # http 요청을 위한 모듈
import bs4 # html
import csv # csv data
from tqdm import tqdm # 반복문 진행 상황을 출력하기 위한 모듈

headers = {'User-Agent': 'Mozilla/5.0'} # 차단 방지를 위한 눈속임용 (프로그램을 사용한다는 것을
# 모르게 하기 위함)
base_url = 'https://search.naver.com/search.naver'
naver_news_search =
'https://search.naver.com/search.naver?where=news&sm=tab_jum&query={}'
file_path = 'naver_news.csv'
# 페이지 1부터 10만까지 긁어옴 (무수히 많이 긁어오겠다는 것)
count = 1
max_count = 100000

#CSV file 생성 및 초기화
with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

#태그 분석
raw = requests.get(url=naver_news_search.format('루시드 에어'), headers=headers)

for itr in tqdm(range(50)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('.group_news .list_news li div a[title]')

    #텍스트 추출
    with open(file_path, 'a', encoding="UTF8") as f:
        csv_file_w = csv.writer(f)

        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(',', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1

#페이지 이동
next_link = soup.find(class_='btn_next')
next_link = next_link.attrs['href']

raw = requests.get(url=base_url + next_link, headers=headers)
```

2015012009 배상훈

```
import requests #http 요청
import bs4 ##html을 파싱
import csv #csv data를 생성
from tqdm import tqdm #반복 문 진행 상황 출력

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = \
'http://search.naver.com/search.naver?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
```

```

max_count = 100000

# base_url: 네이버 뉴스 메인 주소
# naver_news_search: 네이버 뉴스 검색에 사용할 주소
# file_path: 크롤링한 데이터를 저장할 csv file 경로
# count: 저장한 기사 개수
# max_count: 최대 저장 가능한 기사 개수

# naver_news_search 문자열 내부 {}에 검색 키워드를 넣음 ex) = 코로나

with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

raw = requests.get(url=naver_news_search.format('머신러닝'), headers = headers)
for itr in tqdm(range(100)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    # html을 BS를 통해 파싱
    news_titles = soup.select('.group_news .list_news li div a[title]')

    # 뉴스페이지의 html 구조에 맞도록 html 데이터를 파싱
    # '.news .type01 li dt a[title]' : 각각 news(class), type01(class), li(tag),
    dt(tag),
    # a[title](속성)으로 검색하여 가져온다.

    with open(file_path, 'a') as f:
        csv_file_w = csv.writer(f)
        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(' ', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1

    next_link = soup.find(class_='btn_next')
    next_link = next_link.attrs['href']

    raw = requests.get(url=base_url + next_link, headers=headers)

```

김정규

```

import requests
import bs4
import csv
from tqdm import tqdm

headers={'User-Agent': 'Mozilla/5.0'}
base_url='https://search.naver.com/search.naver'
naver_news_search=base_url + '?where=news&query={}'
file_path='naver_new.csv'
count=1
max_count=1000

with open(file_path, 'w') as f:
    csv_file_w=csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

```

```

raw=requests.get(url=naver_news_search.format('머신러닝'), headers=headers)

for itr in tqdm(range(30)):
    soup=bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles=soup.select('.api_subject bx .group_news li div a[title]')

    with open(file_path, 'a', encoding="UTF8") as f:
        csv_file_w=csv.writer(f)

        for i, title in enumerate(news_titles):
            if count>max_count: exit(-1)
            data=[title['title'].replace(', ', ' '), title['href']]
            csv_file_w.writerow(data)
            count +=1

next_link=soup.find(class_='btn_next')
next_link=next_link.attrs['href']

raw=requests.get(url=base_url + next_link, headers=headers)

```

김영재

```

import requests #http 요청을 위한 모듈
import bs4 #html을 파싱(구분하여 나누는 행위) 하기 위한 모듈
import csv #csv data를 생성하기 위한 모듈
from tqdm import tqdm # 반복문 진행 상황을 출력하기 위한 모듈

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver' #네이버 뉴스 메인 주소
naver_news_search = 'https://search.naver.com/search.naver?where=news&sm=tab_jum&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 100000

with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

raw = requests.get(url=naver_news_search.format('아이폰12미니'), headers=headers)
for itr in tqdm(range(30)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser') # html을 BS를 통해 파싱
    news_titles = soup.select('.group_news .bx div a[title]')
    # 뉴스페이지의 html 구조에 맞도록 html 데이터를 파싱'.news .type01 li dt a[title]'\
    # : 각각 news(class), type01(class), li(tag), dt(tag), a[title](속성)으로 검색하여
    가져온다.

    with open(file_path, 'a', encoding = 'utf-8') as f:
        csv_file_w = csv.writer(f)
        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(', ', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1
        next_link = soup.find(class_='btn_next')
        next_link = next_link.attrs['href']

```

```
raw = requests.get(url=base_url + next_link, headers=headers)
```

강만길

```
import requests          #http 요청을 위한 모듈
import bs4               #html을 파싱(구분해서 나누는것)하기 위한 모듈
import csv               #csv data를 생성하기 위한 모듈
from tqdm import tqdm    #반복 문 진행상황을 출력하기 위한

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = 'https://search.naver.com/search.naver?where=news&query={}'
file_path='naver_news.csv'
count = 1
max_count = 100000

raw = requests.get(url=naver_news_search.format('게임'), headers=headers)
for itr in tqdm(range(2)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('.group_news .list_news .bx a[title]')
    with open(file_path, 'a') as f:
        csv_file_w = csv.writer(f)

        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(' ', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1
            next_link=soup.find(class_='btn_next')
            next_link = next_link.attrs['href']

raw = requests.get(url = base_url + next_link, headers = headers)
```

안호균

```
import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search='https://search.naver.com/search.naver?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 100000

with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

raw = requests.get(url=naver_news_search.format('아이폰 12 미니'), headers=headers)

for itr in tqdm(range(30)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('.api_subject_bx .group_news .list_news li a[title]')

    with open(file_path, 'a') as f:
        csv_file_w = csv.writer(f)
```

```

for i, title in enumerate(news_titles):
    if count > max_count: exit(-1)
    data = [title['title'].replace(' ', ' '), title['href']]
    csv_file_w.writerow(data)
    count += 1
#for 문 이어짐

next_link = soup.find(class_='btn_next')
next_link = next_link.attrs['href']

raw = requests.get(url=base_url + next_link, headers=headers)

```

하영우

```

import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = 'https://search.naver.com/search.naver?where=news&query={}'
file_path = 'naver_news.csv'
count = 1
max_count = 100000
wforsearch = input("원하시는 검색어를 입력하세요.\n")
numforrange = int(input("원하시는 범위를 입력하세요.\n"))
raw = requests.get(url=naver_news_search.format(wforsearch), headers=headers)

for itr in tqdm(range(numforrange)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('.list_news li a[title]')

    with open(file_path, 'a', encoding="UTF8") as f:
        csv_file_w = csv.writer(f)
        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(' ', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1
            next_link = soup.find(class_='btn_next')
            next_link = next_link.attrs['href']
            raw = requests.get(url=base_url + next_link, headers=headers)

```

2019080127 주효정

```

import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent' : 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver'
naver_news_search = 'http://search.naver.com/search.naver?where=news&query={}'
file_path = 'naver_news.csv'
count = 1

```

```

max_count = 1000

raw = requests.get(url=naver_news_search.format('머신러닝'), headers=headers)

for itr in tqdm(range(1000)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('.api_subject_bx a[title]')

    with open(file_path, 'a', encoding='UTF-8') as f:
        csv_file_w = csv.writer(f)

        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'], title['href']]
            csv_file_w.writerow(data)
            count += 1

    next_link = soup.find(class_='btn_next')
    next_link = next_link.attrs['href']

    raw = requests.get(url=base_url + next_link, headers=headers)

```

임채현

```

max_count = 100000
naver_news_search = 'https://search.naver.com/search.naver?where=news&query={}'
with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])
raw = requests.get(url=naver_news_search.format('머신러닝'), headers=headers)
for itr in tqdm(range(10)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('.group_news .list_news div a[title]')
    with open(file_path, 'a', encoding='UTF-8') as f:
        csv_file_w = csv.writer(f)

        for i, title in enumerate(news_titles):
            if count > max_count: exit(-1)
            data = [title['title'].replace(',', ' '), title['href']]
            csv_file_w.writerow(data)
            count += 1
            next_link = soup.find(class_='btn_next')
            next_link = next_link.attrs['href']

    raw = requests.get(url=base_url + next_link, headers=headers)

```

다른 사이트에서 크롤링한 예제

이름, 코드 올려주세요.

남상민

```

#네이버 날씨 크롤러
import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent': 'Mozilla/5.0'}
base_url = 'https://search.naver.com/search.naver?where=nexearch&sm=top_hy&fbm=0&ie=utf8&query={}'
file_path = 'weather.csv'
count = 0

data = [0 for _ in range(32)]
tmp = [0 for _ in range(8)]
time = [0 for _ in range(8)]
wth = [0 for _ in range(8)]

#웹사이트를 텍스트로 로딩
raw = requests.get(url=base_url.format('날씨'), headers=headers)

#기사가 max_count개만큼 모일때까지 1000번 찾기
soup = bs4.BeautifulSoup(raw.text, 'html.parser') #로딩된 텍스트를 html형식으로 파싱
weather = soup.select('.list_area li dl dd span') #형식을 만족하는 날씨 탭 파싱

#정보기록할 csv파일 만들기
with open(file_path, 'a') as f:
    csv_file_w = csv.writer(f)

#날씨정보를 모두분리하여 저장
for i, weatherInfo in enumerate(weather):
    # get_text를 이용하여 태그안의 내용을 가져옴
    tempString = str(weatherInfo.get_text()).lstrip().rstrip()
    # 조건을 만족하는 태그안의 내용만 저장
    if ((tempString != "") & (count < 32) & (tempString != '내일') & (tempString != data[count-1])):
        data[count] = tempString
        count += 1
# 온도, 시간, 날씨 데이터 별로 분리하여 저장
for i in range(8):
    tmp[i] = data[4*i+1]
    time[i] = data[4*i+3]
    wth[i] = data[4*i+2]
    csv_file_w.writerow([tmp[i], tmp[i], wth[i]]) #분리한 데이터를 파일에 쓰기
    print(time[i], tmp[i], wth[i])

print('https://search.naver.com/search.naver?where=nexearch&sm=top_hy&fbm=1&ie=utf8&query=%EB%82%A0%EC%94%A8')

```

박성호(무신사 인기차트 크롤러)

```

import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent': 'Mozilla/5.0'}
file_path = 'musinsa_rank.csv'

```

```

with open(file_path, 'w') as f:
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['rank', 'name', 'url'])

for itr in tqdm(range(100)):
    raw = 
requests.get(url=f'https://search.musinsa.com/ranking/best?period=now&mainC
ategory=&subCategory=&price=&newProduct=false&discount=false&soldOut=false&
page={itr+1}&viewType=small&device=&priceMin=&priceMax=', headers=headers)

soup = bs4.BeautifulSoup(raw.text, 'html.parser')
rank_titles = soup.find_all(class_="li_box")

with open(file_path, 'a', encoding='UTF-8') as f:
    csv_file_w = csv.writer(f)

    for i, title in enumerate(rank_titles):
        rank = str(itr*10+(i+1)) + '위'
        title = title.select('.list_info a[title]')
        url = title[0].attrs['href']
        title = title[0].attrs['title']
        data = [rank.replace(',', ' '), title.replace(',', ' '), url]
        csv_file_w.writerow(data)

```

```

2015011336 김경훈
#무신사 '패딩' 검색
import requests
import bs4
import csv
from tqdm import tqdm

headers = {'User-Agent' : 'Mozilla/5.0'}
base_url = 
'https://search.musinsa.com/search/musinsa/goods?q=패딩&list_kind=small&sort
Code=pop&sub_sort=&page='
musinsa_search = 
'https://search.musinsa.com/search/musinsa/goods?q={}&list_kind=small&sortC
ode=&sub_sort=&page=1&display_cnt=90&saleGoods=&includeSoldOut=&popular=&ca
tegory1DepthCode=&category2DepthCodes=&category3DepthCodes=&selectedFilters
=&category1DepthName=&category2DepthName=&brandIds=&price=&colorCodes=&cont
entType=&styleTypes=&includeKeywords=&excludeKeywords=&originalYn=N&tags=&s
aleCampaign=false&serviceType=&d_cat_cd='
file_path = 'musinsa_search.csv'
count = 1
max_count = 100000

with open(file_path, 'w') as f :
    csv_file_w = csv.writer(f)
    csv_file_w.writerow(['title', 'url'])

raw = requests.get(url=musinsa_search.format('패딩'), headers=headers)

for itr in tqdm(range(10)) :
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    musinsa_titles = soup.select('ul li div.li_inner div.article_info

```

```
p.list_info a[title]')
```

```
with open(file_path, 'a') as f :  
    csv_file_w = csv.writer(f)
```

```
    for i, title in enumerate(musinsa_titles) :  
        if count > max_count : exit(-1)  
        data = [title['title'].replace(',', ' '), title['href']]  
        csv_file_w.writerow(data)  
        count += 1  
    next_link = soup.find(class_='fa fa-angle-right paging-btn btn  
next')  
    next_link = next_link.attrs['href']
```

```
raw = requests.get(url=base_url + next_link, headers=headers)
```

주요정

```
import requests  
import bs4  
import csv  
from tqdm import tqdm
```

```
headers = {'User-Agent' : 'Mozilla/5.0'}  
base_url = 'https://www.archdaily.com'  
naver_news_search =  
https://www.archdaily.com/search/all?q=housing&ad_source=jv-header  
file_path = 'archdaily_search'  
count = 1  
max_count = 100
```

```
raw = requests.get(url=archdaily_search.format('housing'), headers=headers)
```

```
for itr in tqdm(range(100)):  
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')  
    archdaily_titles = soup.select('.gridview a[title]')
```

```
with open(file_path, 'a', encoding='UTF-8') as f:  
    csv_file_w = csv.writer(f)
```

```
    for i, title in enumerate(archdaily_titles):  
        if count > max_count: exit(-1)  
        data = [title['title'], title['href']]  
        csv_file_w.writerow(data)  
        count += 1
```

```
next_link = soup.find(class_='btn_next')  
next_link = next_link.attrs['href']
```

```
raw = requests.get(url=base_url + next_link, headers=headers)
```

백은선

```
import requests  
import bs4  
import csv
```

```
from tqdm import tqdm

headers={'User-Agent': 'Mozilla/5.0'}
base_url = "https://www.melon.com/chart/index.htm"
melon_music_search = 'https://www.melon.com/chart/{}/index.htm'
file_path='melon_music.csv'
count = 1
max_count=10

raw =
requests.get(url=melon_music_search.format('day'), headers=headers)

for itr in tqdm(range(1000)):
    soup = bs4.BeautifulSoup(raw.text, 'html.parser')
    news_titles = soup.select('#lst50 td .wrap .wrap_song_info
div span a[title]')

    with open(file_path, 'a', encoding='UTF-8') as f:
        csv_file_w = csv.writer(f)

        for i, title in enumerate(news_titles) :
            if count > max_count: exit(-1)
            data = [title['title'].replace(' ', ' '),
title['href']]
            csv_file_w.writerow(data)
            count += 1
```

[3주차] 오늘의 일정

인공지능 개념 잡기 (발표자료)

1. 71p, 지능(퍼펙트론)

https://docs.google.com/presentation/d/1UO-cWiQ5EI7WCwUxDumArOepw0FTkU_EA6CI8hOzi4M/edit?usp=sharing

2. 85p, 만화로 배우는 인공지능 (4a2b)

https://docs.google.com/presentation/d/1gym_4rX26pCwcRiWT2URhw6lxXfYK4JPapV3A2tmM4Y/edit?usp=sharing

3. 99p, 볼츠만 머신러닝 (젊음)

https://docs.google.com/presentation/d/1CC0ZY3WestBDbS3K7Y-QKM43B0-VKpbHFI0_HkHv6al/edit?usp=sharing

4. 126p, 뇌, 마음, 인공지능(4K)

https://docs.google.com/presentation/d/1HTpEyp50FFlyU9i54HBMCE9qvBlpf1nfdU9qir_2D2E/edit?usp=sharing

5. 153p, 인공지능의 철학 (진진남선)

<https://docs.google.com/presentation/d/14vfXmEI4K-HtFtsJkkLtIDPncSWddpnCuDI9Qy-GLdY/edit?usp=sharing>

6. 자유주제 1 (고인물)

7. ,

8. 자유주제 2 (러닝머신)

https://docs.google.com/presentation/d/1xvzcvAa9Inrw-CFlaD5gOpKXBHt_zn3qAd4qi_3j3Tw/edit?usp=sharing

google slide 로 제작하여 단톡방에 공유 **01010110110101**
첫페이지는 팀원들과 찍은 단체 사진 + 팀원 이름

발표 순서

1. [젊음](#)
2. [4k](#)
3. [런닝머신](#)
4. [진진남선](#)
5. [퍼펙트론](#)
6. [4a2b](#)
7. [고인물](#)

점심

math for AIs

colab 사용법 (254)

<https://colab.research.google.com>

MNIST 데이터셋을 활용한 지도학습 (**easy, advanced**)

MNIST interactive DEMO

<https://transcranial.github.io/keras-js/#/mnist-cnn>

#자동채색 설문조사

https://docs.google.com/forms/d/e/1FAIpQLSfJLw6HGpBljr3I4V2JCu-SzkyqjA1dbu9qMaAjKESVcRjKeA/viewform?usp=sf_link

20201122 오늘의 할 일

지도학습 코드 설명

MNIST데이터를 이용한 손글씨 인식 코드 설명

인공지능을 이용한 성능 개선 평가

- Download Test Jupiter Notebook file [here](#) and upload to [cola.google.com](https://colab.research.google.com/)
- 1. 17시에 자기 팀이 만든 코드를 발표자료로 만들어 소개
- 2. 발표자료에는 변경전과 변경후를 요약하여 표현
- 3. 발표자료 링크는 이 페이지에 팀별로 남김

발표자료의 구성

1. 표지
2. 가장 우수한 결과를 만든 코드를 얻기 까지 어떤 시도를 했는지 설명하기
3. 가장 우수한 결과의 코드를 기존 코드와 비교하고 설명하기
4. 결과 보이기

* 활성화 함수 바꾸는 방법

Activation('relu') 로 된 부분을

Activation(tf.nn.[원하는 활성화 함수 이름]) 으로 바꿔주시면 됩니다.

ex) leaky_relu -> Activation(tf.nn.leaky_relu)

발표자료 링크

1->3 4a2b
2->1 퍼펙트론
3->5 러닝머신
4->7 젊음
5->2 고인물
6->4 4k
7->6 진진남선

고인물:

https://docs.google.com/presentation/d/1oW5TNcYugi2nt6UkJ9AacfMIQ5QyzPu7vqePmAgw_4w/edit?usp=sharing

4A2B:https://docs.google.com/presentation/d/1ETqcgGUCpxFOkzfmJOwLvo0vWM_pf65dj12ZFZdqqE/edit#slide=id.p

4K:https://docs.google.com/presentation/d/1X9mKhvnTXhxBUvpEpxUTZCxiWkKBCoEjmm-lZR_XqfbY/edit?usp=sharing

러닝머신:https://docs.google.com/presentation/d/1xVBerS9tLeOV341wCuw3fGSBleVeUrkYVxi1TI5qX_w/edit?usp=sharing

진진남선:

<https://docs.google.com/presentation/d/1b9mnTm0bgsP-YUtdtXQIKDf2FmD7eOQ8ef5-9UHBAY0/edit?usp=sharing>

젊음:https://docs.google.com/presentation/d/15lknBs3MmCz8iW_9lmhJu97v6qlzvx_DBDLUtR_i5sc/edit#slide=id.p

수업에 관한 설문조사

<https://forms.gle/1d4b7UqpdrupEFRU7>