

```

#if문 ':'사용을 빠뜨리지말고 생각하자
weather = "비"
if weather == "비":
    print("우산을 챙기세요")
elif weather == "미세먼지":
    print("마스크를 챙기세요")
else :
    print("준비물이 없어요")
#if문 ':'사용을 빠뜨리지말고 생각하자
weather = input("오늘 날씨는 어때요?")
if weather == "비":
    print("우산을 챙기세요")
elif weather == "미세먼지":
    print("마스크를 챙기세요")
else :
    print("준비물이 없어요")
temp= int(input("오늘 기온이 어때요?"))
if 30<=temp:
    print("너무 더워요. 나가지 마세요")
elif 10<=temp and 30>temp:
    print("날씨가 좋아요 ")
elif 0<=temp<10:
    print("추우니 외투를 챙겨요.")
else:
    print("너무 추우니 나가지 마세요.")
#for문
#print("대기번호:1")
#print("대기번호:2")
#print("대기번호:3")
#print("대기번호:4")

```

```

for waiting_num in [0,1,2,3,4]:
    print("대기번호:{0}".format(waiting_num))
starbucks=["아이언맨", "토르", "아이엠 그루트"]
for customer in starbucks:
    print("{0},커피가 준비되었습니다.".format(customer))
#while
customer=" 토르"
index=5
while index>=1:
    print("{0},커피가 준비 되었습니다.{1}번 남았어요.".format(customer,index))
    index-=1
    if index == 0:
        print("커피는 폐기처분되었습니다.")

```

#무한 루프에 빠진 명령문.이럴때에는 터미널에서 컨트롤c를 눌러 강제 종료를 시킬수있다.

```
customer="아이언맨"
```

```

index=1
while True:
    print("{0},커피가 준비되었습니다.호출{1}회".format(customer,index))
    index+=1
#맞을때 까지 계속 반복되다가 맞으면 반복문을 탈출 하는 형식
customer="토르"
person="unknown"

while person!=customer:
    print("{0},커피가 준비되었습니다.".format(customer))
    person = input("이름이 어떻게 되세요?")
#continue 와 break문
absent=[2,5]
no_book=[7]
for student in range(1,11):
    if student in absent:
        continue
    elif student in no_book:
        print("오늘 수업 여기까지.{0}는 교무실로 따라와".format(student))
        break
    print("{0},책을 읽어봐".format(student))

#한줄 for
#출석번호가 1,2,3,4 앞에 100을 붙이기로함->101,102,103,104
students=[1,2,3,4,5]
print(students)
students=[i+100 for i in students]
print(students)

#학생이름을 길이로 변환
students=["Iron man", "Thor", "I am groot"]
students=[len(i)for i in students]
print(students)

#학생 이름을 대문자로 바꿔보아라
students=["iron man","thor", "i am groot"]
students=[i.upper() for i in students]
print(students)

#함수
def open_account():
    print("새로운 계좌가 생성되었습니다.")

def deposit(balance,money): #입금
    print("입금이 완료되었습니다. 잔액은{0}원입니다.".format(balance+money))
    return balance+money

```

```

def withdraw(balance, money):#출금
    if balance >=money:#잔액이 출금보다 많으면
        print("출금이 완료되었습니다.잔액은{0}원입니다.".format(balance-money))
        return balance-money
    else:
        print("출금이 완료되지 않았습니다. 잔액은 {0}원입니다".format(balalnce))
        return balance
def withdraw_night(balance,money):#저녁에 출금
    comission=100#수수료100원
    return comission,balance-money-comission

balance=0
balance=deposit(balance,1000)
print(balance)
def profile(name,age,main_lang):
    print("이름:{0}\t나이:{1}\t주 사용 언어:{2}".format(name,age,main_lang))

profile("유재석", 20, "파이썬")
profile("노홍철", 25,"자바")

```

#함수의 기본값사용

```

def profile(name,age=17,main_lang="파이썬"):
    print("이름:{0}\t나이:{1}\t주 사용 언어:{2}".format(name,age,main_lang))

profile("유재석")
profile("노홍철")

```

#키워드.순서가 바뀌어도 함수의 순서에 따라 나온다

#키워드값

```

def profile(name,age,main_lang):
    print(name,age,main_lang)
profile(name="유재석", main_lang="파이썬", age=20)
profile(main_lang="자바", age=25,name="김태호")

```

#가변인자를 통한 변수 *을 사용하여 만들수 있다

```

#def profile(name,age,lang1,lang2,lang3,lang4,lang5):
#    #print("이름:{0}\t나이:{1}\t".format(name,age),end=" ")
#    #print(lang1,lang2,lang3,lang4,lang5)

```

```

def profile(name,age,*language):
    print("이름:{0}\t나이:{1}\t".format(name,age),end=" ")
    for lang in language:
        print(lang, end=" ")
    print()

```

```

profile("유재석",20,"python","Java","C","C++","C#","Javascript")

```

```
profile("김태호",25,"kotlin", "swift")
```

#만약에 유재석이 배운 언어가 하나더 추가된다면 lang6을 적어야하나 그렇게 하면 귀찮기 때문에 가변인자를 사용한다

#지역변수와 전역변수

```
un=10
```

```
def checkpoint(soldiers):#경계근무
```

```
    global gun#전역 공간에 있는 gun사용
```

```
    #gun=20#==>함수 내 변수.지역변수
```

```
    gun=gun-soldiers
```

```
    print("[함수 내 남은총:{0}"].format(gun))
```

```
print("전체 총:{0}"].format(gun))
```

```
checkpoint(2)#2명이 경계 근무 나감
```

```
print("남은 총:{0}"].format(gun))
```

#함수 외부에서 총 10자루인데 함수내부에 20자루라고 하면 남은총 18

#global 을 사용하면 전역변수로 된다.

#Quiz 성별에 따른 표준체중 구하기==>소수 둘째 자리까지(round함수사용)

```
def std_weight(height,gender):
```

```
    if gender=="남자":
```

```
        return height*height*22
```

```
    else :
```

```
        return height*height*21
```

```
height=175#cm
```

```
gender="남자"
```

```
weight=round(std_weight(height/100,gender),2)
```

```
print("키{0}cm{1}의 표준 체중은{2}입니다.".format(height,gender,weight))
```