

```

#print("python","java",sep=","end=" ")
#print("무엇이 더 재밌을까요?")

#import sys
#print("python","Java",file=sys.stdout)
#print("python","Java",file=sys.stderr)

#시험 성적
#scores={"수학":0,"영어":50,"코딩":100}
#for subject,score in scores.items():
#    #print(subject,score)
#    # print(subject.ljust(8),str(score).rjust(4))#==>ljust,rjust는 왼쪽정렬과 오른쪽
정렬을 뜻한다

for num in range(1,21):
    print("대기번호 :",str(num).zfill(3))#zfill=>0을 채울때 사용
answer=input("아무 값이나 입력하세요:")
print(type(answer))
#print("입력하신 값은"+answer+"입니다.")==>표준 입출력에서 input 에서는 문자형으로
저장이된다.
#빈자리는 빈공간으로 두고, 오른쪽 정렬을 하되, 총10자리 공간을 확보
print("{0: >10}".format(500))
#양수일때는 +로 표시, 음수일땐 -로 표시
print("{0: >+10}".format(500))
print("{0: >+10}".format(-500))
#왼쪽 정렬하고,빈칸으로_로 채움
print("{0: _<10}".format(500))
#3자리 마다 콤마를 찍어주기
print("{0:,}".format(1000000000000))
#3자리 마다 콤마찍어주기, +-부호도 붙이기
print("{0:+,}".format(1000000000000))
#돈이 많으면 행복하니까 빈자리는 ^로 채워주기
print("{0: ^<+30}".format(1000000000000))#==>30자리 만큼 있다고 하고
나머지빈자리는 ^로 채움
#소수점 출력
print("{0:f}".format(5/3))
#소수점을 특정 자리수 까지만 표시:
print("{0:.2f}".format(5/3))

#score_file=open("score.txt","w",encoding="utf8")
#print("수학:0",file=score_file)
#print("영어:50",file=score_file)
#score_file.close()

score_file=open("score.txt","a",encoding="utf8")
score_file.write("과학:80")
score_file.write("\n코딩:100")

```

```
score_file.close()
```

```
#score_file=open("score.txt","r",encoding="utf8")  
#print(score_file.read())  
#score_file.close()
```

```
score_file=open("score.txt","r",encoding="utf8")  
print(score_file.readline())  
print(score_file.readline())  
print(score_file.readline())  
print(score_file.readline())  
score_file.close()
```

```
score_file=open("score.txt","r",encoding="utf8")  
while True:  
    line=score_file.readline()  
    if not line:  
        break  
    print(line)  
score_file.close()
```

```
import pickle  
profile_file=open("profile.pickle","wb")  
profile={"이름":"박명수","나이":30,"취미":["축구","농구","코딩"]}  
print(profile)  
pickle.dump(profile,profile_file)#==>profile에 있는 정보를 file 에 저장  
profile_file.close()
```

```
profile_file=open("profile.pickle","rb")  
profile=pickle.load(profile_file)  
print(profile)  
profile_file.close()
```

```
with open("stduy.txt","w",encoding="utf8")as study_file:  
    study_file.write("파이썬을 열심히 공부하고 있어요")  
with open("study.txt","r",encoding="utf8")as study_file:  
    print(stduy_file.read())
```

```
for i in range(1,51):  
    with open(str(i)+"주차.txt","w",encoding="utf8")as report_files:  
        report_files.write("-{0}주차 주간보고-".format(i))  
        report_files.write("\n부서:")  
        report_files.write("\n이름:")  
        report_files.write("\n업무 요약:")  
name="마린"  
hp=40
```

```
damage=5
```

```
print("{0}유닛이 생성되었습니다.".format(name))  
print("체력{0},공격력{1}\n".format(hp,damage))
```

```
tank_name="탱크"  
tank_hp=150  
tank_damage=35
```

```
print("{0}유닛이 생성되었습니다.".format(tank_name))  
print("체력{0},공격력{1}".format(tank_hp,tank_damage))
```

```
def attack(name,location, damage):  
    print("{0}:{1}방향으로 적군을 공격합니다.[공격력  
{2}"].format(name,location,damage))  
    attack(name,"1시",damage)  
    attack(tank_name,"1시",tank_damage)  
#여기서 마린이나 탱크의 수를 늘릴려면 수동으로 하면 되는데 시간이 너무 오래 걸림.이럴때  
사용하는 것이 클래스함수이다.
```

#class 함수 ==> class 라는 것을 만들어서 여러개를 만들때 보다 수월하게 만들수 있다

```
class Unit:
```

```
    def__init__(self,name,hp,damage):  
        self.name=name  
        self.hp=hp  
        self.damage=damage  
        print("{0}유닛이 생성되었습니다.".format(self.name))  
        print("체력{0}, 공격력{1}".format(self.hp,self.damage))
```