

# 최소 화폐 개수로 금액 반환

>> 키보드로 반환할 금액을 입력받는다.

> 반환되는 화폐의 개수가 최소가 되도록 한다.

> 화폐 단위별로 반환되는 개수를 출력한다.

>> 예) 50000원 %d개, 10000원 %d개, 5000원 %d개, 1000원 %d개,

500원 %d개, 100원 %d개, 50원 %d개, 10원 %d개, 5원 %d개, 1원 %d개

```
In [2]: ## [1] 최소 화폐 개수로 금액 반환
## 5만원권 처리
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

coin_50000 = remain // 50000 #내준 개수
print("> 50000원권: %d" %coin_50000)

remain = remain - (coin_50000 * 50000) #남은 액수
print(">> 남은 액수: %d" %remain)
```

> 50000원: 2  
>> 남은 액수: 99998

```
In [4]: ## [2] 최소 화폐 개수로 금액 반환
## 5만원권 처리 > 1만원권 처리
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

coin_50000 = remain // 50000 #내준 개수: 50000으로 나눈 몫
print("> 50000원권: %d" %coin_50000)
remain = remain - (coin_50000 * 50000) #남은 액수: 50000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_10000 = remain // 10000 #내준 개수: 10000으로 나눈 몫
print("> 10000원권: %d" %coin_10000)
remain = remain - (coin_10000 * 10000) #남은 액수: 10000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)
```

> 50000원: 2  
>> 남은 액수: 23456  
> 10000원: 2  
>> 남은 액수: 3456

```
In [6]: ## [3] 최소 화폐 개수로 금액 반환
## 5만원권 처리 > 1만원권 처리 : 산술식 변경
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

coin_50000 = remain // 50000 #내준 개수: 50000으로 나눈 몫
print("> 50000원권: %d" %coin_50000)
remain = remain % 50000 #남은 액수: 50000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_10000 = remain // 10000 #내준 개수: 10000으로 나눈 몫
print("> 10000원권: %d" %coin_10000)
```

```
remain = remain % 10000    #남은 액수: 10000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)
```

```
> 50000권: 2
>> 남은 액수: 23456
> 10000권: 2
>> 남은 액수: 3456
```

```
In [14]: ## [4] 최소 화폐 개수로 금액 반환
## 전권 처리
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

coin_50000 = remain // 50000 #내준 개수: 50000으로 나눈 몫
print("> 50000원권: %d" %coin_50000)
remain = remain % 50000    #남은 액수: 50000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_10000 = remain // 10000 #내준 개수: 10000으로 나눈 몫
print("> 10000원권: %d" %coin_10000)
remain = remain % 10000    #남은 액수: 10000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_5000 = remain // 5000 #내준 개수: 5000으로 나눈 몫
print("> 5000원권: %d" %coin_5000)
remain = remain % 5000    #남은 액수: 5000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_1000 = remain // 1000 #내준 개수: 1000으로 나눈 몫
print("> 1000원권: %d" %coin_1000)
remain = remain % 1000    #남은 액수: 1000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_500 = remain // 500 #내준 개수: 500으로 나눈 몫
print("> 500원권: %d" %coin_500)
remain = remain % 500    #남은 액수: 500으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_100 = remain // 100 #내준 개수: 100으로 나눈 몫
print("> 100원권: %d" %coin_100)
remain = remain % 100    #남은 액수: 100으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_50 = remain // 50 #내준 개수: 50으로 나눈 몫
print("> 50원권: %d" %coin_50)
remain = remain % 50    #남은 액수: 50으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_10 = remain // 10 #내준 개수: 10으로 나눈 몫
print("> 10원권: %d" %coin_10)
remain = remain % 10    #남은 액수: 10으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_5 = remain // 5 #내준 개수: 5로 나눈 몫
print("> 5원권: %d" %coin_5)
remain = remain % 5    #남은 액수: 5로 나눈 나머지
print(">> 남은 액수: %d" %remain)

coin_1 = remain // 1 #내준 개수: 1로 나눈 몫
print("> 1원권: %d" %coin_1)
remain = remain % 1    #남은 액수: 1로 나눈 나머지
print(">> 남은 액수: %d" %remain)
```

```

> 50000원권: 2
>> 남은 액수: 23456
> 10000원권: 2
>> 남은 액수: 3456
> 5000원권: 0
>> 남은 액수: 3456
> 1000원권: 3
>> 남은 액수: 456
> 500원권: 0
>> 남은 액수: 456
> 100원권: 4
>> 남은 액수: 56
> 50원권: 1
>> 남은 액수: 6
> 10원권: 0
>> 남은 액수: 6
> 5원권: 1
>> 남은 액수: 1
> 1원권: 1

```

```

In [14]: ## [5] 최소 화폐 개수로 금액 반환
## 패턴 구성: 변수 통일
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

result = remain // 50000 #내준 개수: 50000으로 나눈 몫
print("> 50000원권: %d" %result)
remain = remain % 50000 #남은 액수: 50000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 10000 #내준 개수: 10000으로 나눈 몫
print("> 10000원권: %d" %result)
remain = remain % 10000 #남은 액수: 10000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 5000 #내준 개수: 5000으로 나눈 몫
print("> 5000원권: %d" %result)
remain = remain % 5000 #남은 액수: 5000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 1000 #내준 개수: 1000으로 나눈 몫
print("> 1000원권: %d" %result)
remain = remain % 1000 #남은 액수: 1000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 500 #내준 개수: 500으로 나눈 몫
print("> 500원권: %d" %result)
remain = remain % 500 #남은 액수: 500으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 100 #내준 개수: 100으로 나눈 몫
print("> 100원권: %d" %result)
remain = remain % 100 #남은 액수: 100으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 50 #내준 개수: 50으로 나눈 몫
print("> 50원권: %d" %result)
remain = remain % 50 #남은 액수: 50으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 10 #내준 개수: 10으로 나눈 몫
print("> 10원권: %d" %result)
remain = remain % 10 #남은 액수: 10으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

```

```

result = remain // 5 #내준 개수: 5로 나눈 몫
print("> 5원권: %d" %result)
remain = remain % 5 #남은 액수: 5로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // 1 #내준 개수: 1로 나눈 몫
print("> 1원권: %d" %result)
remain = remain % 1 #남은 액수: 1로 나눈 나머지
print(">> 남은 액수: %d" %remain)

```

```

> 50000원권: 2
>> 남은 액수: 23456
> 10000원권: 2
>> 남은 액수: 3456
> 5000원권: 0
>> 남은 액수: 3456
> 1000원권: 3
>> 남은 액수: 456
> 500원권: 0
>> 남은 액수: 456
> 100원권: 4
>> 남은 액수: 56
> 50원권: 1
>> 남은 액수: 6
> 10원권: 0
>> 남은 액수: 6
> 5원권: 1
>> 남은 액수: 1
> 1원권: 1

```

```

In [17]: ## [6] 최소 화폐 개수로 금액 반환
## 패턴 구성: 금액도 변수 처리
unit = [50000, 10000, 5000, 1000, 500, 100, 50, 10, 5, 1] #List형 데이터 구조

remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

result = remain // unit[0] #내준 개수: 50000으로 나눈 몫
print("> 50000원권: %d" %result)
remain = remain % unit[0] #남은 액수: 50000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[1] #내준 개수: 10000으로 나눈 몫
print("> 10000원권: %d" %result)
remain = remain % unit[1] #남은 액수: 10000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[2] #내준 개수: 5000으로 나눈 몫
print("> 5000원권: %d" %result)
remain = remain % unit[2] #남은 액수: 5000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[3] #내준 개수: 1000으로 나눈 몫
print("> 1000원권: %d" %result)
remain = remain % unit[3] #남은 액수: 1000으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[4] #내준 개수: 500으로 나눈 몫
print("> 500원권: %d" %result)
remain = remain % unit[4] #남은 액수: 500으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[5] #내준 개수: 100으로 나눈 몫
print("> 100원권: %d" %result)

```

```

remain = remain % unit[5]    #남은 액수: 100으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[6]   #내준 개수: 50으로 나눈 몫
print("> 50원권: %d" %result)
remain = remain % unit[6]    #남은 액수: 50으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[7]   #내준 개수: 10으로 나눈 몫
print("> 10원권: %d" %result)
remain = remain % unit[7]    #남은 액수: 10으로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[8]   #내준 개수: 5로 나눈 몫
print("> 5원권: %d" %result)
remain = remain % unit[8]    #남은 액수: 5로 나눈 나머지
print(">> 남은 액수: %d" %remain)

result = remain // unit[9]   #내준 개수: 1로 나눈 몫
print("> 1원권: %d" %result)
remain = remain % unit[9]    #남은 액수: 1로 나눈 나머지
print(">> 남은 액수: %d" %remain)

```

```

> 50000원권: 2
>> 남은 액수: 23456
> 10000원권: 2
>> 남은 액수: 3456
> 5000원권: 0
>> 남은 액수: 3456
> 1000원권: 3
>> 남은 액수: 456
> 500원권: 0
>> 남은 액수: 456
> 100원권: 4
>> 남은 액수: 56
> 50원권: 1
>> 남은 액수: 6
> 10원권: 0
>> 남은 액수: 6
> 5원권: 1
>> 남은 액수: 1
> 1원권: 1
>> 남은 액수: 0

```

```

In [20]: ## [7] 최소 화폐 개수로 금액 반환
## 패턴 구성: 반복 패턴을 반복문으로 변환
unit = [50000, 10000, 5000, 1000, 500, 100, 50, 10, 5, 1] #List형 데이터 구조

cnt_won = len(unit) #unit의 항목 개수 구하기
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

for i in range(cnt_won): # (0에서부터 cnt_won이 되기 전까지) 값을 i로 반환하며 반복
    result = remain // unit[i] #내준 개수
    print("> %d원권: %d" %(unit[i], result))
    remain = remain % unit[i]    #남은 액수
    print(">> 남은 액수: %d" %remain)

```

```
> 50000원권: 2
> 10000원권: 2
> 5000원권: 0
> 1000원권: 3
> 500원권: 0
> 100원권: 4
> 50원권: 1
> 10원권: 0
> 5원권: 1
> 1원권: 1
```

```
In [21]: ## [8] 최소 화폐 개수로 금액 반환
## 일반화, 구조화 완료
unit = [50000, 10000, 5000, 1000, 500, 100, 50, 10, 5, 1] #List형 데이터 구조

cnt_won = len(unit) #unit의 항목 개수 구하기
remain = int(input("반환할 금액 입력(원)? ")) #남은 액수

for i in range(cnt_won): # (0에서부터 cnt_won이 되기 전까지) 값을 i로 반환하며 반복
    result = remain // unit[i] #내준 개수
    if result != 0:
        print("> %d원권: %d" % (unit[i], result))
    remain = remain % unit[i] #남은 액수
```

```
> 50000원권: 2
> 10000원권: 2
> 1000원권: 3
> 100원권: 4
> 50원권: 1
> 5원권: 1
> 1원권: 1
```

```
In [ ]:
```