

2조

김학균, 김해울, 노용준

```
a=1
qs=1
while(qs!= -1):
    qs = int(input('Enter the number : '))
    for i in range(qs):
        a = a * (i+1)
    if(qs != -1):
        print("%d! = %d" %(qs, a))
    a = 1
```

|

학균 코드

- 장점 : 짧고 보기 편하다.
- 단점 : -1 미만의 숫자를 입력하면 1이 출력된다.

```

while True :
    if n == -1 :
        break
    elif n<0 :
        print("계산할 수 없습니다. 수를 다시 입력해주세요.")
        n = int(input("Enter a number : "))
        continue
    elif n==0 :
        print( n, "! = 1")
        n = int(input("Enter a number : "))
        continue
    else :
        while n>0:
            result = 1
            for i in range (1,n+1):
                result *= i
                if i == n :
                    print( n, "! = ",result)
                    continue
            n = int(input("Enter a number : "))
        continue

```

용준 코드

- 장점 : 모든 경우의 수에 맞춰진 결과를 가져올 수 있다.
- 단점 : 코드가 너무 장황하고 보기 힘들다.

결론

- 보기 편한 학꾸니의 코드에 부족한 기능을 추가

```
result=1
number=1
while(number!= -1):
    number = int(input('Enter the number : '))
    if(number <= -2):
        print("계산할 수 없습니다. 숫자를 다시 입력해주세요")
    elif(number != -1):
        for i in range(number):
            result = result * (i+1)
        print("%d! = %d" %(number, result))
result = 1
```

학균 코드 개선

- 개선안
- 의미를 가지고 있지 않던 qs와 a코드를 number와 result로 수정
- -1 미만의 숫자를 입력했을 때 1이 출력되는 값을 수정

개선 코드 실행 결과

```
8! = 40320
Enter the number : 9
9! = 362880
Enter the number : 10
10! = 3628800
Enter the number : -2
계산할 수 없습니다. 숫자를 다시 입력해주세요
Enter the number : -3
계산할 수 없습니다. 숫자를 다시 입력해주세요
Enter the number : -4
계산할 수 없습니다. 숫자를 다시 입력해주세요
Enter the number : -5
계산할 수 없습니다. 숫자를 다시 입력해주세요
Enter the number : -6
계산할 수 없습니다. 숫자를 다시 입력해주세요
Enter the number : -1
>>> |
```