

while 문을 사용해 1 부터 1000 까지의 자연수 중 3 의 배수의 합을 출력.

```
N = 3
result = 0
while N < 1000:
    result += N
    N += 3

print(result)
```

While문을 이용하여 다음과 같이 별(*)을 출력

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

```
i = 1
while i <= 5:
    print('*' * i)
    i += 1
```

// 포문은 파이썬 강점을 이용한 예로 하나 들어줄게

```
[print('*' * i) for i in range(1, 6)]
```

for 문을 사용해 1 부터 100 까지의 숫자를 출력

```
[print(x) for x in range(1, 101)]
```

A 학급에 총 10 명의 학생이 있다. 이 학생들의 중간고사 점수는 다음과 같 다.
A = [70, 60, 55, 75, 95, 90, 80, 80, 85, 100]
반복문을 이용하여 A 학급의 평균 점수를 구하고, 평균 점수보다 위인 학생들과 평균 점수보다 아래인 학생들을 따로 출력.

```
A = [70, 60, 55, 75, 95, 90, 80, 80, 85, 100]

avg = sum(A) / len(A)

[print(x, end = ' ') for x in A if x > avg]
print()
[print(x, end = ' ') for x in A if x < avg]
```

주어진 자연수가 홀수인지 짝수인지 판별해 주는 함수(is_odd)를 작성하고 짝수이면 1 홀수면 0 을 출력.

```
def is_odd(N):
    if N % 2 == 0:
        print(1) // return 1
    else :
        print(0) // return 0
```

다음과 같은 문자열이 있다.

A:B:C:D

문자열의 split 과 join 함수를 사용하여 위 문자열은 A#B#C#D 로 고치시오

```
print('#'.join("A:B:C:D".split(':')))
```

피보나치 함수 구현

```
dp = [0, 1, 1]
def fibo(N):
    for i in range(3, N + 1):
        dp.append(dp[i - 2] + dp[i - 1])
```

```
N = int(input())
print(dp[N])
```

사용자로부터 2~9의 숫자 중 하나를 입력받아 해당 숫자의 구구단을 한 줄로 출력.

```
N = int(input())
[print(N, 'x', x, '=', N * x) for x in range(1, 10)]
```

계산기 함수 구현

```
def cal(a, b, op):
    if op == '+':
        return a+b
    elif op == '-':
        return a-b
    elif op == '*':
        return a*b
    else:
        return a/b
```

1~100까지의 소수 출력

```
import math

Check = [True] * 101
Check[0], Check[1] = False, False

for i in range(2, int(math.sqrt(100)) + 1):
    for j in range(i*2, 101, i):
        Check[j] = False

[print(x) for x in range(101) if Check[x] is True]
```

