Предположим, вам нужно написать программу для поиска наименьшего множителя, отличного от 1, для целого числа n (предположим, что n >= 2). Вы можете написать простой и интуитивно понятный код, используя оператор break следующим образом:

n = int(eval(input("Enter an integer >= 2: ")))

factor = 2

found = False

while factor <= n and not found:

if n % factor == 0:

found = True

else:

factor += 1

print(f"The smallest factor then 1 for {n} is {factor}\n\n")

n = int(eval(input("Enter an integer >= 2: ")))

factor = 2

while factor <= n:

if n % factor == 0:

break

factor += 1

print(f"The smallest factor then 1 for {n} is {factor}\n\n")

Вы можете переписать код без использования break следующим образом:

Очевидно, что оператор break делает программу в этом примере более простой и удобной для чтения. Тем не менее, вы должны использовать break и continue с осторожностью. Слишком большое количество операторов break и continue создаст цикл с множеством точек выхода и затруднит чтение программы.