

아래는 선형 탐색 알고리즘을 사용한 예시 코드입니다. 0번 인덱스의 원소부터, `len(some_list) - 1` 번째 원소까지 `element` 의 값과의 일치 여부를 확인합니다. 일치하면 해당 인덱스를 리턴해주고, 탐색이 끝났는데도 찾지 못하면 `None` 의 값을 리턴해줍니다.

```
def linear_search(element, some_list):
    for i in range(len(some_list)):
        if element == some_list[i]:
            return i
    return None

print(linear_search(2, [2, 3, 5, 7, 11]))
print(linear_search(0, [2, 3, 5, 7, 11]))
print(linear_search(5, [2, 3, 5, 7, 11]))
print(linear_search(3, [2, 3, 5, 7, 11]))
print(linear_search(11, [2, 3, 5, 7, 11]))
```

물론 while 반복문을 사용하여 문제를 풀 수 있습니다.

```
def linear_search(element, some_list):
    i = 0
    while i < len(some_list):
        if element == some_list[i]:
            return i
        i += 1
    return None

print(linear_search(2, [2, 3, 5, 7, 11]))
print(linear_search(0, [2, 3, 5, 7, 11]))
print(linear_search(5, [2, 3, 5, 7, 11]))
print(linear_search(3, [2, 3, 5, 7, 11]))
print(linear_search(11, [2, 3, 5, 7, 11]))
```

```
0
None
2
1
4
```



수업을 완료하셨으면 체크해주세요.

(/questions?

assignment\_id=402&sort\_by=popular)

(/questions/new?

assignment\_id=402&op1=%ED%94%84%EB%A1%9C%EA%B7%

< 이전 강의  
선형 탐색 (/assignments/401)

다음 강의  
이진 탐색 - 반복문 > (/assignments/403)