

```
1 def search_insert_position(nums, target):
2     # Initialize left and right pointers
3     left, right = 0, len(nums) - 1
4     # Perform binary search
5     while left <= right:
6         # Calculate the middle index
7         mid = (left + right) // 2
8         if nums[mid] == target:
9             return mid
10        elif nums[mid] < target:
11            left = mid + 1
12        else:
13            right = mid - 1
14    return left
15 # Example test cases
16 print(search_insert_position([1, 3, 5, 6], 5)) # Output: 2
17 print(search_insert_position([1, 3, 5, 6], 2)) # Output: 1
18 print(search_insert_position([1, 3, 5, 6], 7)) # Output: 4
19 print(search_insert_position([1, 3, 5, 6], 0)) # Output: 0
```

Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.27.0 -- An enhanced Interactive Python.

```
In [1]: runfile('C:/Users/jff/Desktop/spyder/Position Finder.py', wdir='C:/Users/jff/Desktop/spyder')
2
1
4
0
```

```
In [2]:
```