Algorithm: Linear Search

Key Aspects:
- Most intuitive search algorithm.
- Easy to implement in Python.
- Also called "Sequential Search".
- Could be used when the list contains only a few elements. When the list has many elements, the algorithm is very slow.
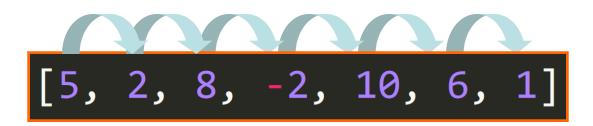
Index is returned

Algorithm:
- Traverse the list using a for loop.
- In every iteration, compare the current value of the list with the target value (the value that you are trying to find).
- If the current value of the list is equal to the target value, return the index of that value.
- Otherwise, if they are different, continue searching through the list.
- If the end of the list is reached without finding a match, the item is not in the list, so return -1.

Time Complexity:
- Worst-Case Time Complexity: O(n) (Linear).
- Average-Case Time Complexity: O(n) (Linear).
- Best-Case Time Complexity: O(1) (Constant).

```
[5, 2, 8, -2, 10, 6, 1]
```

Code:

```python
def linear_search(main_list, item):
    # Check every item in the list
    for i in range(len(main_list)):
        # If the item is the one you are looking for
        if main_list[i] == item:
            # Return the index where the item is located
            return i
    # If the item is not found, return -1
    return -1
```

Example:

```
[5, 2, 8, -2, 10, 6, 1]
```

```
>>> a = [5, 2, 8, -2, 10, 6, 1]
>>> linear_search(a, 10)
4
```

```
Starting Linear Search Algorithm
============ Iteration #0 ============
Current list item: 5
Item searched: 10
Is the current item equal to the item searched? False
This is not the item
============ Iteration #1 ============
Current list item: 2
Item searched: 10
Is the current item equal to the item searched? False
This is not the item
============ Iteration #2 ============
Current list item: 8
Item searched: 10
Is the current item equal to the item searched? False
This is not the item
============ Iteration #3 ============
Current list item: -2
Item searched: 10
Is the current item equal to the item searched? False
This is not the item
============ Iteration #4 ============
Current list item: 10
Item searched: 10
Is the current item equal to the item searched? True
Item found! at index 4
4
```