```
1. Write a program that asks the user to enter a list of at least five integers. Do the following:
(a) Print out the total number of items in the list.
(b) Print out the fourth item (index 3) in the list.
(c) Print out the last three items in the list.
(d) Print out all the items in the list except the first two.
(e) Print out the list in reverse order.
(f) Print out the largest and smallest values in the list.
(g) Print out the sum of all the values in the list.
(h) If the list contains a zero, print out the index of the first zero in the list, and otherwise print out a message
saying there are no zeroes.
(i) Sort the list and print out the list after sorting.
(j) Delete the first item from the (now sorted) list and print out the new list.
(k) Change the second-to-last item in the list to 9876 and print out the new list.
(l) Append the value -500 to the end of the list and print out the new list.
```

In [1]:
```
# (a) Prompt the user to enter a list of integers
integer_list = input("Enter a list of at least five integers, separated by spaces: ").split()
integer_list = [int(num) for num in integer_list]

# (a) Print out the total number of items in the list
print("Total number of items in the list:", len(integer_list))

# (b) Print out the fourth item (index 3) in the list
print("Fourth item in the list:", integer_list[3])

# (c) Print out the last three items in the list
print("Last three items in the list:", integer_list[-3:])

# (d) Print out all the items in the list except the first two
print("Items in the list except the first two:", integer_list[2:])

# (e) Print out the list in reverse order
print("List in reverse order:", integer_list[::-1])

# (f) Print out the largest and smallest values in the list
print("Largest value in the list:", max(integer_list))
print("Smallest value in the list:", min(integer_list))

# (g) Print out the sum of all the values in the list
print("Sum of all values in the list:", sum(integer_list))

# (h) If the list contains a zero, print out the index of the first zero in the list,
#     and otherwise print out a message saying there are no zeroes.
if 0 in integer_list:
    print("Index of the first zero in the list:", integer_list.index(0))
else:
    print("There are no zeroes in the list.")

# (i) Sort the list and print out the list after sorting
sorted_list = sorted(integer_list)
print("List after sorting:", sorted_list)

# (j) Delete the first item from the (now sorted) list and print out the new list
del sorted_list[0]
print("List after deleting the first item:", sorted_list)

# (k) Change the second-to-last item in the list to 9876 and print out the new list
sorted_list[-2] = 9876
print("List after changing the second-to-last item:", sorted_list)

# (l) Append the value -500 to the end of the list and print out the new list
sorted_list.append(-500)
print("List after appending -500:", sorted_list)
```

```
Enter a list of at least five integers, separated by spaces: 6 2 4 5 7
Total number of items in the list: 5
Fourth item in the list: 5
Last three items in the list: [4, 5, 7]
Items in the list except the first two: [4, 5, 7]
List in reverse order: [7, 5, 4, 2, 6]
Largest value in the list: 7
Smallest value in the list: 2
Sum of all values in the list: 24
There are no zeroes in the list.
List after sorting: [2, 4, 5, 6, 7]
List after deleting the first item: [4, 5, 6, 7]
List after changing the second-to-last item: [4, 5, 9876, 7]
List after appending -500: [4, 5, 9876, 7, -500]
```

2. Write a program that asks the user to enter a list of numbers. Then print out the smallest thing in the list and the first index at which it appears in the list.

In [2]:
```
numbers = input("Enter a list of numbers, separated by spaces: ").split()
numbers = [float(num) for num in numbers]

smallest = min(numbers)
index = numbers.index(smallest)

print("Smallest number:", smallest)
print("First index of smallest number:", index)
```

```
Enter a list of numbers, separated by spaces: 5.2 3.1 6.7 9.8 1.1
Smallest number: 1.1
First index of smallest number: 4
```

3. Write a program that asks the user to enter a string of lowercase letters and creates a list containing counts of how many times each letter appears in the string. The first index is how many a's are in the string, the second is how many b's, etc.

In [10]:
```
def count_letters(string):
    counts = [0] * 26  # Initialize the list with 26 zeros for each lowercase letter

    for char in string:
        if 'a' <= char <= 'z':  # Check if the character is a lowercase letter
            index = ord(char) - ord('a')  # Calculate the index based on ASCII values
            counts[index] += 1

    return counts


# Ask the user to enter a string
user_input = input("Enter a string of lowercase letters: ")

# Call the count_letters function and print the result
letter_counts = count_letters(user_input)

# Print the letter counts
for i in range(26):
    letter = chr(ord('a') + i)
    count = letter_counts[i]
    print(f"{letter}: {count}")
```

```
Enter a string of lowercase letters: sivarama krishna reddy
a: 4
b: 0
c: 0
d: 2
e: 1
f: 0
g: 0
h: 1
i: 2
j: 0
k: 1
l: 0
m: 1
n: 1
o: 0
p: 0
q: 0
r: 3
s: 2
t: 0
u: 0
v: 1
w: 0
x: 0
y: 1
z: 0
```

4. Create a dictionary whose keys are the strings 'abc', 'def', 'ghi', 'jkl', and 'mno' and whose corresponding values are 7, 11, 13, 17, and 19. Then write dictionary code that does the following:
(a) Print the value in the dictionary associated with the key 'def'.
(b) Use the keys() method to print out all the keys.
(c) Loop over the dictionary and print out all the keys and their associated values.
(d) Use an if statement to check if the dictionary contains the key 'pqr' and print out an appropriate message indicating whether it does or doesn't.
(e) Change the value associated with the key 'abc' to 23 and then print out all the values in the dictionary using the values() method.

In [11]:
```
# Create the dictionary
my_dict = {'abc': 7, 'def': 11, 'ghi': 13, 'jkl': 17, 'mno': 19}

# (a) Print the value associated with the key 'def'
print("Value associated with 'def':", my_dict['def'])

# (b) Print all the keys
print("All keys:", my_dict.keys())

# (c) Print all keys and their associated values
print("Keys and values:")
for key, value in my_dict.items():
    print(f"Key: {key}, Value: {value}")

# (d) Check if 'pqr' is a key in the dictionary
if 'pqr' in my_dict:
    print("'pqr' is a key in the dictionary.")
else:
    print("'pqr' is not a key in the dictionary.")

# (e) Change the value associated with 'abc' to 23
my_dict['abc'] = 23

# Print all values in the dictionary
print("All values:", my_dict.values())
```

```
Value associated with 'def': 11
All keys: dict_keys(['abc', 'def', 'ghi', 'jkl', 'mno'])
Keys and values:
Key: abc, Value: 7
Key: def, Value: 11
Key: ghi, Value: 13
Key: jkl, Value: 17
Key: mno, Value: 19
'pqr' is not a key in the dictionary.
All values: dict_values([23, 11, 13, 17, 19])
```

In [ ]: