1. Write a Python program to find the second smallest number in a list.

input

second_smallest([1, 2, -8, -2, 0])

output

-2

In [1]:
```
def second_smallest(lst):
    if len(lst) < 2:
        return None

    smallest = min(lst)
    second_smallest = float('inf')

    for num in lst:
        if num > smallest and num < second_smallest:
            second_smallest = num

    return second_smallest
```

In [2]:
```
numbers = [1, 2, -8, -2, 0]
result = second_smallest(numbers)
print(result)
```

```
-2
```

2. Write a Python program to change a given string to a new string where the first and last chars have been exchanged

In [3]:
```
def exchange_first_last_chars(string):
    if len(string) < 2:
        return string

    first_char = string[0]
    last_char = string[-1]
    middle_chars = string[1:-1]
```

```
    new_string = last_char + middle_chars + first_char
    return new_string
```

In [4]:
```
given_string = "Hello, World!"
new_string = exchange_first_last_chars(given_string)
print(new_string)
```

```
!ello, WorldH
```

3. Write a Python function that takes a list of words and returns the length of the longest one

In [5]:
```
def find_longest_word_length(words):
    longest_length = 0

    for word in words:
        current_length = len(word)
        if current_length > longest_length:
            longest_length = current_length

    return longest_length
```

In [6]:
```
word_list = ["apple", "banana", "cherry", "dragonfruit", "elderberry"]
longest_length = find_longest_word_length(word_list)
print(longest_length)
```

```
11
```

4. Write a Python program to remove the nth index character from a nonempty string

In [7]:
```
def remove_nth_character(string, n):
    if n < 0 or n >= len(string):
        return string

    new_string = string[:n] + string[n+1:]
    return new_string
```

In [8]:
```
given_string = "Hello, World!"
index_to_remove = 6
new_string = remove_nth_character(given_string, index_to_remove)
print(new_string)
```

```
Hello,World!
```

5. Check if a given key already exists in a dictionary

input

d = {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

is_key_present(5)

is_key_present(9)

output

Key is present in the dictionary

Key is not present in the dictionary

In [9]:
```
def is_key_present(dictionary, key):
    if key in dictionary:
        return "Key is present in the dictionary"
    else:
        return "Key is not present in the dictionary"
```

In [10]:
```
d = {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
result1 = is_key_present(d, 5)
result2 = is_key_present(d, 9)
print(result1)
print(result2)
```

```
Key is present in the dictionary
Key is not present in the dictionary
```

In [ ]:
```

```