1. .finding maximum and minimum elements of two

numbers

Code:

```python
num1 = int(input("enter the number: "))
num2 = int(input("enter the number: "))
if num1 > num2:
    print("num1 is maximum and num2 is minimum")
else:
    print("num2 is maximum and num1 is minimum")
```

Output:

```
enter the number: 5
enter the number: 2
num1 is maximum and num2 is minimum
```

2. write a program to swap two numbers in a list.

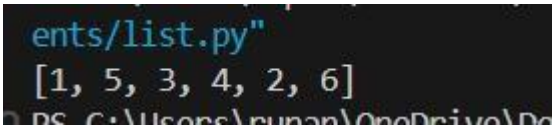
Code :

```python
my_list = [1,2,3,4,5,6]

# Swap the element at index 2 and the element at index 4
temp = my_list[1]
my_list[1] = my_list[4]
my_list[4] = temp

print(my_list)
```

Output:



3. write a program to find the length of the string in 4 ways

Code:

length.py > ...

```python
string = "python assignment!"

# Method 1: Using len() function
length_method1 = len(string)
print(len(string))

# Method 2: Iterating through the string manually
length_method2 = 0
for char in string:
    length_method2 += 1
print(length_method2)

# Method 3: Using a while loop to count characters
length_method3 = 0
while True:
    try:
        char = string[length_method3]
        length_method3 += 1
    except IndexError:
        break
print(length_method3)

# Method 4: Using recursion
def string_length(s):
    if s == "":
        return 0
    else:
        return 1 + string_length(s[1:])

length_method4 = string_length(string)
print(length_method4)
```

Output:

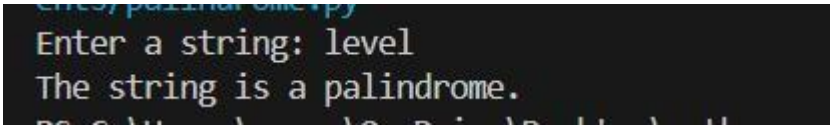
4. write a program to whether the string is palindrome or not

Code:

```python
palindrome.py > ...
def is_palindrome(s):
    # Convert the string to lowercase for case-insensitive comparison
    s = s.lower()
    # Reverse the string using slicing
    reversed_s = s[::-1]
    # Compare the original and reversed strings
    if s == reversed_s:
        return True
    else:
        return False
# Take the input from the user
s = input("Enter a string: ")

# Call the function and print the result
result = is_palindrome(s)
if result:
    print("The string is a palindrome.")
else:
    print("The string is not a palindrome.")
```

Output:



5. write a program to reverse the words in a given

string

Code:

```
reven
1   name=input("enter a string : ")
2   rev=name[::-1]
3   print(rev)
```

Output :

```
ents/reverse.py
enter a string : good morning
gninrom doog
```

6. write a program to remove the ith character in a given string

Code:

```
remove string.py > ...
1   string = input("enter the string: ")
2   i = int(input("enter the index to remove: "))
3   new_word = string[:i] + string[i+1:]
4   print(new_word)
```

Output:

```
enter the string: flowers
enter the index to remove: 4
flowrs
```

7. write a program to check whether the given number is

prime or not

Code:

```python
def is_prime(x):
    if x < 2:
        return False
    elif x == 2:
        return True
    else:
        for n in range(2, int(x**0.5) + 1):
            if x % n == 0:
                return False
        return True
print(is_prime(2))
print(is_prime(5))
```

Output:

```
ents/prime.py"
True
True
```

8. write a program to perform arithmetic operations such as addition,multiplication on complex numbers

Code:

```python
def complex_operations():

    complex1 = 5 + 9j
    complex2 = 7 - 3j

    sum_complex = complex1 + complex2
    print(f"Sum: {complex1} + {co
    (variable) complex2: complex

    product_complex = complex1 * complex2
    print(f"Product: {complex1} * {complex2} = {product_complex}")

# Calling the function to perform complex number operations
complex_operations()
```

Output:

```
ents/operations.py"
Sum: (5+9j) + (7-3j) = (12+6j)
Product: (5+9j) * (7-3j) = (62+48j)
```