Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```
program to rotate the list to given index.Eg: list['a', 'b', 'c', 'd', 'e']Index = 2rotated
= ['a', 'b', 'c', 'd', 'e']
_index = 2

ion_index >= len(my_list):
t("Index out of range")

ted_list = my_list[rotation_index:] + my_list[:rotation_index]
t(rotated_list)

```

input

```
['c', 'd', 'e', 'a', 'b']


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
# A list contains both positive and negative numbers. sort thelist by ignoring sign.Actual l
my_list = [9, -4, 2, 0, -2, 7, -3, 5, -5]
sorted_list = sorted(my_list, key=lambda x: abs(x))
print(sorted_list)

```

input

```
[0, 2, -2, -3, -4, 5, -5, 7, 9]


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | {} Beautify | Language Python 3

main.py

```python
# Write a function that take a sentence and returns thesentence by shuffling the words.
import random

def shuffle_sentence(sentence):
    words = sentence.split()
    random.shuffle(words)
    shuffled_sentence = ' '.join(words)
    return shuffled_sentence
sentence = "This is a sample sentence to demonstrate shuffling words."
shuffled_sentence = shuffle_sentence(sentence)
print(shuffled_sentence)

```

input

```
a to shuffling is This demonstrate sample words. sentence


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
# Write a function to return LCM of two numbers
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a

def lcm(a, b):
    return (a * b) // gcd(a, b)

num1 = 1245
num2 = 1807
result = lcm(num1, num2)
print("LCM of", num1, "and", num2, "is:", result)

```

input

```
LCM of 1245 and 1807 is: 2249715


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
# A list contains words. Write a program to delete the words form list if it contains more
def count_vowels(word):
    vowels = "aeiouAEIOU"
    count = 0
    for char in word:
        if char in vowels:
            count += 1
    return count
def filter_words(words):
    filtered_words = [word for word in words if count_vowels(word) <= 2]
    return filtered_words
word_list = ["apple", "banana", "orange", "watermelon", "kiwi"]
filtered_list = filter_words(word_list)
print(filtered_list)

```

input

```
['apple', 'kiwi']


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
#Write a function to take a n-digits number and check whether it is dominated by even numbe
def digit_dominance(number):
    even_count = 0
    odd_count = 0

    while number > 0:
        digit = number % 10
        if digit % 2 == 0:
            even_count += 1
        else:
            odd_count += 1
        number //= 10

    if even_count > odd_count:
        return "Dominated by even numbers"
    elif odd_count > even_count:
        return "Dominated by odd numbers"
    else:
        return "Balanced"

number = 1234
result = digit_dominance(number)
print(f"{number} is {result}")

number = 2234
result = digit_dominance(number)
print(f"{number} is {result}")

number = 1111
result = digit_dominance(number)
print(f"{number} is {result}")
```

input

```
1234 is Balanced
2234 is Dominated by even numbers
1111 is Dominated by odd numbers


...Program finished with exit code 0
Press ENTER to exit console.
```

Run Debug Stop Share Save { } Beautify Language Python 3

main.py

```python
#write a program to convert a column matrix to row matrixcol_matrix: [1, 2, 3, 4]row_matrix:
def column_to_row(column_matrix):
    row_matrix = [[element] for element in column_matrix]
    return row_matrix
column_matrix = [1, 2, 3, 4]
row_matrix = column_to_row(column_matrix)
print(row_matrix)

```

input

```
[[1], [2], [3], [4]]


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
# write a program to calculate sum of prime numbers within agiven range.
def is_prime(num):
    if num <= 1:
        return False
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            return False
    return True

def sum_of_primes_in_range(start, end):
    prime_sum = 0
    for num in range(start, end + 1):
        if is_prime(num):
            prime_sum += num
    return prime_sum

start_range = 10
end_range = 50
result = sum_of_primes_in_range(start_range, end_range)
print("Sum of prime numbers between", start_range, "and", end_range, "is:", result)
```

input

```
Sum of prime numbers between 10 and 50 is: 311


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
# write a program to add tuple to list
my_list = [(1, 2), (3, 4), (5, 6)]
new_tuple = (7, 8)

my_list.append(new_tuple)

print(my_list)
```

input

```
[(1, 2), (3, 4), (5, 6), (7, 8)]


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

main.py

```python
# write a program to import math module and use• factorial function• gcd function calcula
def factorial(n):
    result = 1
    for i in range(1, n + 1):
        result *= i
    return result

def gcd(a, b):
    while b:
        a, b = b, a % b
    return a

def area_of_circle(radius):
    return 3.14 * radius**2

num = 5
factorial_result = factorial(num)
print("Factorial of", num, "is:", factorial_result)

num1 = 36
num2 = 48
gcd_result = gcd(num1, num2)
print("GCD of", num1, "and", num2, "is:", gcd_result)

radius = 5
area = area_of_circle(radius)
print("Area of circle with radius", radius, "is:", area)

```

input

```
Factorial of 5 is: 120
GCD of 36 and 48 is: 12
Area of circle with radius 5 is: 78.5


...Program finished with exit code 0
Press ENTER to exit console.
```