main.py

Share Run

```python
# Initialize counters and sums
sum_positive = 0
count_positive = 0
sum_negative = 0
count_negative = 0

print("Enter -1 to exit, enter the numbers:")

while True:
    num = int(input())

    if num == -1:
        break

    if num > 0:
        sum_positive += num
        count_positive += 1
    elif num < 0:
        sum_negative += num
        count_negative += 1

# Calculate averages
if count_positive > 0:
    avg_positive = sum_positive // count_positive
```

Output

Clear

```
Enter -1 to exit, enter the numbers:
7
-1
avg negative number is 0, avg positive number is 7

=== Code Execution Successful ===
```

```python
positive_numbers = []
negative_numbers = []

while True:
    num = int(input("Enter number (-1 to exit): "))
    if num == -1:
        break
    if num > 0:
        positive_numbers.append(num)
    elif num < 0:
        negative_numbers.append(num)

avg_pos = sum(positive_numbers) / len(positive_numbers) if positive_numbers else 0
avg_neg = sum(negative_numbers) / len(negative_numbers) if negative_numbers else 0

print(f"avg negative number is {int(avg_neg)}, avg positive number is {int(avg_pos)}")

```

```
Enter number (-1 to exit): 7
Enter number (-1 to exit): 5
Enter number (-1 to exit): -2
Enter number (-1 to exit): 8
Enter number (-1 to exit): -5
Enter number (-1 to exit): 1
Enter number (-1 to exit): -1
avg negative number is -3, avg positive number is 5

=== Code Execution Successful ===
```

```python
num = float(input("Given Number: "))
square = num ** 2
cube = num ** 3

print(f"Square Number: {square}")
print(f"Cube Number: {cube}")

```

```
Given Number: 10
Square Number: 100.0
Cube Number: 1000.0

=== Code Execution Successful ===
```

```python
num = float(input("Given Number: "))
square = num ** 2
cube = num ** 3

print(f"Square Number: {square}")
print(f"Cube Number: {cube}")

```

```
Given Number: 10
Square Number: 100.0
Cube Number: 1000.0

=== Code Execution Successful ===
```

```python
A = int(input("A= "))
B = int(input("B= "))

for i in range(1, B + 1):
    print(f"{A} x {i} = {A*i}")


```

```
A= 5
B= 10
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50

=== Code Execution Successful ===
```

```python
year = int(input("Enter year: "))

if (year % 400 == 0) or (year % 4 == 0 and year % 100 != 0):
    print("Leap Year")
else:
    print("Not Leap Year")

```

```
Enter year: 2008
Leap Year

=== Code Execution Successful ===
```

```python
arr = [1, 2, 3, 4, 1]
unique_arr = list(dict.fromkeys(arr))  # preserves order and removes duplicates
print(f"duplicate array={unique_arr}")

```

```
duplicate array=[1, 2, 3, 4]

=== Code Execution Successful ===
```

```python
from statistics import mean, median, mode

data = [12, 45, 83, 52]

avg_value = (mean(data) + median(data) + mode(data)) / 3
print(f"Output: {int(avg_value)}")

```

```
Output: 36

=== Code Execution Successful ===
```

```python
arr = [1, 8, 3, 4, 0]
arr.sort(reverse=True)
print(arr)

```

```
[8, 4, 3, 1, 0]

=== Code Execution Successful ===
```

```python
set1 = set([2, 3, 4, 5])
set2 = set([3, 4, 8, 6])

intersection = tuple(set1.intersection(set2))
print(intersection)

```

```
(3, 4)

=== Code Execution Successful ===
```