

is true value, then S₁ is
control moves to S₂.
a false, then S₂ is ex-
ecuted, control moves to S₃.
of statement =

level 1 questions
Practice programmes
1) positive_sum = 0
positive_count = 0
negative_sum = 0
negative_count = 0
print("Enter -1 to exit")
while True:
 num = int(input("Enter a number: "))
 if num == -1:
 break
 if num > 0:
 positive_sum += num
 positive_count += 1
 elif num < 0:
 negative_sum += num
 negative_count += 1

chain of if-

if-else



```
if positive_count > 0:  
    avg_positive = positive_sum / positive_count  
    print(f"avg positive number is {int(avg_positive)}")
```

```
else:  
    print("No Positive numbers entered.")
```

```
if negative_count > 0:  
    avg_negative = negative_sum / negative_count  
    print(f"avg negative number is {int(avg_negative)}")
```

```
else:  
    print("No negative numbers entered.")
```

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

```
cube = num**3
```

```
print(f"square Numbers: {square}")
```

```
print(f"cube Number: {cube}")
```

output:

Given number: 0.6

square number: 0.36

cube number: 0.216

```
3) char = input("Enter the character to be  
    printed: ")
```

```
rows = int(input("Number of rows: "))
```


#print the pattern

```
for i in range(1, rows + 1):
```

```
    print((char + " ") * i)
```

output:

enter the character to be printed: a

Number of rows: 5

```
a
a a
a a a
a a a a
a a a a a
```

4.0

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

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

#Print the multiplication table

```
for i in range(1, B + 1):
```

```
    print(f"{A} x {i} = {A * i}")
```

output:

```
A = 7
```

```
B = 5
```

```
7 x 1 = 7
```

```
7 x 2 = 14
```

```
7 x 3 = 21
```

```
7 x 4 = 28
```

```
7 x 5 = 35
```

```
5.) year = int(input("enter a year:"))  
#check leap year conditions  
if (year % 4 == 0 and year % 100 != 0) or  
    (year % 400 == 0):  
    print("leap year")
```

```
else:  
    print("Not a leap year")
```

output:

enter a year: 2000

leap year.

```
6.) array = [1, 2, 3, 4, 1]
```

```
unique_array = list(set(array))
```

```
unique_array.sort()
```

```
print(f"duplicate array = {unique_array}")
```

output:

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

```
7.) num = float(input("enter a number:"))
```

```
if num > 0:
```

```
    print("positive")
```

```
elif num < 0:
```

```
    print("negative")
```

```
else:
```

```
    print("zero")
```


Output:

enter a number = 23
Positive.

```
8.) import statistics
numbers = [12, 45, 83, 52]
mean_val = statistics.mean(numbers)
median_val = statistics.median(numbers)
mode_val = statistics.mode(numbers)
step except statistics.StatisticsError:
    mode_val = 0
average = (mean_val + median_val + mode_val) / 3
print(f"output: {int(average)}")
```

output: 48.

```
9.) array = [1, 8, 3, 4, 0]
sorted_array = sorted(array, reverse = True)
print(f"output: {sorted_array}")
output: [8, 4, 3, 1, 0]
```

```
10.) tuple 1 = (2, 3, 4, 5)
tuple 2 = (3, 4, 8, 6)
intersection = tuple(set(tuple 1) & set(tuple 2))
intersection = tuple(sorted(intersection))
print(f"output: {intersection}")
output: (3, 4)
```


Output:

enter a number = 23
positive.

```
8.) import statistics
numbers = [12, 45, 88, 52]
mean_val = statistics.mean(numbers)
median_val = statistics.median(numbers)
mode_val = statistics.mode(numbers)
except statistics.StatisticsError:
    mode_val = 0
average = (mean_val + median_val + mode_val) / 3
print(f"output: {int(average)}")
```

output: 48.

```
9.) array = [1, 8, 3, 4, 0]
sorted_array = sorted(array, reverse = True)
print(f"output: {sorted_array}")
output: [8, 4, 3, 1, 0]
```

```
10.) tuple 1 = (2, 3, 4, 5)
tuple 2 = (3, 4, 8, 6)
intersection = tuple(set(tuple 1) & set(tuple 2))
intersection = tuple(sorted(intersection))
print(f"output: {intersection}")
output: (3, 4)
```