1. ARITHMETIC OPERATORS

PROGRAM:

```python
a = 46
b = 4

print("For a =", a, "and b =", b,"\nCalculate the following:")

print('1. Addition of two numbers: a + b =', a + b)
print('2. Subtraction of two numbers: a - b =', a - b)
print('3. Multiplication of two numbers: a * b =', a * b)
print('4. Division of two numbers: a / b =', a / b)
print('5. Floor division of two numbers: a // b =',a // b)
print('6. Reminder of two numbers: a mod b =', a % b)
print('7. Exponent of two numbers: a ^ b =',a ** b)
```

OUTPUT:

```
For a = 46 and b = 4
Calculate the following:
1. Addition of two numbers: a + b = 50
2. Subtraction of two numbers: a - b = 42
3. Multiplication of two numbers: a * b = 184
4. Division of two numbers: a / b = 11.5
5. Floor division of two numbers: a // b = 11
6. Reminder of two numbers: a mod b = 2
7. Exponent of two numbers: a ^ b = 4477456
```

2. COMPARISON OPERATORS

PROGRAM:

```python
a = 46     # Initializing the value of a
b = 4      # Initializing the value of b

print("For a =", a, "and b =", b,"\nCheck the following:")

print('1. Two numbers are equal or not:', a == b)
print('2. Two numbers are not equal or not:', a != b)
print('3. a is less than or equal to b:', a <= b)
print('4. a is greater than or equal to b:', a >= b)
print('5. a is greater b:', a > b)
print('6. a is less than b:', a < b)
```

OUTPUT:

```
For a = 46 and b = 4
Check the following:
1. Two numbers are equal or not: False
2. Two numbers are not equal or not: True
3. a is less than or equal to b: False
4. a is greater than or equal to b: True
5. a is greater b: True
6. a is less than b: False
```

3. ASSIGNMENT OPERATORS

PROGRAM:

```python
a = 34
b=6
print('a += b:', a + b)
print('a -= b:', a - b)
print('a *= b:', a * b)
print('a /= b:', a / b)
print('a %= b:', a % b)
print('a **= b:', a ** b)
print('a //= b:', a // b)
```

OUTPUT:

```
a += b: 40
a -= b: 28
a *= b: 204
a /= b: 5.666666666666667
a %= b: 4
a **= b: 1544804416
a //= b: 5
```

4. LOGICAL OPERATORS

PROGRAM:

```python
a = 7
print("For a = 7, checking whether the following conditions are True or False:")
print('\"a > 5 and a < 7\" =>', a > 5 and a < 7)
print('\"a > 5 or a < 7\" =>', a > 5 or a < 7)
print('\"not (a > 5 and a < 7)\" =>', not(a > 5 and a < 7))
```

OUTPUT:

```
For a = 7, checking whether the following conditions are True or False:
"a > 5 and a < 7" => False
"a > 5 or a < 7" => True
"not (a > 5 and a < 7)" => True
```

5. BITWISE OPERATORS

PROGRAM:

```python
a = 7
b = 8
print('a & b :', a & b)
print('a | b :', a | b)
print('a ^ b :', a ^ b)
print('~a :', ~a)
print('a << b :', a << b)
print('a >> b :', a >> b)
```

OUTPUT:

```
a & b : 0
a | b : 15
a ^ b : 15
~a : -8
a << b : 1792
a >> b : 0
```

6. MEMBERSHIP OPERATORS

PROGRAM:

```python
myList = [12, 22, 28, 35, 42, 49, 54, 65, 92, 103, 245, 874]
x = 31
y = 28
print("Given List:", myList)

if (x not in myList):
    print("x =", x,"is NOT present in the given list.")
else:
    print("x =", x,"is present in the given list.")

if (y in myList):
    print("y =", y,"is present in the given list.")
else:
    print("y =", y,"is NOT present in the given list.")
```

OUTPUT:

```
Given List: [12, 22, 28, 35, 42, 49, 54, 65, 92, 103, 245, 874]
x = 31 is NOT present in the given list.
y = 28 is present in the given list.
```

7. IDENTITY OPERATORS

PROGRAM:

```
a = ["Rose", "Lotus"]
b = ["Rose", "Lotus"]

c = a

print("a is c => ", a is c)
print("a is not c => ", a is not c)
print("a is b => ", a is b)
print("a is not b => ", a is not b)
print("a == b => ", a == b)
print("a != b => ", a != b)
```

OUTPUT:

```
a is c =>  True
a is not c =>  False
a is b =>  False
a is not b =>  True
a == b =>  True
a != b =>  False
```

REVERSING STRING

USING FOR LOOP

PROGRAM:

```
def reverse_string(str):
    str1 = ""
    for i in str:
        str1 = i + str1
    return str1

str = "JavaTpoint"
print("The original string is: ",str)
print("The reverse string is",reverse_string(str))
```

OUTPUT:

```
The original string is:  JavaTpoint
The reverse string is tniopTavaJ
```

USING WHILE LOOP

PROGRAM:

```
str = "JavaTpoint"
print ("The original string  is : ",str)
reverse_String = ""
count = len(str)
while count > 0:
    reverse_String += str[ count - 1 ]
    count = count - 1
print ("The reversed string using a while loop is : ",reverse_String)
```

OUTPUT:

```
The original string  is :  JavaTpoint
The reversed string using a while loop is :  tniopTavaJ
```

USING THE SLICE OPERATOR

PROGRAM:

```
def reverse(str):
    str = str[::-1]
    return str

s = "JavaTpoint"
print ("The original string  is : ",s)
print ("The reversed string using extended slice operator  is : ",reverse(s))
```

OUTPUT:

```
The original string  is :  JavaTpoint
The reversed string using extended slice operator  is :  tniopTavaJ
```

USING THE REVERSED() FUNCTION

PROGRAM:

```
def reverse(str):
    string = "".join(reversed(str))
    return string

s = "JavaTpoint"

print ("The original string is : ",s)
print ("The reversed string using reversed() is : ",reverse(s) )
```

OUTPUT:

```
The original string is :  JavaTpoint
The reversed string using reversed() is :  tniopTavaJ
```

USING THE RECURSION

PROGRAM:

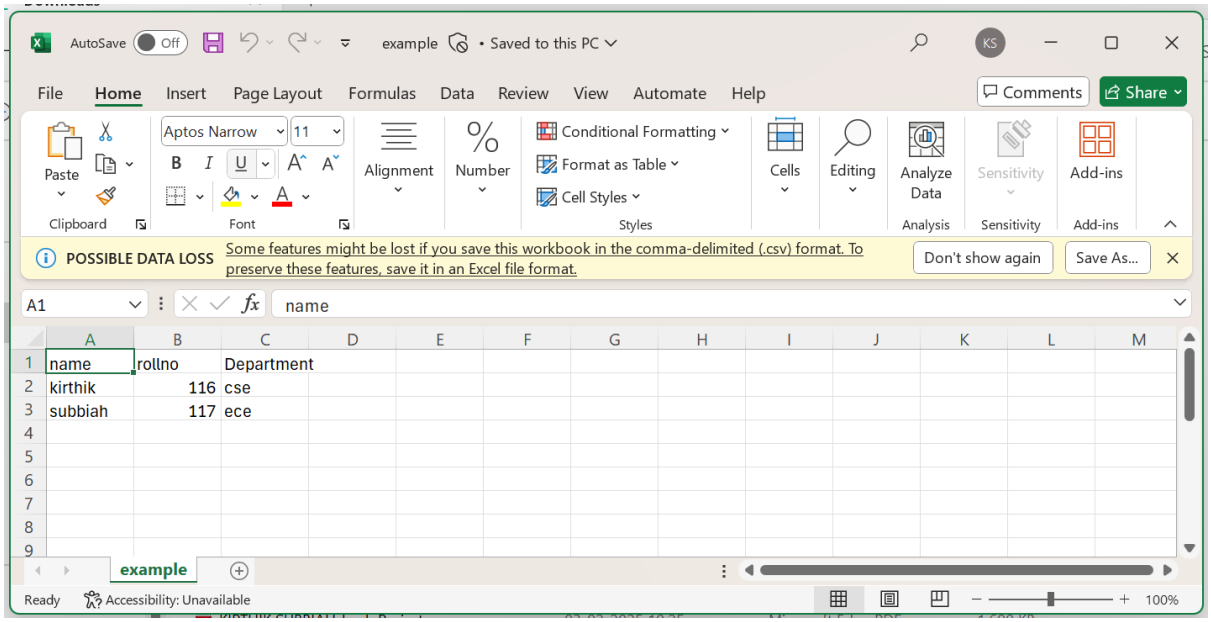
```
def reverse(str):
    if len(str) == 0:
        return str
    else:
        return reverse(str[1:]) + str[0]

str = "Devansh Sharma"
print ("The original string  is : ", str)
print ("The reversed string(using recursion) is : ", reverse(str))
```

OUTPUT:

```
The original string  is :  Devansh Sharma
The reversed string(using recursion) is :  amrahS hsnaveD
```

READ CSV FILE IN PYTHON:

CSV FILE:



PROGRAM:

```
> Users > 289250 > Downloads > csvfile.py > ...
import csv

with open(r'C:\Users\289250\Downloads\example.csv') as csv_file:
    csv_read = csv.reader(csv_file, delimiter=',')
    count_line = 0

    for row in csv_read:
        if count_line == 0:
            print(f'Column names are {", ".join(row)}')
        else:
            print(f'\t{row[0]} roll number is: {row[1]} and department is: {row[2]}.')
        count_line += 1
print(f'Processed {count_line} lines.')
```

OUTPUT:

```
PS C:\Users\289250> & C:/Users/289250/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Users/289250/Downloa
ds/csvfile.py
Column names are ï»¿name, rollno, Department
        kirthik roll number is: 116 and department is: cse.
        subbiah roll number is: 117 and department is: ece.
Processed 3 lines.
PS C:\Users\289250>
```

The if statement

PROGRAM:

main.py

```python
num = int(input("enter the number:"))
if num%2 == 0:
    print("The Given number is an even number")
```

OUTPUT:

```
enter the number:4
The Given number is an even number
```

Program to print the largest of the three number

PROGRAM:

```
a = int (input("Enter a: "));
b = int (input("Enter b: "));
c = int (input("Enter c: "));
if a>b and a>c:
    print ("From the above three numbers given a is largest");

if b>a and b>c:
    print ("From the above three numbers given b is largest");

if c>a and c>b:
    print ("From the above three numbers given c is largest");
```

OUTPUT:

```
Enter a: 3
Enter b: 4
Enter c: 5
From the above three numbers given c is largest
```

Program to check whether a person is eligible to vote or not

PROGRAM:

```
age = int (input("Enter your age: "))
if age>=18:
    print("You are eligible to vote !!");
else:
    print("Sorry! you have to wait !!");
```

OUTPUT:

```
Enter your age: 16
Sorry! you have to wait !!


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

Program to check whether a number is even or not.

PROGRAM:

```
num = int(input("enter the number:"))
if num%2 == 0:
    print("The Given number is an even number")
else:
    print("The Given Number is an odd number")
```

OUTPUT:

```
enter the number:197
The Given Number is an odd number


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

Simple Python program to understand elif statement

PROGRAM:

```python
number = int(input("Enter the number?"))
if number==10:
    print("The given number is equals to 10")
elif number==50:
    print("The given number is equal to 50");
elif number==100:
    print("The given number is equal to 100");
else:
    print("The given number is not equal to 10, 50 or 100");
```

OUTPUT:

```
Enter the number?12
The given number is not equal to 10, 50 or 100


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

Simple Python program to understand elif statement

PROGRAM:

```python
marks = int(input("Enter the marks? "))
if marks > 85 and marks <= 100:
    print("Congrats ! you scored grade A ...")
elif marks > 60 and marks <= 85:
    print("You scored grade B + ...")
elif marks > 40 and marks <= 60:
    print("You scored grade B ...")
elif (marks > 30 and marks <= 40):
    print("You scored grade C ...")
else:
    print("Sorry you are fail ?")
```

OUTPUT:

```
Enter the marks? 76
You scored grade B + ...


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

Python program to show how the for loop works

PROGRAM:

```
main.py
numbers = [4, 2, 6, 7, 3, 5, 8, 10, 6, 1, 9, 2]
square = 0
squares = []
for value in numbers:
    square = value ** 2
    squares.append(square)
print("The list of squares is", squares)
```

OUTPUT:

```
input
The list of squares is [16, 4, 36, 49, 9, 25, 64, 100, 36, 1, 81, 4]


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

Python program to show how if-else statements work

PROGRAM:

```python
string = "Python Loop"
for s in string:
    if s == "o":
        print("If block")
    else:
        print(s)

```

OUTPUT:

```
P
y
t
h
If block
n

L
If block
If block
p
```

Python program to show how to use else statement with for loop

PROGRAM:

```python
tuple_ = (3, 4, 6, 8, 9, 2, 3, 8, 9, 7)
for value in tuple_:
    if value % 2 != 0:
        print(value)
else:
    print("These are the odd numbers present in the tuple")
```

OUTPUT:

```
3
9
3
9
7
These are the odd numbers present in the tuple
```

Python program to show the working of range() function

PROGRAM:

```
main.py
1 print(range(15))
2 print(list(range(15)))
3 print(list(range(4, 9)))
4 print(list(range(5, 25, 4)))
```

OUTPUT:

```
range(0, 15)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
[4, 5, 6, 7, 8]
[5, 9, 13, 17, 21]


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

Python program to iterate over a sequence with the help of indexing

PROGRAM:

```python
tuple_ = ("Python", "Loops", "Sequence", "Condition", "Range")
for iterator in range(len(tuple_)):
    print(tuple_[iterator].upper())
```

OUTPUT:

```
PYTHON
LOOPS
SEQUENCE
CONDITION
RANGE


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

PROGRAM:

main.py

```python
counter = 0
while counter < 10:
    counter = counter + 3
    print("Python Loops")
```

OUTPUT:

```
Python Loops
Python Loops
Python Loops
Python Loops


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

PROGRAM:

```python
counter = 0
while (counter < 10):
    counter = counter + 3
    print("Python Loops")
else:
    print("Code block inside the else statement")
```

OUTPUT:

```
Python Loops
Python Loops
Python Loops
Python Loops
Code block inside the else statement
```

PROGRAM:

main.py

```python
for string in "Python Loops":
    if string == "o" or string == "p" or string == "t":
        continue
    print('Current Letter:', string)
```

OUTPUT:

```
Current Letter: P
Current Letter: y
Current Letter: h
Current Letter: n
Current Letter:
Current Letter: L
Current Letter: s
```

PROGRAM:

main.py

```python
for string in "Python Loops":
    if string == 'L':
        break
    print('Current Letter: ', string)
```

OUTPUT:

```
Current Letter:  P
Current Letter:  y
Current Letter:  t
Current Letter:  h
Current Letter:  o
Current Letter:  n
Current Letter:


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

PROGRAM:

```
main.py
for string in "Python Loops":
    pass
print( 'Last Letter:', string)
```

OUTPUT:

```
Last Letter: s


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

PROGRAM:

main.py

```python
numbers = [3, 5, 23, 6, 5, 1, 2, 9, 8]
sum_ = 0
for num in numbers:
    sum_ = sum_ + num ** 2
print("The sum of squares is: ", sum_)
```

OUTPUT:

```
The sum of squares is:  774


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

PROGRAM:

```python
my_list = [3, 5, 6, 8, 4]
for iter_var in range( len( my_list ) ):
    my_list.append(my_list[iter_var] + 2)
print( my_list )
```

OUTPUT:

```
[3, 5, 6, 8, 4, 5, 7, 8, 10, 6]


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

PROGRAM:

```python
numbers = [3, 5, 23, 6, 5, 1, 2, 9, 8]
sum_ = 0
for num in range( len(numbers) ):
    sum_ = sum_ + numbers[num] ** 2
print("The sum of squares is: ", sum_)
```

OUTPUT:

```
The sum of squares is:  774


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

PROGRAM:

```python
student_name_1 = 'Itika'
student_name_2 = 'Parker'

records = {'Itika': 90, 'Arshia': 92, 'Peter': 46}

def marks(student_name):
    for a_student in records:
        if a_student == student_name:
            return records[a_student]
    return f'There is no student of name {student_name} in the records'
print(f"Marks of {student_name_1} are: {marks(student_name_1)}")
print(f"Marks of {student_name_2} are: {marks(student_name_2)}")
```

OUTPUT:

```
input
Marks of Itika are: 90
Marks of Parker are: There is no student of name Parker in the records


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

Creating String in Python

PROGRAM:

```python
str1 = 'Hello Python'
print(str1)
str2 = "Hello Python"
print(str2)
str3 = '''''Triple quotes are generally used for represent the multiline or docstring'''
print(str3)
```

OUTPUT:

```
input
Hello Python
Hello Python
''Triple quotes are generally used for represent the multiline or docstring
```

Creating String in Python

PROGRAM:

```python
str1 = 'Hello Python'
print(str1)
str2 = "Hello Python"
print(str2)
str3 = '''''Triple quotes are generally used for represent the multiline or docstring'''
print(str3)
```

OUTPUT:

```
input
Hello Python
Hello Python
''Triple quotes are generally used for represent the multiline or docstring
```

PROGRAM:

n.py

```
str = "HELLO"
print(str[0])
print(str[1])
print(str[2])
print(str[3])
print(str[4])
```

OUTPUT:

```
H
E
L
L
O
```

PROGRAM:

```python
str = 'JAVATPOINT'

print(str[-1])
print(str[-3])
print(str[-2:])
print(str[-4:-1])
print(str[-7:-2])
print(str[::-1])
print(str[-12])
```

OUTPUT:

```
T
I
NT
OIN
ATPOI
TNIOPTAVAJ
Traceback (most recent call last):
  File "/home/main.py", line 9, in <module>
    print(str[-12])
          ~~~^^^^^
IndexError: string index out of range
```

PROGRAM:

```python
str = "HELLO"
print(str)
str = "hello"
print(str)
```

OUTPUT:

```
HELLO
hello
```

PROGRAM:

```python
str = "Hello"
str1 = " world"
print(str*3)
print(str+str1)
print(str[4])
print(str[2:4])
print('w' in str)
print('wo' not in str1)
print(r'C://python37')
print("The string str : %s"%(str))
```

OUTPUT:

```
HelloHelloHello
Hello world
o
ll
False
False
C://python37
The string str : Hello
```

PROGRAM:

```
Integer = 10;
Float = 1.290
String = "Devansh"
print("Hi I am Integer ... My value is %d\nHi I am float ... My value is %f\nHi I am string ... My value is
```

OUTPUT:

```
Hi I am Integer ... My value is 10
Hi I am float ... My value is 1.290000
Hi I am string ... My value is Devansh
```

PROGRAM:

```
.py
list_ = [4, 5, 7, 1, 7]
tuple_ = (4, 1, 8, 3, 9)
print("List is: ", list_)
print("Tuple is: ", tuple_)
```

OUTPUT:

```
List is:  [4, 5, 7, 1, 7]
Tuple is:  (4, 1, 8, 3, 9)
```

PROGRAM:

```python
list_ = ["Python", "Lists", "Tuples", "Differences"]
tuple_ = ("Python", "Lists", "Tuples", "Differences")
list_[3] = "Mutable"
print( list_ )
try:
    tuple_[3] = "Immutable"
    print( tuple_ )
except TypeError:
    print( "Tuples cannot be modified because they are immutable" )
```

OUTPUT:

```
['Python', 'Lists', 'Tuples', 'Mutable']
Tuples cannot be modified because they are immutable


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

PROGRAM:

```
main.py (Ctrl+M)
1 print("{} and {} both are the best friend".format("Devansh","Abhishek"))
2 print("{1} and {0} best players ".format("Virat","Rohit"))
3 print("{a},{b},{c}".format(a = "James", b = "Peter", c = "Ricky"))
```

OUTPUT:

```
Devansh and Abhishek both are the best friend
Rohit and Virat best players
James,Peter,Ricky


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

PROGRAM:

```python
list_ = ["Python", "Lists", "Tuples", "Differences"]
tuple_ = ("Python", "Lists", "Tuples", "Differences")
print("Size of tuple: ", tuple_.__sizeof__())
print("Size of list: ", list_.__sizeof__())
```

OUTPUT:

```
Size of tuple:  56
Size of list:  72
```