

TEMPRATURE

```
celsius = float(input("Enter temperature in Celsius: "))
```

```
fahrenheit = (celsius * 1.8) + 32  
celsius_converted =  
(fahrenheit - 32) / 1.8
```

```
print("Fahrenheit is", fahrenheit, "and Celsius is", celsius_converted)
```

```
a = float (input("val")) b=
```

```
(a*1.8)+32 c=(b+32)/1.8
```

```
print("b",b,"a",a)
```

output:

Enter temperature in Celsius: 2

Fahrenheit is 35.6 and Celsius is 2.0000000000000001

PRIME NUMBERS

```
lower = 0 upper = 10 print("Prime numbers between",  
lower, "and", upper, "are:")
```

```
for num in range(lower, upper + 1):  
    if num > 1:        for i in range(2,  
num):                if (num % i) == 0:  
                        break  
    else:  
        print(num)
```

output:

Prime numbers between 0 and 10 are:

2

3

5

7

MATRIX

```
x = [[12, 7, 3],
```

```
[4, 5, 6],  
[7, 8, 9]]
```

```
y = [[5, 8, 1, 2],      [6, 7, 3, 6],  
      [4, 5, 9, 1]]
```

```
result = [[0, 0, 0, 0],  
           [0, 0, 0, 0],  
           [0, 0, 0, 0]]
```

```
for i in range(len(x)):  
    for j in range(len(y[0])):  
        for k in range(len(y)):  
            result[i][j] += x[i][k] * y[k][j]
```

```
for r in result:  
    print(r)
```

```
output: [114,  
160, 60, 69]  
[74, 97, 73, 44]  
[119, 157, 112, 71]
```

STUDENT MARKS

```
sub1 = int(input("Enter marks of Web Design: ")) sub2 =  
int(input("Enter marks of Python Programming: ")) sub3 =
```

```
int(input("Enter marks of Web Lab: ")) sub4 =  
int(input("Enter marks of Python Lab: ")) sub5 =  
int(input("Enter marks of Blockchain: "))
```

```
total = sub1 + sub2 + sub3 + sub4 + sub5  
print("Total Marks:", total)
```

```
average = total / 5  
print("Average Marks:", average)
```

```
if average >= 90:  
    print("Grade: A")  
elif average >= 80:  
    print("Grade: B")  
elif average >= 70:  
    print("Grade: C")  
elif average >= 60:  
    print("Grade: D")  
elif average >= 50:  
    print("Grade: E")  
else:  
    print("Grade: F")
```

output:

Enter marks of Web Design: 56

Enter marks of Python Programming: 45

Enter marks of Web Lab: 55

Enter marks of Python Lab: 77

Enter marks of Blockchain: 55

Total Marks: 288

Average Marks: 57.6

Grade: E

ELECTRICITY BILL

```
def calculate_bill(units):  
    if units <= 100:  
        return units * 10    elif  
        units <= 200:  
            return (100 * 10) + (units - 100) * 15  
    elif units <= 300:  
        return (100 * 10) + (100 * 15) + (units - 200) * 20  
    else:  
        return (100 * 10) + (100 * 15) + (100 * 20) + (units - 300) * 25
```

```
print("Electricity Bill Calculator") units =  
int(input("Enter number of units consumed: ")) bill =  
calculate_bill(units) print("Electricity Bill:", bill)
```

output:

Electricity Bill Calculator

Enter number of units consumed: 30

Electricity Bill: 300

MERGE SORT:

```
def merge_sort(arr):  
    if len(arr) <= 1:  
        return arr  
  
    mid = len(arr) // 2    left =  
    merge_sort(arr[:mid])    right =  
    merge_sort(arr[mid:])  
  
    return sorted(left + right)  
  
arr = [3, 7, 6, -10, 15, 23.5, 55, -13]  
print("Sorted array:", merge_sort(arr))
```

OUTPUT

Sorted array: [-13, -10, 3, 6, 7, 15, 23.5, 55] LIST

```
a = [10, 24, 76, 23, 12] print(max(a))
```

OUTPUT

76

LINEAR SEARCH

```
def linear_search(arr, target):  
    for i in range(len(arr)):  
        if arr[i] == target:
```

```
        return i
    return -1

arr = [5, 3, 7, 1, 9] target
= 7

result = linear_search(arr, target)

if result != -1:
    print("Element found at index:", result)
else:
    print("Element not found in the
    array")
```

OUTPUT:

Element found at index: 2

BINARY SEARCH

```
def binary_search(arr, x):
    low, high = 0, len(arr) - 1

    while low <= high:
        mid = (low + high) // 2

        if arr[mid] == x:
            return mid
        elif
        arr[mid] < x:
```

```
low = mid + 1
```

```
else:
```

```
    high = mid - 1
```

```
    return -1
```

```
arr = [2, 3, 4, 10, 40] x
```

```
= 10
```

```
result = binary_search(arr, x) if
```

```
result != -1:
```

```
    print("Element is present at index", result)
```

```
else:    print("Element is not present in  
array")
```

OUTPUT:

Element is present at index 3

Exception Handling

```
try:
```

```
    numerator = 10
```

```
    denominator = 0
```

```
    result = numerator/denominator
```



```
    print(result)
except:
    print("Error: Denominator cannot be 0.")
```

Classes and Attributes

```
class sampleclass:
    count = 0    # class attribute

    def increase(self):
        sampleclass.count += 1

# Calling increase() on an object
s1 = sampleclass()
s1.increase()
print(s1.count)

# Calling increase on one more
# object
s2 = sampleclass()
s2.increase()
print(s2.count)
```