

```
#1
n = int(input("Enter a number: "))
sum = 0
for i in range(1, n):
    if(n % i == 0):
        sum = sum + i
if (sum == n):
    print(" %d is a Perfect Number" %n)
else:
    print(" %d is not a Perfect Number" %n)
```

```
Enter a number: 6
6 is a Perfect Number
enter a word: 
```

```
#2
def isPalindrome(x):
    return x == x[::-1]
x = input("enter a word: ")
y = isPalindrome(x)

if x:
    print("Yes")
else:
    print("No")
```

```
enter a word: harsh
Yes
```

```
#3
from math import factorial

n = 6
for i in range(n):
    for j in range(n-i+1):
        print(end=" ")

    for j in range(i+1):
        print(factorial(i)//(factorial(j)*factorial(i-j)), end=" ")
    print()
```

```

      1
    1 1
  1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1

```

```

#4
import string

def ispangram(str):
    alphabet = "abcdefghijklmnopqrstuvwxyz"
    for char in alphabet:
        if char not in str.lower():
            return False

    return True

# Driver code
string = 'the quick brown fox jumps over the lazy dog'
if(ispangram(string) == True):
    print("Yes")
else:
    print("No")

```

Yes

```

#5
items=[n for n in input().split('-')]
items.sort()
print('-'.join(items))

```

red-green-black-blue

black-blue-green-red

```

#6
def student_data(student_id, **kwargs):
    print(f'\nStudent ID: {student_id}')
    if 'student_name' in kwargs:
        print(f"Student Name: $ {kwargs['student_name']}")

    if 'student_name' and 'student_class' in kwargs:

```

```

        print(f"\nStudent Name: $ {kwargs['student_name']}")
        print(f"Student Class: $ {kwargs['student_class']}")

student_data(student_id='21109002', student_name='Alisha', student_class
='306')

student_data(student_id='21109017', student_name='Harsh', student_class
='306')

```

Student ID: 21109002
Student Name: \$ Alisha

Student Name: \$ Alisha
Student Class: \$ 306

Student ID: 21109017
Student Name: \$ Harsh

Student Name: \$ Harsh
Student Class: \$ 306
ClassStudent: pass

```

#7
class Student:
    pass
class Marks:
    pass
student1 = Student()
marks1 = Marks()
print(isinstance(student1, Student))
print(isinstance(marks1, Student))
print(isinstance(marks1, Marks))
print(isinstance(student1, Marks))
print("\nCheck whether the said classes are subclasses of the built-in object
class :")
print(issubclass(Student, object))
print(issubclass(Marks, object))

```

True
False
True
False

Check whether the said classes are subclasses of the built-in object class :

True

True

```
#8
def findTriplets(arr, n):
    found = False
    for i in range(0, n-2):
        for j in range(i+1, n-1):
            for k in range(j+1, n):
                if (arr[i] + arr[j] + arr[k] == 0):
                    print(arr[i], arr[j], arr[k])
                    found = True
    if (found == False):
        print("does not exist")
arr = [-25, -10, -7, -3, 2, 4, 8, 10]
n = len(arr)
findTriplets(arr, n)
```

-10 2 8

-7 -3 10

```
#9
class validity:
    def f(str):
        a= ['()', '{}', '[]']
        while any(i in str for i in a):
            for j in a:
                str = str.replace(j, '')
        return not str
s = input("enter : ")
print(s, "-", "is balanced"
      if validity.f(s) else "is Unbalanced")
```

enter : [{()}{()}]

[{()}{()}] - is balanced