

"""Q.1) Write a Python function to check whether a string is a pangram or not.

Note : Pangrams are words or sentences containing every letter of the alphabet at least once.

For example : "The quick brown fox jumps over the lazy dog"""

```
string = "The quick brown fox jumps over the lazy dog"
```

```
def is_pangram(a):  
    a = set(a.lower())  
    b = set("abcdefghijklmnopqrstuvwxyz")  
    if a.issubset(b):  
        return True  
    else:  
        return False
```

```
print(is_pangram(string))
```




The screenshot shows a 'Run' window with a single tab labeled '01'. The command line displays the execution of a Python script: "C:\CDAC DBDA\Python\PythonDAY05\_Test\.venv\Scripts\python.exe" "C:\CDAC DBDA\Python\PythonDAY05\_Test\01.py". The output of the script is 'False', indicating that the provided string is not a pangram. The window also shows 'Process finished with exit code 0'.

"""Q.2) Write a Python program to calculate the sum of the digits in an integer. """

```
a = 1234
```

```
def sumdigits(num):  
    total = 0  
    while num > 0:  
        digit = num % 10  
        total += digit  
        num //= 10  
    return total
```

```
print(sumdigits(a))
```



The screenshot shows a 'Run' window with a single tab labeled '02'. The command line displays the execution of a Python script: "C:\CDAC DBDA\Python\PythonDAY05\_Test\.venv\Scripts\python.exe" "C:\CDAC DBDA\Python\PythonDAY05\_Test\02.py". The output of the script is '10', which is the sum of the digits of the integer 1234. The window also shows 'Process finished with exit code 0'.

"""Q.3) Write a Python program to sort three integers without using conditional statements and loops. [ u can use built in functions for this ]  
"""

```
a = int(input("enter first number: "))  
b = int(input("enter second number: "))  
c = int(input("enter third number: "))  
print(sorted([a, b, c]))  
# print(sorted([a, b, c], reverse=True))
```

```
Run 03 x  
"C:\CDAC DBDA\Python\PythonDAY05_Test\.venv\Scripts\python.exe" "C:\CDAC DBDA\Python\PythonDAY05_Test\03.py"  
enter first number: 45  
enter second number: 21  
enter third number: 63  
[21, 45, 63]  
Process finished with exit code 0
```

"""Q.4) Write a Python function to check whether a number is perfect or not.  
According to Wikipedia : In number theory, a perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself (also known as its aliquot sum). Equivalently, a perfect number is a number that is half the sum of all of its positive divisors (including itself).  
Example : The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and  $1 + 2 + 3 = 6$ . Equivalently, the number 6 is equal to half the sum of all its positive divisors:  $(1 + 2 + 3 + 6) / 2 = 6$ . The next perfect number is  $28 = 1 + 2 + 4 + 7 + 14$ . This is followed by the perfect numbers 496 and 8128."""

```
def is_perfect(n):  
    if n <= 1:  
        return False  
  
    div_sum = 0  
    for i in range(1, n):  
        if n % i == 0:  
            div_sum += i  
  
    return div_sum == n  
print(is_perfect(6))
```

```
Run 04 x  
"C:\CDAC DBDA\Python\PythonDAY05_Test\.venv\Scripts\python.exe" "C:\CDAC DBDA\Python\PythonDAY05_Test\04.py"  
True  
Process finished with exit code 0
```