Task 23

Deep Das

1. Encrypt a message using the Caeser-Cipher Method in Python.

Answer:

CAESER.py > caesar_cipher_encrypt

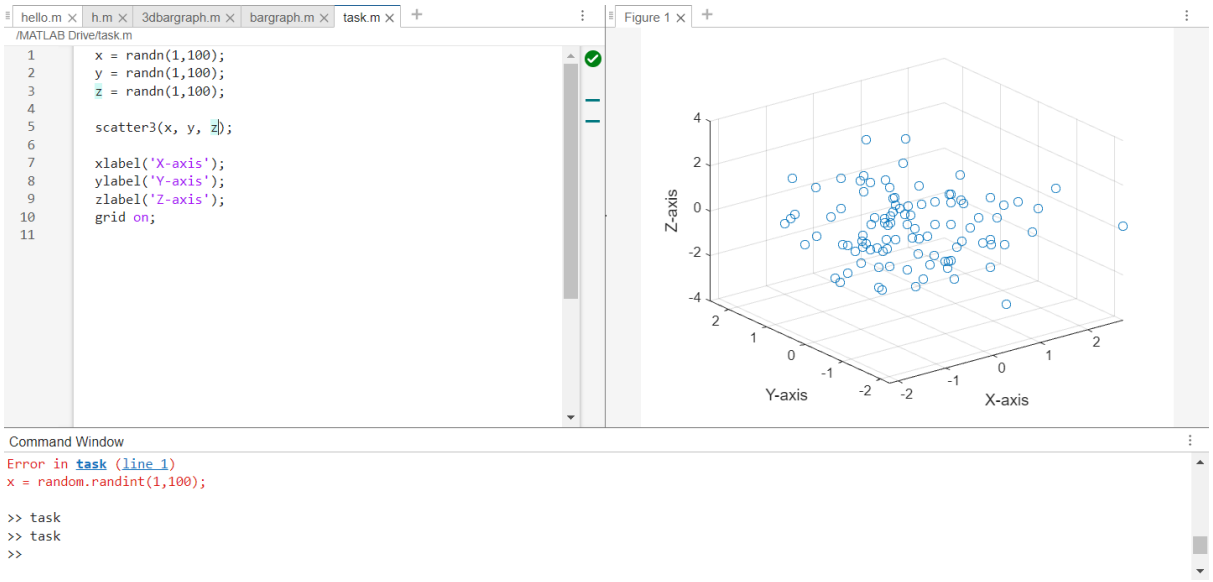
```python
def caesar_cipher_encrypt(text):
    result = ""
    for char in text:
        if char.isalpha():
            if char.isupper():
                result += chr(155 - ord(char))
                # 65 to 90 aave ne A to Z etle add karine minus kairu che.
            else:
                result += chr(219 - ord(char))
        else:
            # biju kai aave to direct muki devanu
            result += char

    return result

message = "HELLO WORLD this IS a pROSoUnD method."
encrypted_message = caesar_cipher_encrypt(message)
print("Original message: ", message)
print("Encrypted message: ", encrypted_message)
message2 = "abcdefghijklmnopqrstuvwxyz."
encrypted_message2 = caesar_cipher_encrypt(message2)
print("Original message: ", message2)
```

PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Original message:  HELLO WORLD this IS a pROSoUnD method.
Encrypted message:  SVOOL DLIOW gsrh RH z kILHlFmW nvgslw.
Original message:  abcdefghijklmnopqrstuvwxyz.
Encrypted message:  zyxwvutsrqponmlkjihgfedcba.
PS D:\aero+>
```

2.create a scatter plot in MATLAB.

Answer:



3. Pattern printing.

Answer:

```
def pattern1():
    n = 5
    for i in range(1, n + 1):
        for iterator in range(n - i):
            print(" ", end="")
        for j in range(1, i + 1):
            print(j, end="")
        for j in range(i - 1, 0, -1):
            print(j, end="")
        print()

    for i in range(n - 1, 0, -1):
        for iterator in range(n - i):
            print(" ", end="")
        for j in range(1, i + 1):
            print(j, end="")
        for j in range(i - 1, 0, -1):
            print(j, end="")
        print()

pattern1()
print()

def pattern2():
    n=5
    for i in range(n):
```

```
        print(" "*(n-i-1)+"1",end="")
        if i>0:
            print(" "*(2*i-1)+str(i+1),end="")
        print()
    for i in range(1,n):
        print(" "*i+"1",end="")
        if i<4:
            print(" "*(2*(n-i-1))+str(n-i))


pattern2()
print()

def pattern3():
    rows=6
    for i in range(rows):
        start = 0 if i%2==0 else 1
        string=""
        for j in range (i):
            if j%2==0:
                string += str(start)
            else:
                string += str(1-start)
        print(string)

pattern3()
print()

def pattern4():
     n = 5
     for i in range(1, n + 1):
         for iterator in range(n - i):
             print(" ",end=" ")
         for j in range(1, i + 1):
             print(j,end=" ")
         for j in range(i - 1, 0, -1):
             print(j,end=" ")
         print()

     for i in range(n - 1, 0, -1):
         for iterator in range(n - i):
             print(" ", end=" ")
         for j in range(1, i + 1):
             print(j, end=" ")
         for j in range(i - 1, 0, -1):
             print(j, end=" ")
         print()
```

```
pattern4()
print()

def pattern5():
    n=5
    for i in range(5):
        str=''
        for j in range(1,n-i+1):
            if i%2==0:
                str+=('1')
            else:
                str+=('0')
        print(str)

pattern5()
print()

def pattern6():
    n=4

    matrix = [[0]*n for _ in range(n)]

    left, right = 0, n-1
    top, bottom = 0, n-1
    num = 1
    while left <= right and top <= bottom:
        #top row
        for i in range(left, right+1):
            matrix[top][i] = num
            num += 1
        top += 1

        #right column
        for i in range(top, bottom+1):
            matrix[i][right] = num
            num += 1
        right -= 1

        #bottom row
        for i in range(right, left-1, -1):
            matrix[bottom][i] = num
            num += 1
        bottom -= 1

        #left column
        for i in range(bottom, top-1, -1):
            matrix[i][left] = num
            num += 1
```

```
        left += 1

    for row in matrix:
        first_cell = True
        row_str = ""
        for cell in row:
            if first_cell==False:
                row_str += " "
            row_str += str(cell)
            first_cell = False
        print(row_str)

pattern6()
```

output :

```
PROBLEMS     OUTPUT     DEBU

    1
   121
  12321
 1234321
123454321
 1234321
  12321
   121
    1

    1
   1 2
  1   3
 1     4
1       5
 1     4
  1   3
   1 2
    1

1
01
101
0101
10101
```

```
        1
      1 2 1
    1 2 3 2 1
  1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
  1 2 3 4 3 2 1
    1 2 3 2 1
      1 2 1
        1

11111
0000
111
00
1

1 2 3 4
12 13 14 5
11 16 15 6
10 9 8 7
PS D:\aero+>
```