Transpose.py

```
#Kirk Martin

#Special Topics

#Due Date 3/2/25

# code:

# Step 1: specify the message

# Step 2 : indicate the number of columns needed.

# Step 3: define encrypted as dividing the string into 8 columns

# Step 4: create a for loop to check the message and divide the characters.



import random


message = " I hope this works in some way shape or form"


# Use a range here for the message to be read . A zero(0) would give no encryption, and a one(1) revealed the actual message!


key = random.randrange(2, 15)

# key = 10


encrypted = [""] * key

#decrypted = [""] / 8

# # test for the number of columns

# print(encrypted)


# test to count the length of the string message = 52
```

```
# print(len(message))


# use the loop to go thorugh the message, indicator will allocate each char into the proper column from 0 - 7


for column in range(key):
  indicator = column


  # while there are still characters left in the message, place each letter in the correct column
  while indicator < len(message):
    encrypted[column] += message[indicator]
    # test:
    # print(encrypted)
    # indicator is = indicator + key
    indicator += key


print("FOR 'ENCRYPTION NON-PKE' HW PART 2: ENCODING A MESSAGE: \n")
print("My encoded message is: \n"+"".join(encrypted)+'\n')
# print("".join(decrypted))
```