**Python Programming Basic Assignment-16**

*# Question1. Write a function that stutters a word as if someone is struggling to read it.*

*# The first two letters are repeated twice with an ellipsis ... and space after each,*

*# and then the word is pronounced with a question mark ?.*

*# Examples*

*# stutter("incredible") ➞ "in... in... incredible?"*

*# stutter("enthusiastic") ➞ "en... en... enthusiastic?"*

*# stutter("outstanding") ➞ "ou... ou... outstanding?"*

*# Hint :- Assume all input is in lower case and at least two characters long.*

In [12]:

**def** stutter(word):

**for** i **in** word:

print(f"{i[:2]}... {i[:2]}... {i}?")

In [13]:

stutter(['incredible','enthusiastic','outstanding'])

in... in... incredible?

en... en... enthusiastic?

ou... ou... outstanding?

In [14]:

*# Question 2.Create a function that takes an angle in radians and*

*# returns the corresponding angle in degrees rounded to one decimal place.*

*# Examples*

*# radians\_to\_degrees(1) ➞ 57.3*

*# radians\_to\_degrees(20) ➞ 1145.9*

*# radians\_to\_degrees(50) ➞ 2864.8*

In [23]:

**import** numpy **as** np

**def** radian\_to\_degree(angle):

**return** round(np**.**degrees(angle),1)

In [24]:

radian\_to\_degree(1)

Out[24]:

57.3

In [25]:

radian\_to\_degree(20)

Out[25]:

1145.9

In [26]:

radian\_to\_degree(50)

Out[26]:

2864.8

In [27]:

*# Question 3. In this challenge, establish if a given integer num is a Curzon number.*

*# If 1 plus 2 elevated to num is exactly divisible by 1 plus 2 multiplied by num, then num is a Curzon number.*

*# Given a non-negative integer num, implement a function that returns True if num is a Curzon number, or False otherwise.*

*# Examples*

*# is\_curzon(5) ➞ True*

*# # 2 \*\* 5 + 1 = 33*

*# # 2 \* 5 + 1 = 11*

*# # 33 is a multiple of 11*

*# is\_curzon(10) ➞ False*

*# # 2 \*\* 10 + 1 = 1025*

*# # 2 \* 10 + 1 = 21*

*# # 1025 is not a multiple of 21*

*# is\_curzon(14) ➞ True*

*# # 2 \*\* 14 + 1 = 16385*

*# # 2 \* 14 + 1 = 29*

*# # 16385 is a multiple of 29*

In [25]:

*# A number N is said to be a Curzon Number if 2\*\*N + 1 is divisible by 2\*N + 1.*

**def** curzon(num):

**if** ((2**\*\***num)**+**1)**%**((2**\***num)**+**1)**==**0:

print('True')

print('Its a curzon number')

**else**:

print('False')

print('Its not a curzon number')

In [26]:

curzon(5)

True

Its a curzon number

In [27]:

curzon(10)

False

Its not a curzon number

In [28]:

curzon(14)

True

Its a curzon number

In [29]:

*# Question 4.Given the side length x find the area of a hexagon.*

*# Examples*

*# area\_of\_hexagon(1) ➞ 2.6*

*# area\_of\_hexagon(2) ➞ 10.4*

*# area\_of\_hexagon(3) ➞ 23.4*

In [35]:

**import** numpy **as** np

**def** area\_of\_hexagon(side):

area**=**(3**\***np**.**sqrt(3)**\***(side**\*\***2))**/**2

**return** round(area,1)

In [36]:

side**=**int(input('Enter the side of a hexagon:'))

area\_of\_hexagon(side)

Enter the side of a hexagon:1

Out[36]:

2.6

In [37]:

side**=**int(input('Enter the side of a hexagon:'))

area\_of\_hexagon(side)

Enter the side of a hexagon:2

Out[37]:

10.4

In [38]:

side**=**int(input('Enter the side of a hexagon:'))

area\_of\_hexagon(side)

Enter the side of a hexagon:3

Out[38]:

23.4

In [5]:

*# Question 5. Create a function that returns a base-2 (binary) representation of a base-10 (decimal) string number.*

*# To convert is simple: ((2) means base-2 and (10) means base-10) 010101001(2) = 1 + 8 + 32 + 128.*

*# Going from right to left, the value of the most right bit is 1,*

*# now from that every bit to the left will be x2 the value, value of an 8 bit binary numbers are*

*# (256, 128, 64, 32, 16, 8, 4, 2, 1).*

*# Examples*

*# binary(1) ➞ "1"*

*# # 1\*1 = 1*

*# binary(5) ➞ "101"*

*# # 1\*1 + 1\*4 = 5*

*# binary(10) ➞ "1010"*

*# # 1\*2 + 1\*8 = 10*

In [44]:

**def** binary(num):

**if** num **>=** 1:

binary(num **//** 2)

print(num **%** 2,end**=**'')

In [45]:

binary(1)

1

In [46]:

binary(5)

101

In [47]:

binary(10)

1010

In [ ]: