Ask the user for a number. Depending on whether the number is even or odd, print out an appropriate message to the user. *Hint: how does an even / odd number react differently when divided by 2?*

Extras:

1. If the number is a multiple of 4, print out a different message.

2. Ask the user for two numbers: one number to check (call it num) and one number to divide by (check). If check divides evenly into num, tell that to the user. If not, print a different appropriate message.

3. If the number if a prime number.

|  |
| --- |
| #INPUTS  print("Program that checks if two numbers are even or odd\n")  num = int(input("Enter one number (bigger than 1) to check: "))  check = int(input("Enter one number to divide by: "))  #TEST LOOPS  if num > 1:  if (num%2) == 0:  print(num,"is even.\n")  else:  print(num,"is odd\n")  if (num%4) == 0:  print(num,"is divisible by 4")  else:  print(num,"is not divisible by 4")  if (num % check) == 0:  print(check,"divides evenly into ", num)  else:  print(check,"doesn't divide evenly into ", num)  for cont in range(2,num):  if (num % cont) == 0:  print(num,"is not a prime number")  break  else:  print(num,"is a prime number")  Unit test  import unittest  class NumbersTest(unittest.TestCase):  def test\_even(self):  #Test that numbers between 0 and 5 are all even.  for i in range(0, 6):  with self.subTest(i=i):  self.assertEqual(i % 2, 0)  def test\_odd(self):  #Test that numbers between 0 and 5 are all odd.  for i in range(0, 6):  with self.subTest(i=i):  self.assertEqual(i % 2, 1)  if \_\_name\_\_ == '\_\_main\_\_':  unittest.main() |

|  |
| --- |
| Problem 2. List Confusion |
| Take two lists, say for example these two: |

a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]

b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

and write a program that returns a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.

Extras:

1. Randomly generate two lists to test this

2. Write this in one line of Python (don’t worry if you can’t figure this out at this point - we’ll get to it soon)

import random

#SET LISTS

a = [1, 1, 2, 2, 3, 5, 8, 13, 21, 34, 55, 89]

b = [1, 2, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

#COMPARE LISTS AND DISPLAY SAME ELEMENTS

c = set(a) & set(b)

print(c)

#RANDOM SUFFLE OF LISTS 'a' & 'b' AND COMPARE ELEMENTS

random.shuffle(a)

random.shuffle(b)

c = set(a) & set(b)

print("Scrambled list 'a':", a, "\nScrambled list 'b':",b,"\nMutual Elem:",c)

Unit test

import unittest

class ListTest(unittest.TestCase):

def setUp(self):

super(ListTest, self).setUp()

self.addTypeEqualityFunc(str, self.assertMultiLineEqual)

def testString(self):

a = [1, 1, 2, 2, 3, 5, 8, 13, 21, 34, 55, 89]

b = [1, 2, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

for count in range(max(len(a),len(b))):

for counte in range(max(len(a),len(b))):

with self.subTest(count=count+counte):

self.assertEqual(a[count], b[counte])

if \_\_name\_\_ == '\_\_main\_\_':

unittest.main()