Module Test – Python 2 – Test 1

String chars

Given a Python function that takes in a list of strings, write a code to return a new list containing only those strings in which all characters are distinct (non-repeating). These strings should be present in the same order in which they are present in the input list.
NOTE: You can assume that the input list contains only lowercase alphabetic characters (a-z).
Input Format
List[str]
Output Format
List[str]
Sample Input 1
['hello', 'world', 'python', 'programming']
Sample Output 1
['world', 'python']
Sample Explanation 1
In this case, the input list contains four strings. The function returns a new list containing only the strings that have all distinct characters. The strings 'hello' and 'programming' have repeated characters, so they are not included in the output list. The strings 'world' and 'python' have all the distinct (non-repeating) characters, so they are included in the output list. They appear in the same order as in the given input list.
Sample Input 2
['abc', 'def', 'ghi', 'jkl']
Sample Output 2
['abc', 'def', 'ghi', 'jkl']
Sample Explanation 2
In this case, all the input strings have only unique (non-repeating) characters, so all of them are part of the output list. In the output list, these strings appear in the same order as in the given input list.
User Code

```
def unique_chars(list_of_words):
  lst = []
  for word in list_of_words:
    to_add = True
    for character in word:
      if word.count(character) > 1:
        to_add = False
        break
    if to_add:
      lst.append(word)
  return lst
```

Categorize the person 2
Problem Description

Write a program to in input from the user an int value(**A**) representing the height of person. You have to print the category of that person.

- If the height is greater than or equal to 195 then that person is “abnormal”.
- If the height is greater than or equal to 165 and less than 195 then that person is “taller”.
- If the height is greater than or equal to 150 and less than 165 then that person is “average”.
- If the height is smaller than 150 then that person is a “dwarf”.

Problem Constraints
50 <= A <= 250
Input Format
One line containing an int value A.
Output Format
A string representing the category of the person.
Example Input
Input 1:
165

Input 2:
205

Input 3:
155
Example Output
Output 1:
taller

Output 2:
abnormal

Output 3:
average
Example Explanation
Explanation 1:
Clearly, 165 is in the range of taller.

Explanation 2:
Clearly, 205 is in the range of abnormal.

Explanation 3:
Clearly, 155 is in the range of average.

User Code

```
def main():
  # YOUR CODE GOES HERE
  # Please take input and print output to standard input/output (stdin/stdout)
  # E.g. 'input()/raw_input()' for input & 'print' for output
  A = int(input())
  if A < 150:
    print("dwarf")
  elif A >=150 and A < 165:
    print("average")
  elif A >= 165 and A < 195:
    print("taller")
  else:
    print("abnormal")


  return 0
```

Maximum no of 1's in a row

Given a 2D List of integers, containing only 1s and 0s, return the index of the row with the maximum number of 1s.

Note: It is given only 1 row among all will have the maximum number of 1s

Input Format:

arr: List[List[int]]

Output Format:

int

Input Sample:

[[0,1,1,1], [0,0,1,1], [0,0,1,1]]

Output Sample:

0

Sample Explanation:

Row 0 has 3 ones whereas

rows 1 and 2 have just 2 ones.

User Code

```
def maxOnes(arr):
 #write code
  initial_sum = 0
  initial_index = 0
  for i in range(len(arr)):
    if sum(arr[i]) > initial_sum:
      initial_sum = sum(arr[i])
      initial_index = i

  return initial_index
```

Dividing x and y

For what distinct values of x and y does the program below output True?

```
x = int(input())
y = int(input())
flag = True
for i in range(2, x + 1):
  if(x % i == 0):
    if(y % i == 0):
      flag = False
print(flag)
```

When x and y have no common divisors except 1.

What calls what?

What would be the output of the code given below?

```
def function_1():
  print("third")
  function_2()

def function_2():
  print("second")
  function_3()

def function_3():
  print("first")

function_1()
```

third
second
first

Oldest Person

The following code uses a list of tuples to store the name and age of some people.

```
people = [('Alice', 25), ('Bob', 32), ('Charlie', 20), ('David', 35)]


oldest_name = __A__

oldest_age = __B__


for name, age in people:

  if age > oldest_age:

    oldest_name = __C__
```

```
    oldest_age = __D__


print("The oldest person is", oldest_name)
```

Which of the following options can fill the blanks to complete the code so that it prints the name of the oldest person in the list?

A-> "", B-> 0, C-> name, D-> age

Inside isTrue

What are the possible value we can set the variable isTrue to from the options given below so that the code below gives us the output as the following string "Inside the if-statement"?

```
isTrue = ____

if bool(isTrue):

  print("Inside the if-statement")

else:

  print("Inside the else-statement")
```

Statement A: Setting isTrue to any integer greater than 0.

Statement B: Setting isTrue to any integer less than 0.

Statement C: Setting isTrue to 0.

Statement D: Setting isTrue to 1.

A, B, D