Python Lists and Strings Coding Questions

**Total Marks:** 30  
(10 questions × 3 marks each)  
  
**Passing Marks:** 12  
**Excellent Marks:** 24 and above

**Difficulty Level:** Easy  
**Time:** 60 minutes

# Instructions

* Questions are marked based on difficulty:
* - Easy questions: 3 marks each
* - Medium questions: 5 marks each
* - Difficult questions: 10 marks each
* Write clean, well-commented code
* Include basic error handling where appropriate
* Test your code with the given test cases
* Bonus question is for extra practice and does not count towards total marks

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## 1: String Length Check [3 marks] Write a function `check\_string\_length(s)` that returns True if the string length is between 5 and 10 characters (inclusive), False otherwise.

***Example:***

Input: "Hello"  
Output: True  
  
Input: "Hi"  
Output: False

# Write your answer here

***Answer: def check\_string\_length(s) :***

***length = len(s)***

***if(length>4 and length<11) :***

***return True***

***else :***

***return False***

# Write your answer here

## 2: List Sum [3 marks] Write a function `sum\_list(numbers)` that takes a list of numbers and returns their sum. Handle empty lists by returning 0.

***Example:***

Input: [1, 2, 3, 4, 5]  
Output: 15  
  
Input: []  
Output: 0

# Write your answer here

***Answer:***

***def sum\_list(numbers) :***

***sum = 0***

***for value in numbers :***

***sum+= value***

***return sum***

# Write your answer here

## 3: String Case Converter [3 marks] Write a function `convert\_case(s)` that converts a string to uppercase if it contains more uppercase letters, and lowercase if it contains more lowercase letters. If equal, return the original string.

***Example:***

Input: "HeLLo"  
Output: "HELLO"  
  
Input: "hello"  
Output: "hello"

# Write your answer here

***Answer:***

# Write your answer here

def isLower(value):

return True if ord(value) >= 97 else False

def isStrLower(value):

lcounter = 0

ucounter =0

for v in value :

if isLower(v) :

lcounter+=1

else :

ucounter+=1

return 1 if(lcounter > ucounter) else -1 if(ucounter > lcounter) else 0

def convert\_case(givenStr) :

if(isStrLower(givenStr) == 1) :

return givenStr.lower()

else :

if(isStrLower(givenStr) == -1) :

return givenStr.upper()

else :

return givenStr

## 4: List Duplicate Remover [3 marks] Write a function `remove\_duplicates(lst)` that removes duplicate elements from a list while preserving the order of first occurrence.

***Example:***

Input: [1, 2, 2, 3, 3, 4, 5, 5]  
Output: [1, 2, 3, 4, 5]  
  
Input: ["apple", "banana", "apple", "cherry"]  
Output: ["apple", "banana", "cherry"]

# Write your answer here

***Answer:***

def remove\_duplicates(lst) :

aList = []

for value in lst :

if value not in aList :

aList.append(value)

return aList

## 5: String Word Counter [3 marks] Write a function `count\_words(s)` that counts the number of words in a string. Words are separated by spaces.

***Example:***

Input: "Hello world"  
Output: 2  
  
Input: "This is a test sentence"  
Output: 5

# Write your answer here

***Answer:***

def count\_words(s) :

if(len(s) >0 and type(s)== str) :

return len(s.split(' '))

else :

return 0

## 6: List Even Numbers [3 marks] Write a function `get\_even\_numbers(lst)` that returns a new list containing only the even numbers from the input list.

***Example:***

Input: [1, 2, 3, 4, 5, 6, 7, 8]  
Output: [2, 4, 6, 8]  
  
Input: [1, 3, 5, 7]  
Output: []

# Write your answer here

***Answer:***

***def get\_even\_numbers(lst) :***

***return list ( filter(lambda x: x%2==0, lst) )***

# Write your answer here

## 7: String Reverser [3 marks] Write a function `reverse\_string(s)` that reverses a string without using the built-in reverse() method.

***Example:***

Input: "hello"  
Output: "olleh"  
  
Input: "python"  
Output: "nohtyp"

# Write your answer here

***Answer:***

def reverse\_string(s) :

return s[::-1]

## 8: List Element Counter [3 marks] Write a function `count\_element(lst, element)` that counts how many times a specific element appears in a list.

***Example:***

Input: lst = [1, 2, 2, 3, 2, 4], element = 2  
Output: 3  
  
Input: lst = ["a", "b", "a", "c", "a"], element = "a"  
Output: 3

# Write your answer here

***Answer:***

def count\_element(lst, element) :

charCountdist = {}

if(len(lst) >0) :

for value in lst :

charCountdist[value] = charCountdist.get(value, 0) + 1

else :

return 0

return charCountdist[element]

## 9: String Vowel Counter [3 marks] Write a function `count\_vowels(s)` that counts the number of vowels (a, e, i, o, u) in a string, ignoring case.

***Example:***

Input: "Hello World"  
Output: 3  
  
Input: "Python Programming"  
Output: 4

# Write your answer here

***Answer:***

# Write your answer here

def count\_vowels(s) :

vowels = ('a', 'e', 'i', 'o', 'u')

counter =0

for char in s :

if char in vowels :

counter+=1

return counter

# Bonus Challenge [Extra Practice]

[Extra Practice]  
Write a function `is\_anagram(s1, s2)` that checks if two strings are anagrams of each other.   
An anagram is a word formed by rearranging the letters of another word, using all the original letters exactly once. For example, "listen" and "silent" are anagrams because they contain the same letters in different order.

***Example:***

Input: s1 = "listen", s2 = "silent"  
Output: True  
  
Input: s1 = "hello", s2 = "world"  
Output: False

# Write your answer here

***Answer:***

def is\_anagram(s1, s2) :

if(type(s1) == str and type(s1)==type(s2) ) :

return sorted(s1.lower()) == sorted(s2.lower())

return False

# Evaluation Criteria

* Correctness (50%)
* - Function works as expected (30%)
* - Handles edge cases correctly (20%)
* Code Readability (30%)
* - Clear variable names (10%)
* - Proper comments (10%)
* - Clean code structure (10%)
* Basic Error Handling (20%)
* - Input validation (10%)
* - Appropriate error messages (10%)

# Marking Scheme

* Easy questions: 3 marks each
* Medium questions: 5 marks each
* Difficult questions: 10 marks each
* Total marks for the paper: 30
* Passing marks: 12
* Excellent marks: 24 and above

***Good Luck!***