

## **Sample Coding Questions**

### **String Manipulation**

# Concatenation

```
first_name = "John"
```

```
last_name = "Doe"
```

```
full_name = first_name + " " + last_name
```

```
print(full_name)
```

# Repetition

```
repeated_text = "Repeat " * 3
```

```
print(repeated_text)
```

# Length

```
text = "Python"
```

```
length = len(text)
```

```
print("Length:", length)
```

# Single quotes

```
single_quoted = 'Hello, world!'
```

```
print(single_quoted)
```

# Double quotes

```
double_quoted = "Hello again!"
```

```
print(double_quoted)
```

# Escaping characters

```
escaped_string = "She said, \"Hello!\""
print(escaped_string)
```

```
escaped_string1 = 'She said, "Hello"'
print(escaped_string1)
```

```
text = " Python Programming "
```

```
text1 = "My university name is :- GLA University"
```

# Remove leading and trailing whitespace

```
trimmed_text = text.strip()
```

```
print(trimmed_text)
```

# Convert to lowercase and uppercase

```
lower_case = text.lower()
```

```
upper_case = text.upper()
```

```
print("Lowercase:", lower_case)
```

```
print("Uppercase:", upper_case)
```

# Replace

```
replaced_text = text.replace("Programming", "Coding")
```

```
print(replaced_text)
```

# Find substring

```
index = text1.find("GLA")
```

```
print("Index of 'GLA':", index)
```

```
text = "hello world"
```

```
capitalized_text = text.capitalize()
```

```
print(capitalized_text) # Output: Hello world
```

```
text = "hello world"
```

```
uppercase_text = text.upper()
```

```
print(uppercase_text) # Output: HELLO WORLD
```

```
text = "Hello World"
```

```
lowercase_text = text.lower()
```

```
print(lowercase_text) # Output: hello world
```

```
text = "abc"
```

```
print(text.isalpha()) # Output: True
```

```
text = "abc123"
```

```
print(text.isalpha()) # Output: False
```

```
number = "123"
```

```
print(number.isdigit()) # Output: True
```

```
text = "abc123"
```

```
print(text.isdigit()) # Output: False
```

```
text = " hello world "
```

```
stripped_text = text.strip()
```

```
print(stripped_text) # Output: "hello world"
```

```
text = "hello,world"
```

```
split_text = text.split(',')
```

```
print(split_text) # Output: ['hello', 'world']
```

```
words = ['hello', 'world']
```

```
joined_text = ','.join(words)
```

```
print(joined_text) # Output: 'hello,world'
```

```
text = "hello world"
```

```
new_text = text.replace("world", "universe")
```

```
print(new_text) # Output: 'hello universe'
```

```
text = "hello world"
print(text.startswith("hello")) # Output: True
```

```
text = "hello world"
print(text.endswith("world")) # Output: True
```

```
text = "hello world"
index = text.find("world")
print(index) # Output: 6
```

```
text = "hello world, hello universe"
count = text.count("hello")
print(count) # Output: 2
```

```
name = "John"
age = 30
message = "My name is {} and I am {} years old.".format(name, age)
print(message) #My name is John and I am 30 years old.
```

```
text = "Hello, World!"
```

```
# Using step argument to take every second character
```

```
result = text[::2]
```

```
print(result) #Hlo ol!
```

```
text = "Hello, World!"
```

```
# Using step argument to reverse the string
```

```
reversed_text = text[::-1]
```

```
print(reversed_text) #!dlroW ,olleH
```

```
text = "Python is amazing"
```

```
# Using step argument to extract odd-indexed characters
```

```
result = text[1::2]
```

```
print(result) #yhni mg
```

```
text = "abcdefghijklmno"
```

```
# Using step argument to skip a certain pattern (in this case, taking every third character)
```

```
result = text[::3]
```

```
print(result) #adgjmno
```

```
text = "Python is great"
```

```
# Using a negative step argument to reverse the string
```

```
reversed_text = text[::-1]
```

```
# Using negative step to extract every second character
```

```
result = reversed_text[::2]
```

```
print(result) #tera sPh
```

```
txt = "GLA2023"
```

```
print(txt.isalnum())
```

```
txt1 = "1234567"  
print(txt1.isnumeric())
```

```
txt2 = "Hello World"  
print(txt2.istitle())
```

```
txt3 = "greetings! welcome to gla university"  
print(txt3.title())
```

```
#https://www.w3schools.com/python/python\_ref\_string.asp
```

```
text = "PythonProgramming"
```

```
# Slicing by index  
substring = text[0:6]  
print(substring)
```

```
# Omitting start or end index  
partial = text[:6]  
print(partial)
```

```
partial2 = text[7:]  
print(partial2)
```

```
# Negative indexing
```



```
last_char = text[-1]
print("Last character:", last_char)
```

```
# Consider the string
st1='Hello welcome to GLA-University'
print("String: ",st1)
```

```
# Slice first 10 characters
print(st1[slice(10)])
```

```
# Slice from last 4th position to 10th position
print(st1[slice(-10,-3)])
```

```
# Slice from first character to 4th character
print(st1[slice(0,5)])
```

```
# Return the entire string using Array Slicing
print(st1[:])
```

```
# Slice from 7th character to 19th character
print(st1[6:20])
```

```
# Slice from 2nd character to 10th character with step as 3
print(st1[1:10:3])
```

# Slice from the first character to 20th character

```
print(st1[:20])
```

# Slice from 7th character to end of the string.

```
print(st1[6:])
```

#Problem Statement: Write Python code to reverse a string.

```
input_str = "hello"
```

```
reversed_str = input_str[::-1]
```

#[::-1]: This is the slice notation. It specifies how you want to extract a

# portion of the string.

#The first colon : indicates the start of the slice.

#The second colon : is the end of the slice. Since we don't specify a value after

# the second colon, it means we want to include all characters up to the end.

#The -1 is the step size. It indicates that we want to iterate over the

# string in reverse order

```
print(reversed_str) # Output: "olleh"
```

#Problem Statement: Write Python code to count the occurrences of a character in a string.

```
input_str = "hello"
```

```
char = "l"
```

```
count = input_str.count(char)
print(count) # Output: 2
```

#Problem Statement: Write Python code to check if a string is a palindrome.

```
input_str = "racecar"
is_palindrome = input_str == input_str[::-1]
print(is_palindrome) # Output: True
```

#Problem Statement: Write Python code to remove leading  
#and trailing whitespaces from a string.

```
input_str = "  hello  "
result = input_str.strip()
print(result) # Output: "hello"
```

#Problem Statement: Write Python code to remove specific characters from a string.

```
input_str = "Hello, World!"
characters_to_remove = ",!"
filtered_str = ''.join(char for char in input_str if char not in
characters_to_remove)
print(filtered_str) # Output: "Hello World"
```

#Problem Statement: Write Python code to replace occurrences of a substring in a string.

```
input_str = "Hello, World!"
substring_to_replace = "World"
replacement = "Universe"
new_str = input_str.replace(substring_to_replace, replacement)
print(new_str) # Output: "Hello, Universe!"
```

#Problem Statement: Write Python code to extract digits and letters from a string.

```
# input_str = "a1b2c3"
# digits = ''.join(char for char in input_str if char.isdigit())
# letters = ''.join(char for char in input_str if char.isalpha())
# print(digits) # Output: "123"
# print(letters) # Output: "abc"
```

```
user_input = input("Enter the string ")
data_alphabets = ""
data_digits = ""
```

```
for i in user_input:
    if i.isalpha():
        data_alphabets+=i
```

```
elif i.isdigit():
```

```
    data_digits+=i
```

```
print(f"All the digits are :- {data_digits}")
```

```
print(f"All the alphabets are:- {data_alphabets}")
```

#Write a Python program that reads a string from the user and counts the number of

#alphabets and digits in the input string.

```
input_string = input("Enter the string containing alpha and digits ")
```

```
num_digits = 0
```

```
num_alphabets = 0
```

```
for entries in input_string:
```

```
    if entries.isalpha():
```

```
        num_alphabets += 1
```

```
    elif entries.isdigit():
```

```
        num_digits += 1
```

```
print(f"Number of digits: {num_digits}")
```

```
print(f"Number of alphabets: {num_alphabets}")
```

#Write a Python program that reads a string from the user and counts the number of

#alphabets, digits and special characters in the input string for the "password" entry.

```
word = input("Enter the password for your profile ")
```

```
num_letters,num_digits,num_special_chars,num_space = 0,0,0,0
```

```
# num_letters = 0
```

```
# num_digits = 0
```

```
# num_special_chars = 0
```

```
# num_space = 0
```

```
for pw in word:
```

```
    if pw.isalpha():
```

```
        num_letters += 1
```

```
    elif pw.isdigit():
```

```
        num_digits += 1
```

```
    elif pw.isspace():
```

```
        num_space +=1
```

```
    else:
```

```
        num_special_chars += 1
```

```
print(f"Number of letters: {num_letters}")
```

```
print(f"Number of digits: {num_digits}")
```

```
print(f"Number of spaces: {num_space}")
```

```
print(f"Number of special characters: {num_special_chars}")
```

```
# Password checker assignment
```

```
# check for strong password entered by the user. Use isalpha(), isdigit(), isspace()  
function in python
```

```
# Conditions to check:-
```

```
# 1. Password atleast 12 characters (comprises of alphabets, digits and special  
characters)
```

```
# 2. Atleast one uppercase and lowercase letter
```

```
# 3. Atleast one digit and special characters (!,@,#,%,? etc)
```

```
#Continue the program until user enter a valid strong password
```

```
while True:
```

```
    # Get password from user
```

```
    password = input("Enter your password: ")
```

```
    # Condition 1: At least 12 characters
```

```
    if len(password) < 12:
```

```
        print("Password should be at least 12 characters long.")
```

```
        continue
```

```
    else:
```

```
        has_uppercase = False
```

```
        has_lowercase = False
```

```
has_digit = False
```

```
has_special = False
```

```
has_space = False
```

```
for char in password:
```

```
    if char.isalpha():
```

```
        if char.isupper():
```

```
            has_uppercase = True
```

```
        elif char.islower():
```

```
            has_lowercase = True
```

```
    elif char.isdigit():
```

```
        has_digit = True
```

```
    elif char.isspace():
```

```
        has_space = True
```

```
    else:
```

```
        has_special = True
```

```
# Conditions 2, 3, and 4
```

```
if has_uppercase and has_lowercase and has_digit and has_special:
```

```
    print("This is a strong password!")
```

```
    break
```

```
else:
```

```
    print("This is not a strong password. Please try again.")
```



#Without using in-built functions

# while True:

# password = input("Enter your password: ")

# if len(password) < 12:

# print("Password should be at least 12 characters long.")

# else:

# has\_uppercase = False

# has\_lowercase = False

# has\_digit = False

# has\_special = False

# for char in password:

# if ('a' <= char <= 'z') or ('A' <= char <= 'Z'):

# if char.isupper():

# has\_uppercase = True

# elif char.islower():

# has\_lowercase = True

# elif '0' <= char <= '9':

# has\_digit = True

# elif char in "!@#\$%^&\*()\_+-=[]{}|;:\",.<>?":

# has\_special = True

```
# if has_uppercase and has_lowercase and has_digit and has_special:
#     print("This is a strong password!")
#     break
# else:
#     print("This is not a strong password. Please try again.")
```