#Creating a string  
pancard\_number="AABGT6715H"

 #Length of the string  
print("Length of the PAN card number:", len(pancard\_number))

#Concatenating two strings  
name1 ="PAN "  
name2="card"  
name=name1+name2  
print(name)

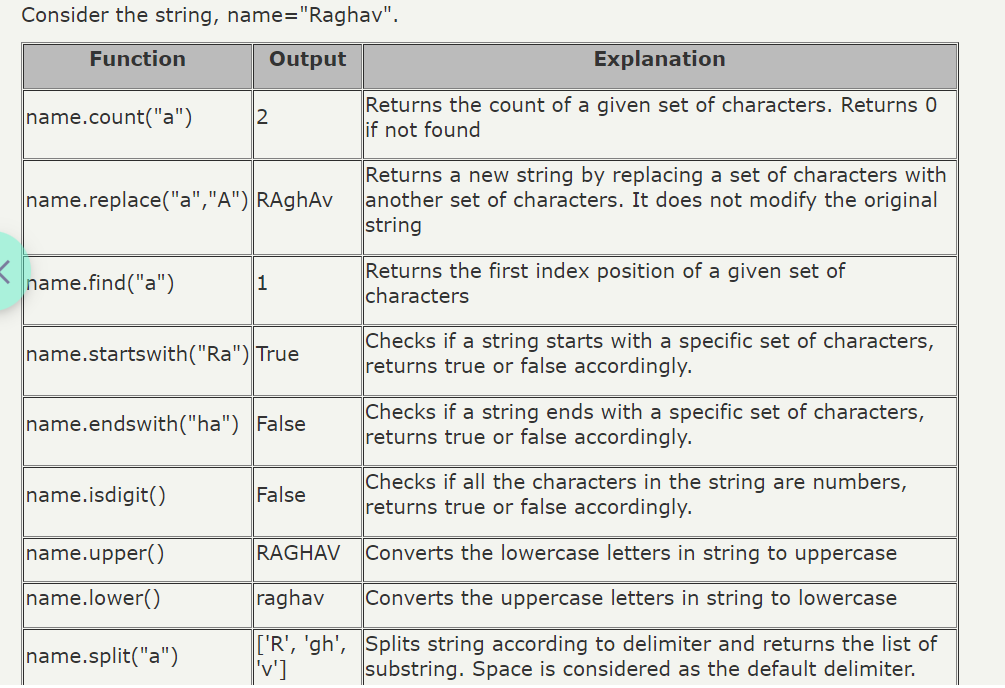
print("Iterating the string using range()")  
for index in range(0,len(pancard\_number)):  
    print(pancard\_number[index])  
  
print("Iterating the string using keyword in")  
for value in pancard\_number:  
    print(value)

print("Searching for a character in string")  
if "Z" in pancard\_number:  
    print("Character present")  
else:  
    print("Character is not present")

//------------  
String data type in Python has many inbuilt functions which make it easier to work with strings.

Consider the string, name="Raghav".

|  |  |  |
| --- | --- | --- |
| **Function** | **Output** | **Explanation** |
| name.count("a") | 2 | Returns the count of a given set of characters. Returns 0 if not found |
| name.replace("a","A") | RAghAv | Returns a new string by replacing a set of characters with another set of characters. It does not modify the original string |
| name.find("a") | 1 | Returns the first index position of a given set of characters |
| name.startswith("Ra") | True | Checks if a string starts with a specific set of characters, returns true or false accordingly. |
| name.endswith("ha") | False | Checks if a string ends with a specific set of characters, returns true or false accordingly. |
| name.isdigit() | False | Checks if all the characters in the string are numbers, returns true or false accordingly. |
| name.upper() | RAGHAV | Converts the lowercase letters in string to uppercase |
| name.lower() | raghav | Converts the uppercase letters in string to lowercase |
| name.split("a") | ['R', 'gh', 'v'] | Splits string according to delimiter and returns the list of substring. Space is considered as the default delimiter. |

//--------------------------------------------  


//-----------------------------------

**Example** -  
  
def count\_occurrences(string, target):

return string.count(target)

def replace\_chars(string, old\_chars, new\_chars):

return string.replace(old\_chars, new\_chars)

def find\_index(string, target):

return string.find(target)

def starts\_with(string, prefix):

return string.startswith(prefix)

def ends\_with(string, suffix):

return string.endswith(suffix)

def is\_all\_digits(string):

return string.isdigit()

def to\_uppercase(string):

return string.upper()

def to\_lowercase(string):

return string.lower()

def split\_string(string, delimiter):

return string.split(delimiter)

# Test the functions

if \_\_name\_\_ == "\_\_main\_\_":

name = "Raghav"

# Testing the functions

print(count\_occurrences(name, "a"))

print(replace\_chars(name, "a", "A"))

print(find\_index(name, "a"))

print(starts\_with(name, "Ra"))

print(ends\_with(name, "ha"))

print(is\_all\_digits(name))

print(to\_uppercase(name))

print(to\_lowercase(name))

print(split\_string(name, "a"))

//===========================================================

name="program"

print(name[0:2])

print(name[:3])

print(name[4:])

print(name[:])

print(name[0:5:2])

print(name[-1:-4:-1])

print(name[-1::-1])

print(name[-1::-1])

s='shubham'

print('p'in s)

print('s' in s)

print('sh' in s)

print('po' not in s)

//===================================================

**MCQ Questions -**

Q1 of 3outlined\_flag

Consider the following list of pan card numbers:  
pancard\_list=["AABGT6715H", "UFFAC4352T", "IFSBD9163K", "JOOEC1225H","RWXAFE187B"]

What is the output of the below two print statements?

print(pancard\_list[3][6], end=" ")

print(pancard\_list[4][3:])

 A. 9 OEC1225H

 B. 2 AFE187B

 C. 9163K O

 D. 225H A

**Answer - B**

 //------------------------------------------------------

Q2 of 3 What is the output of the code given below?

message="welcome to Mysore"  
word=message[-7:]  
if(word=="mysore"):  
    print("got it")  
else:  
    message=message[3:14]  
    print(message)

1. come to Myso  
   B. come to Mys  
   C. lcome to Mys  
   D. lcome to Myso

**Answer - B**

 //---------------------------------------------------

Q3 of 3 What is the output of the below code?

song="JINGLE Bells jingle Bells Jingle All The Way"  
song.upper()  
song\_words=song.split()  
count=0  
for word in song\_words:  
    if(word.startswith("jingle")):  
        count=count+1  
print(count)

A.0  
B.3  
C.2  
D.1

**Answer - D**

//===================================================

Write a python program which displays the count of the names that matches a given pattern from a list of names provided.

Consider the pattern characters to be:

1. "\_ at" where "\_" can be one occurrence of any character

2. "%at%" where "%" can have zero or any number of occurrences of a character

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| [Hat, Cat, Rabbit, Matter] | \_at -> 2 %at% -> 3 |

**Answer -**

//==============================================

1. ) Write a function, check\_palindrome() to check whether the given string is a palindrome or not. The function should return true if it is a palindrome else it should return false.  
     
   Note: Initialize the string with various values and test your program. Assume that all the letters in the given string are all of the same case. Example: MAN, civic, WOW etc.

**Answer -**   
  
//===========================================================

Q.) Problem Statement

Given a string containing uppercase characters (A-Z), compress the string using Run Length encoding. Repetition of character has to be replaced by storing the length of that run.  
  
Write a python function which performs the run length encoding for a given String and returns the run length encoded String.  
  
Provide different String values and test your program

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| AAAABBBBCCCCCCCC | 4A4B8C |
| AABCCA | 2A1B2C1A |

**Answer -**

//=====================================================