

```
#Python Practical Assignment
#-----
#NAME : MALAV THOBHANI
#ROLL NO : 30
#CLASS : MCA-1
```

```
from datetime import date
```

```
date1 = date(2023, 10, 26)
date2 = date(2024, 1, 15)
```

```
difference = date2 - date1
```

```
print(difference)
```

```
81 days, 0:00:00
```

```
import time
```

```
epoch_time_seconds = time.time()
```

```
hours = int(epoch_time_seconds // 3600)
minutes = int((epoch_time_seconds % 3600) // 60)
```

```
print(f"Time since epoch: {hours} hours and {minutes} minutes")
```

```
Time since epoch: 488309 hours and 37 minutes
```

```
from datetime import datetime
```

```
# bday='2002-10-03'
bday=input('Enter birth date in formate (YYYY-MM-DD):')
today=datetime.now()
```

```
bday_formated=datetime.strptime(bday,'%Y-%m-%d')
```

```
diff_day=(today-bday_formated).days
```

```
years = diff_day // 365
remaining_days = diff_day % 365
months = remaining_days // 30
days = remaining_days % 30
```

```
print(f'Years:{years} months:{months} days:{days}')
```

```
Enter birth date in formate (YYYY-MM-DD):2002-10-03
Years:22 months:11 days:23
```

```
import math
angles=[0,30,45,60,90]
```

```
print('Angle \t\t sin \t\t cos \t\t tan ')
```

```
for angle in angles :
    rad=math.radians(angle)
    sin=round( math.sin(rad),2)
    cos= round(math.cos(rad),2)
    if(angle==90):tan='undefiend'
    else :
        tan= round(math.tan(rad),2)
    print(f'{angle} \t\t {sin} \t\t {cos} \t\t {tan}')
```

Angle	sin	cos	tan
0	0.0	1.0	0.0
30	0.5	0.87	0.58
45	0.71	0.71	1.0

60	0.87	0.5	1.73
90	1.0	0.0	undefiend

```
import random
print('Printing 10 random numbers')
for _ in range(10):
    num= random.randint(1,100)
    print(num)
```

```
Printing 10 random numbers
71
55
9
28
60
29
38
48
49
76
```

```
userName=input("Enter your username: ")
password=input("Enter your Password: ")

correct_userName='user1'
correct_password='pass1'

if userName == correct_userName and password == correct_password:
    print("Authentication successful!")
else:
    print("Authentication failed!")
```

```
Enter your username: uss1
Enter your Password: pass1
Authentication failed!
```

```
from cryptography.fernet import Fernet
key=Fernet.generate_key()
cipher=Fernet(key)

correct_username='user123'
correct_password='pass123'.encode()
encrypted_password=cipher.encrypt(correct_password)

username=input("Enter your username: ")
password=input("Enter your password: ").encode()

if(username==correct_username and password==cipher.decrypt(encrypted_password)):
    print('Authentication Successful!')
else:
    print('Authentication Failed!')
```

```
Enter your username: user123
Enter your password: pass123
Authentication Successful!
```

```
import hashlib

# correct username and password [ hashed ]
correct_username='user123'
correct_password=hashlib.sha256('pass123'.encode()).hexdigest()

# user inputs
username=input("Enter Username: ")
password=input("Enter Password: ")

# hash the input password
password_hash=hashlib.sha256(password.encode()).hexdigest()

if(username==correct_username and password_hash==correct_password):
    print('Authentication Successful!')
else:
```

```
print('Authentication Failed!')
```

```
import base64

text='Hello$world'

# convert text into bytes
text_bytes=text.encode('utf-8')
# convert bytes into base64 bytes
base64_bytes=base64.b64encode(text_bytes)

# convert base64 bytes back to string
base64_string=base64_bytes.decode('utf-8')

print('Original string: ',text)
print('Base64 encoded string: ',base64_string)
```

```
Original string: Hello$world
Base64 encoded string: SGVsbG8kd29ybGQ=
```

```
#Exercise 1A:
# Create a string made of the first, middle and last character
str1='malav'
#first character
res=str1[0]
# mid character
mid=int(len(str1)/2)
res=res+str1[mid]
# last character
res=res+str1[-1]
print(res)
```

```
mlv
```

```
# Exercise 1B:
# Create a string made of the middle three characters
str2='JaSonAy'
mid=int(len(str2)/2)
res=str2[mid-1]+str2[mid]+str2[mid+1]
print(res)
```

```
# Exercise 2:
# Append new string in the middle of a given string
s1 = "Ault"
s2 = "Kelly"

mid=int(len(s1)/2)
x=s1[:mid]
x+=s2
x+=s2[mid:]
print(x)
```

```
Son
AuKellylly
```

```
# Exercise 3:
# Create a new string made of the first, middle, and last characters of each input string
s1 = "America"
s2 = "Japan"
res=s1[0]+s2[0]
mid1=int(len(s1)/2)
mid2=int(len(s2)/2)
res+=s1[mid1]+s2[mid2]
res+=s1[-1]+s2[-1]
print(res)
```

```
AJrpan
```

```
# Exercise 4:
# Arrange string characters such that lowercase letters should come first
str1 = 'PyNaTive'
print('Original String:', str1)
lowerlist=[]
upperlist=[]

for c in str1:
    if c.islower():
        lowerlist.append(c)
    else:
        upperlist.append(c)

sorted_list=''.join(lowerlist + upperlist)
print("Result:",sorted_list)
```

Original String: PyNaTive
Result: yaivePNT

```
# Count all letters, digits, and special symbols from a given string

str1 = "P@#yn26at^&i5ve"
def checkLettersDigitsSymbols(str1):
    char=0
    digits=0
    symbol=0
    for c in str1:
        if c.isalpha():
            char+=1
        elif c.isdigit():
            digits+=1
        else:
            symbol+=1

    print(f"Total counts {char} of chars,{digits} digits, and {symbol} symbols")
checkLettersDigitsSymbols(str1)
```

Total counts 8 of chars,3 digits, and 4 symbols

```
# Create a mixed String using the following rules
# Given two strings, s1 and s2.
# Write a program to create a new string s3 made of the first char of s1, then the last char of s2, Next, the second char of s1 and the last char of s2, and so on.

s1 = "Abc"
s2 = "Xyz"
# Expected: 'AzbycX'

# get string length
s1_length = len(s1)
s2_length = len(s2)
result=""

# find length which ever is bigger
length=s1_length if s1_length>s2_length else s2_length

# reverse s2
s2=s2[: : -1]
for i in range(length):
    if i<s1_length:
        result+=s1[i]
    if i<s2_length:
        result+=s2[i]

print(result)
```

AzbycX

```
# String characters balance Test
# Write a program to check if two strings are balanced. For example, strings s1 and s2 are balanced if all the characters in the s1 are present in s2 and vice versa.

def checkCharactersBalanceTest(s1,s2):
    flag=True
```

```

    for i in s1:
        if i in s2:
            continue
        else:
            flag=False
    return flag

s1 = "Yn"
s2 = "PYnative"
flag = checkCharactersBalanceTest(s1, s2)
print("s1 and s2 are balanced:", flag)

s1 = "Ynf"
s2 = "PYnative"
flag = checkCharactersBalanceTest(s1, s2)
print("s1 and s2 are balanced:", flag)

```

```

s1 and s2 are balanced: True
s1 and s2 are balanced: False

```

```

# Find all occurrences of a substring in a given string by ignoring the case
# Write a program to find all occurrences of "USA" in a given string ignoring the case

```

```

str1 = "Welcome to USA. usa awesome, isn't it?"
substring='USA'
# convert string into lowercase
temp_str=str1.lower()
# counts the occurances of substring
counts=temp_str.count(substring.lower())
print(f"The {substring} count is: {counts}")

```

```

The USA count is: 2

```

```

# Calculate the sum and average of the digits present in a string
# Given a string s1, write a program to return the sum and average of the digits that appear in the string, ignoring all other characters

```

```

str1 = "PYnative29@#8496"
sum=0
count=0
for i in str1:
    if i.isdigit():
        sum+=int(i)
        count+=1
print("Sum :",sum, "Average: ", sum/count)

```

```

Sum : 38 Average: 6.333333333333333

```

```

# Write a program to count occurrences of all characters within a string

```

```

str1 = "Apple"
char_dict=dict()
for char in str1:
    count= str1.count(char)
    char_dict[char]=count

print(char_dict)

```

```

{'A': 1, 'p': 2, 'l': 1, 'e': 1}

```

```
# Reverse a given string
str1 = "PYnative"

rev=str1[::-1]
#or
rev2="".join(reversed(str1))
print("Original String is:", str1)

str1 = ''.join(reversed(str1))
print("Reversed String is:", rev)
```

```
Original String is: PYnative
Reversed String is: evitanYP
```

```
# Find the last position of a given substring
# Write a program to find the last position of a substring "Emma" in a given string.

str1 = "Emma is a data scientist who knows Python. Emma works at google."
print("Original String is:", str1)

index = str1.rfind("Emma")
print("Last occurrence of Emma starts at index:", index)
```

```
Original String is: Emma is a data scientist who knows Python. Emma works at google.
Last occurrence of Emma starts at index: 43
```

```
# Split a string on hyphens
# Write a program to split a given string on hyphens and display each substring.

str1="Emma-is-a-data-scientist"
new_str=str1.split('-')

for ch in new_str:
    print(ch)
```

```
Emma
is
a
data
scientist
```

```
# Remove empty strings from a list of strings

str_list = ["Emma", "Jon", "", "Kelly", None, "Eric", ""]
new_str=[]
# using filter
new_str2=list(filter(None,str_list))

# using if condition
for s in str_list:
    if s :
        new_str.append(s)
print("Empty string:",str_list)
print("Non empty string:",new_str)
print("Non empty string:",new_str2)
```

```
Empty string: ['Emma', 'Jon', '', 'Kelly', None, 'Eric', '']
Non empty string: ['Emma', 'Jon', 'Kelly', 'Eric']
Non empty string: ['Emma', 'Jon', 'Kelly', 'Eric']
```

```
# Remove special symbols / punctuation from a string
# Given:
str1 = "/*Jon is @developer & musician"
# Expected Output: "Jon is developer musician"
print("Original string is: ", str1)
import re
# replace special symbols with ''
res=re.sub(r'[\W\s]', '', str1)
print("New string is: ",res)
```

```
Original string is: /*Jon is @developer & musician
New string is: Jon is developer musician
```

```
# Removal all characters from a string except integers
# Given:
str1 = 'I am 25 years and 10 months old'
# Expected Output: 2510

res=""
for ch in str1:
    if ch.isdigit():
        res+=''.join(ch)
print(res)
```

```
2510
```

```
# Find words with both alphabets and numbers
# Write a program to find words with both alphabets and numbers from an input string.
```

```
# Given:
str1 = "Emma25 is Data scientist50 and AI Expert"
# Expected Output:
# Emma25
# scientist50

print("The original string is : " + str1)
res=[]
temp=str1.split(" ")

for item in temp:
    if any(char.isdigit() for char in item) and any(char.isalpha() for ch in item):
        res.append(item)

print("Displaying words with alphabets and numbers")
for i in res:
    print(i)
```

```
The original string is : Emma25 is Data scientist50 and AI Expert
Displaying words with alphabets and numbers
Emma25
scientist50
```

```
# Replace each special symbol with # in the following string
# Given:
str1 = '/*Jon is @developer & musician!!'
# Expected Output: ##Jon is #developer # musician##
```

```
import re
res=re.sub(r'^\w\s',"#",str1)
print("The strings after replacement : ", res)

# or
import string

# Replace punctuations with #
replace_char = '#'

# string.punctuation to get the list of all special symbols
for char in string.punctuation:
    str1 = str1.replace(char, replace_char)

print("The strings after replacement : ", str1)
```

```
The strings after replacement : ##Jon is #developer # musician##
The strings after replacement : ##Jon is #developer # musician##
```