Python Assignment Report
This assignment covers operators, strings, and lists, offering a comprehensive exploration of Python's core functionalities. It significantly enhanced my understanding of Python fundamentals, including arithmetic operations, comparison techniques, and logical operations, alongside practical applications in string formatting and list manipulation.

Approach

1. Operators

Arithmetic Operators: Wrote code to perform fundamental arithmetic operations like addition, subtraction, multiplication, division, exponentiation, and floor division using user inputs.

Code:-
```
a=int(input('enter a number'))#take first no. from user to convert into integer
b=int(input('enter a number'))
adding=a+b
print('adding:',adding)#print the result of operation
substraction=a-b
print('substraction:',substraction)
division=a/b
print('division:',division)
modulas=a%b
print('modulas:',modulas)
multiplication=a*b
print('multiplication:',multiplication)
exponential=a**b
print('exponential:',exponential)
floor=a//b
print('floor:',floor)
```

Comparison Operators: Compared two user inputs to determine if one is greater than, less than, or equal to the other using if-else statements.

Code:-

```
a=int(input('Enter first number'))#taking input from user to convert into integer
b=int(input('enter second number'))
if a>b:#if first is greater than second, it will execute
    print('first number is greater than second number')
elif a==b:
    print('first number is equal to second number')
elif a<=b:
    print('first number is smaller than or equal to second number')
```

Logical Operators: Evaluated multiple user-provided Boolean inputs using logical operations such as AND, OR, and NOT.

Code:-

```
a=bool(input("enter first boolean(True/False):"))
b=bool(input("enter second boolean(true/false):"))
c=bool(input("enter third boolean(true/false):"))
print(a and b and c)
print(a or b or c)
print(not c)
```

2. Strings

String Manipulation: Worked with strings by finding their length, reversing the string, extracting first and last letters, and changing letter cases.

Code:-
Code:-
```
a=input('enter a word:')#find the length of input word
b=len(a)#return the no. of characters in word
c=a[0]#access the first character
d=a[-1]#access last character
e=a[::-1]#reversing the input word
f=a.upper()#input to uppercase
g=a.lower()#input to lowercase
print(a)
print(b)
print(c)
print(d)
print(e)
print(f)
print(g)
```

String Formatting: Used string formatting to create personalized messages by taking user input for name and age.

Code:-
```
Name=input('enter a name:')#to take input from user
Age=int(input('enter the age:'))
print('Hello',Name,'you are',Age,'years old')
```

Substring Search: Employed the find() method to search for a specific word within a sentence, with appropriate if-else statements to confirm its presence.

Code:-

```
sentence=input("enter a sentence:")#enter a sentence
word=input("enter a word to search:")#word they want to search in sentence
if word in sentence:
    print(sentence.index(word))
else:
    print("word not found")
```

3. Lists

List Operations: Appended user inputs (numbers) to a list, then calculated the total sum, maximum, and minimum values from the list.

Code:-

```
a=[] #empty list initialization
for i in range(1,6):#loop 5 times
    b=int(input('enter a number:'))
    a.append(b)#add a
c=sum(a)
d=max(a)
e=min(a)
print(a)
print(c)
print(d)
print(e)
```

List Manipulation: Added and removed items from a list based on user input, primarily focusing on fruit names for practice.

Code:-

```
fruits=['mango','custardapple','pineapple','orange','lemon']#list of 5 fruits
fruits+=['kiwi']#add 1 fruits
fruits.remove(fruits[1])#remove 2nd fruit
print(fruits)
```

Sorting: Sorted the list in both ascending and descending order using Python's sort functions.

Code:-
```
a=[]
for i in range(5):#using a for loop to take 5 inputs from user
    b=input(f"enter the list{i+1}")#appending it to list a
    a.append(b)
a.sort()#list in assending
a.reverse()#list in descending
print(a)
```

List Slicing: Printed specific parts of the list, particularly the first five and last five values.

Code:-

```
nos=[1,2,3,4,5,6,7,8,9,10]
print(nos[:5])#slicing first 5 elements
print(nos[5:])#slicing elements from 5 to end of list
print(nos[2:8])#slicing the elements from index 2 to 7
```

Nested Lists: Implemented a nested list structure where student names were associated with their corresponding marks, demonstrating how to store more complex data.

Code:-
```
 a=[]
for i in range(3):#loop input 3 times
    b=input("enter students name:")#enter student name
    c=[]#nested list
    for j in range(3):#loop the input 3 times
        d=float(input(f"enter mark for {b}'s subject{j+1}:"))#3 times looped
        c.append(d)
    a.append([b,c])
for i in a:
    average=sum(i[1])/len(i[1])#takes average of 3 students with their 3 sub
    print(f"{i[0]}'s average score is:{average}")
```

Key Learnings

Operators: Gained a thorough understanding of how Python handles arithmetic, comparison, and logical operations, which are crucial for decision-making processes in coding.

String Manipulation: Improved my proficiency in manipulating string data, utilizing built-in Python functions for various tasks such as formatting, case conversion, and substring searching.

List Operations: Expanded my knowledge of list manipulation, from simple additions and deletions to more complex tasks like sorting and working with nested structures, which are essential for handling multi-dimensional data

Github repository:- https://github.com/TejasPebam/TejasPebam