Python Assignment Report

Assignment Summary:

This Python project contains 11 questions designed to strengthen your programming skills through coding practice. Each question focuses on a specific skill, such as using basic arithmetic, formal reasoning, and string slicing. It covers concepts such as arithmetic, conditional expressions, Boolean logic, string functions, and function lists. The latter can include nested lists for complex data management. Completing these exercises will give you a solid foundation for solving more complex problems in Python and strengthen your programming skills.

Approach:

1.Operators –

1. QUESTION – Basic Arithmetic Operations:-

In this question, we have to write a code by taking inputs from the user to run basic operations like (Addition, Subtraction, Multiplication, Division, Modulus, Exponential, and Floor Division).

CODE : # Question - 1

```
str1 = int(input("First number"))
str2 = int(input("Second number"))
add = (str1+str2)                    # For addition
print(add)
```

```
sub = (str1-str2)                    # For subtraction
print(sub)
multiply = (str1*str2)               # For multiplication
print(multiply)
divide = (str1/str2)                 # For divison
print(divide)
modulas = (str1%str2)                # For modulas
print(modulas)
exponential = (str1**str2)           # For exponential
print(exponential)
floordivison  = (str1//str2)         # For floor divison
print(floordivison)
```

2. QUESTION - Conditional Statements:-

In this question, we have to write a code with the help of user input to check whether the first number is greater than, equal to, or less than the second number using if-else statements.

Code : # Question - 2

```
num1 = int(input("First number:"))          # Input first number
num2 = int(input("Second number:"))     # Input second number
if num1 > num2:                                 # num1 is greater
print("num1 is greater than num2") # Print num1 is greater than num2
elif num1 == num2:                            # If number are equal
   print("Both number are equal")       # Print both numbers are equal
else:
    print("The num1 is less than num2")  # If num1 is not greater or both the numbers are not equal than print num1 is less
      than num 2
```

3.QUESTION - Boolean Logic

In this question, the user enters three boolean inputs, and the input is compared and printed using the following logical operators: AND, OR, and NOT.

Code : # Question -

```
a = input("enter a boolean value 1:-").strip().lower()=="true"  # Strip() will clear all white spaces and lower will help to
    remove indentation
b = input("enter a boolean value 2:-").strip().lower()=="true"
c = input("enter a boolean value 3:-").strip().lower()=="true"
print(a and b and c)
print(a or b or c)
print(not a)
print(not b)
print(not c)
```

2.STRINGS -

4.QUESTION - String Manipulation

In this code, we have to write a string and perform various tasks on that string, such as finding the length of the string, first and last letters, reversing it, and changing its case.

Code : # Question - 4

```
str1 = input('Enter a word: ')          # Input a word from the user
a = len(str1)                           # Get the length of the string
print(a)
b = str1[:1],str1[-1]    # To get the first and last character of the string
print(b)
c = str1[::-1]           # TO get the first and last character of the string
print(c)
d = str1.upper()                 # To reverse the string
print(d)
e = str1.lower()                    # To convert the string to lowercase
print(e)
```

5.QUESTION - User Input and String Formatting

In this code, we take input from the user of their name and age, and we have to display them in one message.

Code : # Question - 5

```
name = input("Enter your name: ")                    # Input user's name
age = input("Enter your age: ")                      # Input user's age
print("Hello",name, ",you are",age,"years old.")     # To print name and age together in a sentence
```

6.QUESTION - Searching in Strings

In this code, with the help of user inputs, a sentence and a word are provided to find in the sentence. Using index and if-else statements, it will search and output whether the word is found or not.

CODE : # Question - 6

```
str1 = input("Enter a sentence: ")     # Taking input from the user for sentence
str2 = input("Enter a word to search: ")   # Taking input from the user for word
if str2 in str1:
    print(str1.index(word))                  # To get index of the word
else:
    print("Word not found")
```

3.LISTS -

7.QUESTION - List Operations with Numbers

In this question, the program allows the user to input three numbers, and then the program will do the sum of the numbers as well as show the maximum and minimum numbers in the list using a for loop and append.

Code : # Question - 7

```
num1 = []                # Making an empty list to store numbers
for i in range(1,4):               # For the input of three numbers
num2 = int(input('enter a number: '))        # Input of the number
num1.append(num2)            # For adding the number in the list
print(num1)                        # For printing the current list
num3 = sum(num1)                          # For doing addition
print(num3)
```

```
num4 = max(num1)                    # For the highest value
print(num4)
num5 = min(num1)                    # For the lowest value
print(num5)
```

8.QUESTION - List Operations with Strings

In this question, the same method is used as in Question 7 to take user input, but with fruit names as strings, and then the input is appended to the list, allowing adding and removing a fruit from the list.

Code : # Question - 8

```
str1 = []                    # Empty list to store our favorite fruits
for i in range(2):           # For the loop of 2 favorite fruits
str2 = input(f"Your Favorite Fruit {i+1}:")    # Input of favorite fruits
str1.append(str2)    # For adding that favorite fruit in the list
print(f"The First list:{str1}")
str3 = input("Enter the other fruit you want to add:")  # Another input for the favorite fruit in the list
str1.append(str3)            # For that fruit in the list
print(f"added fruit list:{str1}")
str4 = str1.pop(1)       # To remove the second fruit from the list
print(f"The removed and updated list of fruits:{str1}")
```

9.QUESTION - Sorting Numbers

In this code, the program takes five numbers from the user, then stores them in a list, and using the sort() function, it sorts them in ascending and descending order.

Code : # Question - 9

```
numbers = []                 # Empty list for storing numbers
print("Please enter 5 numbers:")       # Input of 5 numbers
for i in range(5):                     # Loop of 5 numbers
num1 = int(input(f"Enter number {i+1}: "))      # Input a number
numbers.append(num1)          # To add that number in the list
asec = sorted(numbers)        # To sort them in ascending order
print("Numbers in ascending order:", asec)
```

```
desc = sorted(numbers, reverse=True)      # To sort them in descending order
print("Numbers in descending order:", desc)
```

10.QUESTION - Slicing Lists

In this code, it extracts the first five elements, the last five elements, and the middle section from the list using slicing syntax.

Code : # Question - 10

```
list1 = ["1","2","3","4","5","6","7","8","9","10"]
x = list1 [:5]            # Extracts the first 5 elements from list1
print(x)
y=list1 [-5:]          # Extracts the last 5 elements from list1
print(y)
z=list1 [2:8]          # Extracts elements from index 2 to 7 (index 8 is not included)
print(z)
```

11.QUESTION - Student Marks and Average Calculation

In this code, the program collects the names and marks of three students and then calculates and prints the average marks of each student using the append and range functions.

Code : # Question - 11

```
a=[]
for i in range(3):
 b=input("Enter Student Name")
 c=[]
 for j in range(3):
 d=int(input("Subjects mark"))
 c.append(d)
 a.append([b,c])
 print(a)
 for k in a:
 avg=sum(k[1])/len(k[1])
```

```
print((k[0]),"'s average score is ",(avg))
```

Key Learning -

1. **String Manipulation:** By investigating various approaches to handling and processing string data, I enhanced my proficiency in manipulating strings using built-in Python functions.

2. **List Operations:** I became more familiar with list operations, such as how to create, edit, sort, and slice lists. I also learned how to work with nested structures, which are used to store more complex data.

3. **Operators:** I learned how to effectively carry out a variety of calculations and checks using arithmetic, comparison, and logical operators.