ALLIANCE UNIVERSITY

Alliance College of Engineering and Design

REPORT

PYTHON

Assignment 1: Exercises on Operators, Strings, and Lists

Submitted by: Noora Syrajudeen

MCA 2024 sec 1

PROV/MCA/7/025

Assignment 1: Exercises on Operators, Strings, and Lists

Python is a high-level programming language known for its simplicity and versatility. Among its various features, operators, strings, and lists play crucial roles in data manipulation and control flow.

Operators

Operators in Python are special symbols that perform operations on variables and values. They can be classified into several categories:

- Arithmetic Operators
- Comparison Operators
- Logical Operators
- Assignment Operators

Strings

Strings are one of the fundamental data types in Python, representing sequences of characters. They are used to handle textual data and can be defined using single quotes (') or double quotes (")

Lists

Lists are versatile data structures in Python that can hold an ordered collection of items. They can contain elements of different data types, including numbers, strings, and even other lists. Lists are defined using square brackets ([]) and support various operations, such as indexing, slicing, appending, and removing elements.

Part 1: Operators

Exercise 1: Arithmetic Operators

Step 1: Get two numbers from the user

Step 2: Perform basic arithmetic operations , modulus , exponentiation operator and floor divison

Step 3: Display the results

Exercise 2: Comparison Operators

Step 1: Get two numbers from the user

Step 2: Check if the first number is greater than the second

Step 3: Check if the first number is equal to the second

#Step 4: Check if the first number is less than or equal to the second

Exercise 3: Logical Operators

Step 1: Take three Boolean values as input

Step 2: Use logical operators to combine the Boolean values. uses the and operator to determine if all three values are True, It uses the or operator to check if at least one value is True , And applies the not operator to each Boolean value

Step 3: Display the results

Part 2: Strings

Exercise 4: String Manipulation

Step 1: Take a string input from the user

Step 2: Display the length of the string using len()

Step 3: Display first and last character using indexing(:)

Step 4: Display String in reverse order using(::-1)

Step 4: Display String in uppercase and lowercase

Exercise 5: String Formatting

#Step 1: get the user's name from user

#Step 2: get the user's age from user

#Step 3: Display the name and age

Exercise 6: Substring Search

Step 1: get a sentence input from the user

Step 2: get a word to search in the sentence

Step 3: Check if the word exists in the sentence , use the .find() method to check if the word exists in the sentence. If the word is found, it returns the starting index .

Part 3: Lists

Exercise 7: List Operations

#Step 1: Create a list of 5 numbers (input from the user)

Step 2: Display the sum of all the numbers in the list. calculates the sum of the list using the sum()

Step 3: Find the largest and smallest number in the list. finds the largest and smallest numbers using the max() and min() functions.

Exercise 8: List Manipulation

Step 1: Create a list of 5 favourite fruits

Step 2: Add one more fruit to the list using .append()

Step 3: Remove the second fruit from the list using .pop()

Exercise 9: Sorting a List

Step 1: Ask the user to input a list of 5 numbers

Step 2: Sort the list in ascending order and descending order using sorted()

Exercise 10: List Slicing

Step 1: Print the first 5 elements , using [:5] retrieves elements from the start of the list up to (but not including) index 5.

Step 2: Print the last 5 elements , using [-5:] retrieves the last 5 elements by using negative indexing.

#Step 3: Print the elements from index 2 to index 7 , using [2:8] retrieves elements starting from index 2 up to (but not including) index 8.

Bonus Challenge

Exercise 11: Nested List

Step 1: Create a list of 5 numbers (input from the user)

Step 2: Display the sum of all the numbers in the list, It calculates the sum of the list using the sum() function

Step 3: The largest and smallest number in the list , It finds the largest and smallest numbers using the max() and min() functions

Conclusion

Understanding operators, strings, and lists is fundamental for effective programming in Python. These concepts form the basis for data manipulation, control flow, and building complex applications.

Basic Arithmetic Operations:

We learned how to use arithmetic operators and handle user input effectively.

Comparison of Two Numbers:

We practiced using comparison operators and conditional statements to derive logical outcomes.

Boolean Logic:

This exercise reinforced our understanding of Boolean values and logical operations.

String Manipulation:

We explored string indexing, slicing, and built-in string methods.

User Interaction:

This highlighted the importance of string formatting and user engagement.

Word Search in a Sentence:

We learned to handle string search operations and the concept of indexing in Python.

GitHub repository Link: