Python Programming Language

Lecture 8: Control Flow Statement, Book Review and Problem Solving

Solving

Hafsa Sultana

Computer Science and Engineering

Today’s Learning Objectives

- Exam Problem Solving
- Control Flow Statement
- For Loop
- Book Review
- Practical Exercises

Problem 1

non_start

Given 2 strings, return their concatenation, except omit the first char of each. The strings will be at least length 1.

non_start('Hello', 'There') → 'ellohere'

non_start('java', 'code') → 'avaode'

non_start('shotl', 'java') → 'hotlava"

Solution 1

non_start

```
# Step 1: Take User input strings
str1 = input("Enter first String: ")          #Hello
str2 = input("Enter second String: ")    #World

# Step 2: Remove the first character from each string
str1_sliced = str1[1:]                              # "ello"
str2_sliced = str2[1:]                              # "orld"

# Step 3: Concatenate the sliced strings
result = str1_sliced + str2_sliced         # "ello" + "orld" = "elloworld"

# Step 4: Print the result
print(result)                                       # Output: "elloworld"
```

Problem 2

sum_double

Given two int values, return their sum. Unless the two values are the same, then return double their sum.

sum_double(1, 2) → 3

sum_double(3, 2) → 5

sum_double(2, 2) → 8

''

Solution 2

sum_double

```
# Step1:  Take user input for num1 and num2
num1 = int(input("Enter the first number (num1): "))
num2 = int(input("Enter the second number (num2): "))

# Step 2: Check if the numbers are equal
if num1 == num2:
    result = 2 * (num1 + num2)
else:
    result = num1 + num2

#Step 3:  Print the result
print("The result is:", result)
```

Problem 3

make_tags

The web is built with HTML strings like "<i>Yay</i>" which draws Yay as italic text. In this example, the "i" tag makes <i> and </i> which surround the word "Yay". Given tag and word strings, create the HTML string with tags around the word, e.g. "<i>Yay</i>".

make_tags('i', 'Yay') → '<i>Yay</i>'

make_tags('i', 'Hello') → '<i>Hello</i>'

make_tags('cite', 'Yay') → '<cite>Yay</cite>'

Solution 3

make_tags

```
# Step 1: Take user input for the tag and the word
tag = input("Enter the tag: ")
word = input("Enter the word: ")

# Step 2: Construct the HTML tag
result = '<' + tag + '>' + word + '</' + tag + '>'

# Step 3: Print the result
print(result)
```

Control Flow Statements

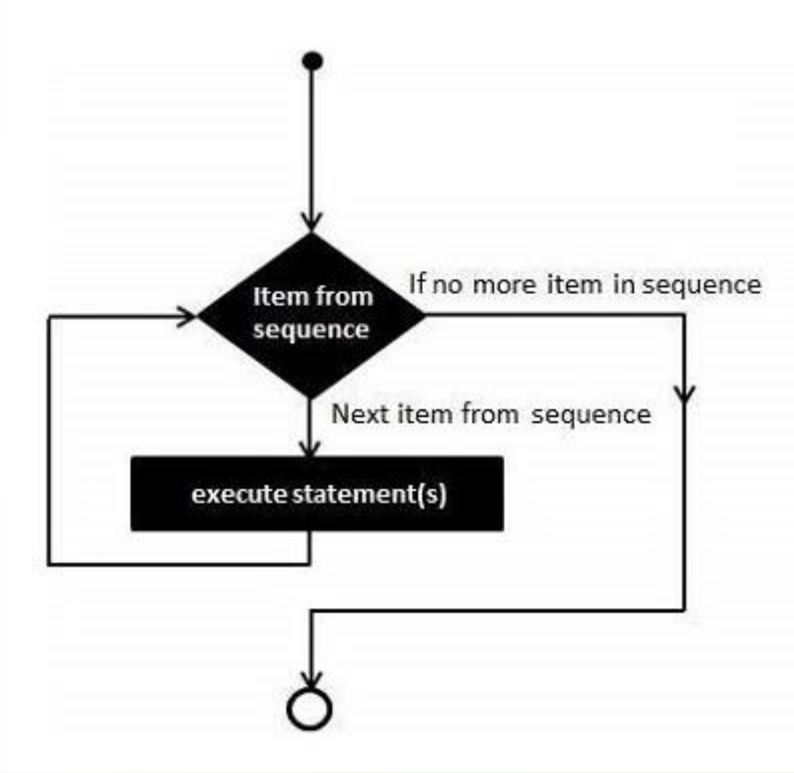
- Control flow statements determine the order in which statements are executed in a program.
- They allow you to make decisions and repeat actions.

Types of Control Flow Statements:

- Conditional Statements: if, elif, else
- Loops: for, while
- Other: break, continue, pass

Introduction to For Loops

- A for loop is used to repeat a block of code a certain number of times.
- It iterates over a sequence (like a list or a range).
- The following diagram illustrates a loop statement :



- Problem: Print "Python is fun!" 7 Times

-
  ```
  for num in range(7):
    print("Python is fun!")
  ```

Basic Structure of a For Loop

- **Syntex:** for variable in sequence:

  # Code to execute
- Variable takes on each value in the sequence one at a time.
- The indented code block runs for each value.

- **Example:** Print 1 to 5 numbers.
- **Normal Code :**

```
print(1)
print(2)
print(3)
print(4)
print(5)
```

- **Using Loop:**

```
for num in range(1,6):
  print(num)
```

Looping Through a String

- Loop through the letters in the word "banana":

-
  ```
  for x in "banana":
    print(x)
  ```

-
  ```
  fruits = ["apple", "banana", "cherry"]
  for x in fruits:
    print(x)
  ```

The range() Function

- The range() function returns a sequence of numbers, starting from 0 by default, increments by 1 (by default), and ends at a specified number.

-
  ```
  for x in range(6):
    print(x)
  ```

-
  ```
  for x in range(1,6):
    print(x)
  ```

-
  ```
  for x in range(1,10,2):
    print(x)
  ```

Practical Example 1

- Sum of First 5 Numbers

- Task: Calculate and print the sum of numbers from 1 to 5.
- Hint: Use a variable to keep track of the sum inside the loop.
- Solution:

```
Total=0
for num in range(1,6):
   Total=Total + num
print(Total)
```

# Output: 15

Practical Example 2

- Print Even Numbers in 1 to 10
- Task: Print even numbers from 1 to 10.
- Hint: Use if num % 2 == 0 inside the loop.

```
for num in range(0, 11, 2):
        print(num)
```

Practical Exercise 3

- Title: Calculate the multiplication of numbers from 1 to 5.
- Input: 5
- Output: 120

Practical Exercise 4

- Title: Print the squares of numbers from 1 to 10
- Input: 10
- Output:

```
1 squared is 1
2 squared is 4
3 squared is 9
4 squared is 16
5 squared is 25
6 squared is 36
7 squared is 49
8 squared is 64
9 squared is 81
10 squared is 100
```

Book Review

Any Question?

References

- https://www.w3schools.com/python/
- https://www.tutorialspoint.com/python/