Python Recap: Lists

School Year 2023-2024

Valsalice

Lists

Modifiable containers for data.

With **variables**:

```
num1 = 42
num2 = 100
num3 = 10

print(num1)
print(num2)
print(num3)
```

With a **list**:

```
nums = [42, 100, 8]

print(nums)
```

Lists

Anatomy of a list:

1. Uses square brackets **[ ]**
2. Elements separated by comma **,**
3. Can take any values

```
nums = [42, 100, 8]
```

```
data = [4.2, "cat", 8]
```

Modifying Lists

Adding new elements:

1. To insert at the back: **append**
2. To insert in any position: **insert**

```
nums = [42, 100]

nums.append(8)
nums.insert(0, 200)
nums.append(51)

print(nums)
```

Exercise

Complete the **1_0.py** & **1_1.py** programs.

- **1_0**: Initialise a list called **nums** with three numbers inside:
  - 6, 90 and 43

- **1_1**: Add the following numbers to the **nums** list: 3, 21, 17
  - **HINT**: Use **append** or **insert**!

Solutions 1_0 & 1_1

1_0

```
nums = [6, 90, 43]

print(nums)
```

1_1

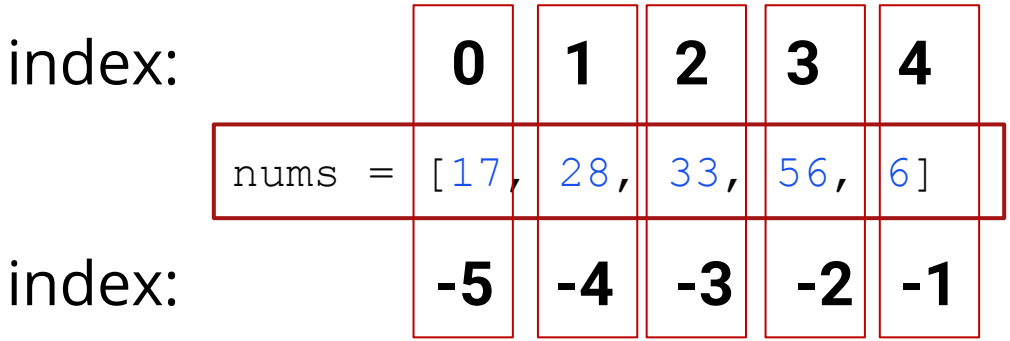
```
nums.append(3)
nums.append(21)
nums.append(17)

print(nums)
```

Accessing List Elements

To access list elements you can use the **[index]** operator.

NOTE: List indices start from **0**




```
print(nums[0])
```

```
print(nums[3])
```

```
print(nums[-2])
```

Exercise

Complete the **1_2.py** & **1_3.py** programs.

- **1_2**: Given the **nums** list defined in previous exercises, print the first, third, last and second-last elements.

- **1_3**: Write a program that generates a list **data** with all the numbers from 1 to 20.

  Then print the 5th, 10th, 15th and last element.

Solutions 1_2 & 1_3

1_2

```
print(nums[0])
print(nums[2])
print(nums[-1])
print(nums[-2])
```

1_3

```
data = []

for x in range(1, 21):
    data.append(x)

print(data[4])
print(data[9])
print(data[14])
print(data[-1])
```

Concatenating Lists

You can concatenate lists with the **extend** function.

Otherwise you can also use addition.

```
left = [1, 2, 3]
right = [4, 5, 6]

left.extend(right)
print(left)
```

```
left = [1, 2, 3]
right = [4, 5, 6]

new = left + right
print(new)
```

Exercise

Complete the **1_4.py** program.

- Create list **numsL** with all numbers from 0 to 9 (inclusive)
- Create list **numsM** with a single number: 10
- Create list **numsR** with all numbers from 11 to 19 (inclusive)

Concatenate the three lists and print the output list out

Solution 1_4

```
numsL = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
numsM = [10]
numsR = [11, 12, 13, 14, 15, 16, 17, 18, 19]

print(numsL + numsM + numsR)
```

Removing List elements

You can remove elements in a list with the **pop** function.

You may optionally pass an index, default is **-1**.

```
data = [4, 8, 12, 16, 20]
data.pop()
print(data)
```

```
data = [4, 8, 12, 16, 20]
data.pop(2)
print(data)
```

```
data = [4, 8, 12, 16, 20]
num1 = data.pop(2)
num2 = data.pop(-2)
print(num1 + num2)
print(data)
```

Exercise

Complete the **1_5.py** program.

1. Create a list `data` that stores the 6 integers between 10 and 15 (inclusive).
2. Using the `data` list, append 2 additional integers: `20` and `65`
3. Then create an empty list `blank`
4. Insert the integer `34` at Index 0 of `data`
5. Remove the last element of `data` and insert it at the beginning of `blank`
6. Print the two lists concatenated

What is the output of the program?

Solution 1_5

```
# 1
data = [10, 11, 12, 13, 14, 15]
# 2
data.append(20)
data.append(65)
# 3
blank = []
# 4
data.insert(0, 34)
# 5
last_element = data.pop()
blank.insert(0, last_element)
# 6
print(blank + data)
```

Output:

```
[65, 34, 10, 11, 12, 13, 14, 15, 20]
```

Additional List Functions

Additional functions that operate on lists

- Get the length of the list: **len**

```
len([4, 8, 10, 12])
```

```
len([-3])
```

```
len([])
```

- Get the max/min elements in a list: **max** and **min**

```
max([4, 8, -2, 0])
```

```
min([4, 8, -2, 0])
```

- Get the sum of all elements in a list: **sum**

```
sum([4, 8, -2, 0])
```

```
sum([-3])
```

Exercise

Complete the **1_6.py** program.

For the following list `nums` calculate the:

- max value
- min value
- mean value (i.e. the average)

HINT: For the average use a combination of `sum` and `len`

Solution 1_6

```
nums = [4, 8, -17, 23, 55]

max_value = max(nums)
min_value = min(nums)
avg_value = sum(nums) / len(nums)

print(max_value)
print(min_value)
print(avg_value)
```

Iterating Lists

Python provides multiple ways to **iterate over lists**.

The most used methodologies are:

Index-iteration:

```
nums = [10, 20, 30, 40]
for i in range(len(nums)):
    print(nums[i])
```

For-each loop:

```
nums = [10, 20, 30, 40]
for num in nums:
    print(num)
```

The output of the two snippets is identical

Exercise

Complete the **1_7.py** & **1_8.py** programs.

- **1_7**: Given the **nums** list use index-based iteration to print out every element in the list

- **1_8**: Given the **nums** list use a for-each loop to print out every element in the list

Solutions 1_7 & 1_8

1_7

```
items = ["apple", "banana", "cherry"]

for i in range(len(items)):
    print(items[i])
```

1_8

```
items = ["apple", "banana", "cherry"]

for item in items:
    print(item)
```

Exercise

Complete the **1_9.py** program.

- Create a list **numbers** with integers from **1** to **10**.
- Use a **for-each loop** to calculate the sum of all even numbers in the list.
- Print the sum.

Solution 1_9

```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

sum_even = 0

for num in numbers:
    if num % 2 == 0:
        sum_even += num

print(sum_even)
```

End of Python Recap

Don’t hesitate to reach out

on Classroom with any questions!