Newton School of Technology

13

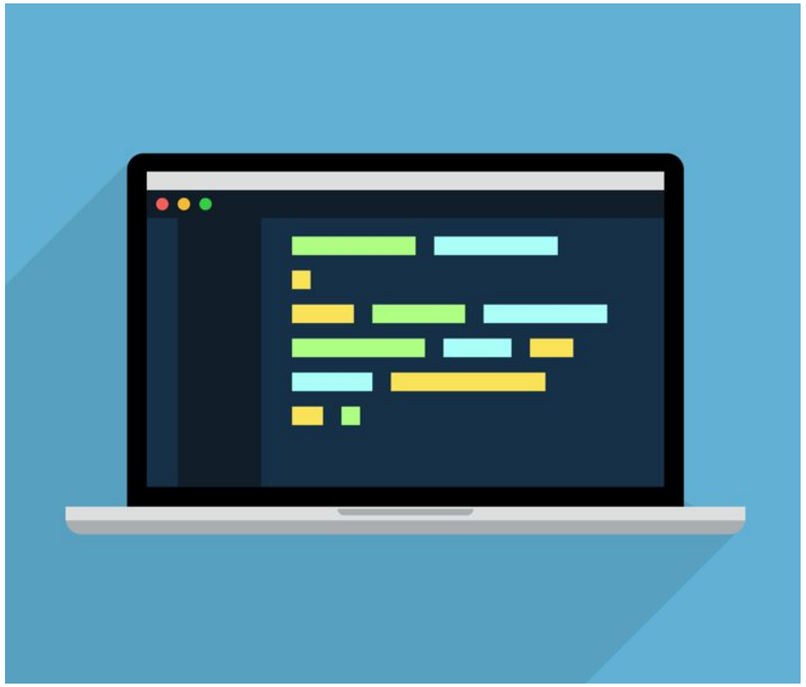
Lists in Python-2

by Gladden Rumao

C01: Problem Solving with Programming

Quick Recap :

- Creating Lists
- Indexing in List
- Accessing Elements in a List
- Slicing in List
- Modifying a List



Newton School of Technology

Question : Holidays - Modify List

Newton School of Technology

Iterating Lists

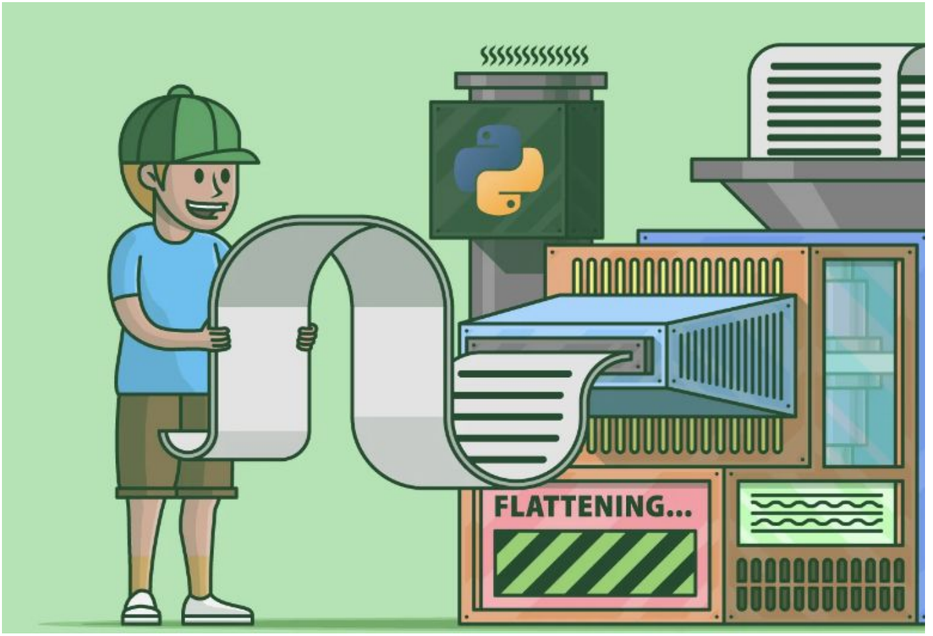
Iterating with ‘for’ loop:

Newton School of Technology

The **for** loop in Python is commonly used for iterating over **sequences** like lists. It iterates through **each element** in the list sequentially.

Syntax:

```python
for element in list:
    # do something with element
```



Example using ‘for’ loop

Newton School of Technology

Code:

```python
# Example of iterating over a list using for loop
fruits = ['apple', 'banana', 'cherry']

# Using for loop
for fruit in fruits:
    print(fruit)
```

Output:

```
apple
banana
cherry
```

Newton School of Technology

Question : Search by Title

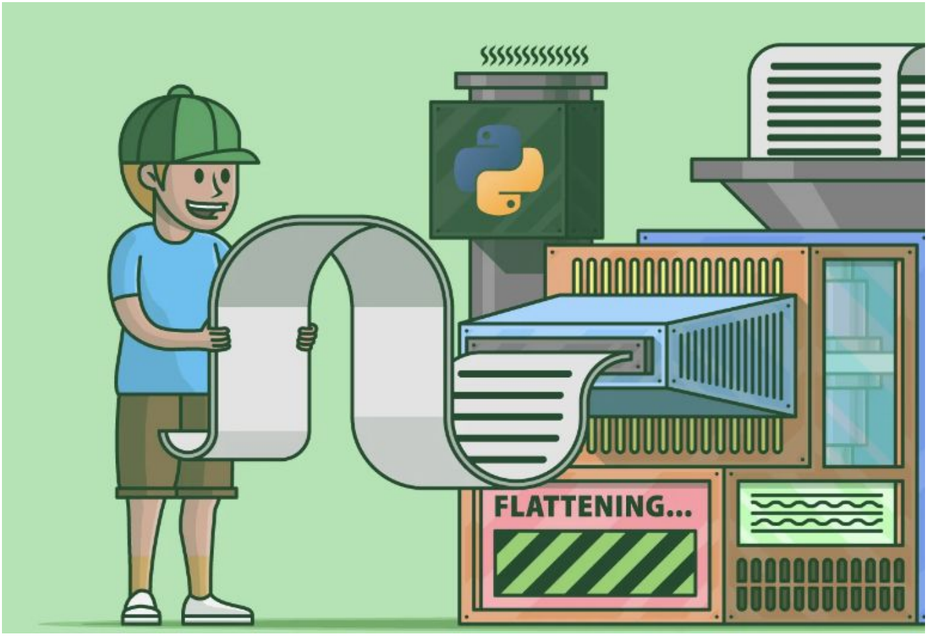
Iterating with ‘while’ loop:

Newton School of Technology

The **while** loop in Python executes a **block of code** as long as a specified condition is **true**. You can use it to iterate over a list by **manually** managing an **index** variable.

Syntax:

```python
index = 0
while index < len(list):
    element = list[index]
    # do something with element
    index += 1
```



Example using ‘while’ loop

Newton School of Technology

Code:

```python
# Example of iterating over a list using while loop
fruits = ['apple', 'banana', 'cherry']

# Using while loop
index = 0
while index < len(fruits):
    print(fruits[index])
    index += 1
```

Output:

```
apple
banana
cherry
```

Newton School of Technology

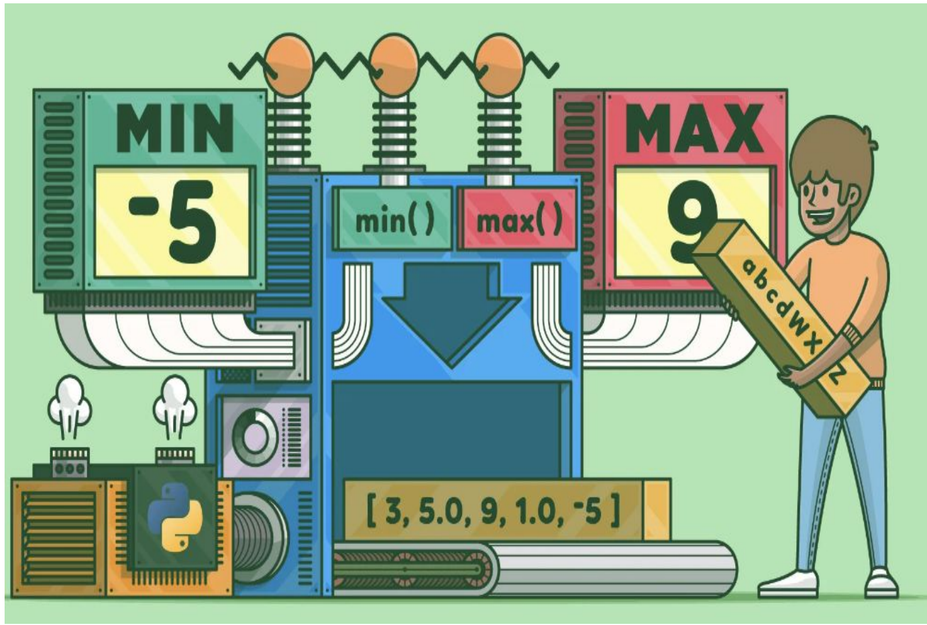
Built in Functions with Lists

max() and min()

Newton School of Technology

Purpose: Finding maximum or minimum value in a list.

Syntax:
max_value= max(list)
min_value = min(list)




```python
numbers = [5, 2, 9, 1, 7]
max_number = max(numbers)  # Output: 9
min_number = min(numbers)  # Output: 1
```

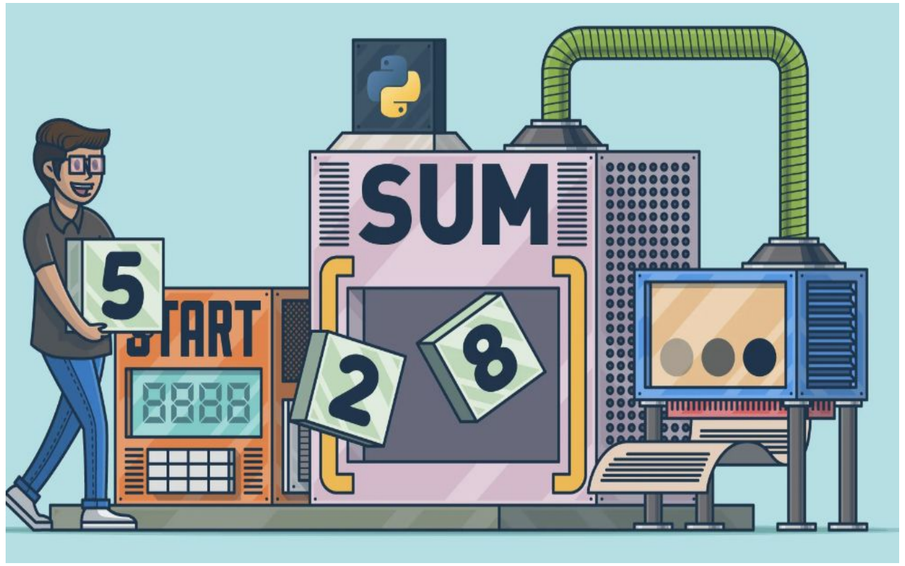
sum()

Newton School of Technology

Purpose: calculating sum of all the elements in the list.

Syntax:
total=sum(list)

```python
numbers = [5, 2, 9, 1, 7]
total_sum = sum(numbers)  # Output: 24
```



any() and all()

Newton School of Technology

Purpose: Checking conditions across all elements in a list.

Syntax:

any_condition = any(condition for element in list)

all_condition = all(condition for element in list)

any() and all()

Newton School of Technology

```
numbers = [5, 2, 9, 1, 7]

# Check if any element is greater than 10
any_greater_than_10 = any(num > 10 for num in numbers)  # Output: False

# Check if all elements are greater than 0
all_greater_than_0 = all(num > 0 for num in numbers)  # Output: True
```

Quiz Time!

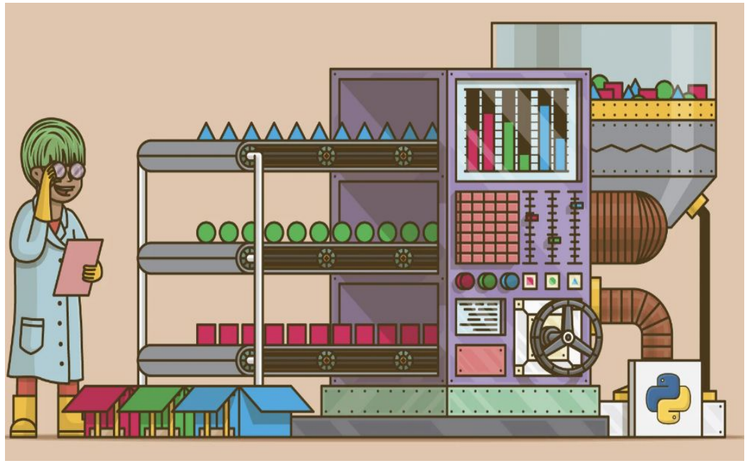
Newton School of Technology

Important List methods

sort() function :

This **modifies** the **original** list to sort its elements in **ascending** order.

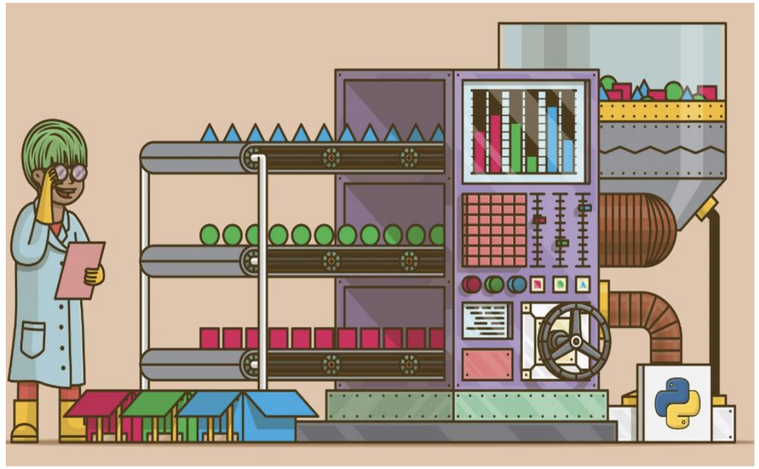
```python
my_list = [3, 1, 4, 1, 5, 9, 2]
my_list.sort()
print(my_list)  # Output: [1, 1, 2, 3, 4, 5, 9]
```



sorted() function :

Newton School of Technology

This function **returns** a **new list** with the elements sorted **without modifying** the original list.



```python
my_list = [3, 1, 4, 1, 5, 9, 2]
sorted_list = sorted(my_list)
print(sorted_list)  # Output: [1, 1, 2, 3, 4, 5, 9]
```

reverse() function :

Newton School of Technology

This method **reverses** the elements of a list **in place**.

```python
my_list = [1, 2, 3, 4, 5]
my_list.reverse()
print(my_list)  # Output: [5, 4, 3, 2, 1]
```

count() function :

Newton School of Technology

This method **counts** the number of **occurrences** of a specified **element** in a list.

```python
my_list = [1, 2, 2, 3, 2, 4, 2, 5]
count_of_twos = my_list.count(2)
print(count_of_twos)  # Output: 4
```

index() function :

Newton School of Technology

This method **returns** the **index** of the **first occurrence** of a specified **element** in a list.

python

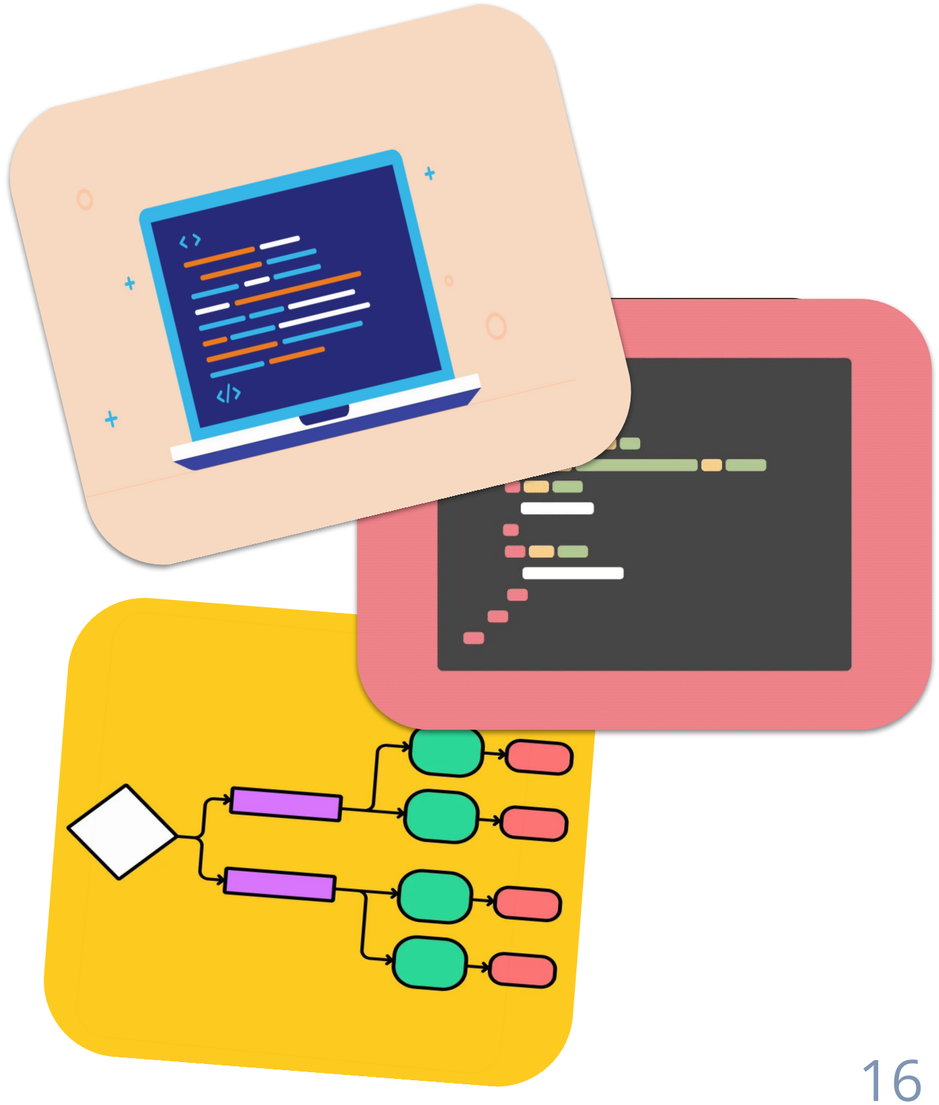
```python
my_list = ['a', 'b', 'c', 'b', 'd']
index_of_b = my_list.index('b')
print(index_of_b)  # Output: 1
```

Newton School of Technology

Question – Traffic Monitoring

Summary

- **Iterating Lists using for/while loops:** Using loops to iterate through each item in a list sequentially**.**
- **Built-in List Functions:**
  - **min(), max(), any(), all(), sum()**
- **List Methods:**
  - **sort(), sorted(), reverse(), count(), index()**



Thank You!