Lists

Summary:

- How are lists defined: `list0 = [“a”, 1, 2, “b”]`

- Lists can have elements of all data types:

    - String Data Type: `list1 = [“abc”, “xyz”, “pqr”]`
    - Int Data Type: `list2 = [1, 2, 3, 4]`
    - Float Data Type: `list3 = [3.14, 1.09, 2.13]`
    - Bool Data Type: `list4 = [True, False, False, True]`

- A single list can have elements of different data types:

```
list5 = [“abc”, 2, “xyz”, 3.14, True, 7, 9.14, False]
```

- Another way of defining lists:

```
list6 = list((“abc”, 2, “xyz”, 3.14, True, 7, 9.14, False))
```

- Lists are ordered:

The list `[1, 2, 3, 4]` is different from the list `[2, 1, 4, 3]`.

- Lists can have duplicate elements:

Example: `[1, 1, 2, 3]` and `[1, 2, 3, 1]` are valid lists, and are different from `[1, 2, 3]`.

- Lists are changeable: The elements of a list can be changed and remove, and new elements can be added too.

- Accessing Elements of a List:

    - Indexing:

        - As mentioned earlier that the elements of a list are ordered, their numbering is known as ‘indexes’.

        - The indexes of a list start from 0 instead of how our usual counting starts from 1.

        - Example: In the list, `list7 = [5, 4, 9, 2]`, `list7[0] = 5`, `list7[1] = 4`, `list7[2] = 9`, `and list7[3] = 2.`

        - Indexing can be negative also, i.e., indexing that starts from the last element instead of the first element.

        - Negative indexing starts from -1, and continues negatively, i.e., -2, -3, -4, -5, and so on.

        - Example: In the list, `list7 = [5, 4, 9, 2]`, `list7[-1] = 2`, `list7[-2] = 9`, `list7[-3] = 4`, `and list7[-4] = 5.`

- Adding Elements to a List:

    - Append Function:

        - Append is an in-built function in python which lets us add an element to the end of a list.

        - Append function takes only one argument which is the element that has to be added.

        - Example: `list8 = [‘a’, ‘b’, ‘c’]`
          `list8.append(‘d’)`

          The new value of `list8` is: `[‘a’, ‘b’, ‘c’, ‘d’]`

    - Insert Function:

        - Insert is an in-built function in python which lets us add an element to a list as the position of our choice.

        - Insert function takes two arguments, first one being the index of the position where the element has to be added, and the second one being the element that has to be added.

        - Example: `list8 = [‘a’, ‘b’, ‘c’]`
          `list8.insert(0, ‘d’)`

          The new value of `list8` is: `[‘d’, ‘a’, ‘b’, ‘c’]`

    - Example: `list9 = [‘x’, ‘y’, ‘z’]`
      `list9.insert(-1, ‘w’)`

      The new value of `list9` is: `[‘x’, ‘y’, ‘z’, ‘w’]`

- Removing Elements of a List:

  - Pop Function:

    - Pop is an in-built function in python that removes an element of the list using index position.

    - When no argument is passed through the pop function, by default, the element which gets removed is the last one. Otherwise, the element at the index value passed as the argument gets removed.

    - Example: `list10 = [‘x’, ‘y’, ‘z’]`
      `list10.pop()`

      The new value of `list10` is: `[‘x’, ‘y’]`

    - Example: `list11 = [‘x’, ‘y’, ‘z’]`
      `list11.pop(0)`

      The new value of `list11` is: `[‘y’, ‘z’]`

  - Remove Function:

    - Remove is an in-built function in python that removes an element of the list using its value.

    - Remove function takes a single argument, which is the value of the element that has to be removed.

    - If multiple elements with the same value exists, the one occurring first gets removed.

    - Example: `list12 = [‘x’, ‘y’, ‘z’]`
      `list12.remove(‘y’)`
      The new value of `list12` is: `[‘x’, ‘z’]`

    - Example: `list13 = [‘x’, ‘y’, ‘x’, ‘z’]`
      `list13.remove(‘x’)`
      The new value of `list13` is: `[‘y’, ‘x’, ‘z’]`

Exercises:

- Exercise 1: Write a program to swap the first element of the list with the last element of the list.

- Exercise 2: Define a list of integers. Write a program to remove all odd elements of the list.

- Exercise 3: Use the list `li0` given below, and remove all the empty spaces from the list.
  `li0 = ['Python', ' ', ' ', 'is', ' ', 'the', ' ', 'best', ' ', ' ']`

- Exercise 4: Write a program to make a list of all numbers from 1 to 100, where if the number is a prime number, it gets added at the end, otherwise it gets added in the front.