More List methods

```
# Max function
# Min function
# Sum function
# Count
# Index
# Reverse
# L + L1
# In operator in list
```

Lists Continued

Challenge: Given a list give another list with square of each element

```
runs = [100, 150, 99, 20, 99, 200, 99, 120]

# getting square of each elements

l = [2, 4, 6]

# iterate on the list

for i in l:
    print(i)
```

```
2
4
6
```

```
# squared of each elements

for i in l:
    print(i * i)
```

```
4
16
36
```

```
# adding in new list

new = []

for i in l:
    # append them in new
    new.append(i * i)
```

```
print(new)
```

```
[4, 16, 36]
```

Taking List as input

```
# 3
# 2 4 5

# Length is n
n = int(input())
```

```
"1 2 3 4".split()
```

```
['1', '2', '3', '4']
```

```
l = input().split()

new = []

for i in l:
    new.append(int(i))
new
```

```
 1 2 3
```

```
[1, 2, 3]
```

```
# Final code

# Length is n
n = int(input())

l = input().split()

new = []

for i in l:
    new.append(int(i))
new
```

```
 3
 2 4 6
```

```
[2, 4, 6]
```

```
# Using map

n = int(input())

li = list(map(int, input().split()))
```

```
 3
 2 4 6
```

```
# Square of each element

l = [2, 3, 4]

def square(i):
    return i * i

square(2)
```

```
4
```

```
square(3)
```

```
9
```

```
square(4)
```

```
16
```

```
new = []

for i in l:
    new.append(square(i))

new
```

```
[4, 9, 16]
```

```
l
```

```
[2, 3, 4]
```

```
new = list(map(square, l))

new
```

```
[4, 9, 16]
```

Updating a list

```
runs
```

```
[100, 150, 99, 20, 99, 200, 99, 120]
```

```
runs[0]
```

```
100
```

```
runs[1]
```

```
150
```

```
runs[0] = 150
```

```
runs
```

```
[150, 150, 99, 20, 99, 200, 99, 120]
```

```
# Quizzes
```

```
l = input()
#input = 5 4 6
print(type(l))
```

```
 5 4 6
```

```
<class 'str'>
```

```
user_values = [2, 5, 9]
user_values[2] = user_values[2] + 1

print(user_values)
```

```
[2, 5, 10]
```

```
user_values = [3, 5, 9]
user_values[1] = user_values[1] + 1
user_values[2] = user_values[2] + 2

print(user_values)
```

```
[3, 6, 11]
```

```
user_values = [1, 6, 8]
user_values[1] = user_values[0]

print(user_values)
```

```
[1, 1, 8]
```

```
user_values = [3, 6, 7]
user_values[1] = user_values[2]
user_values[2] = user_values[0]

print(user_values)
```

```
[3, 7, 3]
```

Wrtie a python function to swap two values in a list

```
# swap them

a = 3
b = 4

# Multiple assignment in python

a, b = 4, 7

print(a, b)
```

```
4 7
```

```
# a = b
# b = a

print(a, b)
```

```
4 7
```

```
temp = a
a = b
b = temp

print(a, b)
```

```
7 4
```

```
# Python way of swapping

a, b = b, a

print(a, b)
```

```
7 4
```

```
# Quiz

l = [1, 5, 2, 3]
l.append(7)
l.insert(0, 5)
l[1] = l[2]
l[2], l[0] = l[0], l[2]
print(l)
```

```
[5, 5, 5, 2, 3, 7]
```

```
def swap(li, index1, index2):
    li[index1], li[index2] = li[index2], li[index1]
    return li

l = [1, 5, 2, 3]

print(swap(l, 2, 3))
```

```
[1, 5, 3, 2]
```

Removing data

- pop
- remove
- del

```
## pop element: It also returns the element

# quiz

li
```

```
[2, 4, 6]
```

```
deleted = li.pop()

deleted
```

```
6
```

```
li
```

```
[2, 4]
runs
[150, 150, 99, 20, 99, 200, 99, 120]
runs.pop(0)
150
runs
[150, 99, 20, 99, 200, 99, 120]


## remove element: using an element
## Remove method removes the first occurance of that element
## remove method doesn't return the value
# quiz
runs
[150, 99, 20, 99, 200, 99, 120]
runs.remove(99)
deleted = runs.remove(99)
deleted
runs
[150, 20, 200, 99, 120]

99 in runs
True



# del
a = 4
print(a)
```

```
4
```

```
runs
```

```
[150, 20, 200, 99, 120]
```

```
# del runs[0]
```

```
runs
```

```
[20, 200, 99, 120]
```

```
del runs
```

```
# runs
```

```
# Quiz
```

```
l = [1, 2, 3, 3, 5, 6, 7, 5]
l.pop(5)
print(l)
```

```
[1, 2, 3, 3, 5, 7, 5]
```

```
# To remove 2nd position element 5:
```

```
l = [1, 5, 7]
x = 2

l.pop(1)
```

```
5
```

```
l
```

```
[1, 7]
```

```
l = [1, 2, 3, 3, 5, 6, 7, 5]
l.remove(5)
```

```
l
```

```
[1, 2, 3, 3, 6, 7, 5]
```

Linear search

- Write a function for linear search on a list

```
# First time sachin scored 99

runs = [100, 150, 99, 20, 99, 200, 99, 120]

# Search space
# target

target = 99

for i in range(len(runs)):
    if runs[i] == target:
        print(i)
        break
```

```
2
```

```
def linear_search(li, target):
    # run a loop in range of li

    for i in range(len(li)):
        if li[i] == target:
            return i
    # After iteration if element is not there
    return "Not found"

linear_search(runs, 990)
```

```
'Not found'
```

```
# Index method

runs.index(99)
```

```
2
```

Count of 99 runs by Sachin

```
runs
```

```
[100, 150, 99, 20, 99, 200, 99, 120]
```

```
runs.count(99)
```

```
3
```

```
# HW: Do it using a loop
```

Write a function to find if a list has same consecutive elements

```
l = [1, 2, 3, 5, 1, 2, 4, 1, 1, 2, 3]

# consecutive elements

for i in range(len(l) - 1):
    if l[i] == l[i + 1]:
        print(i)
```

```
7
```

```
# HW: Solve the above question using function
```