Introduction

```
runs = [0, 99, 100, 15, 16, 56, 34]

print(runs)
```

```
[0, 99, 100, 15, 16, 56, 34]
```

```
runs = [0, 99,100, 15,16, 56, 34]

print(runs)
```

```
[0, 99, 100, 15, 16, 56, 34]
```

```
runs = [0,
        99,
        100,
        15,
        16,
        56,
        34]
print(runs)
```

```
[0, 99, 100, 15, 16, 56, 34]
```

List Indexing

```
runs = [0, 99, 100, 15, 16, 56, 34]

print(runs[0])
print(runs[1])
print(runs[2])
print(runs[3])
print(runs[4])
print(runs[5])
print(runs[6])
```

```
0
99
100
15
```

```
16
56
34
```

Quizzes

```
l = [5, 1, -2, 2, 3, 4]
print(l[2])
```

```
-2
```

```
l = [1, 2, 5, 3, 2, 6, 5, 4]
```

Double-click (or enter) to edit

List Append

```
runs = [0, 99, 100, 15, 16, 56, 34]
print(runs)
```

```
[0, 99, 100, 15, 16, 56, 34]
```

```
runs.append(59)
print(runs)
```

```
[0, 99, 100, 15, 16, 56, 34, 59]
```

List Iteration

Iteration over the values

```
l = [1, 3, 5, 7, 9]

for x in l: # l is the name of the list
  print(x)
```

```
1
3
5
7
9
```

```
l = [1, 3, 5, 7, 9]

for x in l: # l is the name of the list
  print(x, end=' ')
```

```
1 3 5 7 9
```

```
l = [1, 3, 5, 7, 9]

x = []
for y in l: # l is the name of the list
  # print(y)
  x.append(y)
  # print(x)
  # print()

print(x)
```

```
[1, 3, 5, 7, 9]
```

```
print(len(l))
```

```
5
```

Double-click (or enter) to edit

Quiz

```
def mystery(lst1, lst2):
  print(len(lst1))
  print(len(lst2))

  if len(lst1) != len(lst2):
    print('Lengths are not equal')
    return True

  ## ITERATION OVER THE INDEX (POSITION)
  for i in range(len(lst1)):
    print(i, end=' ')
```

```
  print()

  for i in range(len(lst1)):
    if lst1[i] != lst2[i]:
      print('The list are not equal at index:', i)
      return False

  print('The 2 list are equal')
  return True


mystery([1,2,3], [1,4,3])
```

```
    3
    3
    0 1 2
    The list are not equal at index: 1
    False
```

Type

```
l = [1, 2, 3]
print(type(l))
```

```
    <class 'list'>
```

Negative Indexing

```
# Print the last element


runs = [0, 100, 99, 77, 65]

print(runs[len(runs) - 1]) # runs[index for the last element]
```

```
    65
```

```
print(runs[-1])
```

```
    65
```

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

```
    77
```

```
print(runs[-3])
```

```
    99
```

List Operations

- insert
- pop

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

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

```
l.insert(3, 4)
print(l)
```

```
[1, 2, 3, 4, 5, 7]
```

Problem Solving:

1. Sum
2. Remove that
3. Reverse

Learnings:

1. Take input in a single line
2. Removing the element at a given position => pop
3. Break our code into functions, make it more readable

```
l = input()
print(l)
```

```
     3 10 20 30
     3 10 20 30
```

```
print(type(l))
```

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

```
x = l.split()
print(x)
```

```
     ['3', '10', '20', '30']
```

```
res = []
for i in x:
    # print(i)
    # print(type(i))
    res.append(int(i))

print(res)
```

```
     [3, 10, 20, 30]
```

```
res.pop(0)
print(res)
```

```
     [10, 20, 30]
```

Quiz

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

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

Up Next:

- More operations: remove, remove vs pop
- Mutability
- References
- List Slicing
- Max and Min in List
- Searching in List
- Enumerate