List in Python (Part-1)

What are Lists?

List is a data type where you can store multiple items under 1 name. More technically, lists act like dynamic arrays which means you can add more items on the fly.

 Alt text

Characterstics of a List:

- Ordered
- Changeble/Mutable
- Hetrogeneous
- Can have duplicates
- are dynamic
- can be nested
- items can be accessed
- can contain any kind of objects in python

Creating a List

In [ ]:
```
# Empty
print([])

# 1D -> Homo
print([1,2,3,4,5])

# 2D
print([1,2,3,[4,5]])

# 3D
print([[[1,2],[3,4]]])

# Heterogenous
print([1,True,5.6,5+6j,'Hello'])

# Using Type conversion
print(list('hello'))
```

```
[]
[1, 2, 3, 4, 5]
[1, 2, 3, [4, 5]]
[[[1, 2], [3, 4]]]
[1, True, 5.6, (5+6j), 'Hello']
['h', 'e', 'l', 'l', 'o']
```

Accessing Items from a List

- Indexing
- Slicing

In [ ]:
```
# indexing
L = [[[1,2],[3,4]],[[5,6],[7,8]]]
L1 = [1,2,3,4]

# Positive Indexing
print(L1[1:4])

print(L[0][0][1]) # for 2

#How to extract 6
print(L[1][0][1])
```

```
[2, 3, 4]
[]
2
6
```

In [ ]:
```
# Negative indexing
L = [[[1,2],[3,4]],[[5,6],[7,8]]]
L1 = [1,2,3,4]

print(L[-1])

# how to extract 8 with negative
print(L[-1][-1][-1])
```

```
[[5, 6], [7, 8]]
8
```

In [ ]:
```
# Slicing
L = [1,2,3,4,5,6]

print(L[::-1])
```

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

Adding Items to a List

In [ ]:
```
# Apend -> The append method is used to add an item to the end of a list.
L =  [1,2,3,4,5]
L.append(True)
print(L)
```

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

In [ ]:
```
# Extend -> The extend method is used to append elements from an iterable (e.g., ar
L = [1,2,3,4,5]
L.extend([2])
L
```

Out[ ]:
```
[1, 2, 3, 4, 5, 2]
```

In [ ]:
```
# insert -> The insert method allows you to add an item at a specific position in t
l = [1,2,3,4]

l.insert(1,100)
print(l)
```

```
[1, 100, 2, 3, 4]
```

Editing items in a List

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

# editing with indexing
l[-1] = 300
print(l)

# editong with slicing
l[1:4] = [200,300,400]
print(l)
```

```
[1, 2, 3, 4, 5, 300]
[1, 200, 300, 400, 5, 300]
```

Deleting items from a List

In [ ]:
```
# del -> The del statement is used to remove an item from a list based on its index
l = [1,2,3,4,5]

#indexing
del l[2]
print(l)

# slicing
del l[2:4]
print(l)
```

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

In [ ]:
```
# remove -> The remove method is used to remove the first occurrence of a specific
l = [1,2,3]

l.remove(2)

print(l)
```

```
[1, 3]
```

In [ ]:
```
# pop -> The pop method is used to remove and return an item from the list based or
# If you don't provide an index, it will remove and return the last item by default
L = [1,2,3,4,5]

L.pop()

print(L)
```

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

In [ ]:
```
# clear -> The clear method is used to remove all items from the list, effectively
L = [1,2,3,4,5]

L.clear()

print(L)
```

```
[]
```

In [ ]:
```

```

In [ ]:

In [ ]: