2021-11-28

1.LIST

2.MOCK TEST

Presenter: Nguyen Khoa

1. LIST

1.1. append()

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

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

```
a.append([1, 9])
a
```

Syntax

```
list.append(obj)
```

```
a.append("a_string")
a
```

```
a.append(0.89)
a
```

```
a.append(0, 89)
```

1.2. extend()

Syntax

```
list.extend(sequence)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
print(a)
print(b)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

b.extend(a)
print(a)
print(b)
```

1. LIST

1.3. pop()

Syntax

```
list.pop(obj = list[-1])
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
a
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89]
```

```
a.pop()
a
```

```
a = a[:-1]
a
```

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

```
a.pop()
a
```

```
a.pop()
a
```

1.4. remove()

Syntax

```
list.remove(obj)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
a
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89]
```

```
a.remove(2)
a
```

```
a.remove([1, 9])
a
```

```
a.remove("a_string")
a
```

1.4. remove()

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
print(a)
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89, 2, 2, 2, 2, 3, 3]
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
print(a)
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89, 2, 2, 2, 2, 3, 3]
```

1.5. del()

Syntax

```
del(obj)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
a
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89]
```

```
del(a[2])
a
```

```
del(a[4])
a
```

1.5. del()

```
del(a)
a
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
del(a[2:7:2])
a
```

1.5. del()

```
del(a)
a
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
del(a[2:7:2])
a
```

1.6. index()

Syntax

```
list.index(obj)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
a
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89, 2, 2, 2, 2, 3, 3]
```

```
a.index([1, 9])
```

```
a.index(2)
```

1.6. index()

```
for index, element in enumerate(a):
    print(index, element)
```

```
[index for index, element in enumerate(a) if element == 2]
```

```
index for index, element in enumerate(a) if element == 2
```

```
for index, element in enumerate(a):
    if element == 2:
        print(index, end="\t")
```

1.7. del()

Syntax

```
list.insert(index, obj)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
a
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89]
```

```
a.insert(3, 2001)
a
```

```
a.insert(7, 2001)
a
```

1.8. sort()

Syntax

```
list.sort()
```

```
a = [3, 2, 1, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
a
```

```
[3, 2, 1, 4, [1, 9], 'a_string', 0.89, 2, 2, 2, 2, 3, 3]
```

```
a.sort()
a
```

```
a[:3].sort()
a
```

1.9. reverse()

Syntax

```
list.reverse()
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
a
```

```
[1, 2, 3, 4, [1, 9], 'a_string', 0.89, 2, 2, 2, 2, 3, 3]
```

```
a.reverse()
a
```

1.10. len()

Syntax

```
len(list)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
len(a)
```

```
b = [2, 2, 2, 2, 3, 3]
len(b)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
b = [2, 2, 2, 2, 3, 3]

a.extend(b)
len(a)
```

1.11. max()

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
max(a)
```

```
b = [2, 2, 2, 2, 3, 3]
max(b)
```

1.12. min()

Syntax

```
min(list)
```

```
a = [1, 2, 3, 4, [1, 9], 'a_string', 0.89]
min(a)
```

```
b = [2, 2, 2, 2, 3, 3]
min(b)
```

1.13. Slicing

Syntax

```
a[low:high:step] if step>0
a[high:low:step] if step<0
```

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
a
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
```

```
a[1:3]
```

```
a[1:9:2]
```

```
a[:9:2]
```

```
a[::2]
```

1.13. Slicing

Syntax

```
a[low:high:step] if step>0
a[high:low:step] if step<0
```

```
a[:-2]
```

```
a[-2:]
```

```
a[::-1]
```

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
a[-3::-2]
```

```
a[:-5:-1]
```