Les Listes

Declaration liste:

```
> l = list()
> l = []
> l = ["one", "two", 1, 100.5] # avec initialisation
> l[1]      # access list
   two
```

→ liste est mutable : add, Delete, Edit

→ print(*l) # afficher liste sans [ ]

Slice list:

```
> l [start : stop : step]  # slice d'une liste
```

stop à stop-1

default value = 1

```
> l [ : ]  # une copie de tout la liste
```

start ou stop ou step peut être negative

```
> l [ : : -1]  # tous les elements inversé
> l [1 : : -1]  # Les deux premiers inversé
> l [ : -3 : -1]  # les deux derniers inversé
> l [-3 : : -1]  # tous sauf les deux derniers inversé
```

Fonctions:

```
> del l[2]   # supprimer element index 2
> len(l)     # nbres elements
> min(l)     # minimum
> max(l)     # maximum
> sum(l)     # somme
```

Methode liste:

Append():

> l.append(element)
→ append element to the end of list

element: required, any type (str, number, objet ...)

extend():

> l.extend(iterable)
→ add specified list elements to a list

iterable: required, any iterable (list, set, tuple ...)

Remove()

> l.remove(element)
→ remove the first occurence of the element with the value
element: required, any type (str, number, list ...)

Sort():

> l.sort(reverse=True|False, key=myFunc)
→ sorts the list ascending by default
→ make a function to decide the sorting criteria.

reverse: Optional, True: to sort descending, Defaults = False
key : Optional, specify the sorting criteria.

reverse()

> l.reverse()
→ reverse the sorting order of the elements.
No parameters

clear():

> l.clear()
→ remove all the items from a list
No parameters

copy():

> l.copy()
→ return a copy of list, shallow copy
no parameters.

count()

> l.count(value)
→ return the number of elements with the specified value.
value: required, Any type

index():

> l.index(element)
→ returns the position at the first occurence of element
element: required, any Type (str, number, list)

insert( )

> l.insert (pos, element)

→ inserts the specified value at the specified position

Pos : required, a number specified which position to insert
elent : required, a element of any type (str, num, obj. etc

Pop( )

> l.pop (pos)

→ return element position Pos and remove it

Pos : optional, position of element
default value is -1, the last items