

# List Arguments Summarize

Questo PDF presenta un sunto di tutti gli argomenti che saranno coperti durante le lezioni riguardanti le liste.

- **List Structure**

- Sintassi `[1, 2, 3, 4, 5, ...]`
- Contenuto di ogni tipo `[1, "uno", 2, [1, 2], "tre"]`
- Differenze con le stringhe
  - Le stringhe sono *iterables* delimitati dai **doppi apici**
  - Le liste sono *iterables* delimitati dalle **parentesi quadre**
  - Le liste possono contenere ogni tipo di *object* (str, list, int), mentre le stringhe no. Da notare che se io avessi una stringa come questa `"ciao1[2, 2, 3]"`, dovrei considerare `1` o `[2, 2, 3]` come delle sottostringhe, quindi *str object* e non *list* o *int*.
  - Le stringhe e le liste differiscono anche per **ibuilt-in methods**

- **List Properties**

- Ordinamento non **in-structure**. Posso creare una lista non ordinata come questa `[3, 1, 2]`
- **Indexing**. Posso accedere ad un elemento della lista tramite il suo indice
- **Extensibility**. Posso aumentare o decrementare la sua dimensione

- **Loop on lists**

- Scansionare una lista utilizzando un **For Loop**

```
l = [1, 2, 3, 4, 5, "uno", "due", "tre", "quattro", "cinque"]
for x in l:
    print(x)
```

output:

```
1
2
3
4
5
uno
due
tre
quattro
cinque
```

- Scansionare una lista utilizzando un **While Loop**

```
l = [1, 2, 3, 4, 5, "uno", "due", "tre", "quattro", "cinque"]
index=0
while index <= len(l) - 1:
    print(l[index])
    index = index + 1
```

```
output:
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
uno
```

```
due
```

```
tre
```

```
quattro
```

```
cinque
```