NESTED LISTS

"This is no cave." -Han

WHAT'S INSIDE A LIST

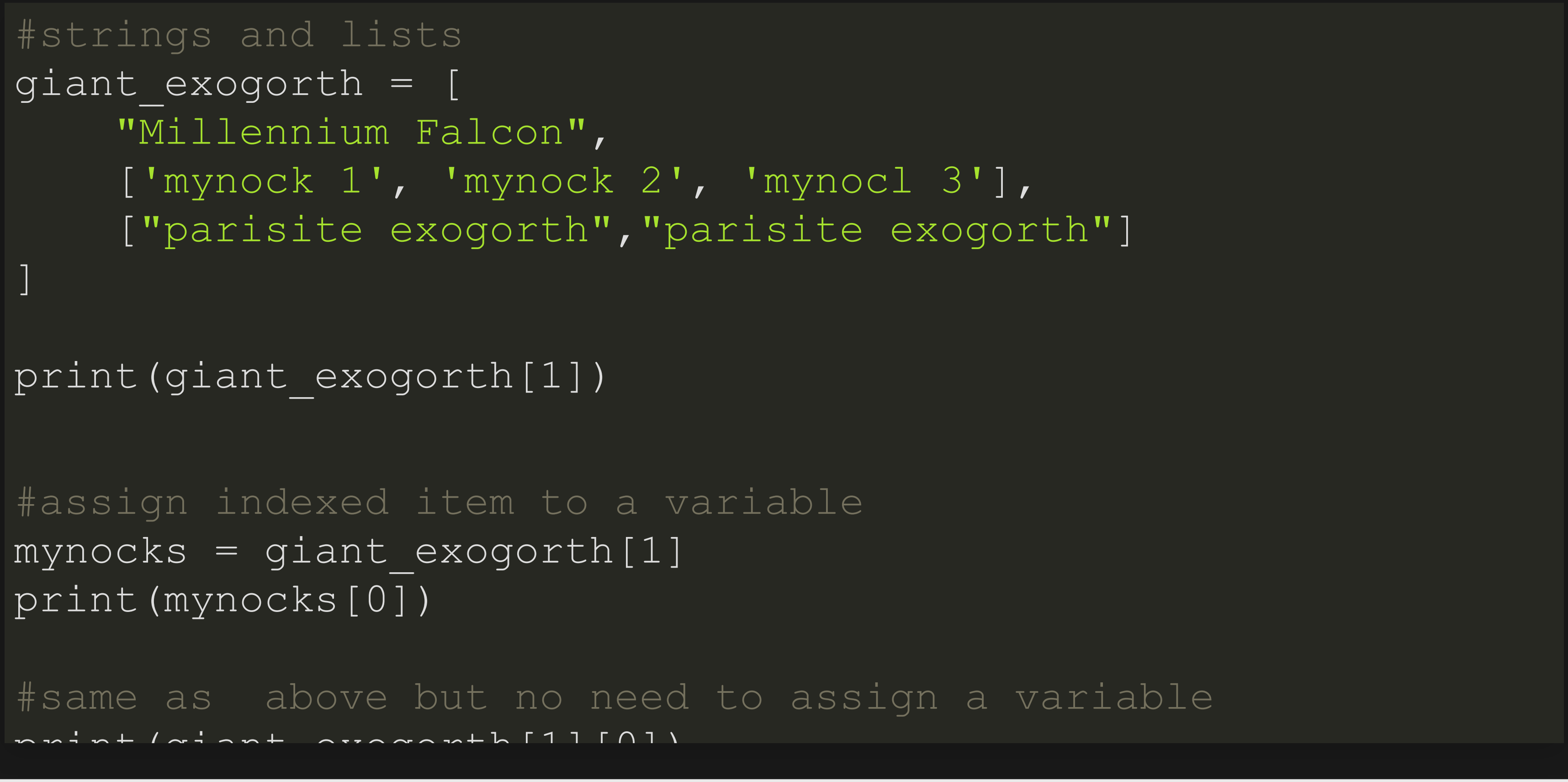
Lists can have more than just strings inside the list. It can actually have any python type, and it can be mixed and matched.

```python
#lists with differrent types inside
#integers and floats
episode_numbers_order = [4, 5, 6, 1, 2, 3, 7, 3.9, 8, 3.5, 9]

#episode 1-9 (True, False, None = Didn't Appear)
Luke_whines_in_episode = [None, None, None, True, True, False,


#strings and lists
giant_exogorth = [
    "Millennium Falcon",
    ['mynock 1', 'mynock 2', 'mynocl 3'],
    ["parisite exogorth","parisite exogorth"]
]
```

ⓘ Lists can have any type that is available for python



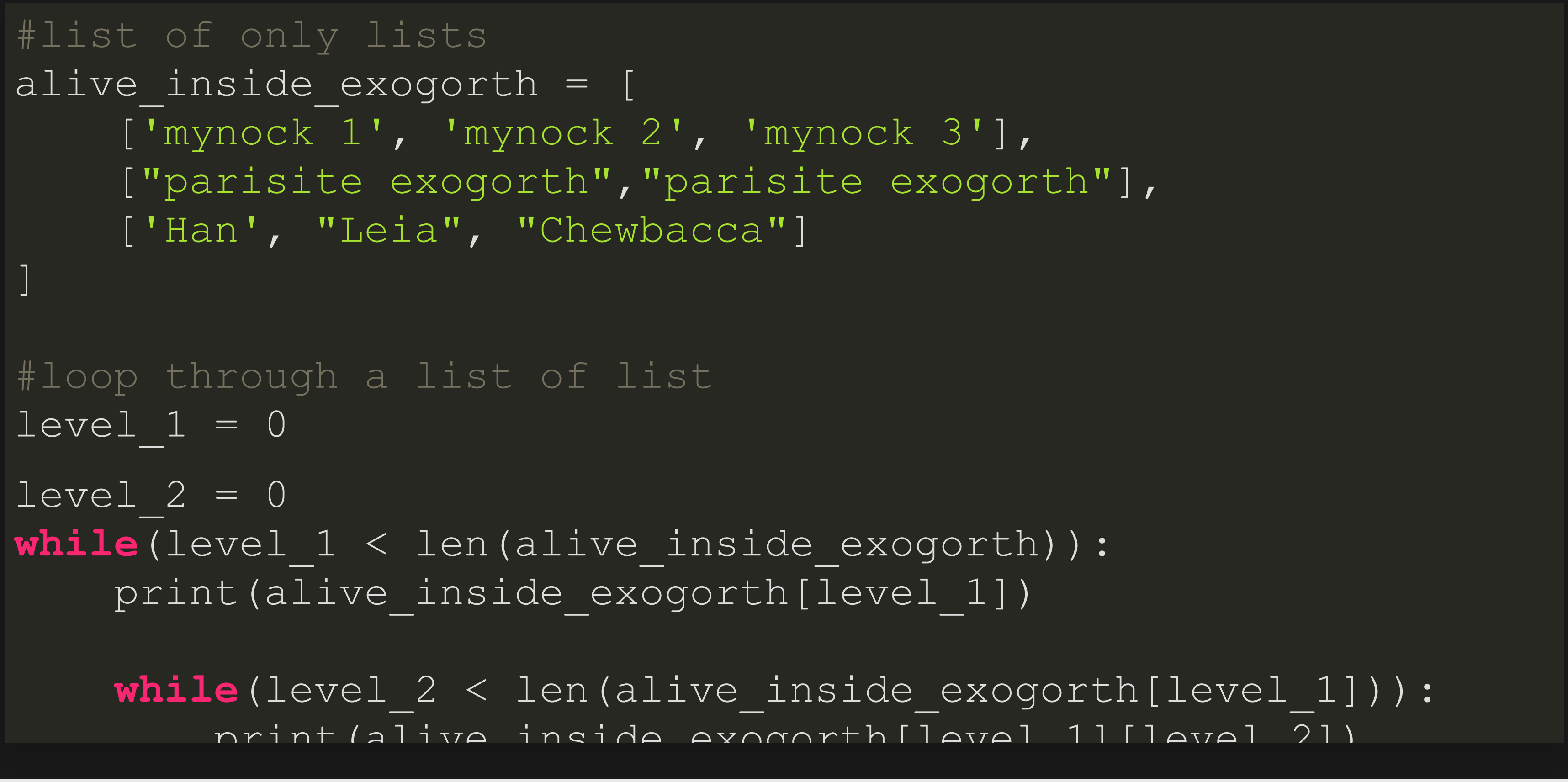
```
#strings and lists
giant_exogorth = [
    "Millennium Falcon",
    ['mynock 1', 'mynock 2', 'mynocl 3'],
    ["parisite exogorth","parisite exogorth"]
]

print(giant_exogorth[1])


#assign indexed item to a variable
mynocks = giant_exogorth[1]
print(mynocks[0])

#same as  above but no need to assign a variable
print(giant_exogorth[1][0])
```

You can access the list from within a list by calling the index of the item you want inside the index of the list.



```
#list of only lists
alive_inside_exogorth = [
    ['mynock 1', 'mynock 2', 'mynock 3'],
    ["parisite exogorth","parisite exogorth"],
    ['Han', "Leia", "Chewbacca"]
]

#loop through a list of list
level_1 = 0
level_2 = 0
while(level_1 < len(alive_inside_exogorth)):
    print(alive_inside_exogorth[level_1])

    while(level_2 < len(alive_inside_exogorth[level_1])):
        print(alive_inside_exogorth[level_1][level_2])
```

ⓘ If indexes are needed then the while loop will give you the index.

ⓘ If you just need to do something with the content then for - in is the right choice.