

Python (Programming Language)

Teran Subasinghe

Check Python is Installed

- Goto CMD
- Goto the Installation Directory
 - Ex: C:\Python31\
- Type
 - Python.exe -V (To check version)
 - Python (To execute python)

What is Python (IDLE)

- IDLE lets you write code in its full-featured code editor as well as experiment with code at the Python Shell.
- You'll use the code editor later in this book but, when learning Python, IDLE's shell really rocks, because it lets you try out new Python code as you go.
- When you first start IDLE, you are presented with the “triple chevron” prompt (`>>>`) at which you enter code. The shell takes your code statement and immediately executes it for you, displaying any results produced on screen.

This is your
"movies" list in
code.

```
movies = ["The Holy Grail", "The Life of Brian", "The Meaning of Life"]
```

Item #0

Item #1

Item #2

This is your "movies"
list in memory.



2

1

0

Each data item
in the list has a
numeric OFFSET
associated with it.

Python starts counting
from zero.

Creating Lists

- `len(movies)`
 - Return the Length of List
- `movies.append("Fist of Fury")`
 - Add another Value to End of the List
- `movies.pop()`
 - Delete the last Item from the List
- `movies.extend(["Dragon","Dear"])`
 - Add another List to an Existing List
- `movies.remove("Dragon")`
 - Remove a specific data item from a List
- `movies.insert(0,"jungle book")`
 - Add a data item to a specific location of a List
- **Python Lists can contain data of Mixed Type**

Lists Cont..

For Loops work with lists of any size

The keyword "for" indicates the start of the loop and comes before the target identifier.

→ **for**

target identifier

The keyword "in" separates the target identifier from your list.

in

list

A colon ":" follows your list name and indicates the start of your list-processing code.

:

The list-processing code MUST be indented under the for loop.

list-processing code

The target identifier is like any other name in your code. As your list is iterated over, the target identifier is assigned each of the data values in your list, in turn. This means that each time the loop code executes, the target identifier refers to a different data value.

For Loop and While Loop

- An alternative to using for is to code the iteration with a while loop. Consider these two snippets of Python code, which perform the same action:

When you use "while", you have to worry about "state information," which requires you to employ a counting identifier.

```
count = 0
while count < len(movies):
    print(movies[count])
    count = count+1
```



```
for each_item in movies:
    print(each_item)
```

When you use "for", the Python interpreter worries about the "state information" for you.

These while and for statements do the same thing.

Store Lists with in Lists

```
movies = [  
    "The Holy Grail", 1975, "Terry Jones & Terry Gilliam", 91,  
    ["Graham Chapman",  
     ["Michael Palin", "John Cleese", "Terry Gilliam", "Eric Idle", "Terry Jones"]]]
```

The start of the first, outer list

The end of all the lists is here.

The start of the second, inner list: "movies[4]"

The start of the third, inner inner list: "movies[4][1]"

This looks a little weird...until you remember that there are three opening square brackets, so there must also be three closing ones.

So, a list within a list is possible, as is a list within a list within a list (as this example code demonstrates). In fact, it's possible to nest lists within lists to most any level with Python. And you can *manipulate* every list with its own list methods and *access* it with the square bracket notation:

```
print(movies[4][1][3])
```

A list within a list within a list

Eric Idle

Eric is this deeply nested, so he can't possibly be idle. 😊

Check List for a List

- To Check whether it's a list
 - `isinstance(listName,list) -> Boolean`
- To Access a nested List and Print the content
 - For each_item in movies :
 - if `isinstance(each_item,list) :`
 - for nested_item in each_item :
 - `print(nested_item)`
 - else :
 - `print(each_item)`