

The different ways of for ...

- Iterating through a list can be done in different ways
- Simple is usually better
- ... but you should understand "strange" constructs, too ...

```

L = [3,1,4,1,5]
L2 = "abcde"
print("Ex1")
for e in L:
    print(e)

print("\nEx2")
for c in L2:
    print(c)

print("\nEx3")
for i in range(len(L2)):
    print("i =", i, L2[i])

print("\nEx4")
i = 0
for c in L2:
    print("i =", i, c)
    i = i + 1

print("\nEx5: print L2 backwards")
mylen = len(L2)
for i in range(mylen):
    print("i =", i, L2[mylen-i-1])

print("\nEx6: print L2 backwards")
mylen = len(L2)
for i in range(mylen-1, -1, -1):
    print("i =", i, L2[i])

print("\nEx7")
l_tmp = L
while l_tmp != []:
    e = l_tmp[0]
    print(e)
    l_tmp = l_tmp[1:]

print("\nEx8")
l_tmp = L2
while l_tmp != "":
    e = l_tmp[0]
    print(e)
    l_tmp = l_tmp[1:]

```

Sequences: Lists and Strings

- finite sequence of elements/objects: **o, o, o, ..., o**
- Example: **Strings**
 - S = "Hello World! FOO BAR"**

H	e	l	l	o		W	o	r	l	d	!		F	O	O		B	A	R
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

-20 -2 -1

- Access elements of sequence via an **index**
 - 0, 1, 2, ..., **N-1** where **N** is the **length** of the sequence
 - S[0] **first** element of sequence (here: string S has **index 0**)
 - S[1] **second** element has **index 1**
 - S[i] (**i+1**)-st element of S has **index i**

Slicing sequences ...

if S is a sequence (string, list)...

- ...we can do S[start : end]
- yielding the **slice** from index *start* to *end-1* !
- can also start from the back of the sequence: [-1], [-2], ...

defaults:

- for *start*: **0** S[:3]
- for *end*: **len(S)** S[1:]

create a copy of food string

... and more operations on strings and lists

- String concatenation:**
 - S3 = S1 ⊕ S2
 - "Hi " + "there!" → "Hi there!"
- String indexing:**
 - S[i]
 - "foobar"[3] → "b"
- String slicing:**
 - S[m : n]
- String length:**
 - len(S)
- Iterating over:**
 - for x in S:
- Initialization:**
 - S = "" L = []
- Testing membership (new!)**
 - e in X → True/False
- Others (on strings!):**
 - .split(), .upper(), ...