

Lists and Loops



Sarah Holderness

Author

@dr_holderness



A List is a Container of Things

```
empty = []
```

empty list

```
words = ['LOL', 'IDK', 'TBH']
```

list of strings

```
nums = [5, 10, 15]
```

list of numbers

```
mixed = [5, 'SDK', 1.5]
```

list of mixed items

```
lists = [ ['A', 'B', 'C'], ['D', 'E', 'F'] ]
```

list of lists

Creating a List of Internet Slang Acronyms

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH']
```



*We're compiling a list of
acronyms that we'll define later*

An Item's Index is its Position

```
acronyms = [ 'LOL', 'IDK', 'SMH', 'TBH' ]
```

 0 1 2 3

```
print(acronyms[0])
```

 1st item

> LOL

An Item's Index is its Position

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH']
```

 0 1 2 3

```
print(acronyms[3])
```

4th item

> TBH

Note: if you want the n th item then use index $[n-1]$

Creating a List and Adding Items

```
acronyms = []
```



We can create an empty list

```
acronyms.append('LOL')  
acronyms.append('IDK')
```



And then add each item individually

```
print(acronyms)
```

```
> ['LOL', 'IDK']
```

We can see the 2 items in the list

Creating a List with Items and Then Adding Items

```
acronyms = ['LOL', 'IDK', 'SMH']
```

```
acronyms.append('BFN')  
acronyms.append('IMHO')
```

```
print(acronyms)
```

*We haven't called a method like this before.
Let's take a closer look...*

```
> ['LOL', 'IDK', 'SMH', 'BFN', 'IMHO']
```

Calling a Method

```
acronyms.append('BFN')
```

*The list you
want to change*

dot

*method
name*

*What you want to
add to the list in
parentheses ()*

Removing Items

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH', 'BFN']  
acronyms.remove('BFN') OR del acronym[4] ◀...  
print(acronyms)
```

*You can use either
remove or del depending
on whether you know the
value or the index.*

```
> ['LOL', 'IDK', 'SMH', 'TBH']
```

*You can see 'BFN'
was removed*

Check if Exists in List



> True

Check if Exists in List

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH']  
word = 'BFN'
```

```
if word in acronyms:  
    print(word + ' is in the list')  
else:  
    print(word + ' is NOT in the list')
```

False

So this line is run

> BFN is NOT in the list

Printing a List

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH']
```

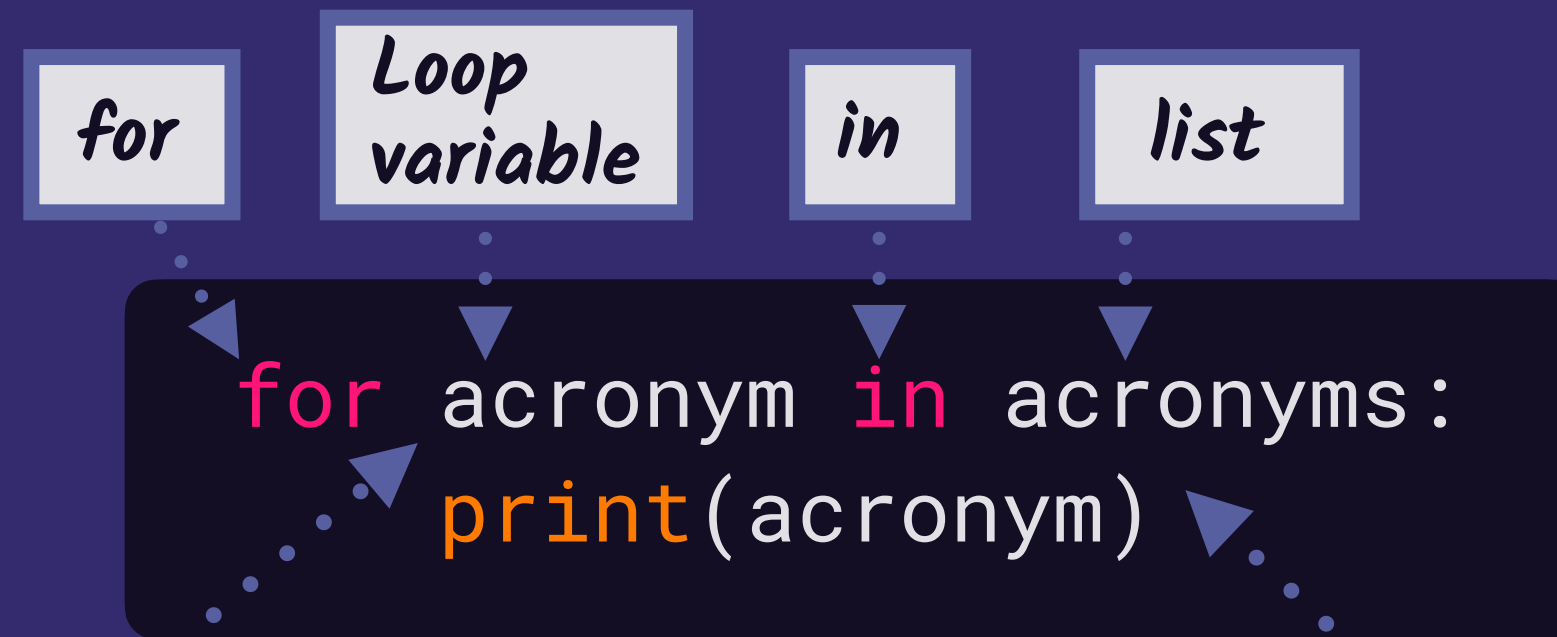
```
print(acronyms)
```

```
> ['LOL', 'IDK', 'SMH', 'TBH']
```

What if we want to print each acronym on a separate line?

We need a loop 

The Syntax of a for loop



acronym is a temporary variable that holds one of the acronyms in the list for each run

Like saying "do this" for each string acronym in our acronyms list

For Loop: Looping Over Each Item in a List

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH']
```

```
for acronym in acronyms:  
    print(acronym)
```

> LOL

IDK

SMH

TBH

1st loop

2nd loop

3rd loop

4th loop

For Loop: Looping Over Each Item in a List

```
acronyms = ['LOL', 'IDK', 'SMH', 'TBH']
```

```
for acronym in acronyms:  
    print(acronym)
```



Notice how the code block you want repeated inside the loop is indented, just like in an if statement.

> LOL

IDK

SMH

TBH

1st loop

2nd loop

3rd loop

4th loop

Up Next:

Demo: Sum a List of Expenses



Loops Using range()



Sarah Holderness

Author

@dr_holderness



Adding Input to the Expenses Calculator

expenses.py

We want the user to be able to enter their own expenses

```
expenses = [10.50, 8.50, 5.30, 15.05, 20.00, 5.00, 3.00]
```

```
total = sum(expenses)
```

```
print("You spent $", total, " on lunch this week.", sep='')
```

Adding Input to the Expenses Calculator

expenses.py

```
expenses = []  
expenses.append(float(input("Enter an expense:\n")))  
expenses.append(float(input("Enter an expense:\n")))  
expenses.append(float(input("Enter an expense:\n")))  
expenses.append(float(input("Enter an expense:\n")))  
expenses.append(float(input("Enter an expense:\n")))  
expenses.append(float(input("Enter an expense:\n")))  
expenses.append(float(input("Enter an expense:\n")))  
...
```

With our current set of tools, we would type input 7 times.

Is there a way we can loop 7 times instead and ask for input inside the loop?

The range() Function

```
>>> range(7)
```

(0, 1, 2, 3, 4, 5, 6)

```
>>> range(0, 7, 1)
```

(0, 1, 2, 3, 4, 5, 6)

Start

Stop

Step

*Note: Start and Step are optional.
Start is 0 by default, Step is 1.*

*Note: The sequence starts at 0 and
ends at 6, but there are 7 numbers.*

The range() Function

```
>>> range(7)
```

(0, 1, 2, 3, 4, 5, 6)

```
>>> range(0, 7, 1)
```

(0, 1, 2, 3, 4, 5, 6)

```
>>> range(2, 14, 2)
```

(2, 4, 6, 8, 10, 12)

Start

Stop

Step

*We get even numbers
starting at 2 and
stopping before 14*

The Syntax of a for loop

```
for i in range(7):  
    print(i)
```

```
> 0  
  1  
  2  
  3  
  4  
  5  
  6
```

We can then use a for loop like we've seen with the sequence generated by range().

This let's us loop a certain number of times, which is what needed to enter expenses...

Adding Input to the Expenses Calculator

expenses.py

```
total = 0
expenses = []
for i in range(7):
    expenses.append(float(input("Enter an expense:")))

total = sum(expenses)

print("You spent $", total, sep='')
```

Adding Input to the Expenses Calculator

expenses.py

```
total = 0
expenses = []
for i in range(7):
    expenses.append(float(input("Enter an expense:")))

total = sum(expenses)

print("You spent $", total, sep='')
```

```
> python3 expenses.py
Enter an expense:10
Enter an expense:5
Enter an expense:20
Enter an expense:12
Enter an expense:13
Enter an expense:8
Enter an expense:4
You spent $72
```


Adding Input to the Expenses Calculator

expenses.py

```
total = 0
expenses = []
for i in range(7):
    expenses.append(float(input("Enter an expense:")))

total = sum(expenses)

print("You spent $", total, sep='')
```

*What if we want the user to
enter the number of expenses?*

```
> python3 expenses.py
Enter an expense:10
Enter an expense:5
Enter an expense:20
Enter an expense:12
Enter an expense:13
Enter an expense:8
Enter an expense:4
You spent $72
```

Adding Input to the Expenses Calculator

expenses.py

```
total = 0
expenses = []
num_expenses = int(input("Enter # of expenses:"))
```

Adding Input to the Expenses Calculator

expenses.py

```
total = 0
expenses = []

num_expenses = int(input("Enter # of expenses:"))
for i in range(num_expenses):
    expenses.append(float(input("Enter an expense:")))

total = sum(expenses)

print("You spent $", total, sep='')
```

```
> python3 expenses.py
Enter # of expenses:5
Enter an expense:5
Enter an expense:20
Enter an expense:12
Enter an expense:13
Enter an expense:8
You spent $58
```

Up Next:

Demo: Loan (Mortgage) Calculator

