# Introduction to Lists

## Why is this important?

In our everyday life we use a lot of lists: Todo Lists, Shopping Lists, Rankings etc. In programming this is no different, we need lists to store data.

In Python, a list is a data structure that holds items that are of the same data type.

Lists are crucial because they are so versatile. They can be used to store data that is related to each other, and they can be used to store data that is not related to each other.

They're used in every program that you'll write, so that's pretty much why they're important.

## What are we going to do?

We're going to take a look at a grocery store list and we're going to see how we can access items in the list, modify items in the list, and get the length of the list. We're also going to see how we can get a part of the list at a given index.

## Steps

### 1. Let's create a file called `intro_to_lists.py`

### 2. Let's create our first list

We're going to create a list of grocery items that we need to buy.

- using the `[]` brackets, we can create a list of items that are separated by commas.

```
# our grocery list.
groceries = ['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
print("Our grocery list is: ")
print(groceries)
```

- The output should look like the following

```
$ python intro_to_lists.py
Our grocery list is:
['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
```

- Note you can see here that every item in the list is a string.
    - now you might be thinking yourself, it would be really handy if I could access the items in that list. Well you can!
    - There's something called the "index" it starts at 0 and goes up by 1 for every item in the list.

## 2. Let's Access some of the items in the list.

- Let's say we want to access the first item in the list, we can do that by using the index which always begins at 0.
    - the way you access the item is by using the `[]` brackets next to the variable name (here `groceries`) and putting the index of the item you want to access in the brackets.

```
# our grocery list.
groceries = ['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
print("Our grocery list is: ")
print(groceries)

# Let's check the first item in our list.
print("the first item in our grocery list is: ")
print(groceries[0])
```

- the output should look like the following

```
$ python intro_to_lists.py
Our grocery list is:
['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
the first item in our grocery list is:
lettuce
```

- So you can see here that `groceries[0]` is `lettuce` and `groceries[1]` is `tomatoes` and so on, but let's try it out.
- Let's take a look at how we can select the third item in the list which is `bread`. Now if we want the third item our index would be `2` because remember the index starts at 0.

```
# our grocery list.
groceries = ['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
print("Our grocery list is: ")
print(groceries)

# Let's check the first item in our list.
print("the first item in our grocery list is: ")
print(groceries[0])
# Let's check out the third item in our list.
print("the first item in our grocery list is: ")
print(groceries[2])
```

- here's the output.

```
$ python intro_to_lists.py
Our grocery list is:
['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
the first item in our grocery list is:
lettuce
the first item in our grocery list is:
bread
```

- A few notes about accessing items in a list.
    - You can access the last items in a list by using the index `-1`.
        - so if we wanted to access the last item in our list we would do `groceries[-1]` and it would return `apples`.
- Try to access something at index `42` and see what happens.
    - You'll see here that you're going to get an `IndexError` exception.
        - we'll talk about how to handle exceptions in the coming week.

# 3. Let's change some items in the list.

Let's say you get a call from our partner and they tell you that they specifically wanted "romaine lettuce" instead of "lettuce". They also let us know that they wanted a "baguette" instead of "bread". Let's make this change before the last step so we can see the changes.

- How we change items in the list is by accessing an item with a specific index and then assigning it a new value using the `=` operator.

```
groceries = ['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
print("Our grocery list is: ")
print(groceries)

# Modifying items in the list.
groceries[0] = 'romaine lettuce' # changing the first item
groceries[2] = 'baguette' # changing the third item
print("Our modified grocery list is: ")
print(groceries)
# other code here removed for brevity.
```

- you can see here that if want to change items that on the left of the `=` operator we have `groceries[0]` and on the right we assign it a new value `'romaine lettuce'`.
    - we do the same thing with the third item in the list.
- the output should look like the following

```
$ python intro_to_lists.py
Our grocery list is:
['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
Our modified grocery list is:
['romaine lettuce', 'tomatoes', 'baguette', 'milk', 'chicken', 'apples']
the first item in our grocery list is:
romaine lettuce
the first item in our grocery list is:
baguette
```

- Note you can see the the items are changed in the list and stay changed when we access them later on.

# 4. Let's take a look at how we can "slice" items in the list.

Let's say during the conversation with our partner they asked us "what were the first three items on the list" they also asked us what's the "last two items on the list." (they're fairly nosy). We can do this by using the **:** operator within the `[]` brackets. We're going to do this after the previous steps.

- Let's take a look at how to do this.

```
# our grocery list.
groceries = ['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
print("Our grocery list is: ")
print(groceries)

# other code removed for brevity

# Slicing lists.
print("The first three items in our list are: ")
print(groceries[0:3]) # this will print the first three items in the list.

print("The last two items in our list are: ")
print(groceries[-2:]) # this will print the last two items in the list.
```

- you can see here that we're using the `:` operator to slice the list.
    - the first number is the index of the first item we want to include.
    - the second number is the index of the last item we want to include.
    - the first number is inclusive, the second number is exclusive.
        - so if we want to include the first three items we would do `groceries[0:3]` and it would include the items at index `0`, `1`, and `2`.
        - if we want to include the last two items we would do `groceries[-2:]` and it would include the items at index `-2` and `-1`.
- Taking a look at the output should look like the following.

```
$ python intro_to_lists.py
Our grocery list is:
['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
Our modified grocery list is:
['romaine lettuce', 'tomatoes', 'baguette', 'milk', 'chicken', 'apples']
the first item in our grocery list is:
romaine lettuce
the first item in our grocery list is:
baguette
The first three items in our list are:
['romaine lettuce', 'tomatoes', 'baguette']
The last two items in our list are:
['chicken', 'apples']
```

- Note you can see here that the first three items are `['romaine lettuce', 'tomatoes', 'baguette']` and the last two items are `['chicken', 'apples']`.

## 5. Let's get the length of a list.

Let's say during this conversation with our partner they asked us "how many items are on the list?" We can do this by using the `len()` function. We're going to do this after the previous steps.

- Let's take a look at how to do this, you pass the list into the brackets of `len()` function (we've seen `print` and `input` as a function before, and we'll go deeper on them later on).

```
groceries = ['lettuce', 'tomatoes', 'bread', 'milk', 'chicken', 'apples']
print("Our grocery list is: ")
print(groceries)

# other code removed for brevity

# Getting the length of a list.
print("The length of our grocery list is: ")
print(len(groceries)) # this will print the length of the list.
```

# Conclusion

We've taken a look here at some of the fundamentals of lists:

- how to create a list.
- how to access items in a list.
- how to modify items in a list.
- getting the length of a list.
- getting part of a list at given indices.

Next, we're going to take a look at list methods and how we can use them to remove, add, and sort items in a list. We're also going to check how we can check if something is in a list.