

Starter: There are four mistakes in this program. What are the mistakes and how would you fix them?

```
carrots = input('How many carrots do you have? ')
rabbits = 6

if rabbits < carrots:
    print('There are not enough carrots')
if rabbits > carrots:
    print('There are too many carrots')
else:
    print('You have the right number of carrots')
```



Python Session 4

This session:

1. Lists
2. Dictionaries

By the end of this session you will be able to:

- Create lists to store multiple data values
- Identify different list functions
- Combine lists and for loops
- Construct dictionaries to structure data

Lists

List: an ordered collection of values

List are written inside square brackets and separated by commas

A list of integers

```
In [ ]: lottery_numbers = [4, 8, 15, 16, 23, 42]
```

A list of strings

```
In [ ]: student_names = ['Diedre', 'Hank', 'Helena', 'Salome']
```

Lists can be made up of values of one or more data types

```
In [ ]: person = ['Jess', 32]
```


List values can be accessed using their **index** in square brackets

```
In [1]: student_names = ['Diedre', 'Hank', 'Helena', 'Salome']  
print(student_names[2])
```

Helena

List indexes start counting from 0

```
In [2]: student_names = [  
        'Diedre',    # index 0  
        'Hank',      # index 1  
        'Helena',    # index 2  
        'Salome'     # index 3  
    ]  
  
    print(student_names[0])
```

Diedre

You can also set the values in lists using their indexes, similar to how you would set a variable

```
In [3]: student_names = [  
        'Diedre',    # index 0  
        'Hank',      # index 1  
        'Helena',    # index 2  
        'Salome'     # index 3  
    ]  
  
    student_names[1] = 'Joshua'
```

Exercise 4.1: When I'm travelling in the winter I often forget to pack warm clothes. Let's write a program to help me to remember the right clothes.

The program should check if the first item in the `clothes` list is `"shorts"`. If it is it should change the value to `"warm coat"`.

```
In [ ]: clothes = [  
        "shorts",  
        "shoes",  
        "t-shirt",  
        ]
```

Solution

```
In [4]: clothes = [  
        "shorts",  
        "shoes",  
        "t-shirt",  
        ]  
  
        if clothes[0] == 'shorts':  
            clothes[0] = 'warm coat'  
  
        print(clothes)
```

```
['warm coat', 'shoes', 't-shirt']
```

List Functions

There are functions designed for lists

- `len()`: the number of items in a list
- `max()`: The biggest value in a list
- `min()`: The smallest value in a list

In [5]: `costs = [1.2, 4.3, 2.0, 0.5]`

```
print(len(costs))  
print(max(costs))  
print(min(costs))
```

4

4.3

0.5

Functions for changing the order of a list

- `sorted()`: Sorts the
- `reversed()`: Reverses the order of a list

In [7]: `costs = [1.2, 4.3, 2.0, 0.5]`

```
print(sorted(costs))  
print(list(reversed(costs)))
```

```
[0.5, 1.2, 2.0, 4.3]
```

```
[0.5, 2.0, 4.3, 1.2]
```


Exercise 4.2: Make a list of game scores. Using list functions write code to output information of the scores in the following format:

```
Number of scores: 10  
Highest score: 200  
Lowest score: 3
```

Extension: Output all of the scores in descending order

append() and **in**

You can check if a value is in a list using the `in` operator. If the value is in the list it will result in `True` and `False` if it is not.

```
In [8]: student_name = input('Which student are you looking for? ')
students = [
    'Diedre', 'Hank', 'Helena', 'Salome',
]

if student_name in students:
    print('{} is in the class'.format(student_name))
else:
    print('{} is not in the class'.format(student_name))
```

```
Which student are you looking for? Hank
Hank is in the class
```

The `.append()` method is used to add items to a list

```
In [9]: students = [  
        'Diedre', 'Hank', 'Helena', 'Salome',  
        ]  
        student_name = input('What is the name of the new student? ')  
        students.append(student_name)  
        print(students)
```

```
What is the name of the new student? Fred  
['Diedre', 'Hank', 'Helena', 'Salome', 'Fred']
```

Exercise 4.3: Whenever I'm shopping and I buy some bread I always forget to buy butter. Create a list and if 'bread' is in the list, add 'butter' to the shopping list.

Try running the program with and without bread in the list to check that your program works.

Remember the `in` operator checks if an item is in a list and the `.append()` method adds an item to a list.

Solution

```
In [10]: shopping_list = [  
    'bread',  
    'cheese',  
    'pop tarts',  
    'carrots',  
]  
  
if 'bread' in shopping_list:  
    shopping_list.append('butter')  
  
print(shopping_list)
```

```
['bread', 'cheese', 'pop tarts', 'carrots', 'butter']
```

To check if an item is not in a list

```
In [11]: fridge = [  
          'cheese',  
          'pizza',  
          'coke',  
        ]  
  
if 'milk' not in fridge:  
    print('You have no milk in the fridge')
```

You have no milk in the fridge

For Loops ♥ Lists

Using lists and for loops together

```
In [12]: student_names = ['Diedre', 'Hank', 'Helena', 'Salome']  
  
for student_name in student_names:  
    print(student_name)
```

```
Diedre  
Hank  
Helena  
Salome
```

Counting the total number of items in a list using a for loop

```
In [13]: student_names = ['Diedre', 'Hank', 'Helena', 'Salome']  
count = 0  
  
for student_name in student_names:  
    count = count + 1  
  
print(count)
```

4

Exercise 4.4: I want to work out how much money I've spent on lunch this week. I've created a list of what I spent each day.

Write a program that uses a `for` loop to calculate the total cost

```
In [ ]: costs = [8.30, 7.12, 5.01, 1.00, 0.99, 5.92, 3.50]  
        total_cost = 0
```

Solution

```
In [14]: costs = [8.30, 7.12, 5.01, 1.00, 0.99, 5.92, 3.50]
total_cost = 0

for cost in costs:
    total_cost = total_cost + cost

print(total_cost)
```

31.839999999999996

There is an easier way to do the last program without a for loop. The `sum()` function can be used to add up all of the values in a list:

```
In [15]: costs = [8.30, 7.12, 5.01, 1.00, 0.99, 5.92, 3.50]
total = sum(costs)

print(total)
```

```
31.839999999999996
```

Dictionaries

Dictionary: Stores a collection of labelled items. Each item has a *key* and a *value*

```
In [ ]: person = {  
        'name': 'Jessica',  
        'age': 23,  
        'height': 172  
    }
```

Values in a dictionary are accessed using their keys

```
In [16]: person = {  
          'name': 'Jessica',  
          'age': 23,  
          'height': 172  
        }  
  
        print(person['name'])
```

Jessica

Exercise 4.5: Print the values of `name` , `post_code` and `street_number` from the dictionary

```
In [ ]: place = {  
    'name': 'The Anchor',  
    'post_code': 'E14 6HY',  
    'street_number': '54',  
    'location': {  
        'longitude': 127,  
        'latitude': 63,  
    }  
}
```

Extension: Print the values of `longitude` and `latitude` from the inner dictionary

Solution

```
In [17]: place = {  
    'name': 'The Anchor',  
    'post_code': 'E14 6HY',  
    'street_number': '54',  
    'location': {  
        'longitude': 127,  
        'latitude': 63,  
    }  
}  
  
print(place['name'])  
print(place['post_code'])  
print(place['street_number'])
```

The Anchor
E14 6HY
54

Extension:

```
In [18]: print(place['location']['longitude'])  
         print(place['location']['latitude'])
```

```
location = place['location']
```

```
print(location['longitude'])  
print(location['latitude'])
```

127

63

127

63

Dictionaries in Lists

Putting dictionaries inside a list is very common

```
In [19]: people = [  
    {'name': 'Jessica', 'age': 23},  
    {'name': 'Trisha', 'age': 24},  
    ]  
  
    for person in people:  
        print(person['name'])  
        print(person['age'])
```

```
Jessica  
23  
Trisha  
24
```

Exercise 4.6: Using a for loop, output the values name , colour and price of each dictionary in the list

```
In [ ]: fruits = [  
    {'name': 'apple', 'colour': 'red', 'price': 0.12},  
    {'name': 'banana', 'colour': 'yellow', 'price': 0.2},  
    {'name': 'pear', 'colour': 'green', 'price': 0.19},  
]
```

Solution

```
In [20]: fruits = [  
    {'name': 'apple', 'colour': 'red', 'price': 0.12},  
    {'name': 'banana', 'colour': 'yellow', 'price': 0.2},  
    {'name': 'pear', 'colour': 'green', 'price': 0.19},  
]  
  
for fruit in fruits:  
    print(fruit['name'])  
    print(fruit['colour'])  
    print(fruit['price'])
```

```
apple  
red  
0.12  
banana  
yellow  
0.2  
pear  
green  
0.19
```

Random Choice

The `choice()` function in the random module returns a random item from a list

```
In [21]: import random

colours = ['red', 'green', 'blue']
chosen_colour = random.choice(colours)

print(chosen_colour)
```

blue

Exercise 4.7: Write a program to create a random name. You should have a list of random firstnames and a list of lastnames. Choose a random item from each and display the result.

Extension: Using list of verbs and a list of nouns, create randomised sentences

Recap

This session:

1. Lists
2. Dictionaries

Question 1: What shape brackets are used for creating a list and what shape brackets are used for creating a dictionary?

Question 2: What is the result of this program?

```
In [ ]: cheeses = [  
        'brie',  
        'cheddar',  
        'wensleydale',  
        'edam',  
    ]  
  
    print(cheeses[4])
```

Question 3: This program raises an error when I run it. What do I need to change to get it to run?

```
In [ ]: trees = [  
        {'leaf_colour': 'green', 'height': 2120},  
        {'leaf_colour': 'green', 'height': 2300},  
  
        new_tree = {  
            'leaf_colour': 'green',  
            'height': 1020  
        }  
  
        trees.append(new_tree)  
  
        print(trees)
```

Homework: Session 4 homework questions in your student guide