Level 4

samkary Parmaga

FOR LEARN BASIC PYTHON

Lesson 1

# Tuple

**Introduction:**

In the previous level, you learned a storage method, which is a list, and you learned some functions to deal with the list, such as how to add an element, how to delete it, how to find it, how to arrange it, etc. In this level, you will learn different storage methods, such as tuple, dictionary, set. You will also learn how to save yourself writing codes and how to create a function.

**What is a tuple?**

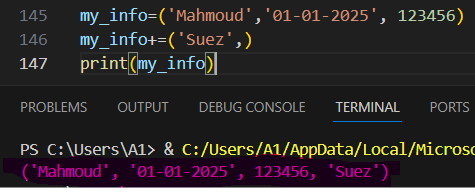
A tuple is a collection that is ordered and unchangeable. Once you create a tuple, you can’t add or remove items from it.

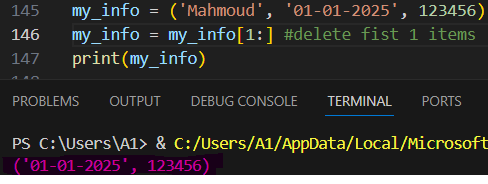
Note: Tuples are written with round brackets

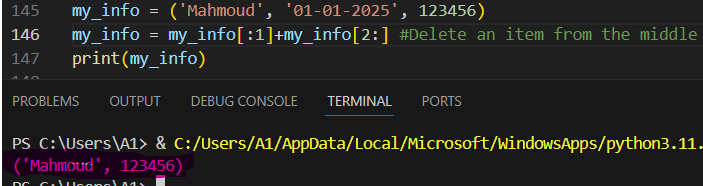
But what does unchangeable(immutable) mean ?

The meaning is that you cannot add an element, delete an element, or change its value, but you can add or delete indirectly.

Example:







**Use case in real life:**

**Think about your national ID — your name, birthday, and ID number never change. That's like a tuple.**

**Syntax and Example:**

**python**

**my\_info = ("Mahmoud", "2005-01-01", "123456")**

**print(my\_info[0])**

**Output:**

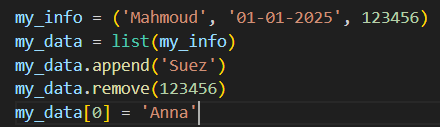
**Mahmoud**

**But how do I update a value in the tuple?**

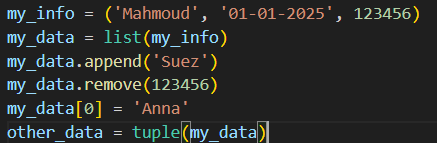
1. Convert tuple into list:



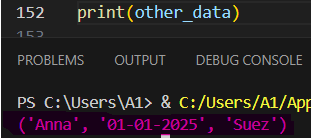
1. Add item or remove item or update item in list:



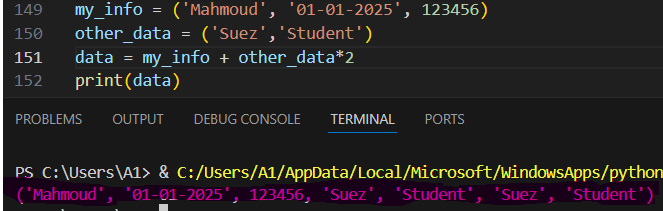
1. Convert list into tuple:



1. Print tuple:

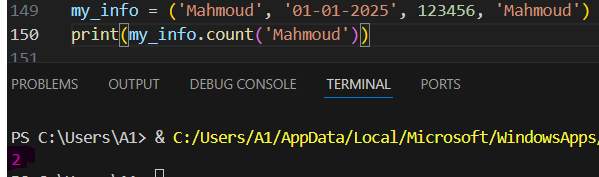


A tuple can do more than one thing. It can even add two tuples or make two into one tuple, like:

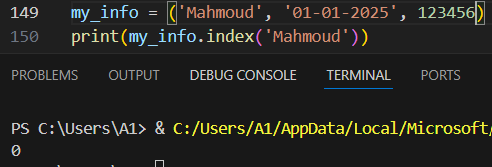


**Tuple Methods:**

1. **Count():** Returns the number of times a specified value occurs in a tuple like:



1. **Index():** Searches the tuple for a specified value and returns the position of where it was found like:



**Finish the lesson**

Lesson 2

**Set**

**Introduction:**

In the previous lesson, you learned about a data type, which is tuples. In this lesson, you will study the third type, which is used only to store data without a similar stored value.

**What is a set?**

A set is a collection that is unordered, does not allow duplicates, and items cannot be accessed by index.

**Use case in real life:**

**Imagine a checklist for your daily routine. You don’t want to repeat tasks, and you only care if something is done or not — not its order.**

**Example:**

**python**

**daily\_tasks = {"wake up", "brush teeth", "go to school"}**

**print(daily\_tasks)**

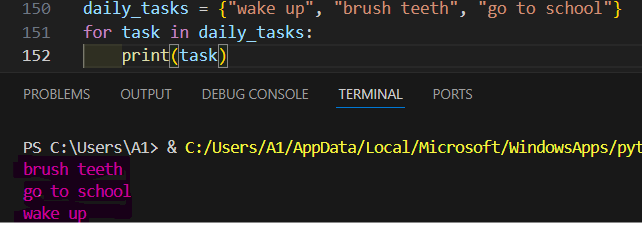
**Output:**

**{'wake up', 'brush teeth', 'go to school'}**

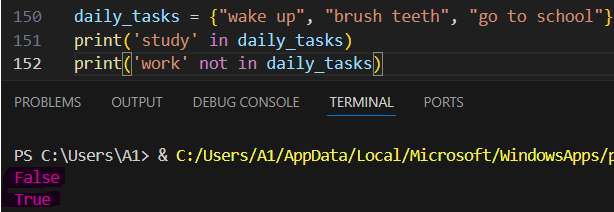
**But how can you access item in the set** http://cdn.shopify.com/s/files/1/1061/1924/products/Thinking_Face_Emoji_grande.png?v=1480481060 ?

To access any item in the set, you have two ways:

1. For loop

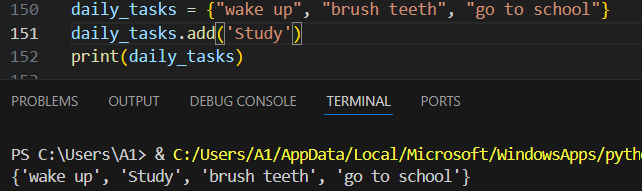


1. Mempersip operator(in-not in)

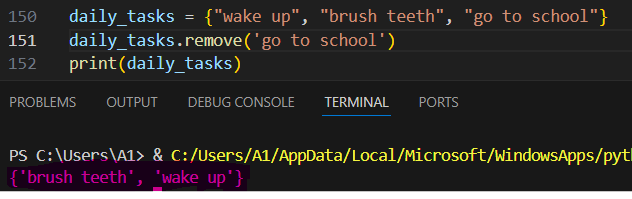


**Set Methods:**

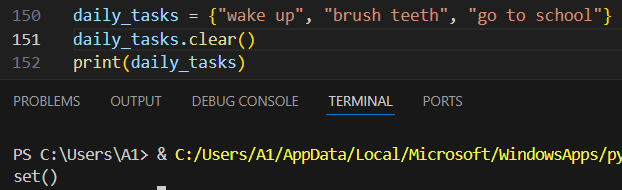
1. **Add ():** add an element to the set

****

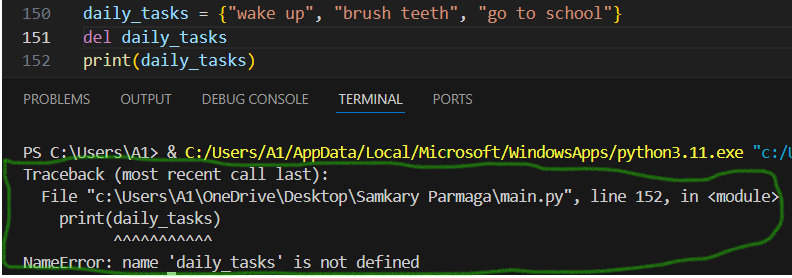
1. **Remove ():** remove a specific element in the set

****

1. **Clear ():** remove all items in the set



1. **Del:** delete the set completely

****

**Finish the lesson**

Lesson 3

**Dictionary**

**Introduction:**

In the previous lesson, you learned that in the set you add elements, but the most important thing is that it is not duplicate. Also, it does not allow you to search for a word through it, or you can allow it through the index. In this lesson, we learn a more professional way to solve these problems, which is the dictionary.

**What is a dictionary?**

A dictionary stores data in key-value pairs. Each key must be unique.

**Use case in real life:**

Think about a contact list — you search for a name (key), and you get the phone number (value).

**Example:**

**python**

**contact = {"name": "Ali", "phone": "01123456789"}**

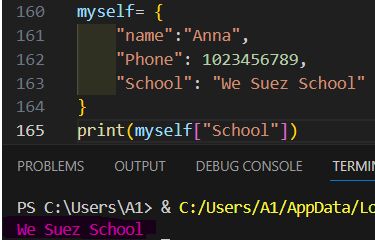
**print("Phone:", contact["phone"])**

**Output:**

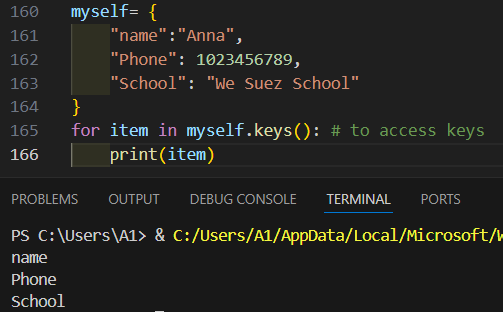
**Phone: 01123456789**

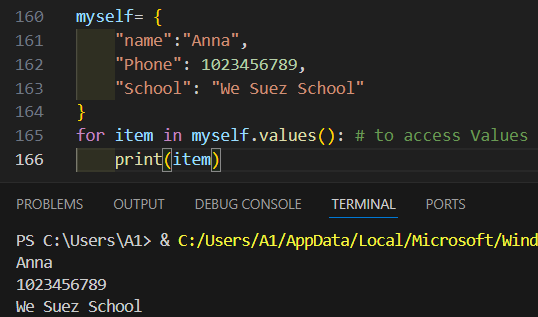
**How to access any item in the dictionary?**

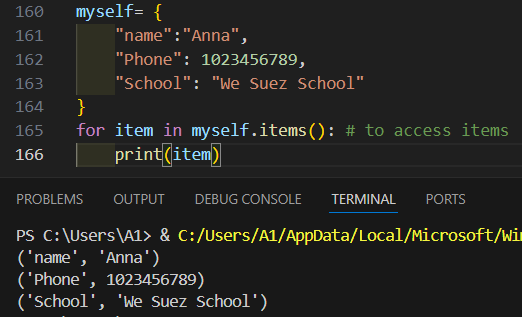
* **By referring or searching for the name of the key:**



* **For loop:**

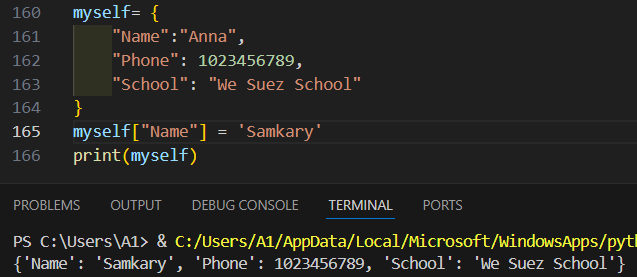




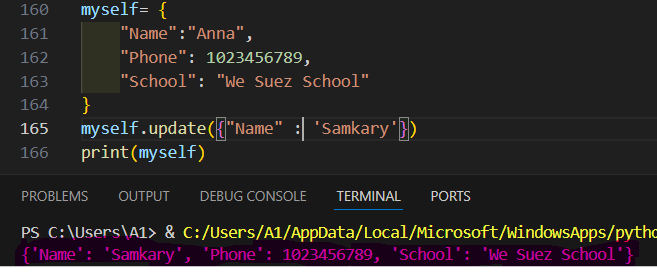


**How to change the value of a key in a dictionary?**

* referring to its key name:



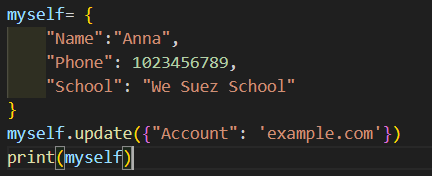
* Using update() method**:**



**Dictionary methods:**

1. Update (): Add and update any item in the dictionary

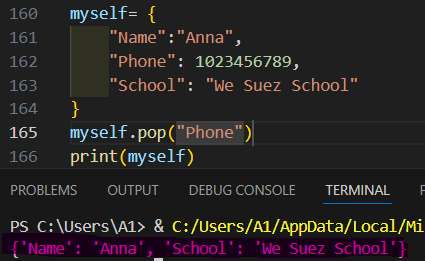
Syntax: update(key, value)



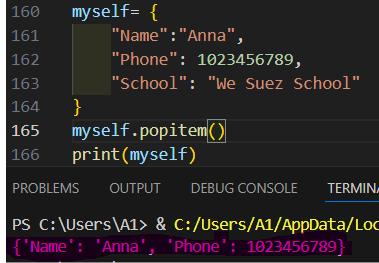
Output:

{'Name': 'Anna', 'Phone': 1023456789, 'School': 'We Suez School', 'Account': 'example.com'}

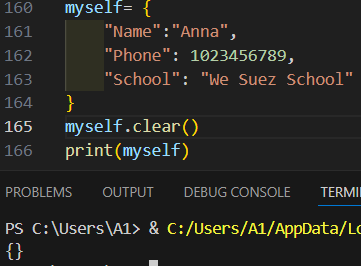
1. Pop (): Removes the element with the specified key



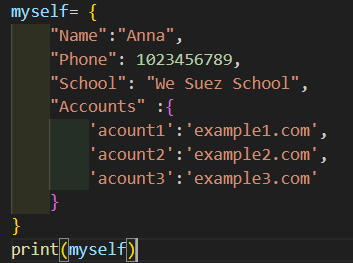
1. Popitem (): Removes the last inserted key-value pair



1. Clear ():Removes all the elements from the dictionary



You can work on more than one dictionary within one dictionary, and this is called: nested dictionary example:



Output:

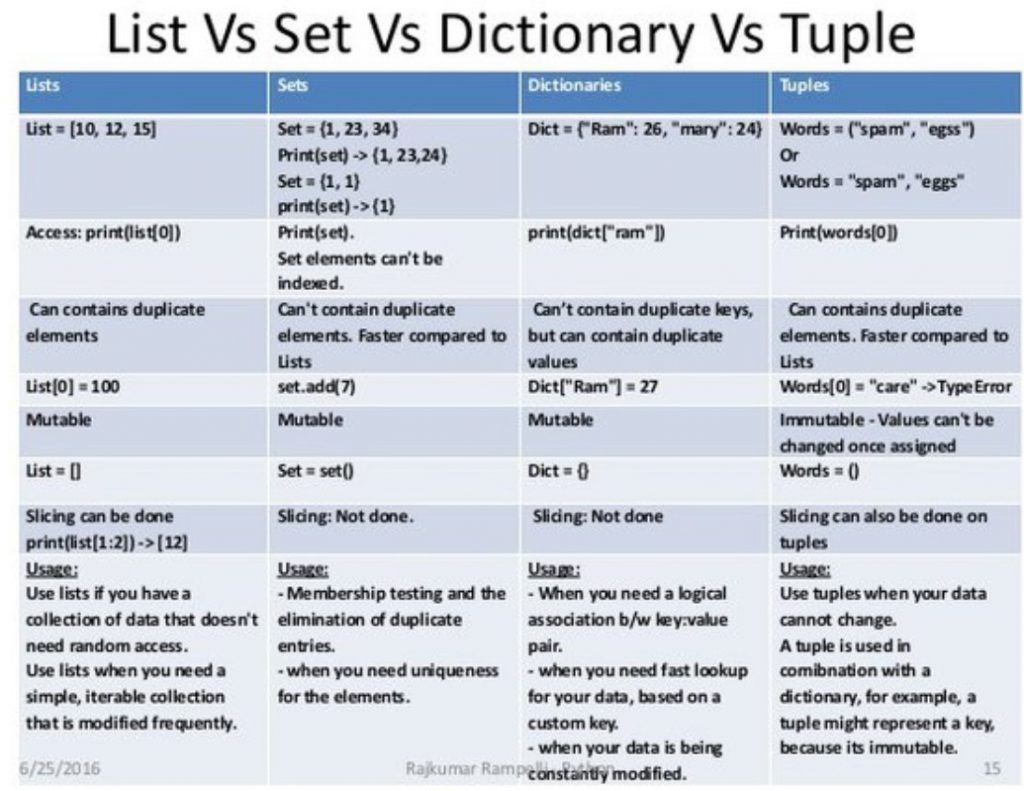
{'Name': 'Anna', 'Phone': 1023456789, 'School': 'We Suez School', 'Accounts': {'acount1': 'example1.com', 'acount2': 'example2.com', 'acount3': 'example3.com'}}

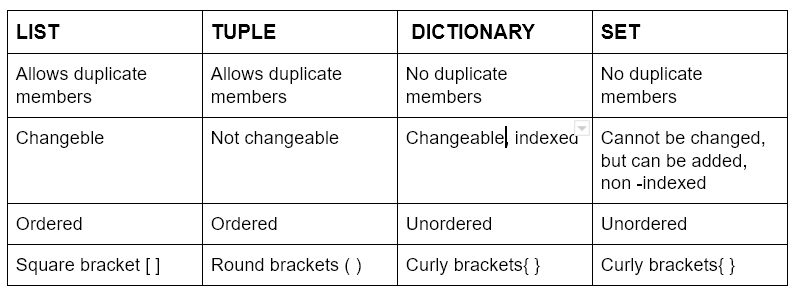
**Finish the lesson**

Lesson 4

**Functions**

**Remember:**





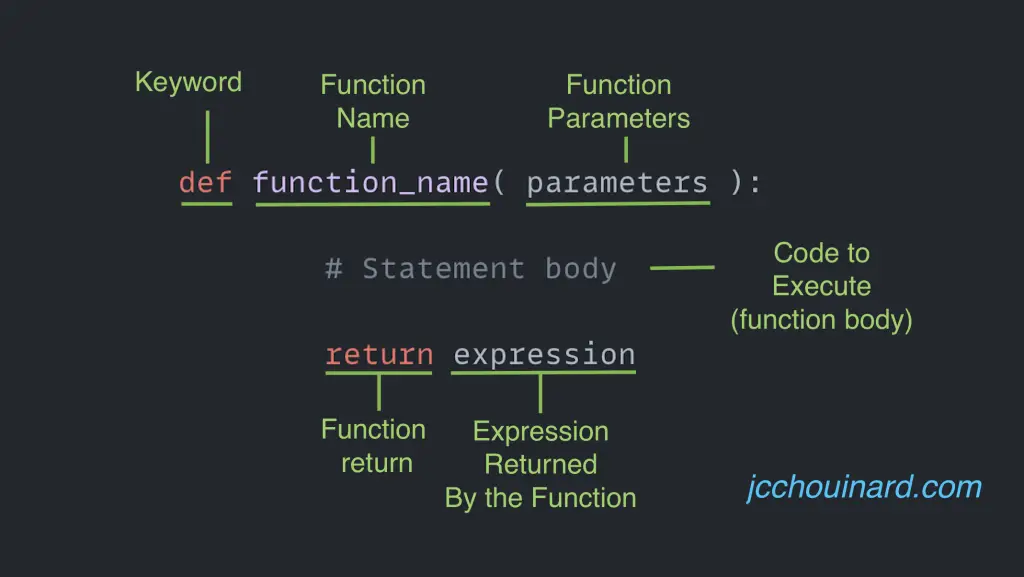
**What is a function?**

A function is a block of code that runs when you call it. It helps you avoid repeating code.

**Use case in real life:**

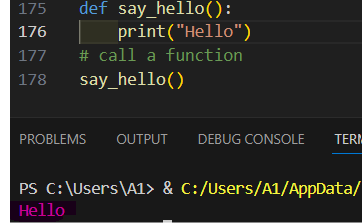
A vending machine is like a function. You press a button (call the function), and it gives you a drink (result).

**Syntax:**



**Basic Function:**

**python**

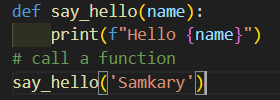


**Why function?**

* Code Reusability
* Code Organization
* Easy to Maintain
* division of tasks (Modularity)
* Facilitate testing

**What is a parameter and arguments?**

Parameter

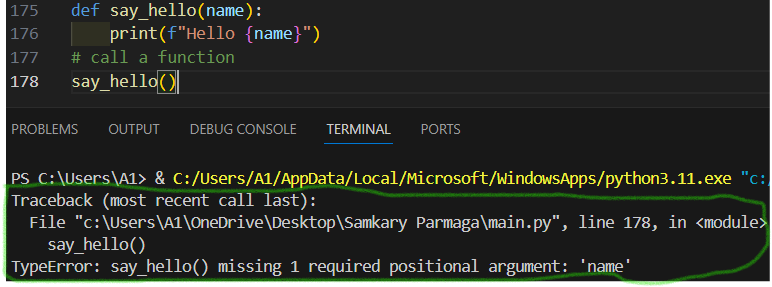


Arguments

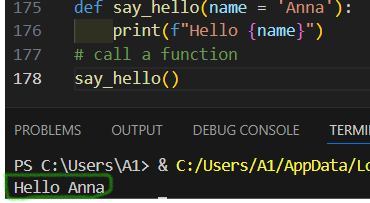
A parameter is the variable listed inside the parentheses in the function definition.

An argument is the value that is sent to the function when it is called.

When setting a parameter, you must ensure that the argument has the same number as the parameter.

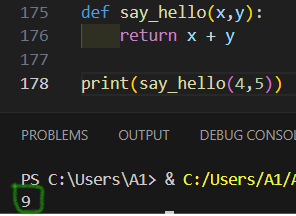


**You can solve this problem by setting a default value for the parameter.**



**Return values:**

**Return:** Used to retrieve a value from a function.



**Finish the lesson**

Lesson 5

**Project level4**

This is final level: you require to create this is project: -

**Build a simple To-Do List using a set and multiple functions. The user should be able to:**

* **Add a task**
* **Delete a task**
* **Show all tasks**
* **Mark a task as done**
* **Check if a task is done**

**Output:**

**Added task: Finish homework**

**Added task: Clean room**

**Your To Do List:**

**- Finish homework [✗]**

**- Clean room [✗]**

**Marked as done: Finish homework**

**✅ 'Finish homework' is done.**

**Deleted task: Clean room**

**Your To Do List:**

**- Finish homework [✓]**