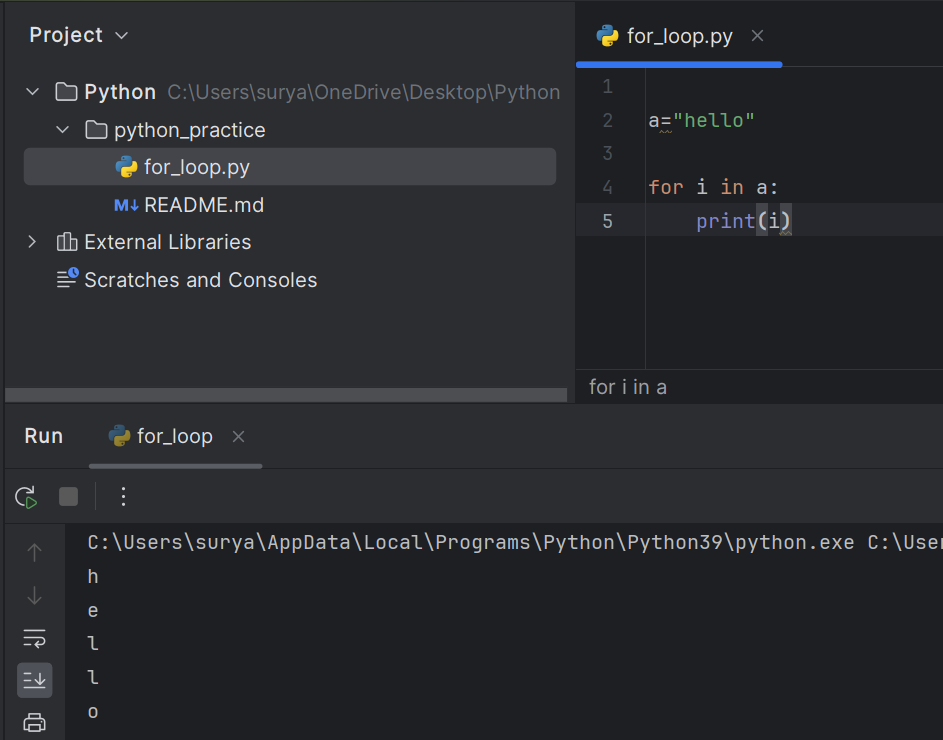
Python

**Loop:**

Execute a block of code repeatedly based on a condition

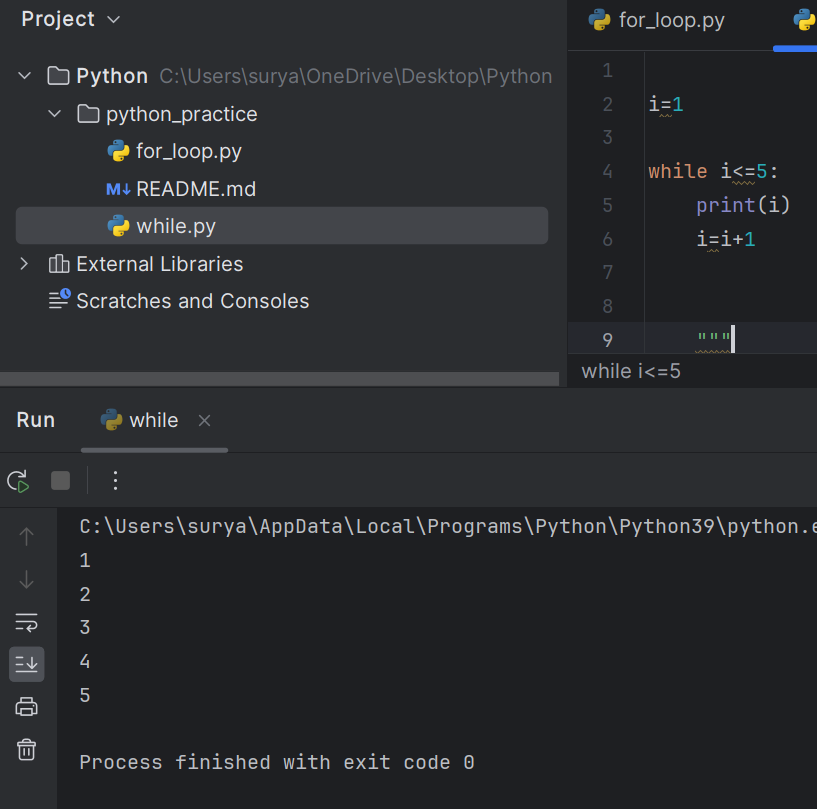
**For Loop:**

We know in advance how many times we want to execute a statement



**While Loop**:

Execute the loop until the condition is get satisfied



**Different of for and while:**

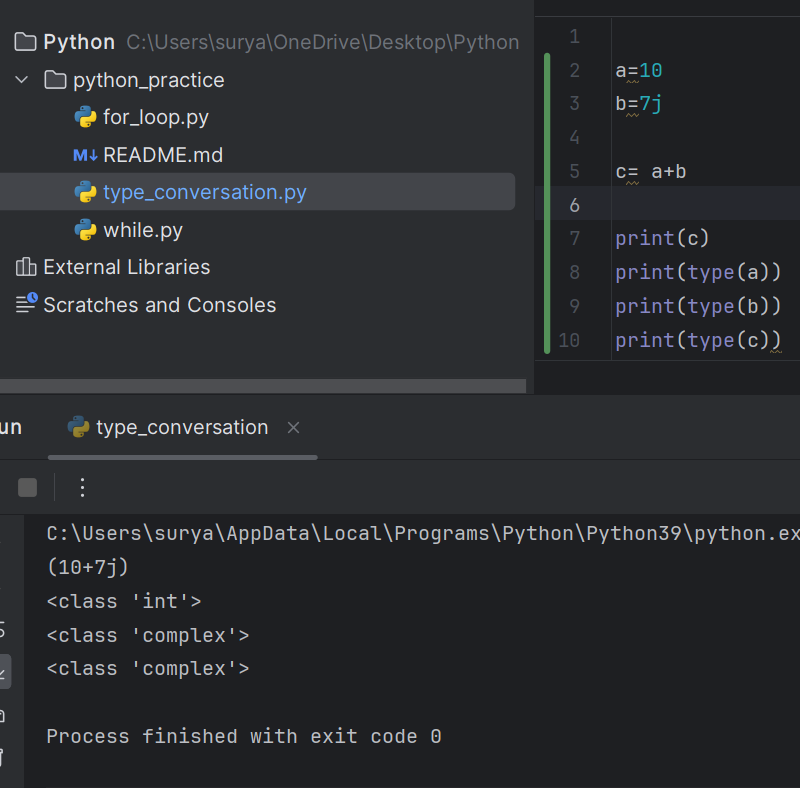
**For:**

* Best for iterating over a known sequence (list, tuple, string, etc.) or a specific range of values.

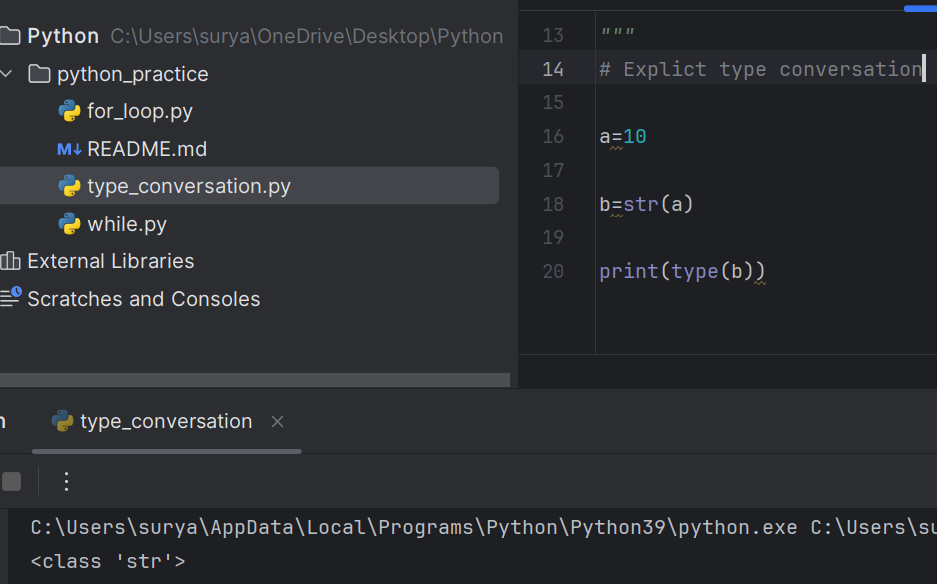
**While:**

|  |
| --- |
| * Best for repeating an action while a condition is True, typically when the number of iterations is unknown.   **Type Conversation:**  Type conversion in Python is the process of converting one data type into another. Python provides two types of type conversion:   1. **Implicit Type Conversion**: Python automatically converts one data type to another without any user intervention. 2. **Explicit Type Conversion (Type Casting)**: The programmer manually converts one data type to another using functions like int(), float(), str(), etc.   **Implicit Type Conversion:** |

|  |
| --- |
|  |



**Explicit Type Conversion:**



**Data types:**

Data type specify which type of value the variable have to store

1. **Numeric Type:**

Int

Float

Complex

1. **Sequence Type:**

List

Tuple

String

1. **Mapping Type**

Dict

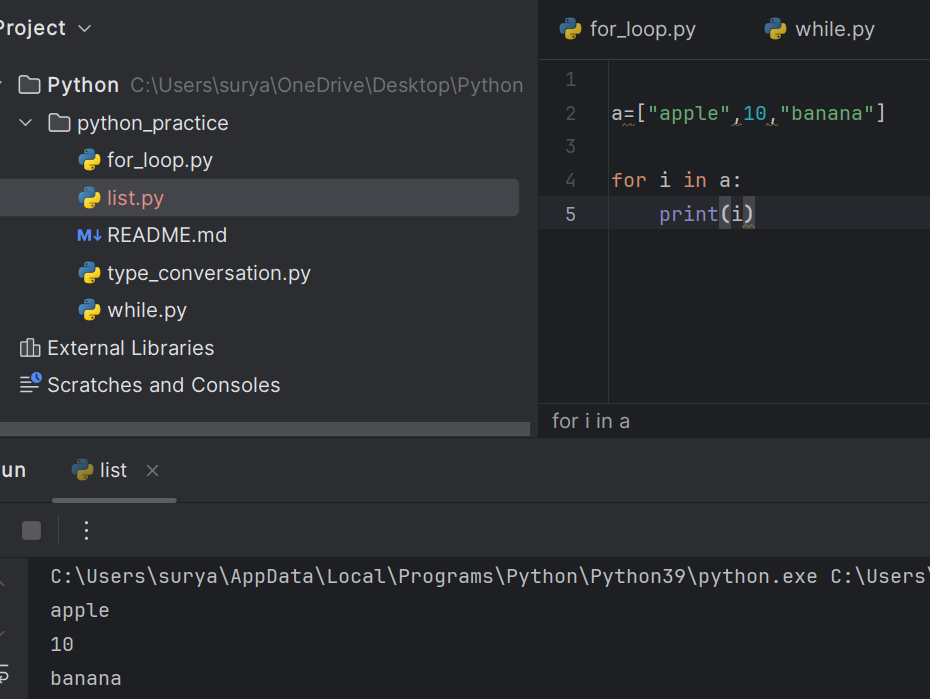
1. **Set Type**
2. **Boolean**
3. **None**

**List Type:**

Ordered collection of elements

Mutable

Allow Duplicates



**List Methods:**

Append

Count

Index

Extend

Insert

Pop

Remove

Clear

Copy

Reverse

Sort

**Tuples:**

Ordered collection of elements

Allow duplicates

Immutable

**Set:**

Unordered collection of elements

Mutable

Does not allow Duplicates

**Dictionary:**

The values that are stored in a key value pair

Values may duplicate but the key must be unique

**Real time usage of list tuple set and dictionary:**

**List:** Shopping cart, Todo list

**Tuple:** Database records, Configuration setting

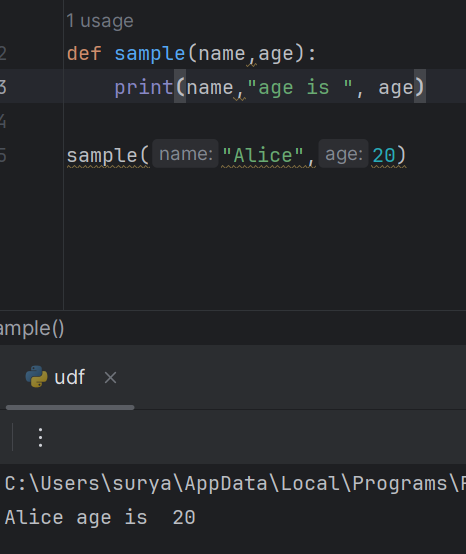
**Set**: The thing which we need not to get duplicate records we use set

**Dict**: Count a frequency of word

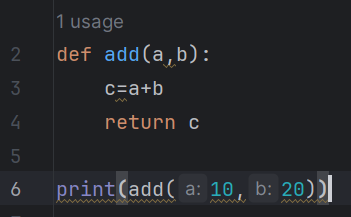
**User define functions:**

User define functions are block of code that performs a specific task

**Function with parameters:**

****

**Returning Values:**

* Return statement allows a function to send back a result to the caller
* ****After a return statement is executed, the function terminates, and control is passed back to the caller.

**Lambda:**

A lambda function is a small anonymous function**.**

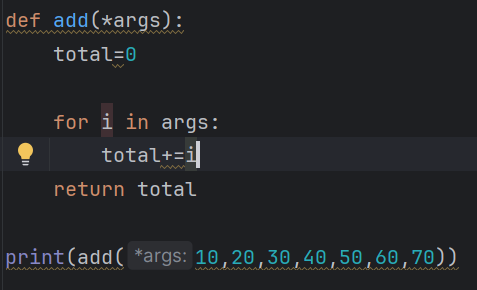
**lambda *arguments:* *expression***

**x= lambda a,b : a+b**

**print(x(10,20))**

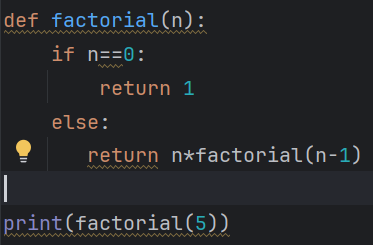
**Arbitrary function:**

It is used to give a multiple argument to a function



**Recursion function:**

It calls itself in order to solve a problem

****

**String:**

String is a immutable data type

**String Methods:**

Upper

lower

captalize

casefold

swapcase

isupper

islower

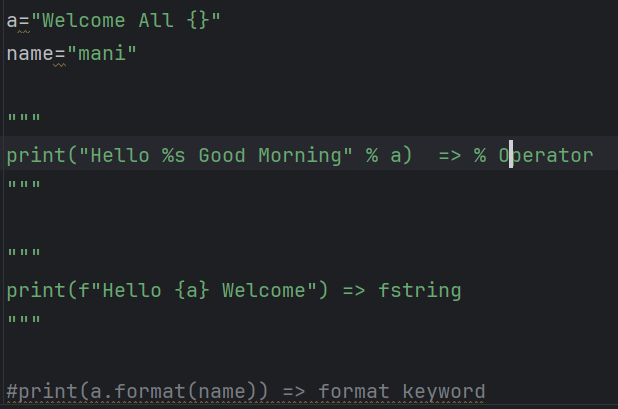
title

count

index

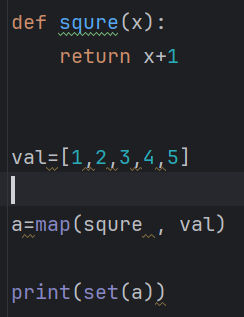
join

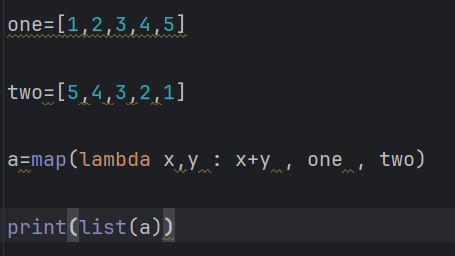
strip

**String Format: **

**Map Function**:

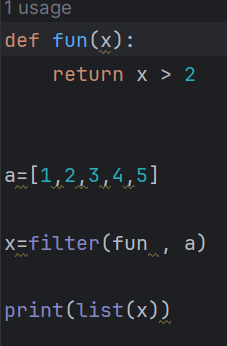
Instead of loop we use iterate the value using these functions

 Map is used to apply a function on each and every element of the iteratable



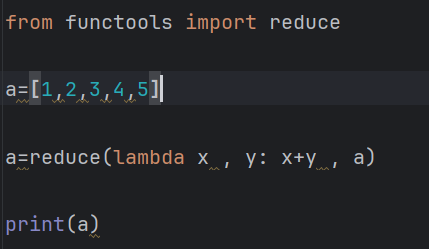
**Filter:**

It also used to apply a function to each element but it filters the values based on conditions



**Reduce:**

Reduce function reduce a list of items in a single unit of result

****

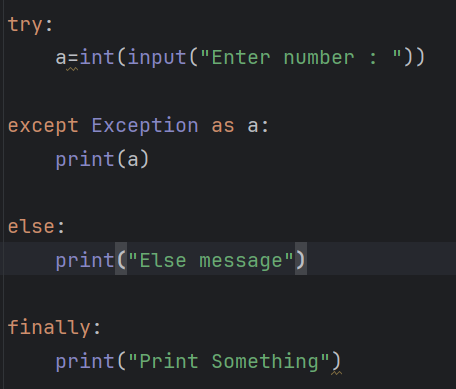
**Expectation handling:**

In Program the error may arise in 3 ways:

**Compile time error**

**Logical error**

**Run time error**

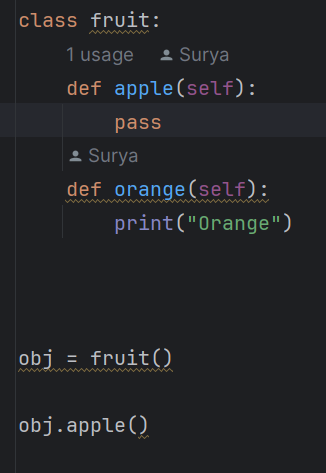
Expectation handling is used to avoid run time errors

**Class and object:**

Python is an object-oriented programming language.

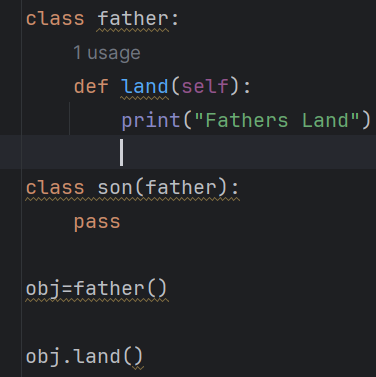
Almost everything in Python is an object, with its properties and methods.

A Class is like an object constructor, or a "blueprint" for creating objects.



**Inheritance:**

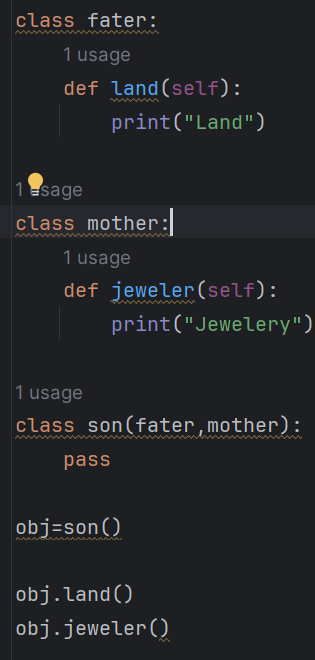
* Inheritance in Python allows a class (child class) to inherit properties and methods from another class (parent class).
* Inheritance provides the reusability of a code. We don’t have to write the same code again and again**.**

****

**Single inheritance:**

A child class inherits from only one parent class

**Multiple Inheritance:**

 A child class can inherit from more than one parent class.

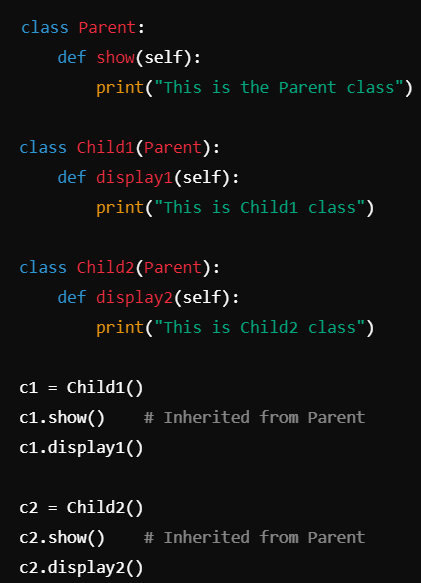
**Multilevel Inheritance:**

A class can inherit from a child class, creating a chain of inheritance.



**Hierarchical Inheritance**:

A same Parent class is inherent by multiple child class

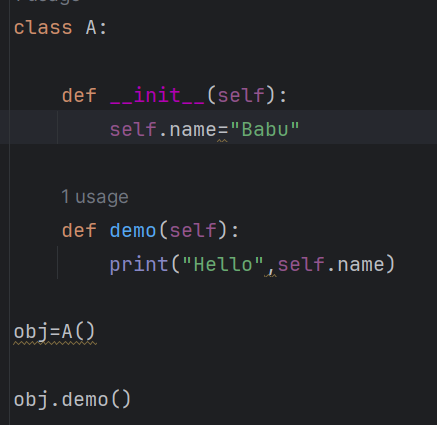


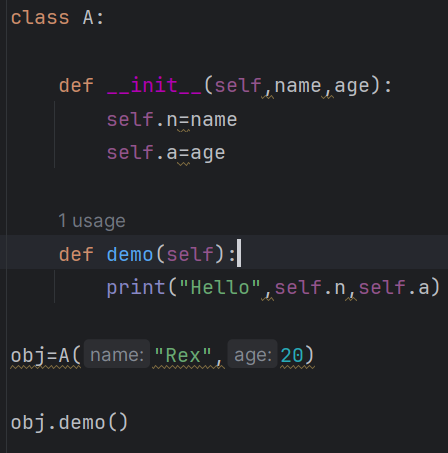
**Hybrid Inheritance:**

A combination of multiple types of inheritance.

**Constructor:**

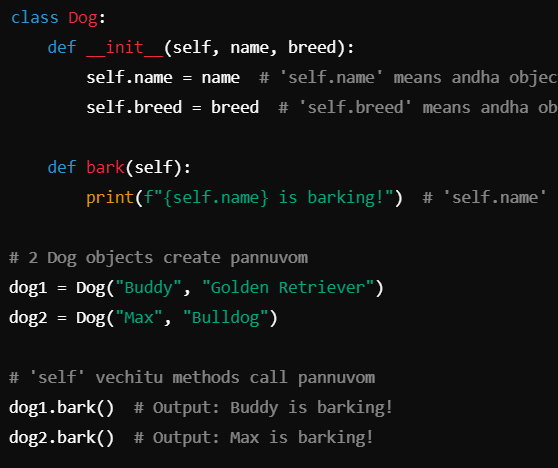
The constructor is used to initialize variables (attributes) in a class function**.**

****

****

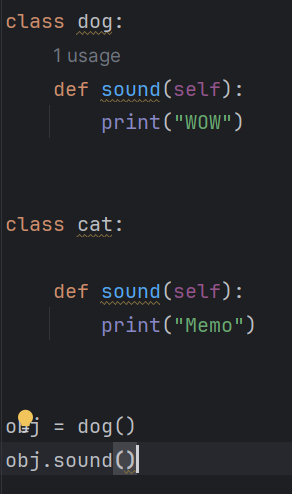
**Self-Keyword:**

Self-keyword refers current object of a class

****

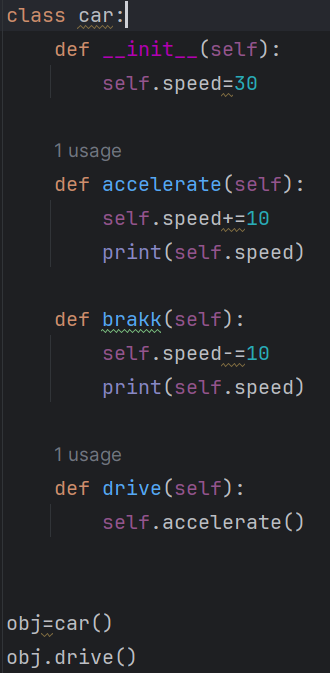
**Polymorphism:**

Same function used in different class for different purpose

****

**Abstraction:**

**Hiding the internal functionality and shows only the necessary information is called abstraction**



**Encapsulation:**

* Wrapping up of data and methods into a single unit
* It Prevents a data modification by limiting the access to variables and methods

**Method overriding:**

 When a sub class provides its own version of a method that already present in a parent class