Python

Python is an object oriented programming language which is platform independent which is used to develop applications for web, scientific, mathematical computations, AI, Datascience, Machine Learning.

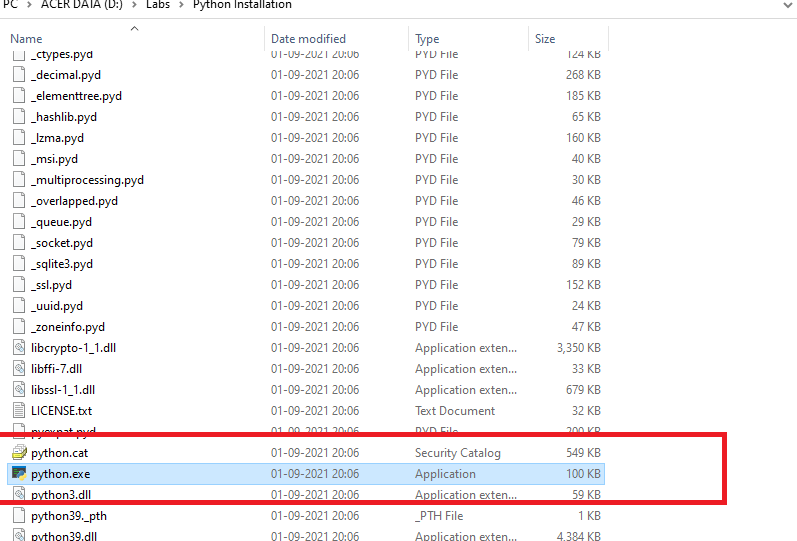
It is easier to write & understand, you don’t have to compile you can directly run it.

It is one of the more widely used object oriented programming language after Java

Downloading python

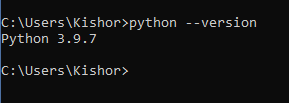
* You can download the installation zip from the python website
* Unzip the installation folder

Note: You can run the python program without compiling, using python command, once you get the installation directory you can notice a python.exe file



You need to set this location in the system environment variable

How to verify the python installation



Python enables programmers to write programs in a compact & better readable way, most of the codes in python are shorter than C, C++ and Java

Features of python

1. It is very simple to understand
2. Free & Open Source
3. Portable: You can move code from one platform to another platform without changing
4. Most of the complex operations can be done in single statement
5. It is interactive which allows you to directly write & run programs without any editor, i.e., it gives you the terminal where you can write python programs & run it
6. It is object oriented
7. Simple syntax
8. It is not strict like you don’t have to write semi-colons, no need to write types for variables, returns types for functions
9. It forces you to follow some good practices like proper indentation, which makes beginners to learn the language easily

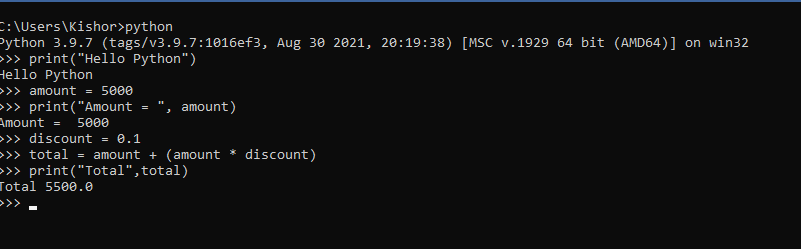
Applications using python

1. Amazon
2. Instagram
3. Mozilla

Python programs

# It is used to comment, it’s a single line comment, python has only single line comment, but multiline comment is written in different way ‘’’….‘’’

Simple calculation in python



Notice that variables are not created with any datatypes and also no python file is created.

To come out from the python terminal use Control + Z and hit enter.

When you are using print(), you can only concatenate string not other types of values

name = “Alex”

print(“Name = “+name); // ok

salary = 35000

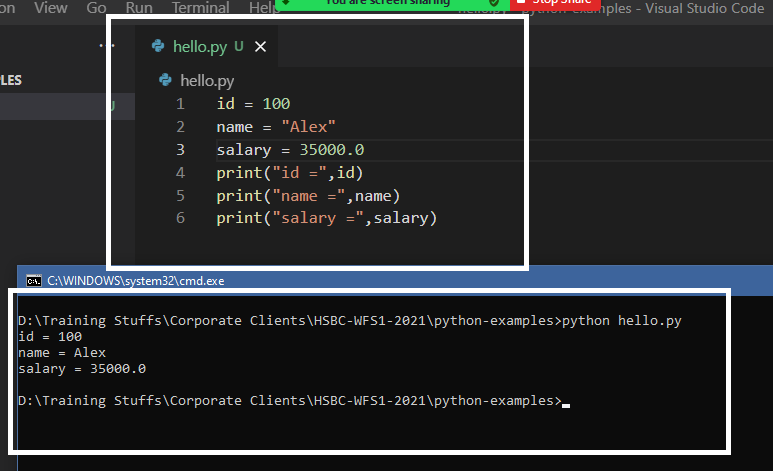
print(“Salary = “+salary); // gives error

Python has an inbuilt function type(variable\_name) which returns the type of value a variable has stored.

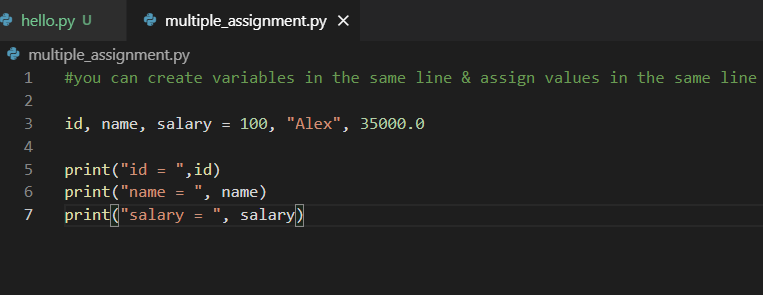
You can perform assignment to multiple variables in the same line in python

a,b,c = 10, 20, 30

Here a = 10, b = 20, c = 30



Multiple assignment in the same line



How to take input from the keyboard

We have input() it waits for the input from the user, you can also pass a message in the input like input(“Enter value”) which is displayed while asking the input

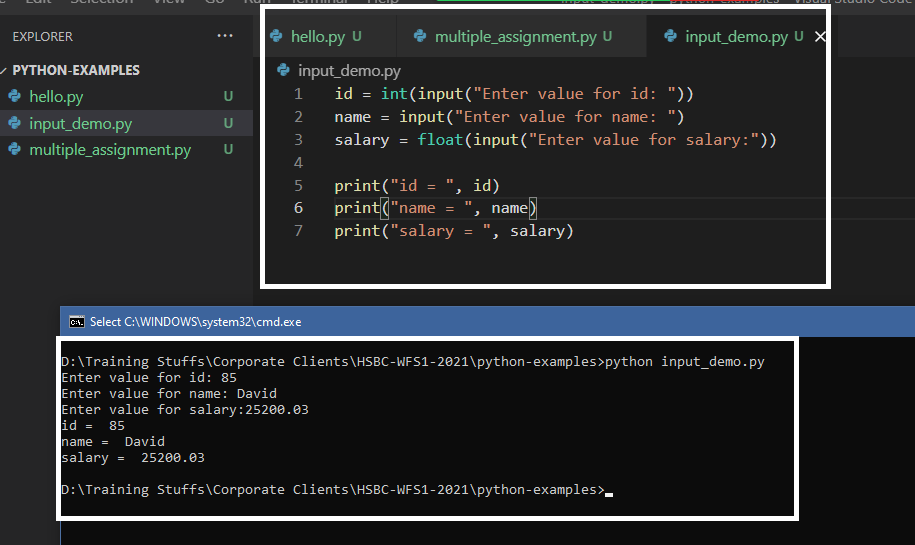
input() returns string by default, but if you want a specific type of value you can use the type like int, float,

x = input(“Enter value”)

y = int(input(“Enter value”))

z = float(input(“Enter value”))

Taking input example



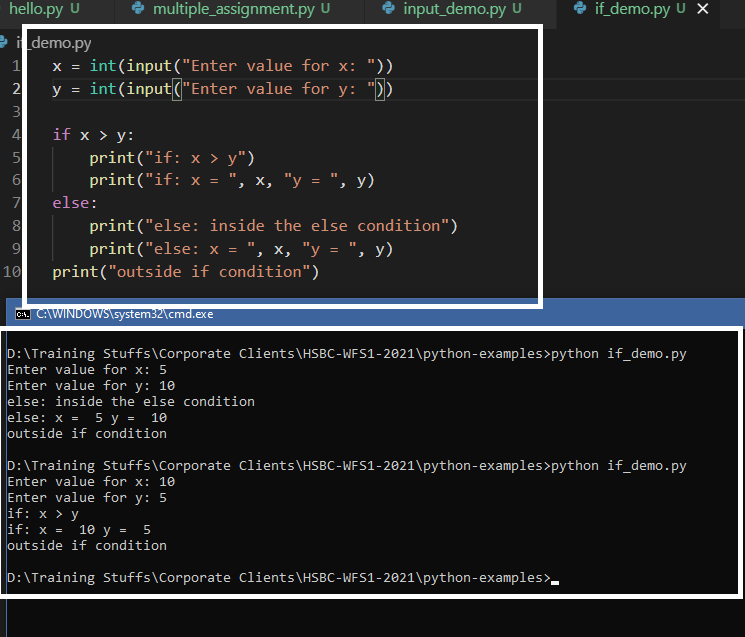
Operators

+, -, \*, /, =, !=, ==, <, >, <=, >=, |, &, %, ^

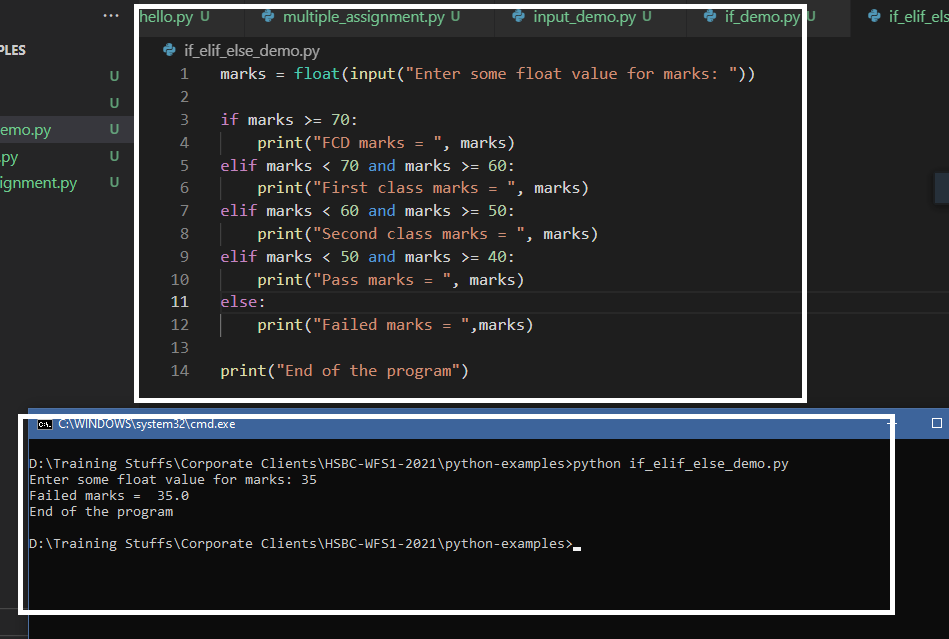
Control Statements

1. if
2. if elif elif else
3. for loop
4. while loop

if else example

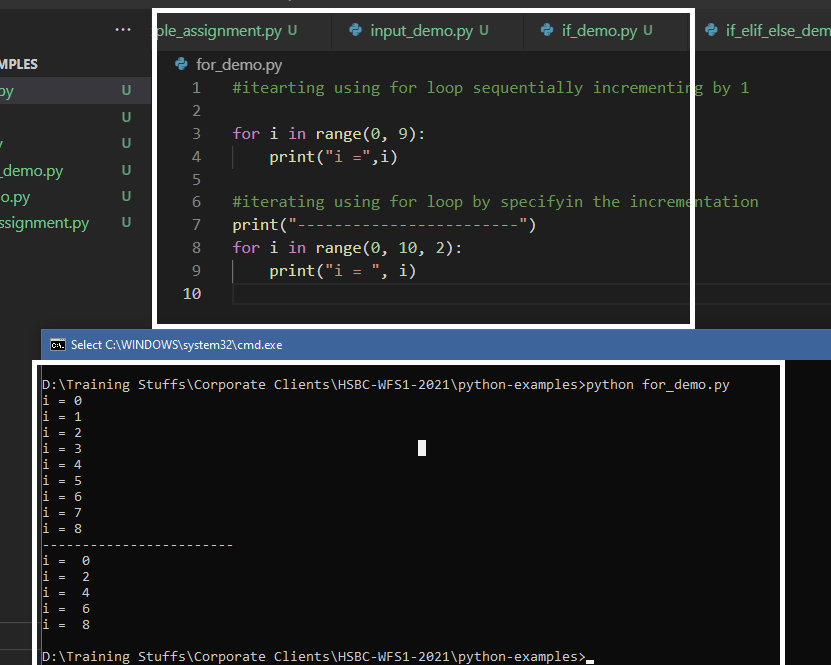


if elif elif example

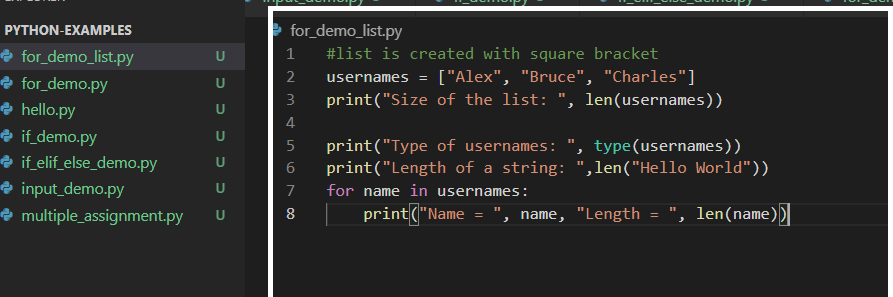


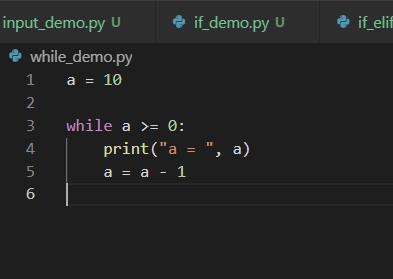
for loop

If you want to iterate over the list or tuples or dictionary or set you can use for loop and also you want to execute some statements sequentially then also you can use for loop

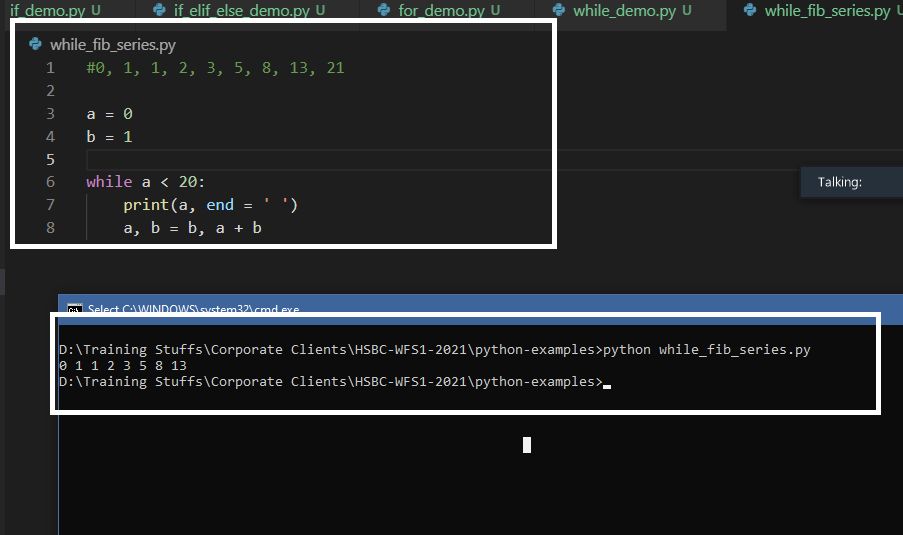


Iterating over the list



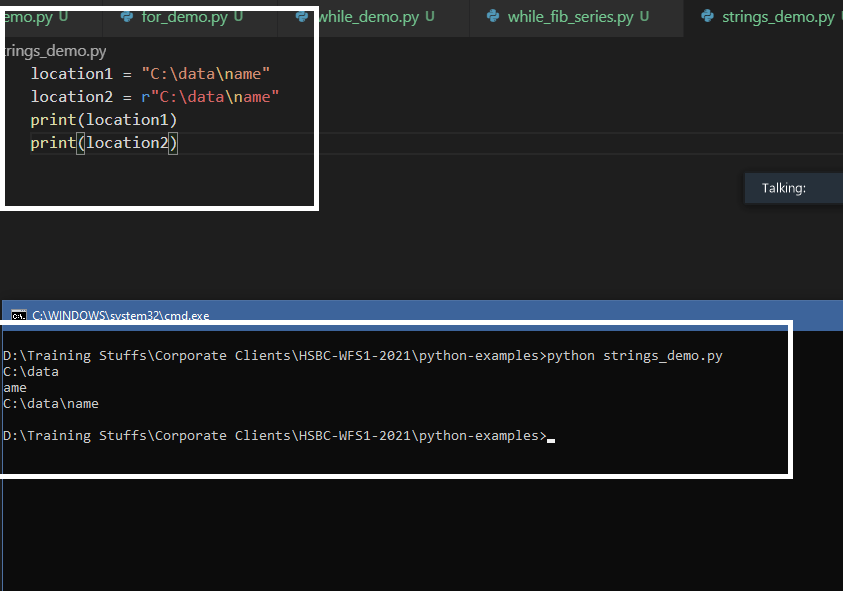
while loop  


Fibonacci series with while loop



Strings:

Use ‘r’ in the beginning of the string to consider the raw string



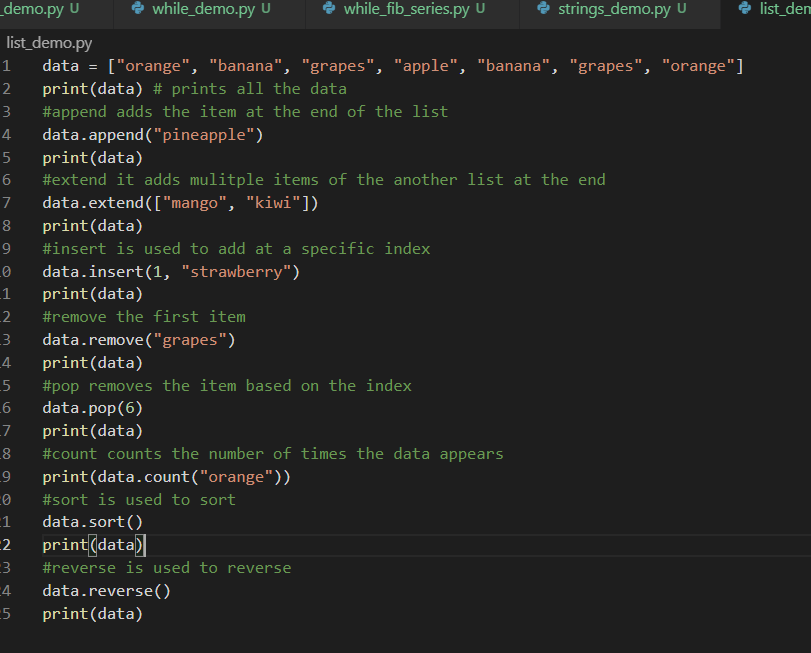
Datastructures in Python

It is used to maintain data in different way, following are the datastructures

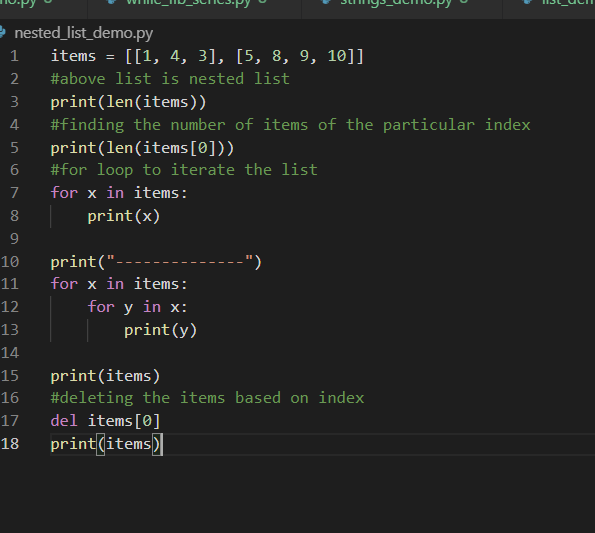
1. list []- stores the data in a sequential order & supports duplicates and mutable i.e., you can modify the same list
2. tuples ()- stores the data in a sequence order, supports duplicates and immutable i.e., you can’t modify the same tuple
3. set {} - stores only unique data, order is not maintained
4. dictionary {} - stores data in key value pairs, where key must be unique

Suppose you store the data in a list you can use some of the methods like

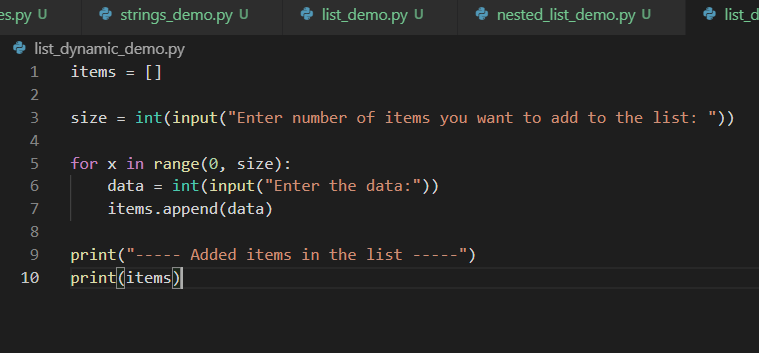
append(), pop(), remove(), index(), count(), extend(), insert(), reverse(), sort() and so on



Nested List: A list inside another list

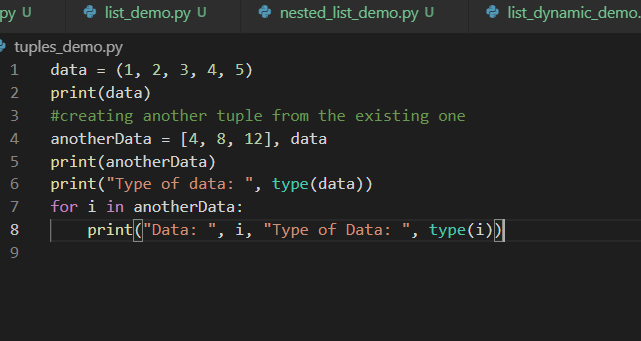


Adding the items at runtime

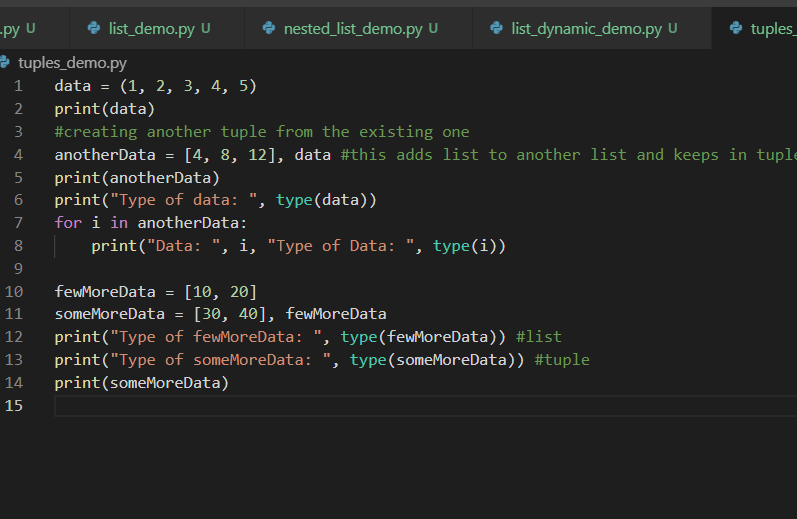


Tuples: They are immutable and can store heterogenous data

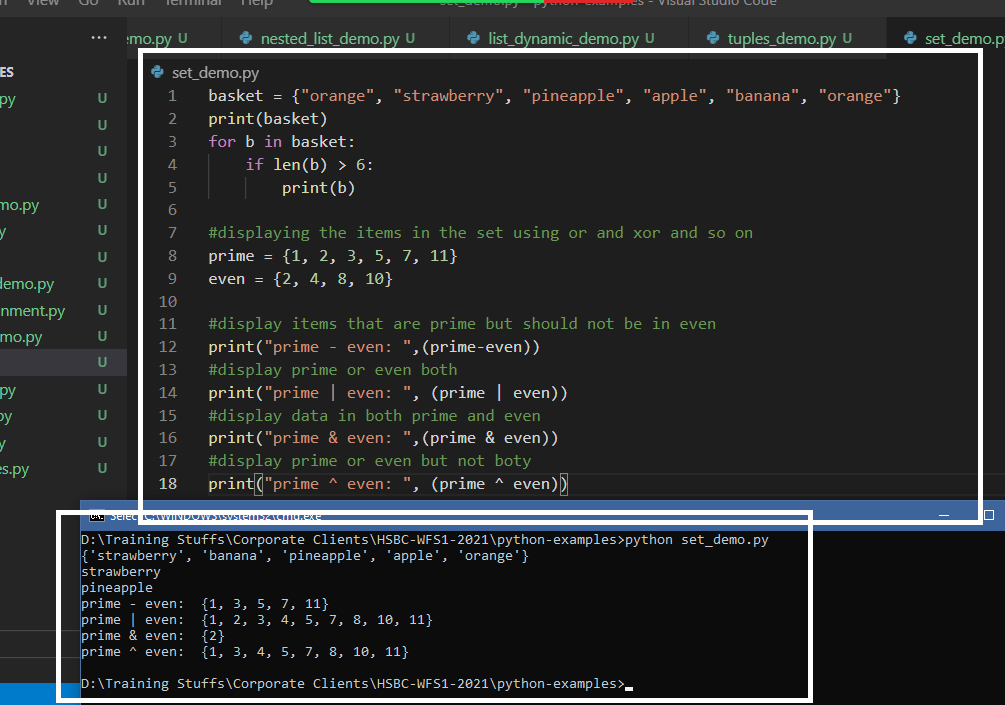
Heterogenous means you can have a list, set, dictionary inside the tuple



Tuple taking heterogenous data



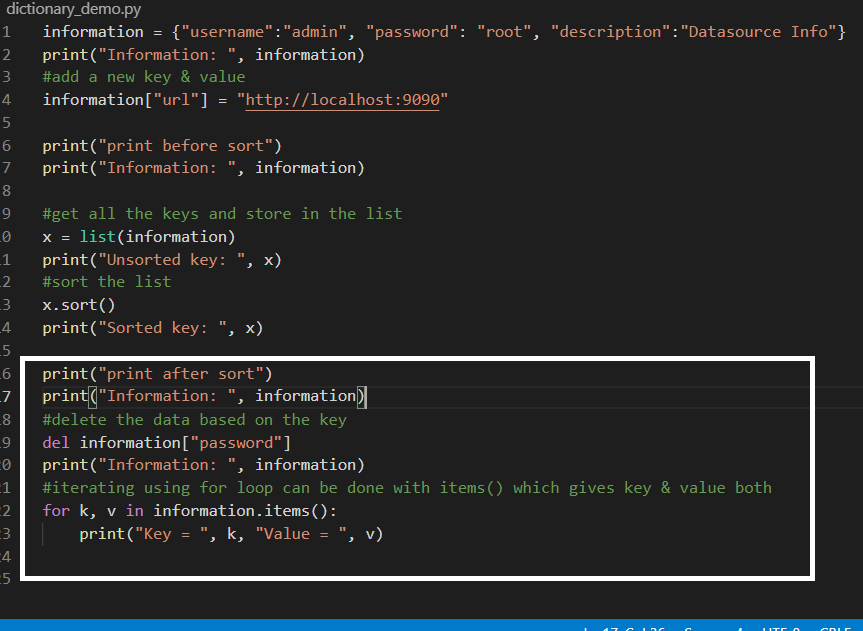
Set {}: It stores only unique data, it doesn’t maintain duplicates & the order



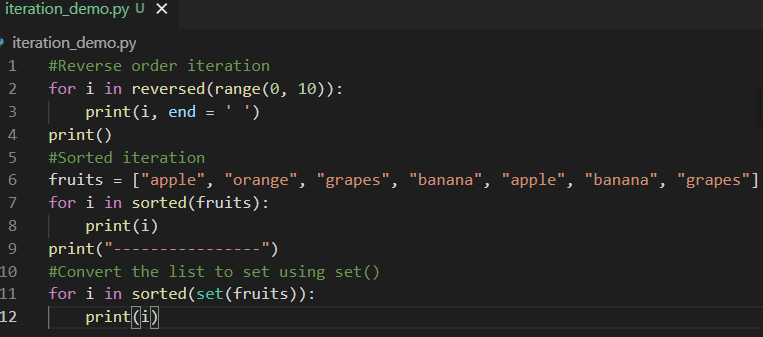
Dictionary:

It stores data in key value pairs

{key : value, key : value}

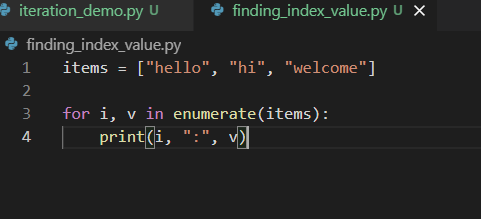


How to iterate in reverse order

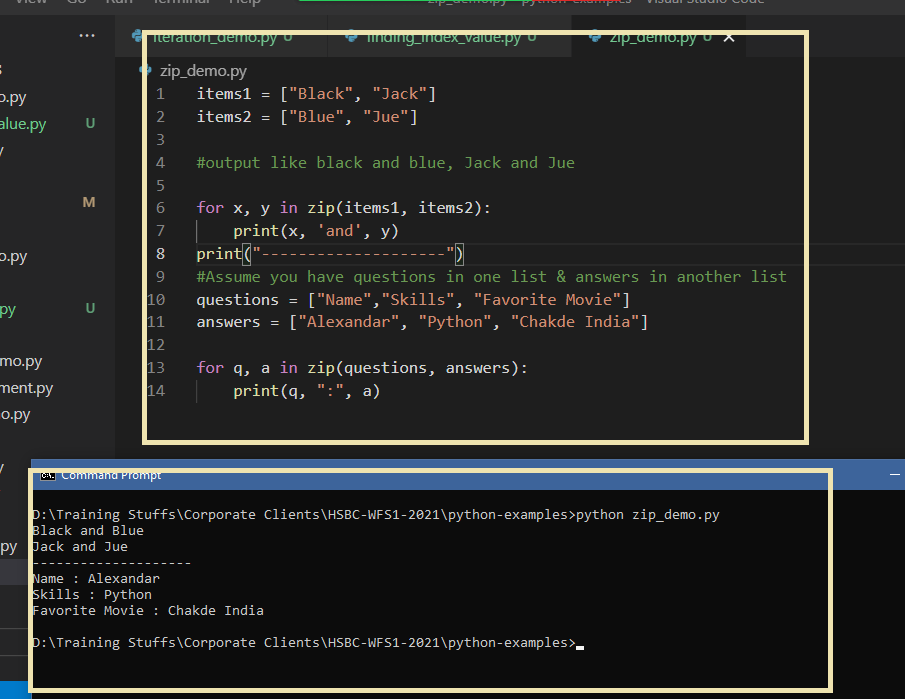


How to find the index and value both while iterating

We can use enumerate() to find the index & value both



We have zip() to iterate over multiple datastructure



Functions in python:

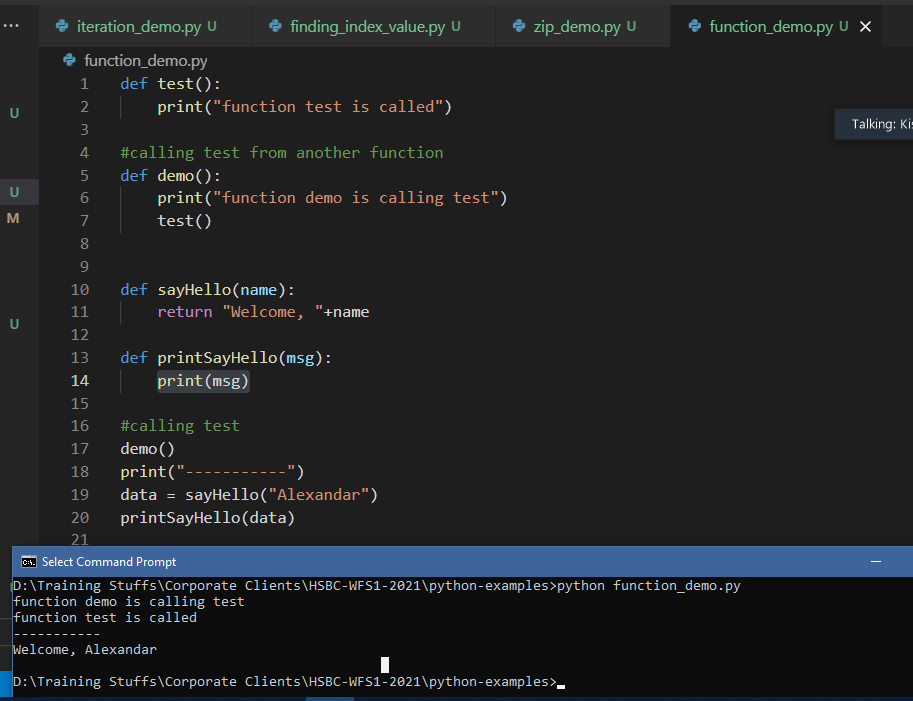
def function-name(arguments):

statements

<<return values>>

ex:

function sayHi(name):  
 return “Hello “+name  
   
  
function printData(i):  
 print(i)



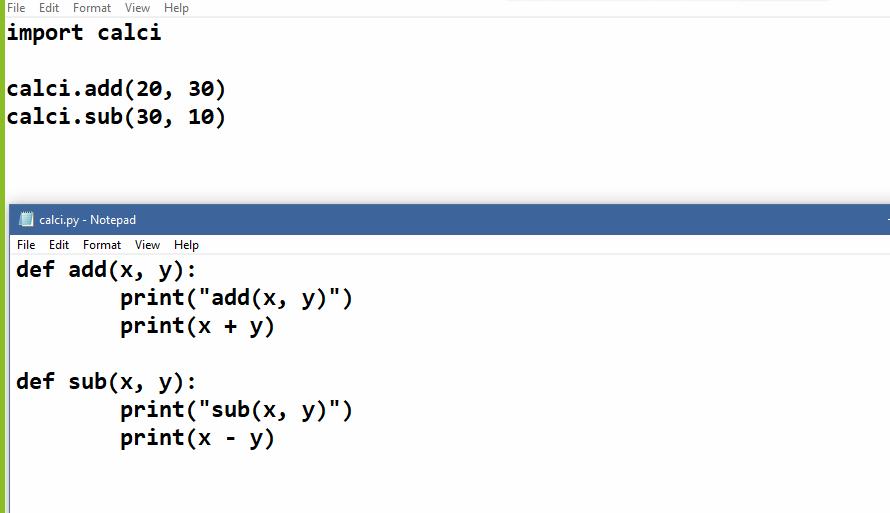
Exercise:

1: Add items 2: Delete item 3: Search item 4: Display all items 5: Exit

The above option should ask user to enter any 1 option and based on that you can call the function to add data to list, delete data from list, search data from list, display all data in the list and exit

The options repeats until you enter 5

Modules  
Collection of python programs which you can import in other python programs



Modules are just collection functions in a python file that can be imported in other python program

You can also use alias name while importing

import calci as c

c.add(30, 20)

Exception Handling

We handle exceptions inorder to avoid abnormal termination of the program, exceptions occurs at runtime, Unlink in Java we don’t have 2 types of exceptions in python, we have only runtime exceptions. Different exceptions we get on different scenarios

1. ZeroDivisionError: When you divide a number by 0
2. NameError: When you use the variables which are not defined
3. OSError: When you try to perform a file handling operation
4. ValueError: When you try to assign incompatible values, ex: you are trying to assign input at runtime and converting to a specific type but the content is incompatible
5. TypeError: When you try to concatenate the string with other types
6. Exception: This can represent all types of exception

In python we have totally 4 keywords in exception handling

1. try: block of code that can cause error
2. except: similar to catch block in Java, C#, and other languages
3. raise: similar to throw keyword
4. finally: executed regardless of whether or not exception handled

Note: In python exceptions are propagated automatically to the caller.

In Python you write the codes that generate exception in try block and you can have one or more except block to handle the exception

Ex1:

try:  
 stmts  
except:  
 stmts

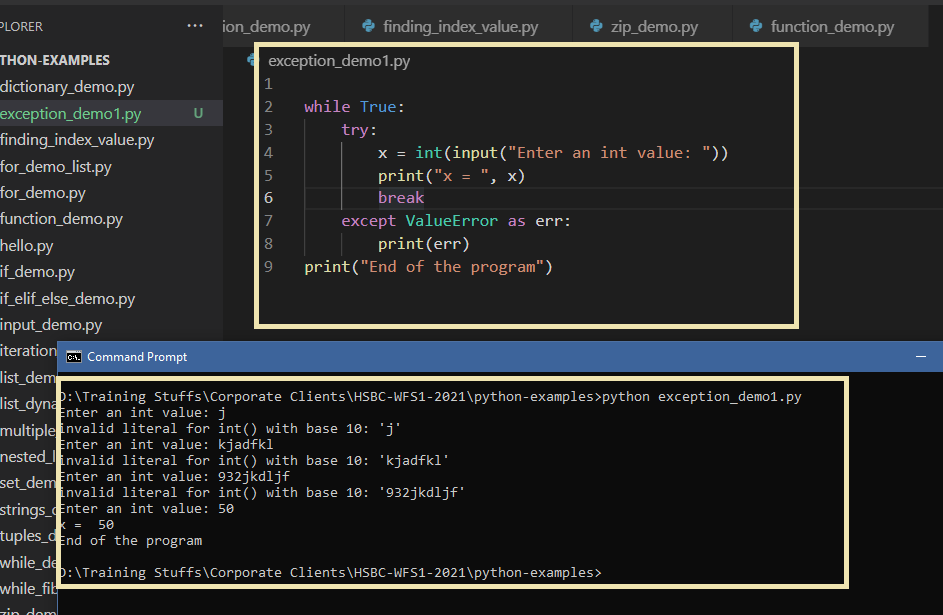
Ex2:  
try:  
 stmts  
except ValueError as e:  
 stmts  
except ZeroDivisionError as e:  
 stmts

You can write finally block at the end

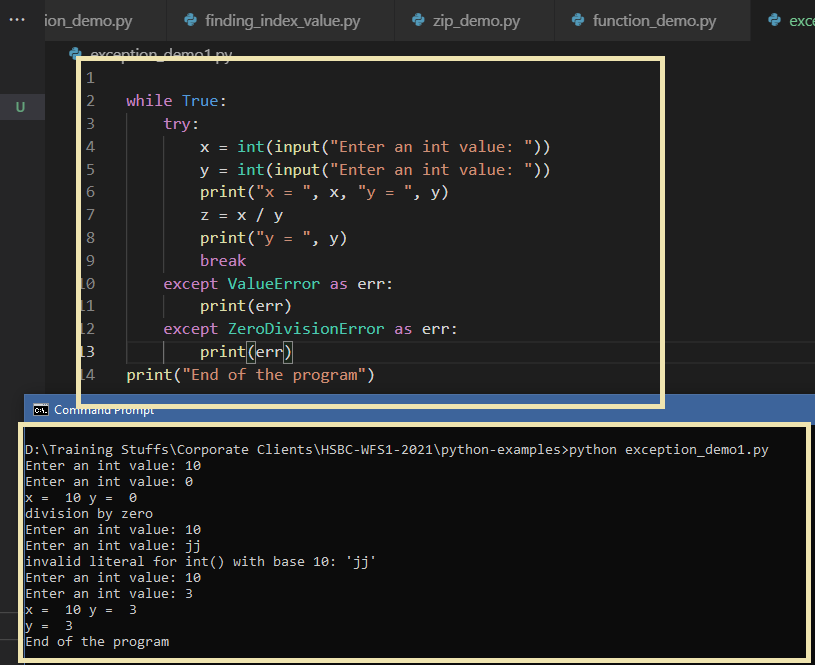
try:  
 ….  
except:  
….

finally:  
…..

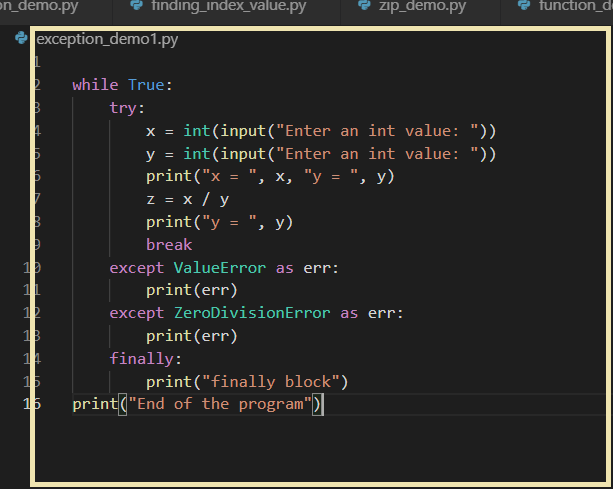
Simple try & except statements to continuously loop until you enter an int value



Multi-except statement



Using finally block



Exercise: For the same above program, instead of while True, try to create a function and call that function for invalid inputs

Other errors like OSError, TypeError all are some of the inbuilt exceptions.

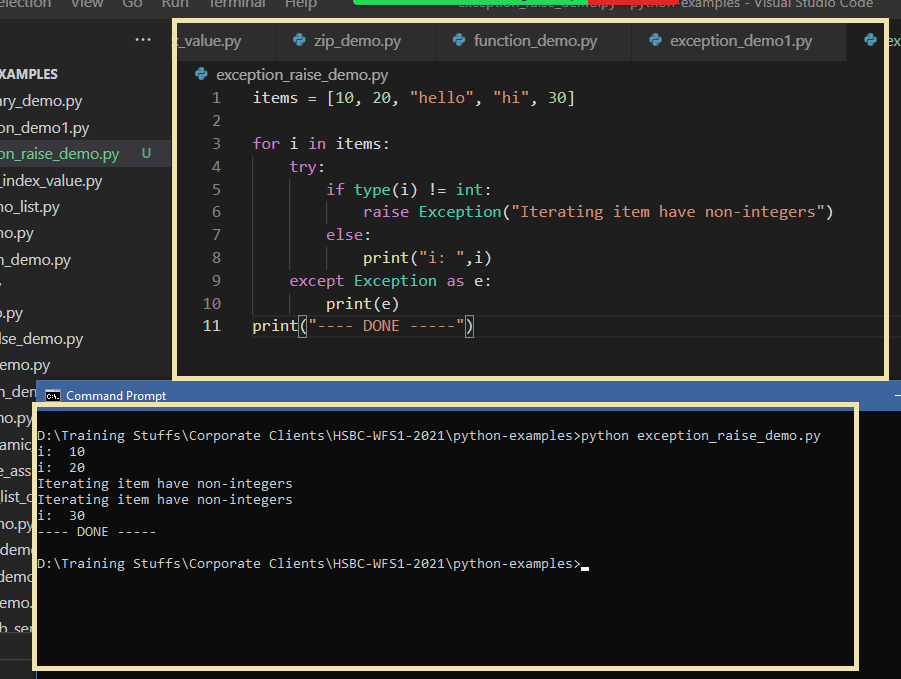
OSError occurs when you are trying to perform some I/O operations

try:  
 f = open(“file-names”)  
except OSError as e:  
 #If in case file is not found you will get OSError which can be handled using except OSError

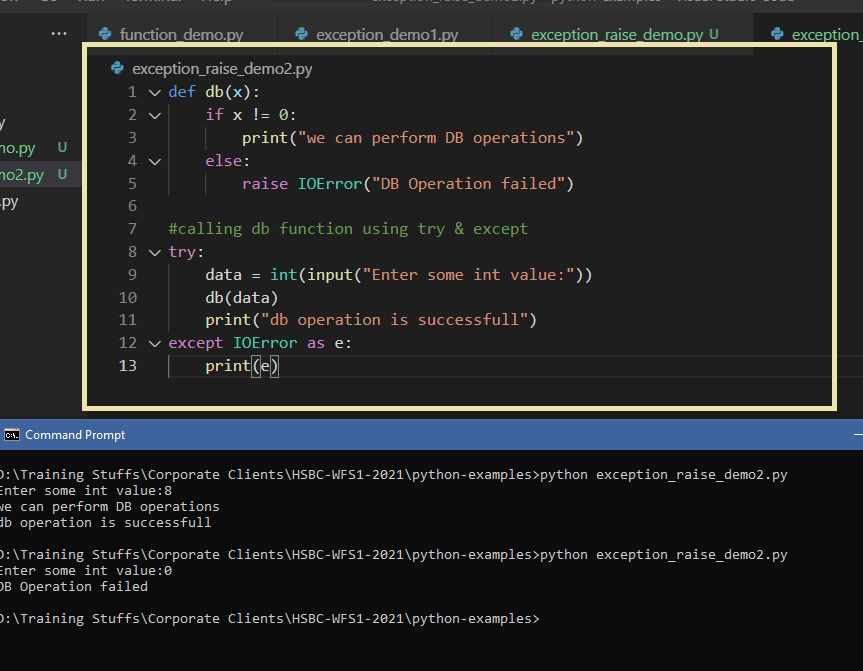
raise exception:

It is used to manually generate the exception when certain condition is met, it is similar to throw keyword in Java

raise exception\_name(‘…..’)



raise Exception will automatically propagates the exception to the caller



How to handle files i.e., reading the files & writing to the files

In Python performing operations on file is very easy

#you need to open the file with some mode like r, w, a and so on

r: read mode

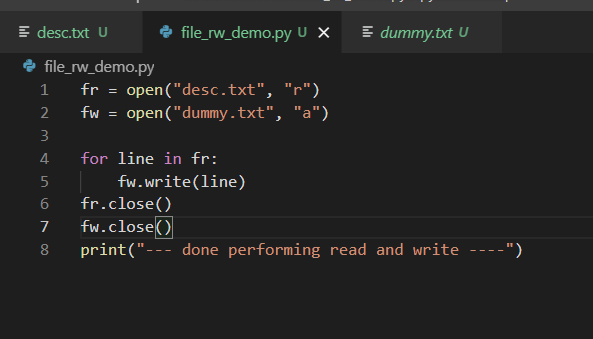
w: write mode / overwrite mode

a: append mode

f = open(“file.ext”, “r”)  
f = open(“file.ext”, “w”)

To perform operations on the file we need to call some functions like readline(), write(), read(), seek()

File Read & Write



seek() is to point to the particular character and read() is to read certain range of characters after the seek()

