```
class Customer:
    def __init__(self,cid,name,opamount):
        self.cid=cid
        self.name=name
        self.balance=opamount
    def deposit(self,amount):
        self.balance+=amount
    def withdraw(self,amount):
        self.balance-=amount
    def balance(self):
        return self.balance

c=Customer(1234,"Amit Kumar",5000)
c.deposit(3000)
c.withdraw(1200)
b=c.balance
```

print("Balance is",b)

Exception Handling

An exception is a runtime error that we can trap using try-except block

Use **Exception** class to get the error

exceptionhandling.py - C:/Users/admin/Desktop/Python/exceptionhandling.py (3.7.0)

File Edit Format Run Options Window Help

```
try:
    n=int(input("Enter a number "))
    print("number is",n)
except Exception as e:
    print("OOPs! an error has occured")
    print("Error reported : ",e)
```

Ln: 6 Col: 32

Working with strings

- Contains alphanumeric data
- Use len() function to get length of the string
- Use slicing to extract part of the string
- First character starts from 0 index
- We can also traverse using for loop
- We can also apply all basic function of the strings
  - upper()
  - lower()
  - title()

```
>>> s="Mohan Prasad, ABES Engineering College"

>>> len(s)

38

>>> s[0]

'M'

>>> s[0:4]

'Moha'
```

```
>>> s[-1]
'e'
>>> for ch in s:
...    print(ch)
```

Assignment

WAP to input an alphanumeric string from user and show sum of all the digits present in the string

Using count() function

Returns total number of occurances of given string in some string

```
>>> s
'NH-24, Ghaziabad 201009'
>>> s.count('a')
3
```

Using find() function

Used to find as string in another string

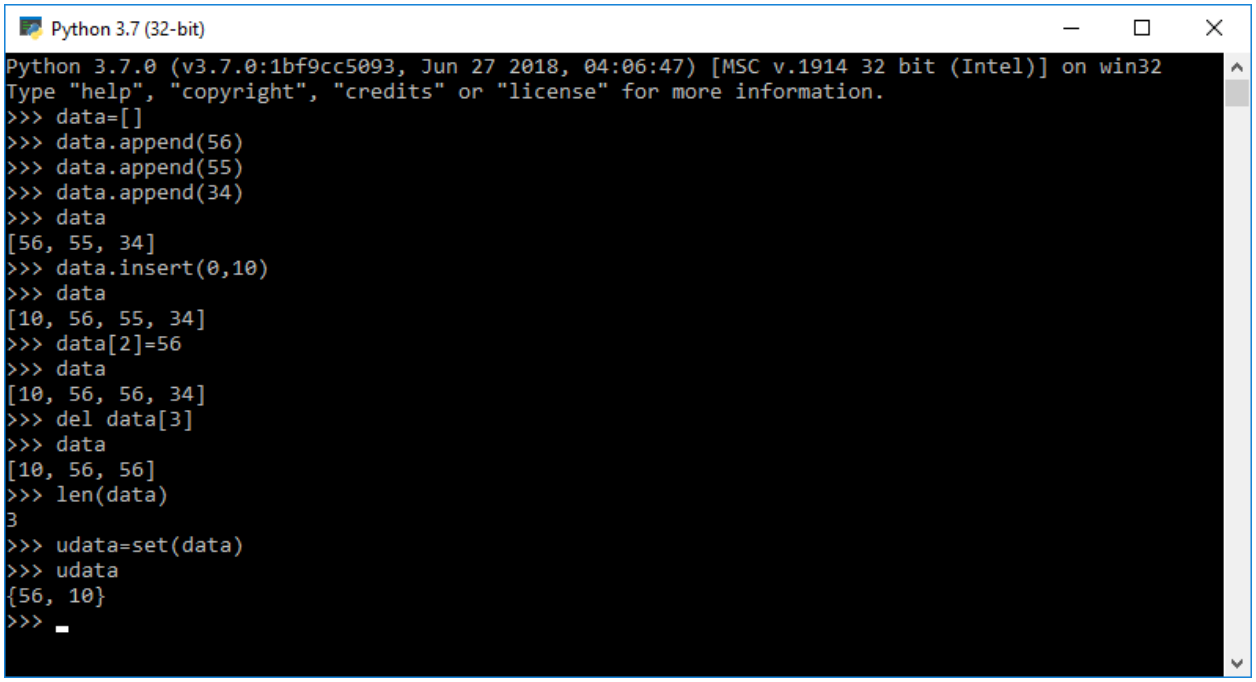
```
>>> s
'NH-24, Ghaziabad 201009'
>>> s.find('a')
9
>>> s.find('a',10)
12
```

Working with data using collections

1. tuple – for fixed data… use ()
2. list – for dynamic single set of data … use []
3. dictionary – for dynamic key/value pairs … use {}
4. set – for unique data

Using list

1. append(data)
    a. To add the data from bottom line
2. insert(location, data)
    a. To add given data on given location
3. count()
    a. To get count of given item
4. index()
    a. To return the index of searched item
5. sort()
    a. To automatically sort data in ascending order
6. Use **del** operator to delete specific elements or while list
7. Use len() function to get count of data in some list



Assignment

WAP to input as many numbers as user wants until -999 get pressed.

Show all the numbers along with sum of given numbers

dynamiclist.py - C:/Users/admin/Desktop/Python/dynamiclist.py (3.7.0)

File Edit Format Run Options Window Help

```
data=[]
print('Enter few numbers... -999 for quit ')
while True:
    n=int(input())
    if n==-999: break
    data.append(n)

s=0
for d in data:
    print(d)
    s+=d
print('sum is',s)
```

Ln: 15 Col: 0

Working with dictionary

- Use {} to make the dictionary
- Use **key:value** format to provide direct values
- Use **variable[key]=value** for dyanamic data
- Use keys() function to get the keys only
- Use values() function get values only
- Use items() function to get key/value pairs

```
Python 3.7 (32-bit)
>>> marks={'Rohit':56,'Nitin':99,'Kapil':87}
>>> marks
{'Rohit': 56, 'Nitin': 99, 'Kapil': 87}
>>> for k in marks.keys():
...     print(k)
...
Rohit
Nitin
Kapil
>>> for v in marks.values():
...     print(v)
...
56
99
87
>>> for k,v in marks.items():
...     print(k,v)
...
Rohit 56
Nitin 99
Kapil 87
>>> marks['Riya']=99
>>> for k,v in marks.items():
...     print(k,v)
...
Rohit 56
Nitin 99
Kapil 87
Riya 99
>>>
```

Working with modules

A module is python file having a set of classes and functions. Modules can be user defined or system defined.

Example of system defined module

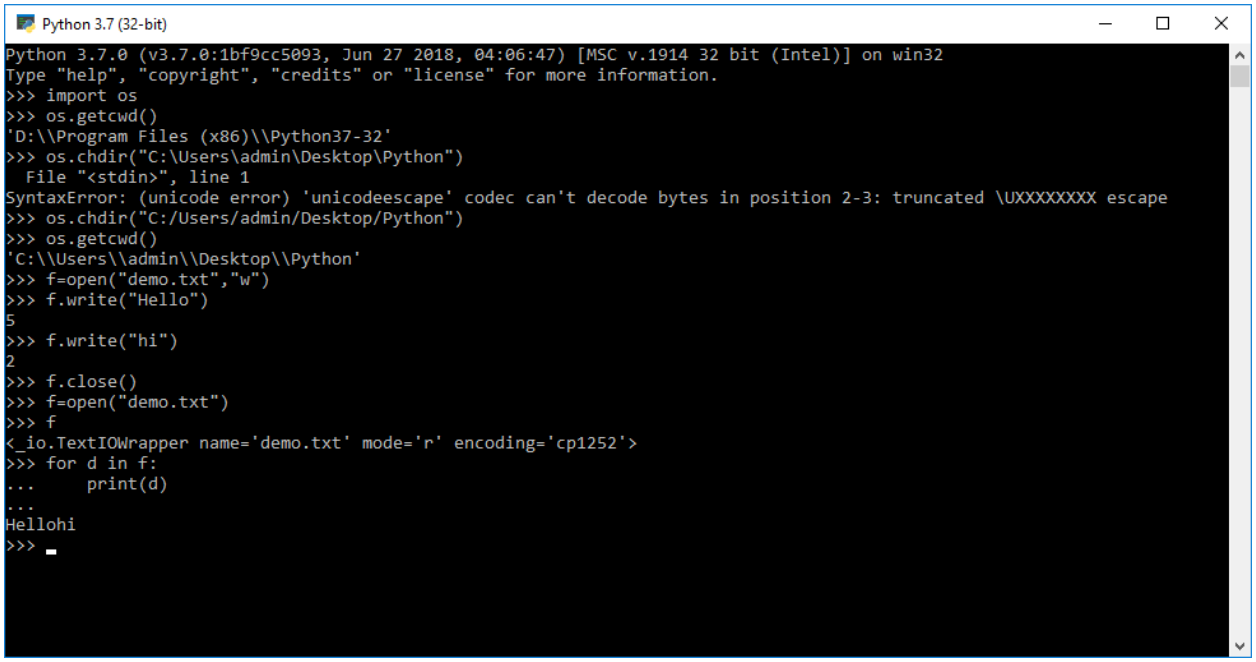
os

math

json

Use **import** keyword to import a module

import os



What are packages?

Special files placed on some server having the modules. We need to fetch those packages from server to local machine using PIP command.

pip install pymsql

pip install pymongo

pip install pandas

pip install numpy

pip install matplotlib