

Chapter -04 Using Python Libraries

Practical Session -I:

Program 1:

Create a module **lengthconversion.py** that stores functions for various lengths conversion e.g.,

- `miletokm()` to convert miles to kilometre
- `kmtomile()` to convert kilometres to miles
- `feettoinches()`
- `inchestofoot()`

It should also store constant values such as value of (mile in kilometres and vice versa).

[1 mile =1.609344 kilometre; 1 foot =12 inches]

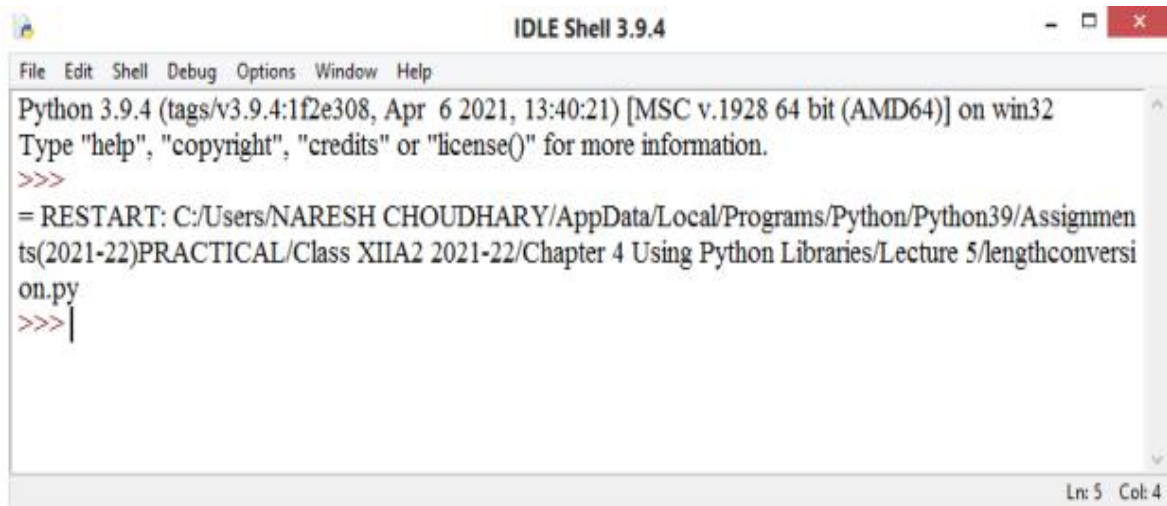
Help() function should display proper information

Code:

```
# lengthconversion.py
"""This module stores various conversion functions to convert
distances into different units"""
def miletokm(d):
    #Conversion function 1
    """ returns miles converted to kilometers"""
    return d * ONE_MILE
def kmtomile(d):
    # Conversion function 2
    """ returns kilometers converted to miles"""
    return d / ONE_MILE
def feettoinches(a):
    # Conversion function 3
    """ returns feet converted to inches"""
    return a * ONE_FEET
def inchestofoot(a):
    # Conversion function 4
    """ returns inches converted to inches"""
    return a / ONE_FEET

#Constants
ONE_MILE =1.609344          # =1.609
ONE_FEET=12                 # =12 inches
```

Now run the module as:



```
IDLE Shell 3.9.4
File Edit Shell Debug Options Window Help
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/NARESH CHOUDHARY/AppData/Local/Programs/Python/Python39/Assignments(2021-22)PRACTICAL/Class XIIA2 2021-22/Chapter 4 Using Python Libraries/Lecture 5/lengthconversion.py
>>>|
```

Now Type help() function/command on the interactive mode:



```
IDLE Shell 3.9.4
File Edit Shell Debug Options Window Help
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/NARESH CHOUDHARY/AppData/Local/Programs/Python/Python39/Assignments(2021-22)PRACTICAL/Class XIIA2 2021-22/Chapter 4 Using Python Libraries/Lecture 5/lengthconversion.py
>>> import lengthconversion
>>> help(lengthconversion)
Help on module lengthconversion:

NAME
    lengthconversion

DESCRIPTION
    This module stores various conversion functions to convert
    distances into different units

FUNCTIONS
    feett inches(a)
        returns inches converted to inches

    kmtomile(d)
        returns kilometers converted to miles

    miletokm(d)
        returns miles converted to kilometers

DATA
    ONE_FEET = 12
    ONE_MILE = 1.609344

FILE
    c:\users\naresh choudhary\appdata\local\programs\python\python39\assignments(2021-22)practical\class xia2 2021-22\chapter 4 using python libraries\lecture 5\lengthconversion.py

>>>|
```

Program 2:

Create a **module massconversion.py** that stores function for *mass conversion* e.g.,

- *kgtotonne()* to convert kg to tonnes
 - *tonnetokg()* to convert tonne to kg
 - *kgto pounds()* to convert kg to pound
 - *poundtokg()* to convert pound to kg
- (Also store constants 1 kg =0.001 tonne, 1 kg= 2.20462 pound)

help() function should give proper information about the module

Code:

```
# massconversion.py
"""This module stores various conversion functions to convert
masses into different units"""

def kgtotonne(m):
    #Conversion function 1
    """ returns kilograms converted to tonnes"""
    return m * ONE_KG_TONNE

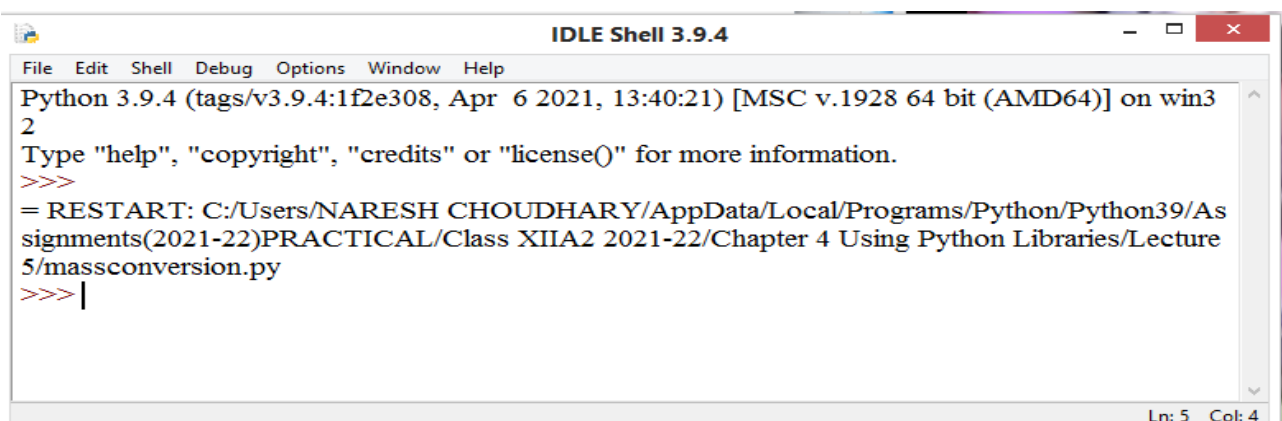
def tonnetokg(m):
    # Conversion function 2
    """ returns tonnes converted to kilograms"""
    return m / ONE_KG_TONNE

def kgtopound(a):
    # Conversion function 3
    """ returns kilograms converted to pounds"""
    return a * ONE_KG_POUND

def poundtokg(a):
    # Conversion function 4
    """ returns pounds converted to kilograms"""
    return a / ONE_KG_POUND

#Constants
ONE_KG_TONNE =0.001      #=0.001 TONNE
ONE_KG_POUND =2.20462    # =2.20462
```

Now run the module massconversion.py as

A screenshot of the IDLE Shell 3.9.4 window. The title bar reads 'IDLE Shell 3.9.4'. The menu bar includes 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The shell area shows the following text: 'Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32', 'Type "help", "copyright", "credits" or "license()" for more information.', and a prompt '>>>' followed by a new line. Below this, it says '= RESTART: C:/Users/NARESH CHOUDHARY/AppData/Local/Programs/Python/Python39/As signments(2021-22)PRACTICAL/Class XIIA2 2021-22/Chapter 4 Using Python Libraries/Lecture 5/massconversion.py' followed by a prompt '>>>' and a cursor. The status bar at the bottom right shows 'Ln: 5 Col: 4'.

Now type help () function/command on the interactive mode:



```
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/NARESH CHOUDHARY/AppData/Local/Programs/Python/Python39/Assignments(20
21-22)PRACTICAL/Class XIIA2 2021-22/Chapter 4 Using Python Libraries/Lecture 5/massconversion.py
>>> import massconversion
>>> help(massconversion)
Help on module massconversion:

NAME
    massconversion

DESCRIPTION
    This module stores various conversion functions to convert
    masses into different units

FUNCTIONS
    kgtopound(a)
        returns kilograms converted to pounds

    kgtotonne(m)
        returns kilograms converted to tonnes

    poundtokg(a)
        returns pounds converted to kilograms

    tonnetokg(m)
        returns tonnes converted to kilograms

DATA
    ONE_KG_POUND = 2.20462
    ONE_KG_TONNE = 0.001

FILE
    c:\users\naresh choudhary\appdata\local\programs\python\python39\assignments(2021-22)practical\class xia
    2 2021-22\chapter 4 using python libraries\lecture 5\massconversion.py

>>> |
```

Ln: 37 Col: 4

Program 3:

Create a **module tempcon.py** that stores function for *temperature conversion* e.g.,

➤ `to_centigrade()` to convert Fahrenheit to Centigrade

➤ `to_fahrenheit()` to convert Centigrade to Fahrenheit

(Also store two constants `FREEZING_C = 0.0` `FREEZING_F = 32.0`)

help() function should give proper information about the module

Code:

```
#tempcon.py
```

```
"""Conversion functions between fahrenheit and centigrade"""
```

```
def to_centigrade(x):
```

```
    """Returns: x converted to centigrade """
```

```
    return 5 * (x-32)/9.0
```

```
def to_fahrenheit(x):
```

```
    """Returns: x converted to fahrenheit"""
```

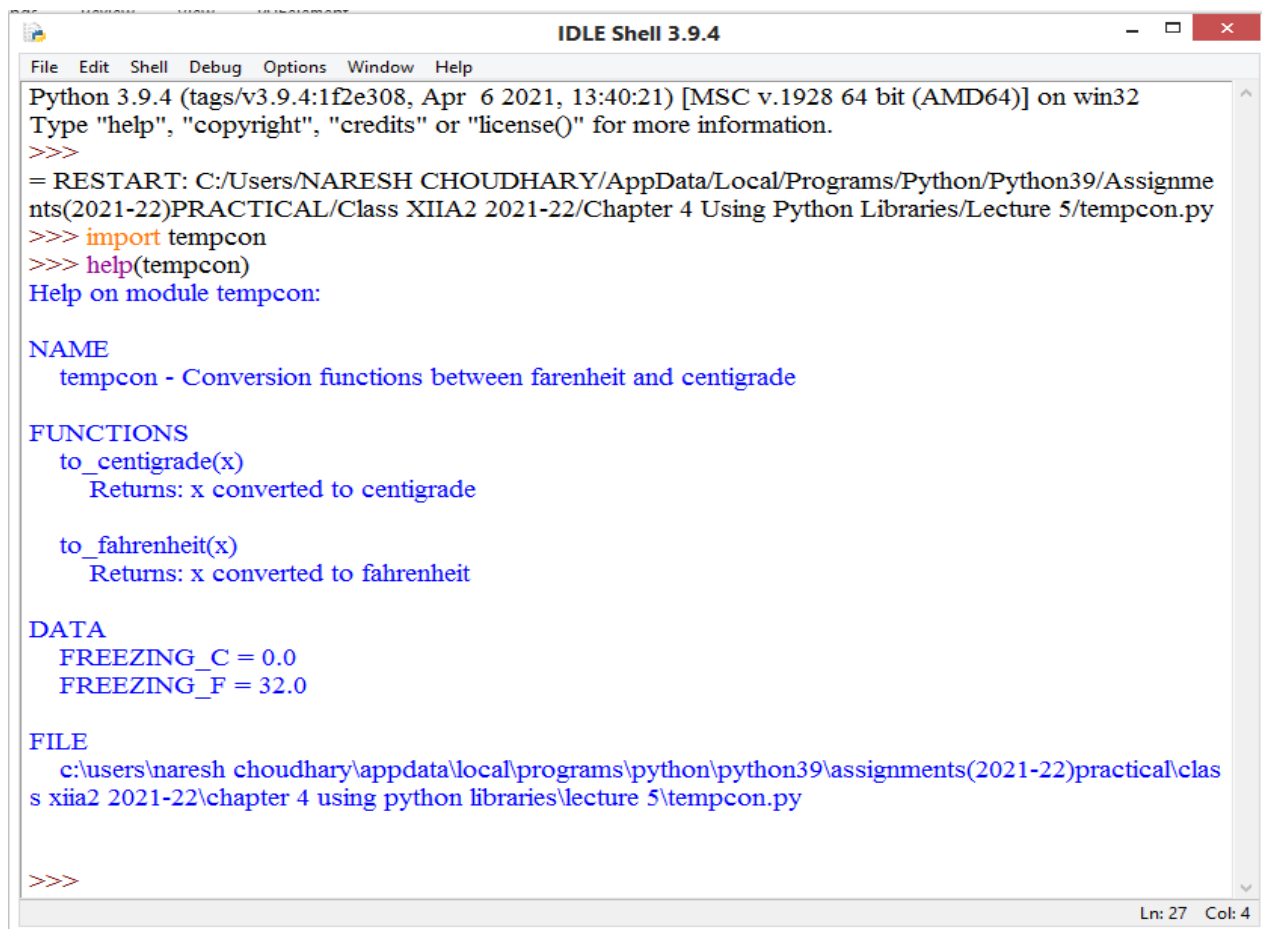
```
    return 9 * x /5.0 +32
```

```
#Constants
```

```
FREEZING_C = 0.0 #Water freezing temp. (in celcius)
```

```
FREEZING_F= 32.0 #Water freezing temp. (in fahrenheit)
```

Now run the module tempcon.py and type help() command as



```
IDLE Shell 3.9.4
File Edit Shell Debug Options Window Help
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/NARESH CHOUDHARY/AppData/Local/Programs/Python/Python39/Assignme
nts(2021-22)PRACTICAL/Class XIIA2 2021-22/Chapter 4 Using Python Libraries/Lecture 5/tempcon.py
>>> import tempcon
>>> help(tempcon)
Help on module tempcon:

NAME
tempcon - Conversion functions between fahrenheit and centigrade

FUNCTIONS
to_centigrade(x)
    Returns: x converted to centigrade

to_fahrenheit(x)
    Returns: x converted to fahrenheit

DATA
FREEZING_C = 0.0
FREEZING_F = 32.0

FILE
c:\users\naresh choudhary\appdata\local\programs\python\python39\assignments(2021-22)practical\clas
s xia2 2021-22\chapter 4 using python libraries\lecture 5\tempcon.py

>>>
```

Program 4:

Create a **package conversion** which include the above two (**lengthconversion.py**, **massconversion.py** and **temcon.py**) modules as like structure given below:

```
Conversion
/
/____Temperature
/      /____tempconversion.py
/      /______init__.py
/
/____Length
/      /____lengthconversion.py
/      /______init__.py
/
/____Mass
/      /____masscon.py
/      /______init__.py
```

Make sure that above package meets the requirements of being a **Python package**. Also, program should be able to import above package and and/or its modules using **import command**.

Solution

Structure of *Conversion package* will be displayed as:



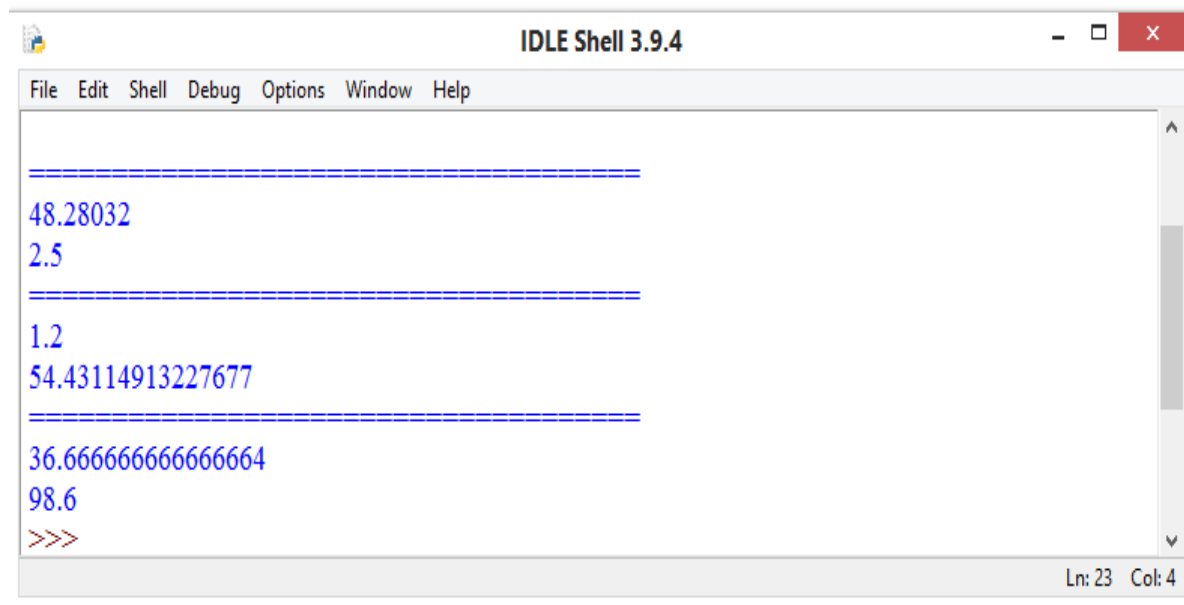
Code:-

```
# Call_conversion.py
import Conversion.Length.lengthconversion
import Conversion.Mass.massconversion
import Conversion.Temperature.temcon
print("\n=====")
print(Conversion.Length.lengthconversion.miletokm(30))
print(Conversion.Length.lengthconversion.feettoinches(30))
print("=====")

print(Conversion.Mass.massconversion.kgtotonne(1200))
print(Conversion.Mass.massconversion.poundtokg(120))
print("=====")

print(Conversion.Temperature.temcon.to_centigrade(98))
print(Conversion.Temperature.temcon.to_fahrenheit(37))
```

Output:-



The screenshot shows a window titled "IDLE Shell 3.9.4" with a menu bar (File, Edit, Shell, Debug, Options, Window, Help) and a text area. The text area contains the following output:

```
=====
48.28032
2.5
=====
1.2
54.43114913227677
=====
36.666666666666664
98.6
>>>
```

The status bar at the bottom right indicates "Ln: 23 Col: 4".