Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

Aim: Write a python program to define a module and import a specific function in that module to another program

IDE:

Python Modules

As our program grows bigger, it may contain many lines of code. Instead of putting everything in a single file, we can use modules to separate codes in separate files as per their functionality. This makes our code organized and easier to maintain.

Module is a file that contains code to perform a specific task. A module may contain variables, functions, classes etc. Let's see an example,

Let us create a module. Type the following and save it as

example.py

```
def add(a,b):
    result = a+b
    return result
```

```
import example as addition
a = addition.add(4,5)
print(a)
```

Output

Import Python Standard Library Modules

The Python standard library contains well over 200 modules. We can import a module according to our needs. Suppose we want to get the value of pi, first we import the math module and use math.pi. For example,

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

#import standard math module

import math

use math.pi to get value of pi

print("The value of pi is", math.pi)

Output

```
In [3]:
   ...: import math
   ...: print("The value of pi is", math.pi)
The value of pi is 3.141592653589793
```

Python import with Renaming

In Python, we can also import a module by renaming it. For example,

import module by renaming it

import math as m

print(m.pi)

Output

```
In [3]:
   ...: import math
   ...: print("The value of pi is", math.pi)
The value of pi is 3.141592653589793
```

Python from...import statement

We can import specific names from a module without importing the module as a whole. For example,

import only pi from math module

from math import pi

print(pi)

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

Output

```
In [4]:
   ...: from math import pi
   ...: print(pi)
3.141592653589793
```

Import all names

In Python, we can import all names(definitions) from a module using the following construct:

import all names from the standard module math

from math import *

print("The value of pi is", pi)

Output

```
In [5]: from math import *
   ...: print("The value of pi is", pi)
The value of pi is 3.141592653589793
```

The dir() built-in function

In Python, we can use the dir() function to list all the function names in a module.

We can use dir in math module in the following way:

print(dir(math))

Output

```
In [6]:
   ...: print(dir(math))
['__doc__', '__loader__', '__name__', '__package__', '__spec__', 'acos', 'acosh', 'asin', 'asinh',
'atan', 'atan2', 'atanh', 'cbrt', 'ceil', 'comb', 'copysign', 'cos', 'cosh', 'degrees', 'dist', 'e',
'erf', 'erfc', 'exp', 'exp2', 'expm1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma',
'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan', 'isqrt', 'lcm', 'ldexp', 'lgamma',
'log', 'log10', 'log1p', 'log2', 'modf', 'nan', 'nextafter', 'perm', 'pi', 'pow', 'prod', 'radians',
'remainder', 'sin', 'sinh', 'sqrt', 'sumprod', 'tan', 'tanh', 'tau', 'trunc', 'ulp']
```

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

Built-in modules

Some examples of Python built-in modules include “os”, “sys”, “math”, and “datetime”.

help('modules')

Output:

```
7f0197f6d050da244d93__mypyc codecs                matplotlib          snappy
IPython             codeop              matplotlib_inline   sndhdr
OpenSSL             collections         mccabe              sniffio
PIL                 colorama            mdit_py_plugins     snowballstemmer
PyQt5               colorcet            mdurl               socket
__future__          colorsys            menuinst            socketserver
__hello__           comm                mimetypes           socks
__phello__          commctrl            mistune             sockshandler
_abc                compileall          mkl                 sortedcontainers
_aix_support        concurrent          mkl_fft             soupsieve
_argon2_cffi_bindings conda               mmap                sphinx
_ast                conda_build         mmapfile            sphinxify
_asyncio            conda_content_trust mmsystem            sphinxthread
_bisect             conda_env           modulefinder        spyder
_black_version      conda_index         more_itertools      spyder_kernels
_blake2             conda_libmamba_solver mpmath              sqlalchemy
_brotli             conda_pack          msgpack             sqlite3
_bz2                conda_package_handling msilib               sre_compile
_cffi_backend       conda_package_streaming msvcrt              sre_constants
_codecs             conda_token         multidict           sre_parse
_codecs_cn          conf                multipledispatch    ssl
_codecs_hk          configparser        multiprocessing     sspi
_codecs_iso2022     constantly          mypy                sspicon
_codecs_jp          contextlib          mypy_extensions     stack_data
_codecs_kr          contextvars         mypyc               stat
_codecs_tw          contourpy           nacl                statistics
_collections        cookiecutter        navigator_updater   statsmodels
_collections_abc    copy                nbclient            streamlit
_compat_pickle      copyreg             nbconvert           string
_compression        cpuinfo             nbformat            stringprep
_contextvars        crypt               nest_asyncio        struct
_csv                cryptography        netbios             subprocess
_ctypes             cssselect           netrc               sunau
_ctypes_test        csv                 networkx            sympy
_datetime           ctypes              nltk                symtable
_decimal            curl                nntplib             sys
_distutils_hack     curses              notebook            sysconfig
_elementtree        cwp                 notebook_shim       tables
_functools          cycler              nt                  tabnanny
_hashlib            cytoolz             ntpath              tabulate
_heapq              dask                ntsecuritycon       tarfile
_imp                dask_expr           nturl2path          tblib
_io                 dataclasses         numba               telnetlib
```

Marwadi University **Faculty of Engineering & Technology** **Department of Information and Communication Technology**		
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

```
_json                   datashader              numbergen               tempfile
_locale                 datetime                numbers                 tenacity
_lsprof                 dateutil                numexpr                 terminado
_lzma                   dbi                     numpy                   test
_markupbase             dbm                     numpydoc                test_pycosat
_md5                    dde                     odbc                    text_unidecode
_msi                    debugpy                 opcode                  textdistance
_multibytecodec         decimal                 openpyxl                textwrap
_multiprocessing        decorator               operator                this
_nsis                   defusedxml              optparse                threading
_opcode                 diff_match_patch        os                      threadpoolctl
_operator               difflib                 overrides               three_merge
_osx_support            dill                    packaging               tifffile
_overlapped             dis                     pandas                  time
_pickle                 distributed             pandocfilters           timeit
_plotly_future_         distro                  panel                   timer
_plotly_utils           docstring_to_markdown   param                     tinycss2
_py_abc                 doctest                 paramiko                tkinter
_pydatetime             docutils                parsel                  tldextract
_pydecimal              dotenv                  parso                   tlz
_pyio                   email                   partd                   token
_pylong                 encodings               pathlib                 tokenize
_pytest                 ensurepip               pathspec                toml
_queue                  entrypoints             patsy                   tomli
_random                 enum                    pdb                     tomlkit
_sha1                   erfa                    perfmon                 tomllib
_sha2                   errno                   pexpect                 toolz
_sha3                   et_xmlfile              pickle                  tornado
_signal                 executing               pickleshare             tqdm
_sitebuiltins           fastjsonschema          pickletools             trace
_socket                 faulthandler            pip                     traceback
_sqlite3                filecmp                 pipes                   tracemalloc
_sre                    fileinput               pkce                    traitlets
_ssl                    filelock                pkg_resources           truststore
_stat                   flake8                  pkginfo                 tty
_statistics             flask                   pkgutil                 turtle
_string                 fnmatch                 platform                turtledemo
_strptime               fontTools               platformdirs            twisted
_struct                 fractions               plistlib                types
_symtable               frozendict              plotly                  typing
_system_path            frozenlist              pluggy                  typing_extensions
_testbuffer             fsspec                  ply                     tzdata
_testcapi               ftplib                  poplib                  uc_micro
_testclinic             functools               posixpath               ujson
```

```
_testconsole            gc                      pprint                  unicodedata
_testimportmultiple     genericpath             profile                 unicodedata2
_testinternalcapi       gensim                  prometheus_client       unidecode
_testmultiphase         getopt                  prompt_toolkit          unittest
_testsinglephase        getpass                 protego                 urllib
_thread                 gettext                 pstats                  urllib3
_threading_local        git                     psutil                  uu
_tkinter                gitdb                   pty                     uuid
_tokenize               glob                    ptyprocess              venv
_tracemalloc            graphlib                pure_eval               w3lib
_typing                 greenlet                py                      warnings
_uuid                   gzip                    py_compile              watchdog
_warnings               h11                     pyarrow                 wave
_weakref                h5py                    pyasn1                  wcwidth
_weakrefset             hashlib                 pyasn1_modules          weakref
_win32sysloader         heapdict                pyclbr                  webbrowser
_winapi                 heapq                   pycodestyle             webencodings
_winxptheme             hmac                    pycosat                 websocket
_wmi                    holoviews               pycparser               werkzeug
_xxinterpchannels       html                    pyct                    whatthepatch
_xxsubinterpreters      http                    pycurl                  wheel
_yaml                   httpcore                pydantic                widgetsnbextension
_zoneinfo               httpx                   pydantic_core           win2kras
abc                     hvplot                  pydeck                  win32api
adodbapi                hyperlink               pydispatch              win32clipboard
aext_assistant          idlelib                 pydoc                   win32com
aext_assistant_server   idna                      pydoc_data              win32con
aext_core               imagecodecs             pydocstyle              win32console
aext_core_server        imageio                 pyexpat                 win32cred
aext_panels             imagesize               pyflakes                win32crypt
aext_panels_server      imaplib                 pygments                win32cryptcon
aext_share_notebook     imblearn                pylab                   win32ctypes
aext_share_notebook_server imghdr                  pylint                     win32event
aext_shared             importlib               pylint_venv             win32evtlog
afxres                  importlib_metadata      pyls_spyder             win32evtlogutil
aifc                    incremental             pylsp                   win32file
aiobotocore             inflection              pylsp_black             win32gui
aiohttp                 iniconfig               pylsp_jsonrpc           win32gui_struct
aioitertools            inspect                 pyodbc                  win32help
aiosignal               intake                  pyparsing               win32inet
alabaster               intervaltree            pytest                  win32inetcon
altair                  io                      pythoncom               win32job
anaconda_anon_usage     ipaddress               pythonjsonlogger        win32lz
anaconda_catalogs       ipykernel               pytoolconfig            win32net
```

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

```
anaconda_cloud_auth  ipykernel_launcher  pytz                 win32netcon
anaconda_navigator   ipython_genutils    pyviz_comms          win32pdh
anaconda_project     ipywidgets          pywin                win32pdhquery
annotated_types      isapi               pywin32_bootstrap    win32pdhutil
antigravity          isort               pywin32_testutil     win32pipe
anyio                isympy              pywintypes           win32print
appdirs              itemadapter         pywt                 win32process
archspec             itemloaders         qdarkstyle           win32profile
argon2               itertools           qstylizer            win32ras
argparse             itsdangerous        qtawesome            win32rcparser
array                jedi                qtconsole            win32security
arrow                jellyfish           qtpy                 win32service
ast                  jinja2              queue                win32serviceutil
astroid              jmespath            queuelib             win32timezone
astropy              joblib              quopri               win32trace
astropy_iers_data    json                random               win32traceutil
asttokens            json5               rasutil              win32transaction
async_lru            jsonpatch           re                   win32ts
asyncio              jsonpointer         referencing          win32ui
atexit               jsonschema          regcheck             win32uiole
atomicwrites         jsonschema_specifications regex              win32verstamp
attr                 jupyter             regutil              win32wnet
attrs                jupyter_client      repo_cli             win_inet_pton
audioop              jupyter_console     reprlib              winerror
automat              jupyter_core        requests             winioctlcon
autopep8             jupyter_events      requests_file        winnt
babel                jupyter_lsp         requests_toolbelt    winperf
base64               jupyter_server      rfc3339_validator    winpty
bcrypt               jupyter_server_terminals rfc3986_validator   winreg
bdb                  jupyterlab          rich                 winsound
binaryornot          jupyterlab_plotly   rlcompleter          winxpgui
binascii             jupyterlab_pygments rope                 winxptheme
binstar_client       jupyterlab_server   rpds                 wrapt
bisect               jupyterlab_widgets  rtree                wsgiref
black                jwt                 ruamel_yaml          xarray
blackd               keyring             runpy                xdrlib
bleach               keyword             s3fs                 xlwings
blib2to3             kiwisolver          sched                xlwingsjs
blinker              lazy_loader         scipy                xml
bokeh                lazy_object_proxy   scrapy               xmlrpc
boltons              lckr_jupyterlab_variableinspector seaborn              xxlimited
botocore             lib2to3             secrets              xxlimited_35
bottleneck           libarchive          select               xxsubtype
```

```
bottleneck           libarchive          select               xxsubtype
brotli               libmambapy          selectors            xyzservices
bs4                  lief                semver               yaml
builtins             linecache           send2trash           yapf
bz2                  linkify_it          service_identity     yapf_third_party
cProfile             llvmlite            servicemanager       yapftests
cachetools           lmdb                setuptools           yarl
calendar             locale              shelve               zict
certifi              locket              shlex                zipapp
cffi                 logging             shutil               zipfile
cgi                  lxml                signal               zipimport
cgitb                lz4                 sipbuild             zipp
chardet              lzma                site                 zlib
charset_normalizer   mailbox             six                  zmq
chunk                mailcap             skimage              zoneinfo
click                markdown            sklearn              zope
cloudpickle          markdown_it         slugify              zstandard
cmath                markupsafe          smart_open
cmd                  marshal             smmap
code                 math                smtplib

Enter any module name to get more help.  Or, type "modules spam" to search
for modules whose name or summary contain the string "spam".
```

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

Let's find the area of the circle : $a = \pi r^2$

Python Code

r = int(input("enter value of r"))

A = r*r*pi

Print(a)

Output:

```
In [15]:
    ...: r = int(input("enter  value of  r"))
    ...: a = r*r*pi
    ...: print(a)
enter  value of  r5
78.53981633974483
```

Print the values of positive and negative infinity.

import math

print (math.inf)

print (-math.inf)

Output

```
In [8]: import math
   ...: print (math.inf)
   ...: print (-math.inf)
inf
-inf
```

List of Mathematical function in Math Module

pow(x,y), sqrt(x), trunc(x), cos(x), sin(x), tan(x), degrees(x), radians(x), exp(x), log2(x), log10(x)

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

Post Lab Exercise:

a. Write a Python program to convert degree to radian

Code:

```
from math import pi
x=int(input("enter x"))
y=(x*pi)/180
print(y)
```

Output:

```
In [17]: from math import pi
    ...: x=int(input("enter x"))
    ...: y=(x*pi)/180
    ...: print(y)
enter x180
3.141592653589793
```

b. Make a simplest possible Python program that calculates and prints the value of the formula

$$y = 6x^2 + 4sin(x)$$

Code:

```
import math
x=int(input("enter value:"))
y=6*x**2+4 * math.sin(x)
print(y)
```

Output:

```
In [18]: import math
    ...: x=int(input("enter value:"))
    ...: y=6*x**2+4 * math.sin(x)
    ...: print(y)
enter value:17
1730.1544100324818
```

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**

c. Write a Python function that evaluates the mathematical functions $f(x) = cos(2x), f'(x) = -2\sin(2x)$, $and\ f''(x) = -4\cos(2x)$.

Return these three values. Write out the results of these values for $x = \pi$

Code:

```
from math import pi
import math
f=math.cos(2*pi)
f1=-2*math.sin(2*pi)
f2=-4*math.cos(2*pi)
print(f)
print(f1)
print(f2)
```

Output:

```
In [19]: from math import pi
    ...: import math
    ...: f=math.cos(2*pi)
    ...: f1=-2*math.sin(2*pi)
    ...: f2=-4*math.cos(2*pi)
    ...: print(f)
    ...: print(f1)
    ...: print(f2)
1.0
4.898587196589413e-16
-4.0
```

Github link: https://github.com/JenishDesai5115/PWP_postlabs

Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Engineering & Technology** **Department of Information and Communication Technology**	
Subject: Programming With Python (01CT1309)	**Aim:** Write a python program to define a module and import a specific function in that module to another program	
Experiment No: 08	**Date:**	**Enrollment No:92510133025**