Python Notes.

* Python Modules :-

Definition :- Modules refer to a file containing Python statements and definitions

- A module can define functions, classes and variables.
- A module can also include runnable code.

* A file containing Python code, for example : `example.py`, is called a module, and its module name would be `example`.

Example :- Let us create, a module, Type the following and save it as :- example.py.

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

→ We can define our most used functions in a module and import it, instead of copying their definitions into different programs.

add() ← function defined inside a module named example.

STUDYBYNOTES

* <u>Need of Module :-</u>

○ Python has the feature of module in which programmer can put definitions in a file and use them in a script.

○ The definitions written inside a module can be imported into the main module by using the import command.

* <u>Creating a Module</u> :- ○ from the interpreter.
  ○ from another script file.

A module is simply a Python file with a .py extension that can be imported inside another Python Program.

The name of the Python file becomes the module name.

Example :- lets create a simple module called JP.

[JP.Py]

```
clef Hello ():
    print ("jpwebdevelopers")

location = "malout"
```

The above example shows the creation of a simple named "jp" -> name of the Python file is JP.py

* **Import Module:-**

To use the above created module, create a new Python file in the same directory and import jp module using the import statement

example   <- keyword
```
        import jp
        jp. Hello()
```

```
        print (jp. location)
```

Output
```
        jpwebclevelopers
        Malout
```

-> We use the import keyword to do this.
-> To import our previously defined module jp.

```
import jp
```

* Importing Modules:-

We can import a module using the import statement followed by module name to be imported..

It can access the definitions inside it using the dot operator.

example:

← module name

```
import math
print(" value of pi is ", math.pi)
```

Output
value of pi is 3.14159265

* import with renaming:-

we can import a module by renaming it as follows:-

```
import math as m
print(" The value of pi is ", m.pi)
```

→ we have renamed the math module as m.

* Python [from] ... import Statement

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

example:- # import only pi from math module

from [math] import [pi]
Print(" The value of pi is ", pi)

* import all names:-

We can import all names (definitions) from a module.

Example:- from math import *
Print(" The value of pi is ", pi)