How importing works in python

Importing in the python is the process of loading code from Python module into the current script. This allows you use the functions and variables defined in the module in your current script, as well as any additional modules that the imported module may depend on.

To import a module in Python, you use the import statement followed by the name of the module. For example to import the math module, which contains a variety of mathematical functions, you would use the following statement.

Example:

```
import math
```

After importing math module.

Example2:

```
import math
square=math.sqrt(9)
print(square)
```

From keyword

You can also import specific functions or variable from a module using keyword. For example to import only the sqrt function from the math module you would write

```
from math import sqrt
sqr=sqrt(9)
print(sqr)
```

Importing everything

It’s possible to import all function and variables from a module using you wildcard. However this is generally not recommended as it can lead to confusion and make it harder to understand where specific function and variables are coming from.

```
from math import*
s=sqrt(9)*pi
print(s)
```

The as keyword

Example

```
from math import sqrt as sq
result=sq(9)
print(result)
```

The dir function

Finally, Python has a built in function called dir that you can use to view the names of all the functions and variables defined in module. This can be helpful and understanding the contains of a new module.

Example

```

import pandas
print(dir(pandas))
```

Source Code

```
# import math
# square=math.sqrt(9)
# print(square)
# from math import sqrt
# sqr=sqrt(9)
# print(sqr)
```

```
# from math import*
# s=sqrt(9)*pi
# print(s)
# from math import sqrt as sq
# result=sq(9)
# print(result)

# import pandas
# print(dir(pandas))
import munawar as mun
print(mun.Welcome)
print(mun)
```

Thank You