

## Modules and Packages in Python:

### Python Modules:

A module is a Python file containing Python statements and definitions. For example, a file `even_odd.py` is a module, and we call it `'even_odd'`. We put similar code together in one module. This helps us modularize our code, and make it much easier to deal with. And not only that, a module grants us reusability. With a module, we don't need to write the same code again for a new project that we take up. Same like the classes we can import the libraries too basically libraries are also a type of module which can be used in our program.

### Examples:

- We created a program of even and odd and this is the code that it holds

```
def check():  
    a = int(input('Enter a number: '))  
    if a % 2 == 0:  
  
        print("Even")  
    else:  
        print("Odd")
```

- To call function `check()`, we do the following:

```
from even_odd import check  
check()
```

- Now let's do it with already built-in library

**Math.py**

```
import math
print(math.factorial(6))
print(math.pi)
print(math.e)

# u can use anything means all the functions that math library contains
as we have imported the whole library
```

- If you want to import only limited functions from math library or only a single function u can do this:

```
from math import factorial
print(factorial())
# now here u can only use factorial function as u have import only
factorial function from math.py
# if u want to print any other function it will give u error
print(pi)
```

- If u do not want to write (math.function\_name) again and again just simply do this:

```
from math import *
print(factorial(7))
print(pi)
print(tan(60))
```

## Python Packages:

A package, in essence, is like a directory holding subpackages and modules. While we can create our own packages, we can also use one from the Python Package Index (PyPI) to use for our projects.

To import a package, we type the following:

- `import Game.Sound.load`
- We can also import it giving it an alias:
- `import Game.Sound.load as load_game`
- You can't import a function using the dot operator(.) For that, you must type this:
- `from Game.Sound.load import volume_up`
- A package must have the file `__init__.py`, even if you leave it empty.
- But when we import a package, only its immediate modules are imported, not the sub-packages. If you try to access those, it will raise an `AttributeError`.

### **Difference between Modules and Packages in Python:**

So, now that we've revised both modules and packages, let's see how they differ:

- A module is a file containing Python code. A package, however, is like a directory that holds sub-packages and modules.
- A package must hold the file `__init__.py`. This does not apply to modules.
- To import everything from a module, we use the wildcard `*`. But this does not work with packages.