Experiment Number – 13

Title – Write a program to demonstrate creating and importing a package

**Theory**

We usually organize our files in different folders and subfolders based on some criteria, so that they can be managed easily and efficiently. For example, we keep all our games in a Games folder and we can even subcategorize according to the genre of the game or something like this. The same analogy is followed by the packages in [Python](https://www.geeksforgeeks.org/python-programming-language/).

A package is a container that contains various functions to perform specific tasks. For example, the math package includes the sqrt() function to perform the square root of a number. While working on big projects, we have to deal with a large amount of code, and writing everything together in the same file will make our code look messy. Instead, we can separate our code into multiple files by keeping the related code together in packages.

A **python package** is a collection of modules. Modules that are related to each other are mainly put in the same package. When a module from an external package is required in a program, that package can be imported and its modules can be put to use. Any Python file, whose name is the module’s name with the **.py** extension, is a **module**.

A package is a directory of Python modules that contains an additional \_\_init\_\_.py file, which distinguishes a package from a directory that is supposed to contain multiple Python scripts. Packages can be nested to multiple depths if each corresponding directory contains its own \_\_init\_\_.py file.

When you import a module or a package, the object created by Python is always of type module.

**Creating Packages**

We have included a **\_\_init\_\_.py,** file inside a directory to tell Python that the current directory is a package. Whenever you want to create a package, then you have to include **\_\_init\_\_.py** file in the directory. You can write code inside or leave it as blank as your wish. It doesn't bothers Python.

**Follow the below steps to create a package in Python**

* Create a directory and include a **\_\_init\_\_.py** file in it to tell Python that the current directory is a **package**.
* Include other sub-packages or files you want.
* Next, access them with the valid **import** statements.

### Understanding \_\_init\_\_.py

\_\_init\_\_.py helps the Python interpreter recognize the folder as a package. It also specifies the resources to be imported from the modules. If the \_\_init\_\_.py is empty this means that all the functions of the modules will be imported. We can also specify the functions from each module to be made available.

For example, we can also create the \_\_init\_\_.py file for the above module as:

**\_\_init\_\_.py**

|  |
| --- |
| from mod1 import gfg  from mod2 import sum |

This \_\_init\_\_.py will only allow the gfg and sum functions from the mod1 and mod2 modules to be imported.

### Import Modules from a Package

We can import these Python modules using the [from…import statement](https://www.geeksforgeeks.org/import-module-python/) and import statement.

**Syntax:**

import package\_name.module\_name

from package\_name import module\_name