



## Session 9 - Modules

1. Types of modules
    - Built-in Modules
    - Installed Modules
    - user created modules
  2. Getting started with the PyAutoGui Module
  3. Creating a simple Modules
- 

**Modules are one of the biggest reason why python is so famous.**

You can literally think of any task and you can find an existing module which will make your life a lot easier.

### 1. Built-in Modules

Few Modules come in handy in everyday programming. So these moduels comes preinstalled when you install python. So you can directly install and start using these modules.

Below link shows a list of all the moduels that comes preinstalled with python

[Built-in Modules](#)

### Lets try using the Time module

To import a module we can use the following **syntax** (there are multiple ways to import a module this is just one of them)

```
import module_name
```

```
In [1]: import time
```

This makes all the function defined inside the time module available to us.

Once imported we can call any function we want.

```
In [2]: print("hello")
time.sleep(2)
print("world")
```

```
hello
```

world

If the run the above cell you will get hello printed on the screen follwed by thw world after a 2 second delay.

well that is what the sleep function do, it makes the python code to wait for some time.

**This brings us to the next important thing while using modules that is reading the documentation.**

**!! Always read the documention before you start working with a module.**

Time Module

## 2. Installed Modules

All Modules apart from the Built-in ones needs to be installed to be able to use them. Almost all the modules createdd by the users can be found on the [PyPi](#) website

**Installing modules is pretty straight foward**

### **For Windows**

- open command prompt and type

```
pip install module_name
```

### **For Mac**

- open Terminal and type

```
pip3 install module_name
```

**NOTE : The instruiucion to install a module can be found on the PyPi project page itself. See the below image**

# PyAutoGUI 0.9.52

✓

Latest version

`pip install PyAutoGUI`

Released: Oct 7, 2020

PyAutoGUI lets Python control the mouse and keyboard, and other GUI automation tasks. For Windows, macOS, and Linux, on Python 3 and 2.

## Navigation

Project description

Release history

Download files

## Project description

### PyAutoGUI

PyAutoGUI is a cross-platform GUI automation Python module for human beings. Used to programmatically control the mouse & keyboard.

`pip install pyautogui`

We can open a command prompt or terminal on windows to use the command shown to install the module

```
Command Prompt
Microsoft Windows [Version 10.0.18362.900]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\REJIN>pip install pyautogui
```

To verify if the module is installed properly we can try importing the module

```
import pyautogui
```

If you don't get any output it means the module is installed properly.

If you get a `ModuleNotFoundError` Error check the command prompt or terminal for the reason

```
In [3]: import pyautogui
```

**Well what is the package that we just installed??**

As the PyPi website states its a cross platform modules that allows is to control the keybaord and mouse programatically.

**As with any other Modules read the documentation before you start using it.**

[PyAutoGui Documentaion](#)

**Lets try to bring the search bar up on ---> windows.**

we need to press the `win+q` key to make that happen

```
In [4]: import pyautogui

pyautogui.keyDown('win')
pyautogui.press('q')
pyautogui.keyUp('win')
```

If you run the above cell it should bring up the search bar

**Lets try to bring up the spotlight on ---> Mac.**

```
In [5]: import pyautogui

pyautogui.keyDown('command')
pyautogui.press('space')
pyautogui.keyUp('command')
```

**Wait It didn't work!!**

Well in mac you need to provide some extra access for the pyautogui module to be able to automate things.

You can follow these steps to provide it access.

goto system preferecnes ---> unlock the lock ---> give access to the app you are using to run the code ---> save changes ---> lock

## 3. User defined module

Here you can create your own module. Let's try to make own module for calculator. We will use separate file to define functions used in the calculator And then will import the file into main (execution file)

## Home work

**1. Write a python script to automatically send a mail to yourself using PyAutoGui**

1. open google chrome
2. open Gmail
3. click on compose mail
4. Fill in the email ID, Subject and Body

5. send Mail using the shortcut

In [ ]: