# Sharing code via Git Hub and introduction to creating a Python library (structure and Python PyPi Packaging)

Using Github:

* What is GitHub? - <https://www.youtube.com/watch?v=w3jLJU7DT5E>
* Tutorial: Github Tutorial For Beginners - Github Basics for Mac or Windows & Source Control Basics: <https://www.youtube.com/watch?v=0fKg7e37bQE>
* Tutotial 2 - iteractive: <https://try.github.io/> to get to know it.
* If you want to go further in here, have a look into these two course:
  + codeacademy: <https://www.codecademy.com/learn/learn-git>
  + coursera: <https://www.coursera.org/learn/data-scientists-tools/lecture/SqWSK/introduction-to-github>

## Steps For Installing Git for Windows

https://phoenixnap.com/kb/how-to-install-git-windows

Main git commands:

* git init
* git add \*
* git commit -m "first commit"
* git remote rm origin
* git remote add origin https://github.com/...   
  (Replace https://github.com/... with your own github project https path created on the github website - if you haven't created, please do for this workaround to work)

GIt & Github cheat sheet: <https://training.github.com/downloads/github-git-cheat-sheet.pdf>

Python package structure - Organize Your Code: <https://www.youtube.com/watch?v=qmsTqQbcBNM>

**Detailed instruction for creating Python Pypi Library:**

Sharing Python code using package "pip install" is what makes Python great and it is a bonus part of this individual exercise sharing your **Method of linear congruences** function.

The following instruction is intended to help you with this task of creating your python package. Expect to spend 3 to 4 hours following the instructions below as you'll probably need to watch some video instructions and maybe get stuck in a point where you need to return to very beginning. It is a bit challenging to do, so please be patient, and if you get stuck, consider starting from scratch again!

Recommended name of the package: **ie\_mbdbl2018\_mfalonso** (replace mfalonso with your own IE urser name)

Your package must have the following function:

def  linear\_congruence\_random\_generator():  
    """ function to return a random number using the Method of linear congruences."""

   #YOUR CODE GOES HERE  
    return(my\_random\_number)

**Instruction to create your package:**  
HELPUL VIDEOS:

* If you are using **Mac**, on top of following the instructions below please watch this video from a colleague student who kindly recorded instruction for Mac:  <https://youtu.be/MuhOtYknkZc> (he only forgot to run the first line, to upload the package to pypi)
* You can watch this video if you need any help: <https://www.youtube.com/watch?v=nelRslDOK_Y> (but careful pypi has changed servers since the video was recorded as explained here:[https://stackoverflow.com/questions/46008952/pypi-assertionerror-unsupported-schema](https://campus.ie.edu/webapps/discussionboard/do/%20https:/stackoverflow.com/questions/46008952/pypi-assertionerror-unsupported-schema))

INSTRUCTIONS - PLEASE READ THEM ALL ONCE BEFORE STARTING TO FOLLOW THEM:

* + Register (create an account) on Pypi (pip repository): https://pypi.org/account/register/
  + Install Git - https://git-scm.com/book/en/v2/Getting-Started-Installing-Git
    - On Windows, install chocolatey with administrator permission on the CMD console:

@"%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile -InputFormat None -ExecutionPolicy Bypass -Command "iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))" && SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"

(reference: https://chocolatey.org/docs/installation#installing-chocolatey)  
Then run:  
choco install git

* + Clone this github minimal python package I have created: https://github.com/manoelgadi/pythonpackagesample - enter on console (any directory you wish t work) - CMD (for windows) or shell for Linux/Mac
    - git clone https://github.com/manoelgadi/pythonpackagesample
  + Edit the file .pypirc with your credentials to https://pypi.org; next, move ONLY the file .pypirc to your user directory, on windows (C:\Users\<username>), linux (/home/<username>) and Mac (/Users/<username>); if you are in Linux or Mac change the permission of the file granting all permissions to the file .pypirc : (1st: cp ./.pypirc ~/.pypirc ; 2nd: sudo chmod 777~/.pypirc )
* Change the directory name from pythonpackagesample for ie\_mbdbl2018\_mfalonso  (replace mfalonso with your own IE urser name) . Now open all files, and replace  pythonpackagesample with ie\_mbdbl2018\_mfalonso  everywhere (replace mfalonso with your own IE urser name).
* Edit the setup.py file, changing author and author\_email.
* Open the file \_\_init\_\_.py and replace the current text returned with "This is Paul Smith first package that anyone can install it! How cool!!!"

PUBLISHING ON GITHUB, HERE IS A QUICK GUIDE ON HOW TO DO IT (THIS PART IS OPTIONAL, YOU CAN SKIP IT IF YOU FACING PROBLEMS AND CAN'T PROCEED) :

* Register (create an account) for Git Hub if you don't have one: https://github.com/
* Then run:
  + git init
  + git add \*
  + git commit -m "1st commit"
  + git push -u origin master  
    (reference video: how to create the repository in Github on Windows with PowerShell console: <https://www.youtube.com/watch?v=Y4uqOVl3VMY>)
* Log into your Git Hub account, find the project there and copy the project url.
* Come back to setup.py and replace the url with the url you have just copied.

IF YOU HAVE PROBLEMS PUBLISHING ON GIT - PLEASE HAVE A LOOK INTO THIS: Maybe it is because your cloned files are linked to my git not yours.:

* git push origin master

PUBLISHING YOUR CODE ON PYPI:

* python setup.py sdist upload -r pypi

(please don't use twine, the process is different with twine)

Now you can test your code by running: "pip install ie\_mbdbl2018\_mfalonso"   (replace mfalonso with your own IE urser name)

You can also login into pypi  https://pypi.org/manage/projects/ and check your project there ;-)

Congratulation or creating your Python package!

Welcome to the Python community of contributors ;-)