Name: 潘氏玄女英

Student ID: 0781532

Machine Learning on Material Informatics
Development Environment For Machine Learning
Homework 01

TASK

1. Install Miniconda/Anaconda on your own computer. And try to rebuild the environment we used in the class.

2. Create an account and initial a project on one of online code management (github, gitlab or bitbucket). The commit history of your repository is considered to be part of evaluation.

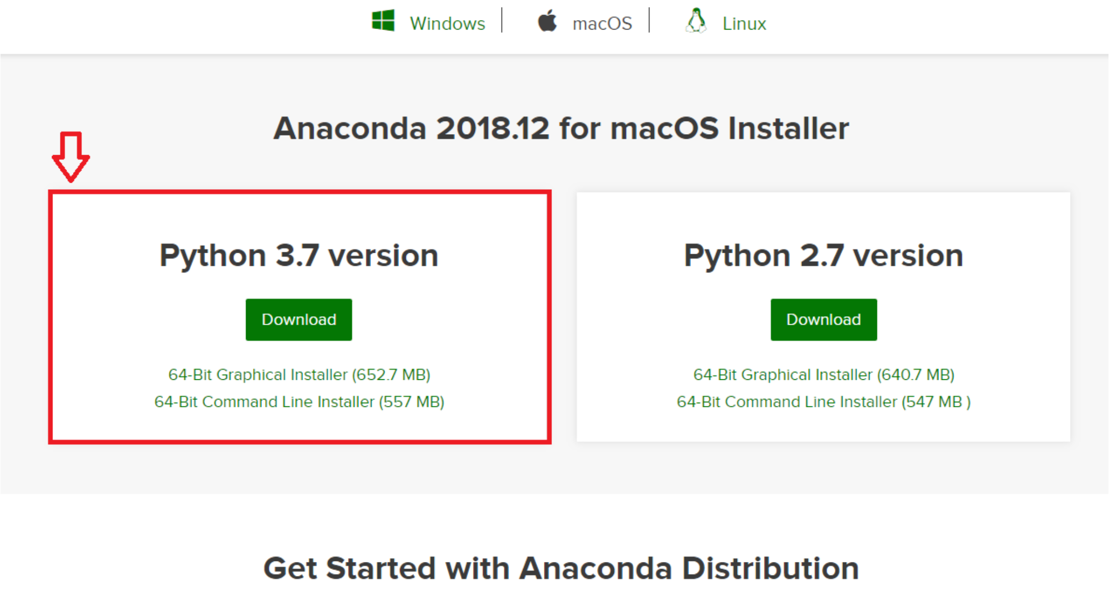
3. Download Linear Regression Example from scikit-learn and open it
   https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html#

SOLUTION

1. **Install Miniconda/Anaconda on your own computer.**

   Install and use Machine Learning with Anaconda python on window

   Step1: Download Python 3.7 version



Anaconda 2018.12 for macOS Installer

Python 3.7 version

64-Bit Graphical Installer (652.7 MB)
64-Bit Command Line Installer (557 MB)

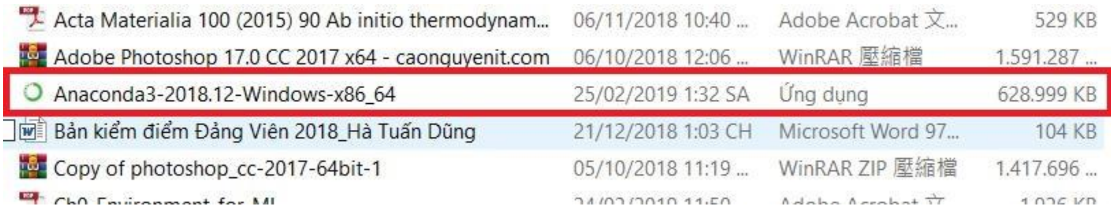
Python 2.7 version

64-Bit Graphical Installer (640.7 MB)
64-Bit Command Line Installer (547 MB)

Get Started with Anaconda Distribution

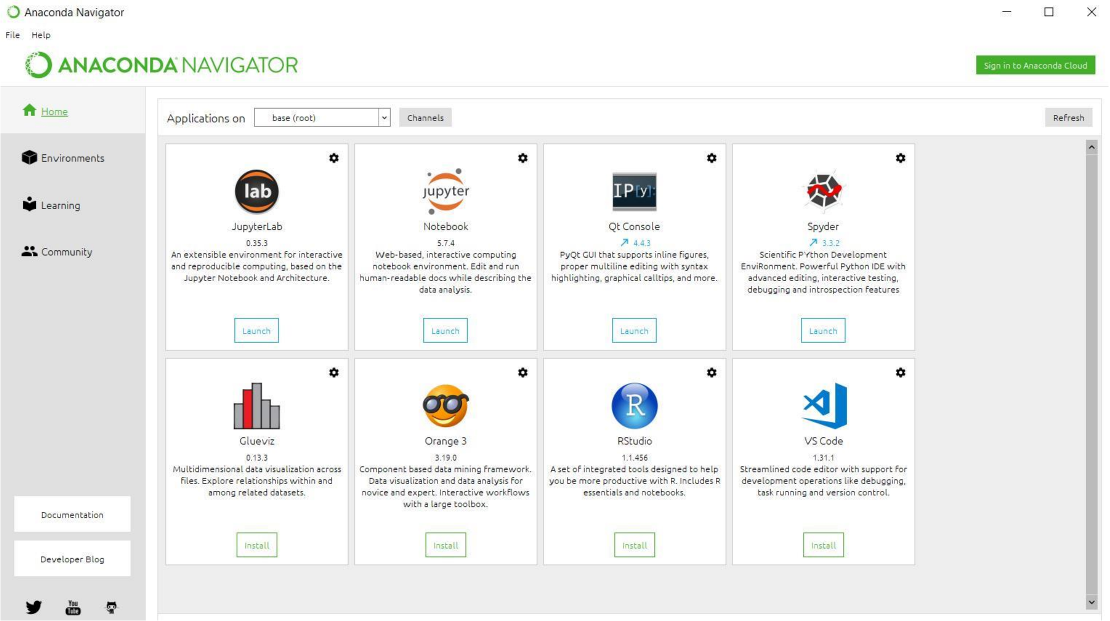
https://www.anaconda.com/distribution/#download-section

Step2: Install Anaconda3-2018.12-Windows-x86_64



Acta Materialia 100 (2015) 90 Ab initio thermodynam...	06/11/2018 10:40 ...	Adobe Acrobat 文...	529 KB
Adobe Photoshop 17.0 CC 2017 x64 - caonguyenit.com	06/10/2018 12:06 ...	WinRAR 壓縮檔	1.591.287 ...
Anaconda3-2018.12-Windows-x86_64	25/02/2019 1:32 SA	Ứng dụng	628.999 KB
Bản kiểm điểm Đảng Viên 2018_Hà Tuấn Dũng	21/12/2018 1:03 CH	Microsoft Word 97...	104 KB
Copy of photoshop_cc-2017-64bit-1	05/10/2018 11:19 ...	WinRAR ZIP 壓縮檔	1.417.696 ...
Ch0 Environment for ML	24/02/2019 11:50 ...	Adobe Acrobat 文...	1.026 KB

After the installation is completed, the anaconda has the following interface:



Applications on: base (root)

JupyterLab 0.35.3 — An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook 5.7.4 — Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Qt Console 4.4.3 — PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder 3.3.2 — Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Glueviz 0.13.3 — Multidimensional data visualization across files. Explore relationships within and among related datasets.

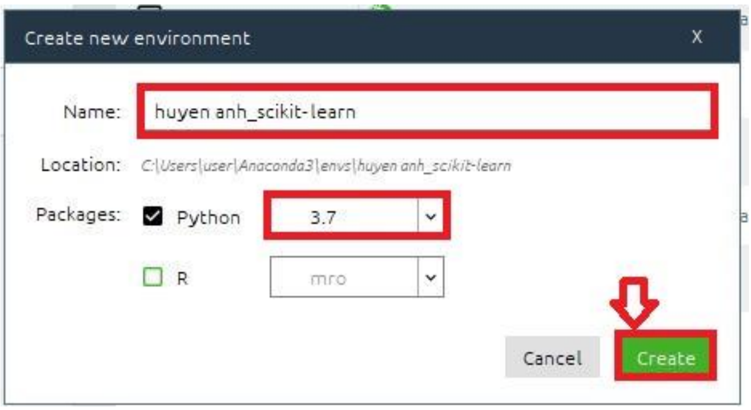
Orange 3 3.19.0 — Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

RStudio 1.1.456 — A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

VS Code 1.31.1 — Streamlined code editor with support for development operations like debugging, task running and version control.

Step3: Click **Environments** >> Create a **New Environments** >> **Name**: huyenanh-scikit-learn

Packages: choose **Python 3.7**



Create new environment

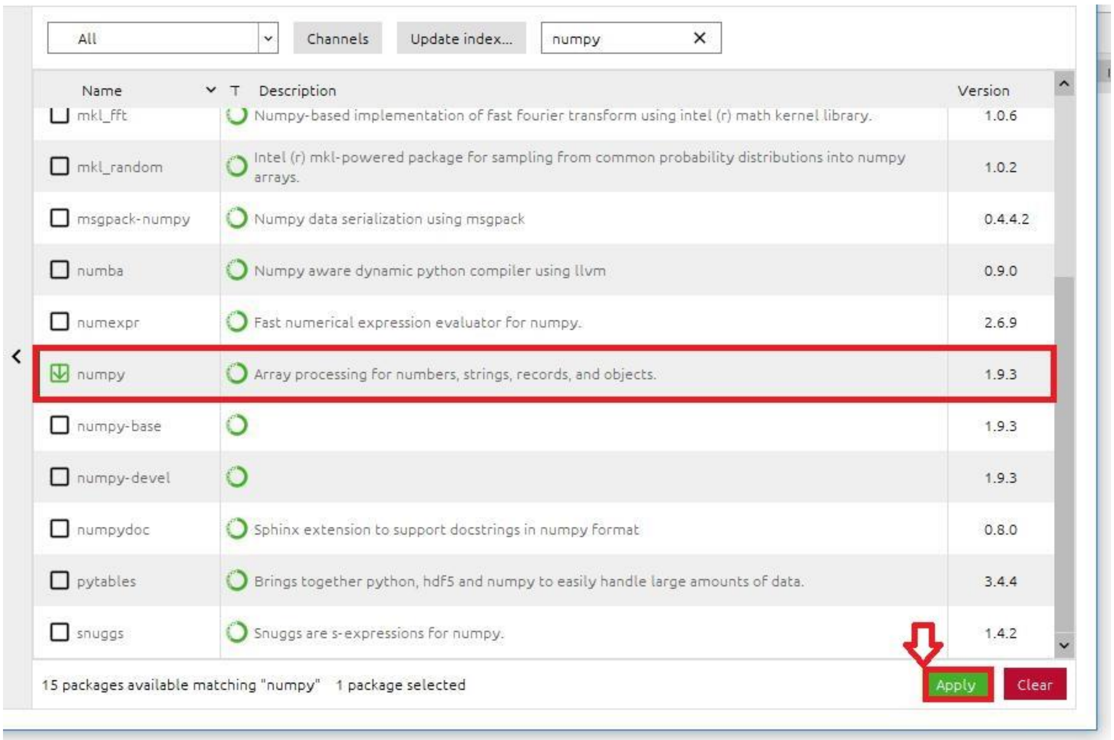
Name: huyen anh_scikit-learn

Location: C:\Users\user\Anaconda3\envs\huyen anh_scikit-learn

Packages: ☑ Python 3.7

□ R mro

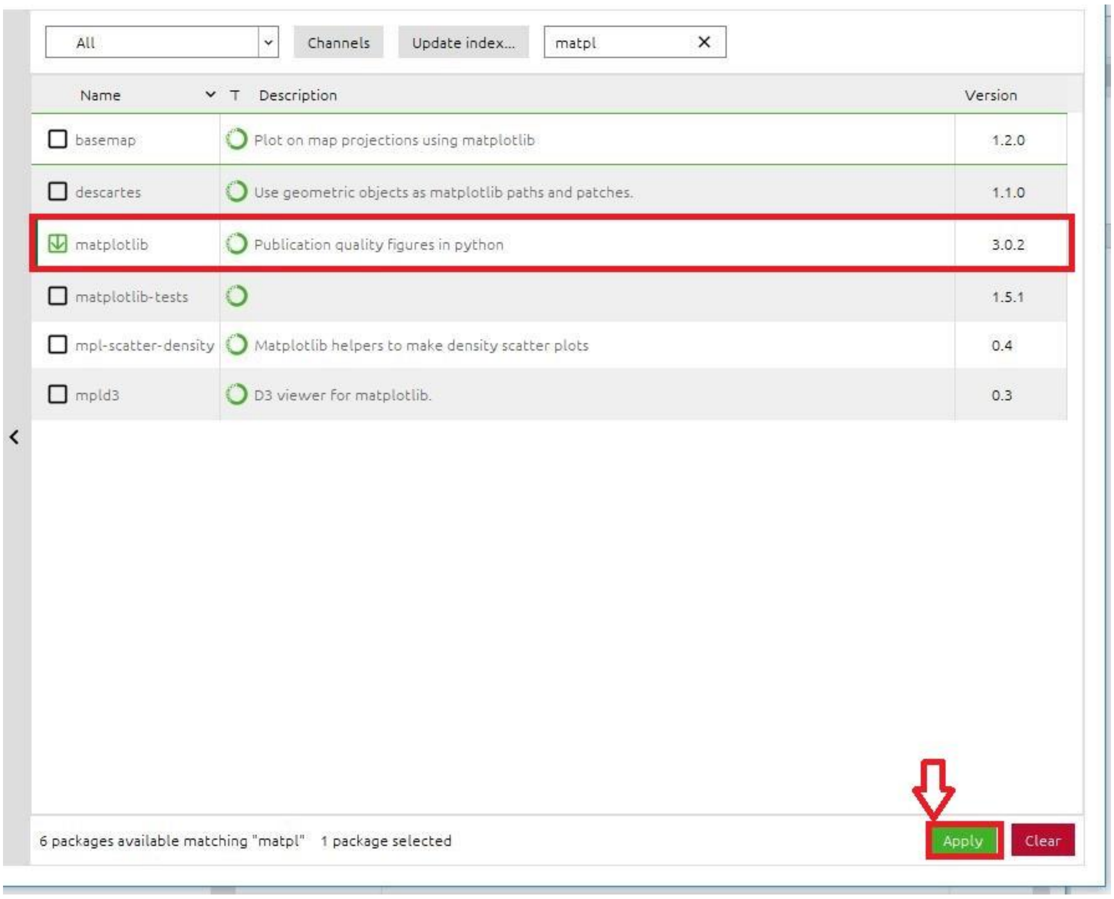
- **Install numpy scipy matplotlib:** click **numpy** choose **Apply**



	Name	T	Description	Version
□	mkl_fft	○	Numpy-based implementation of fast fourier transform using intel (r) math kernel library.	1.0.6
□	mkl_random	○	Intel (r) mkl-powered package for sampling from common probability distributions into numpy arrays.	1.0.2
□	msgpack-numpy	○	Numpy data serialization using msgpack	0.4.4.2
□	numba	○	Numpy aware dynamic python compiler using llvm	0.9.0
□	numexpr	○	Fast numerical expression evaluator for numpy.	2.6.9
☑	numpy	○	Array processing for numbers, strings, records, and objects.	1.9.3
□	numpy-base	○		1.9.3
□	numpy-devel	○		1.9.3
□	numpydoc	○	Sphinx extension to support docstrings in numpy format	0.8.0
□	pytables	○	Brings together python, hdf5 and numpy to easily handle large amounts of data.	3.4.4
□	snuggs	○	Snuggs are s-expressions for numpy.	1.4.2

15 packages available matching "numpy"   1 package selected

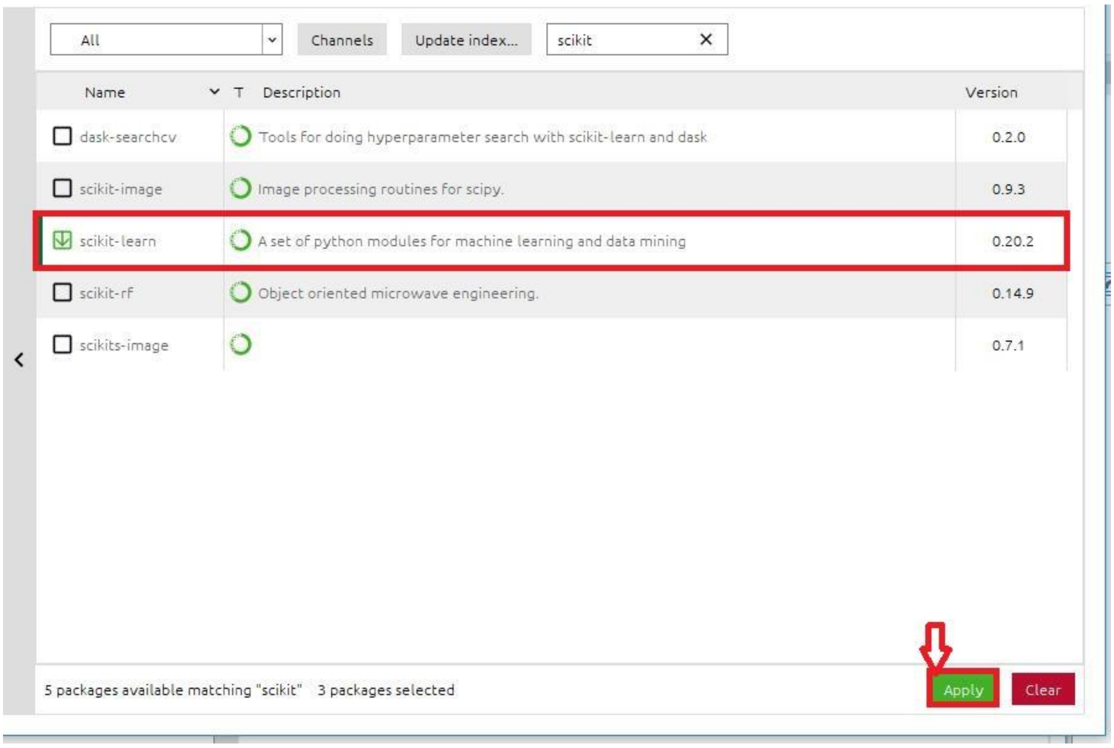
click **matplotlib** choose **Apply**



	Name	T	Description	Version
□	basemap	○	Plot on map projections using matplotlib	1.2.0
□	descartes	○	Use geometric objects as matplotlib paths and patches.	1.1.0
☑	matplotlib	○	Publication quality figures in python	3.0.2
□	matplotlib-tests	○		1.5.1
□	mpl-scatter-density	○	Matplotlib helpers to make density scatter plots	0.4
□	mpld3	○	D3 viewer for matplotlib.	0.3

6 packages available matching "matpl"   1 package selected

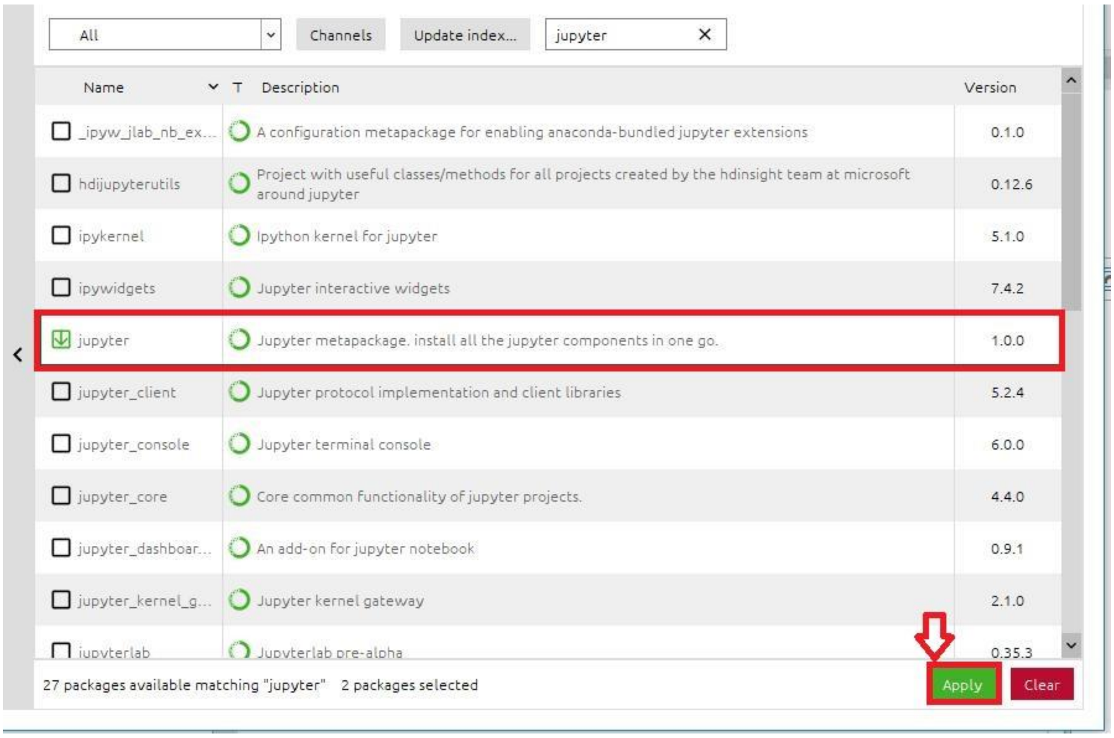
- **Install scikit-learn:** Click **scikit-learn** choose **Apply**



	Name	T	Description	Version
□	dask-searchcv	○	Tools for doing hyperparameter search with scikit-learn and dask	0.2.0
□	scikit-image	○	Image processing routines for scipy.	0.9.3
☑	scikit-learn	○	A set of python modules for machine learning and data mining	0.20.2
□	scikit-rf	○	Object oriented microwave engineering.	0.14.9
□	scikits-image	○		0.7.1

5 packages available matching "scikit" 3 packages selected

- **Install jupyter or spyder:** Click **jupyter** choose **Apply**

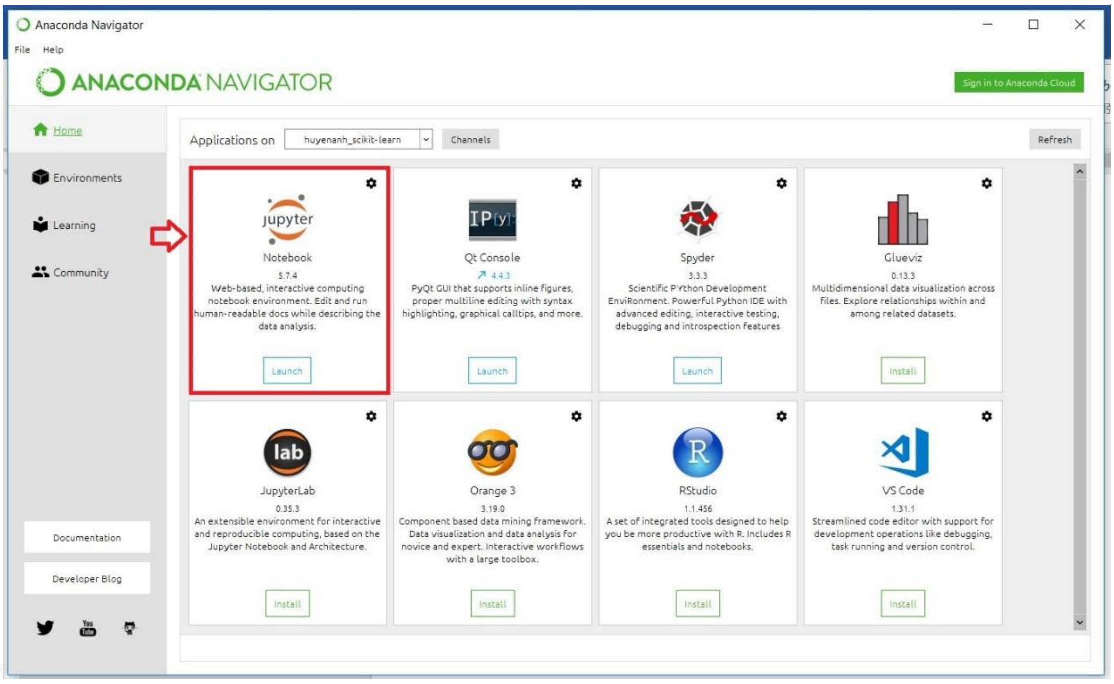


	Name	T	Description	Version
□	_ipyw_jlab_nb_ex...	○	A configuration metapackage for enabling anaconda-bundled jupyter extensions	0.1.0
□	hdijupyterutils	○	Project with useful classes/methods for all projects created by the hdinsight team at microsoft around jupyter	0.12.6
□	ipykernel	○	Ipython kernel for jupyter	5.1.0
□	ipywidgets	○	Jupyter interactive widgets	7.4.2
☑	jupyter	○	Jupyter metapackage. install all the jupyter components in one go.	1.0.0
□	jupyter_client	○	Jupyter protocol implementation and client libraries	5.2.4
□	jupyter_console	○	Jupyter terminal console	6.0.0
□	jupyter_core	○	Core common functionality of jupyter projects.	4.4.0
□	jupyter_dashboar...	○	An add-on for jupyter notebook	0.9.1
□	jupyter_kernel_g...	○	Jupyter kernel gateway	2.1.0
□	jupyterlab	○	Jupyterlab pre-alpha	0.35.3

27 packages available matching "jupyter" 2 packages selected

Step 4: Test

- Click **Home** >> Launch **jupyter Notebook**



Notebook 5.7.4 — Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Qt Console 4.4.3 — PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder 3.3.3 — Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

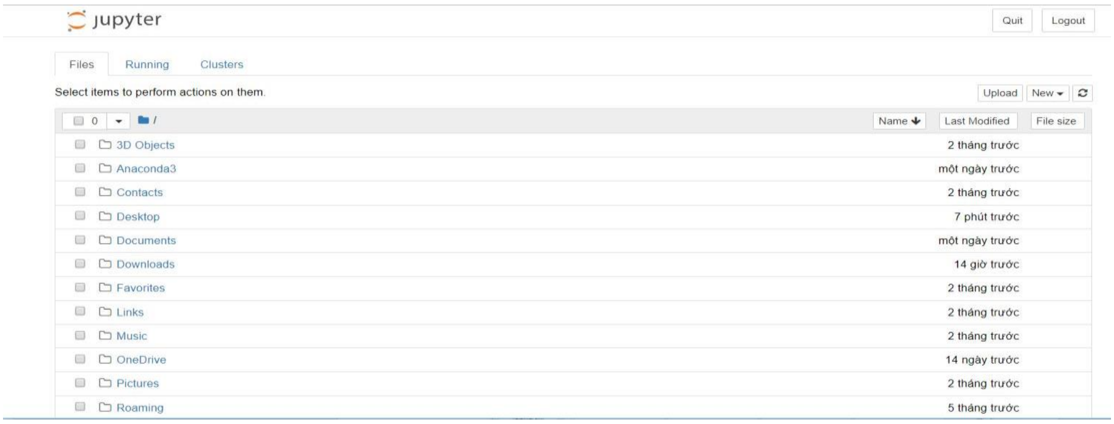
Glueviz 0.13.3 — Multidimensional data visualization across files. Explore relationships within and among related datasets.

JupyterLab 0.35.3 — An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Orange 3 3.19.0 — Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

RStudio 1.1.456 — A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

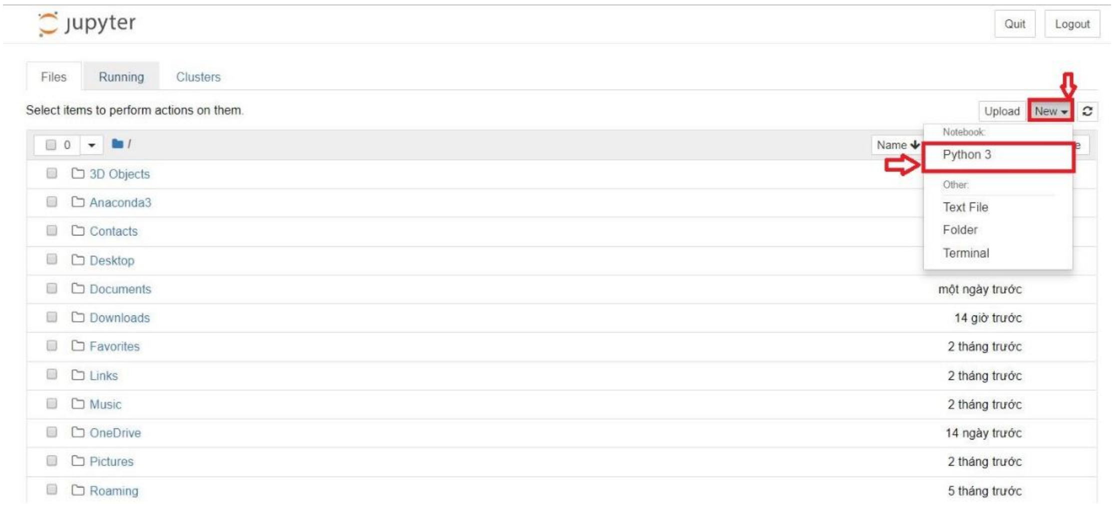
VS Code 1.31.1 — Streamlined code editor with support for development operations like debugging, task running and version control.



Select items to perform actions on them.

Name	Last Modified	File size
3D Objects	2 tháng trước	
Anaconda3	một ngày trước	
Contacts	2 tháng trước	
Desktop	7 phút trước	
Documents	một ngày trước	
Downloads	14 giờ trước	
Favorites	2 tháng trước	
Links	2 tháng trước	
Music	2 tháng trước	
OneDrive	14 ngày trước	
Pictures	2 tháng trước	
Roaming	5 tháng trước	

- Click **New** >> Choose **Python 3**



Select items to perform actions on them.

Name	Last Modified
3D Objects	
Anaconda3	
Contacts	
Desktop	
Documents	một ngày trước
Downloads	14 giờ trước
Favorites	2 tháng trước
Links	2 tháng trước
Music	2 tháng trước
OneDrive	14 ngày trước
Pictures	2 tháng trước
Roaming	5 tháng trước

Step 5: Try to run some code

- Rename Notebook
- In[1]: %matplotlib inline
- In[2]: import numpy as np

  import matplotlib.pyplot as plt

  Compute the x and y coordinates for points on a sine curve

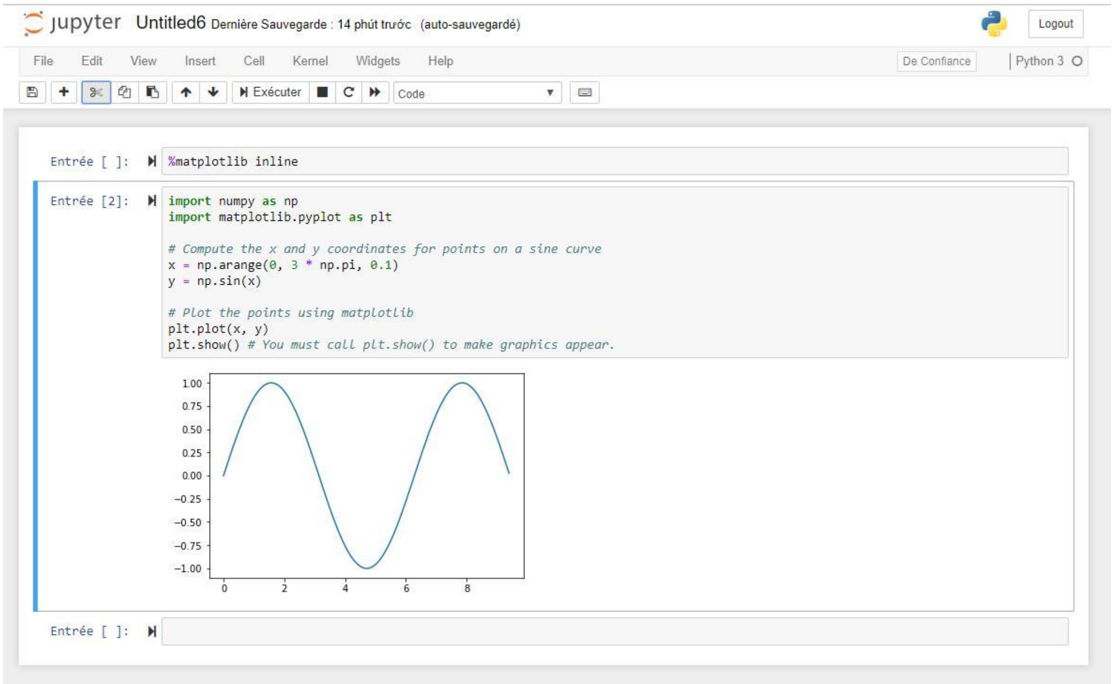
  x = np.arange(0, 3 * np.pi, 0.1)

  y = np.sin(x)

  Plot the points using matplotlib

  plt.plot(x, y)

  plt.show() # You must call plt.show() to make graphics appear.
- Run
- **Result**



```
Entrée [ ]:  %matplotlib inline
```

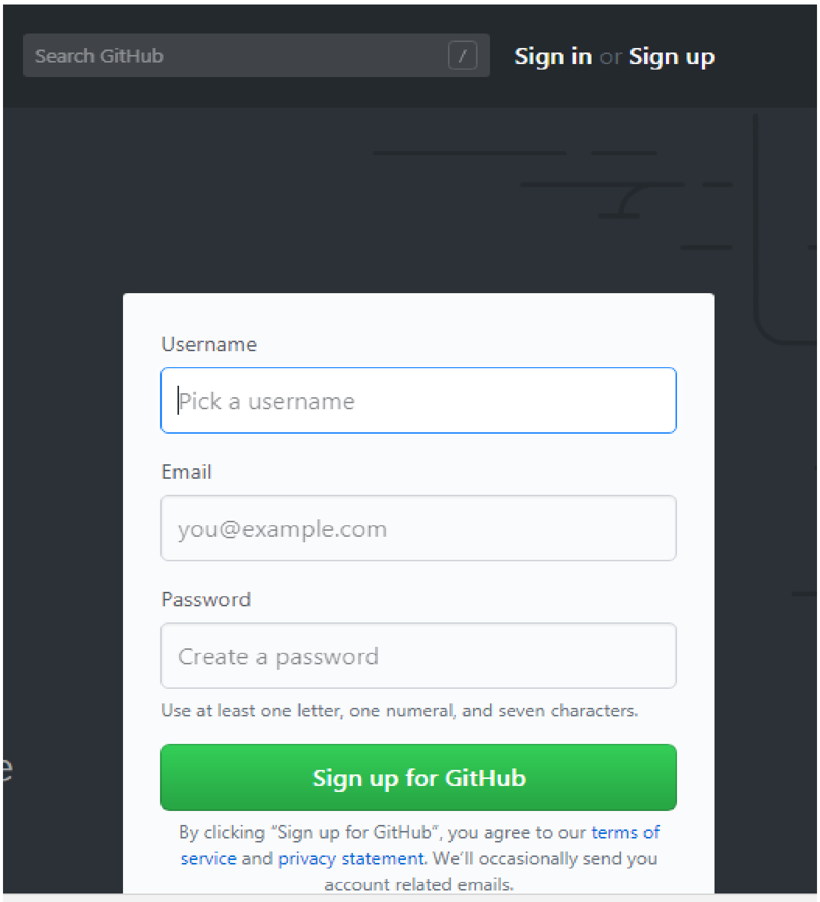
```
Entrée [2]:  import numpy as np
             import matplotlib.pyplot as plt

             # Compute the x and y coordinates for points on a sine curve
             x = np.arange(0, 3 * np.pi, 0.1)
             y = np.sin(x)

             # Plot the points using matplotlib
             plt.plot(x, y)
             plt.show() # You must call plt.show() to make graphics appear.
```

```
Entrée [ ]:
```

2. Create an account and initial a project on one of online code management use Github

Step 1: Create an account

- Go to github.com and start the registration process for an account.
  Usernam: 0781532
  Password:
  Email: tuyenbkdn@gmail.com
- Verify via email address if required.



Username

Pick a username

Email

you@example.com

Password

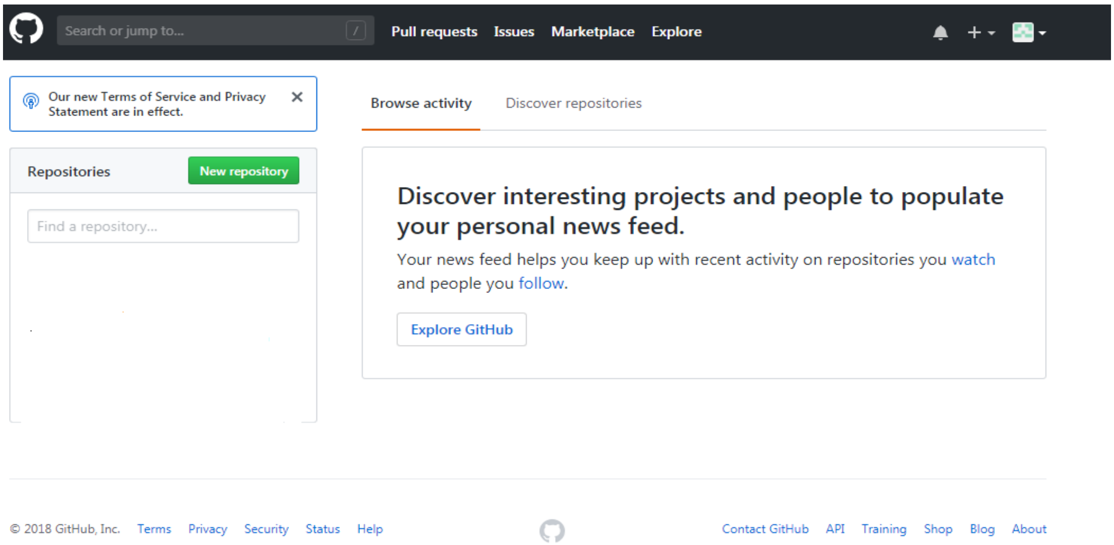
Create a password

Use at least one letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our terms of service and privacy statement. We'll occasionally send you account related emails.

Step 2: Github interface after registration:

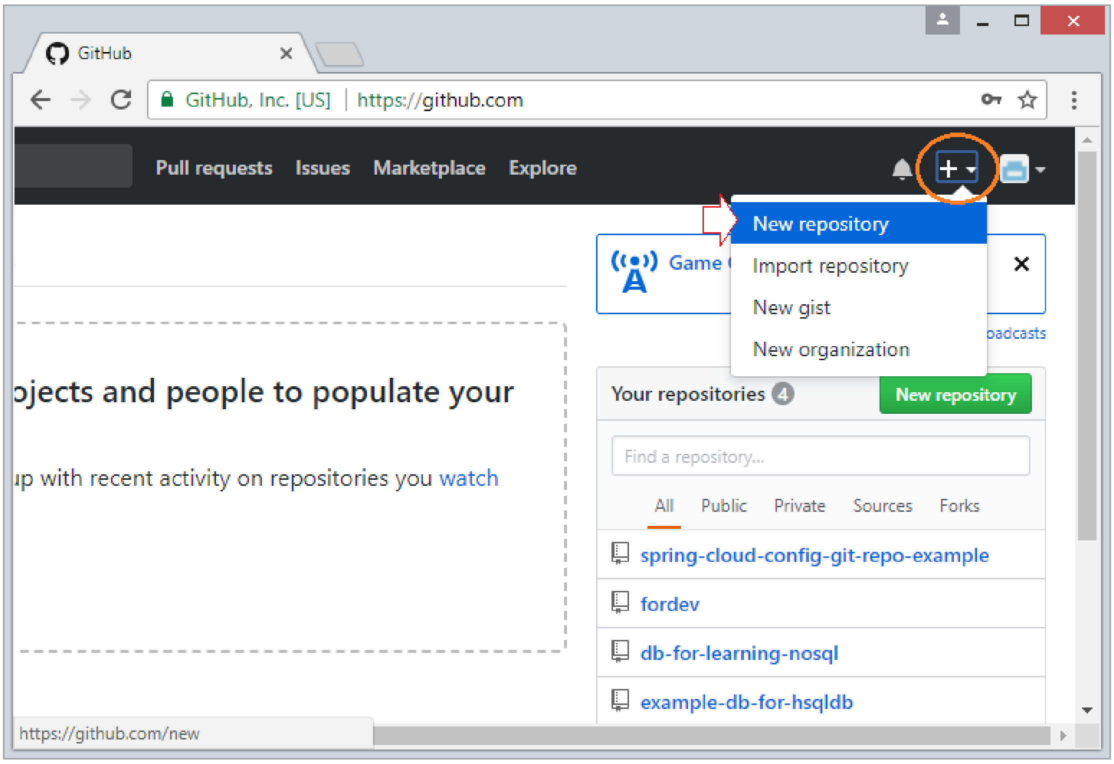


Our new Terms of Service and Privacy Statement are in effect.

Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you watch and people you follow.

Create registration



jects and people to populate your

p with recent activity on repositories you watch

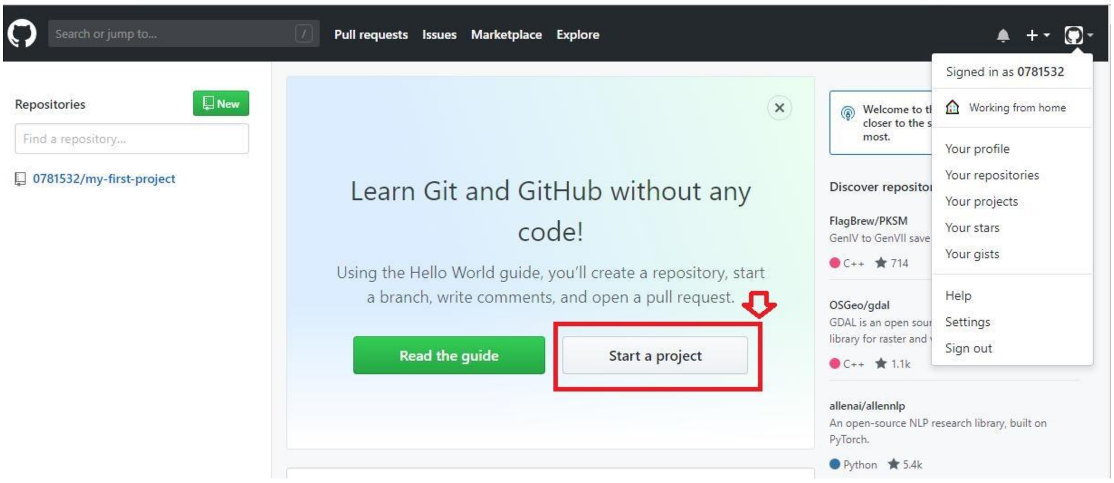
Step 3: log in to github: https://github.com

- Name account: 0781532

-

Step 4: initial a project on one of online code management use **Github**

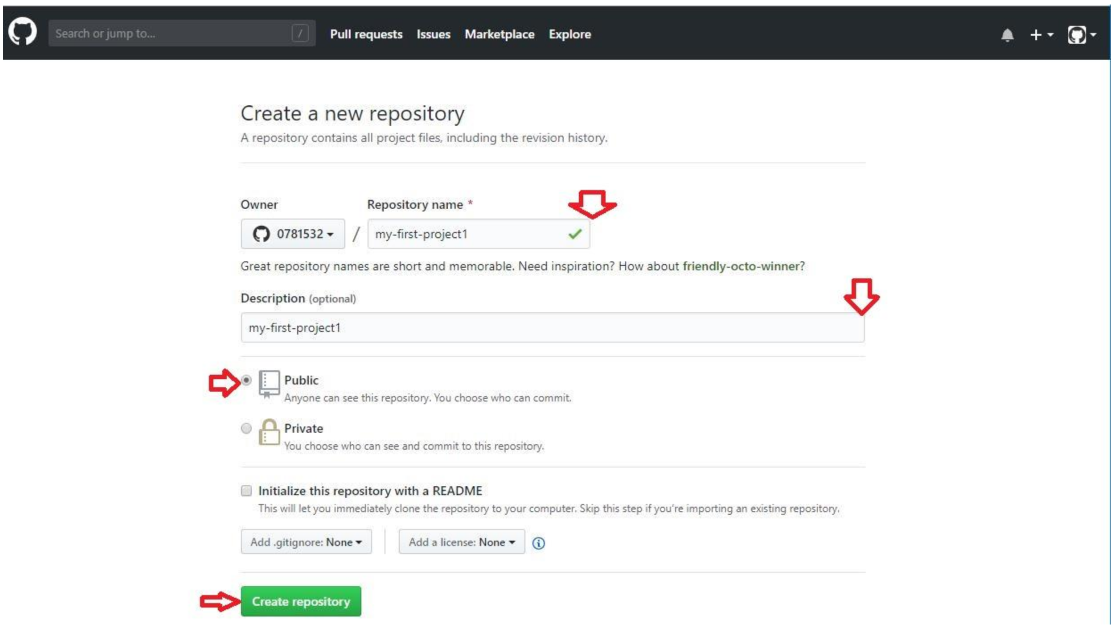
Initial a project >> choose **Star a project**



Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Step 5: Name and describe the projest



Create a new repository

A repository contains all project files, including the revision history.

Owner: 0781532 / Repository name *: my-first-project1

Great repository names are short and memorable. Need inspiration? How about friendly-octo-winner?

Description (optional): my-first-project1

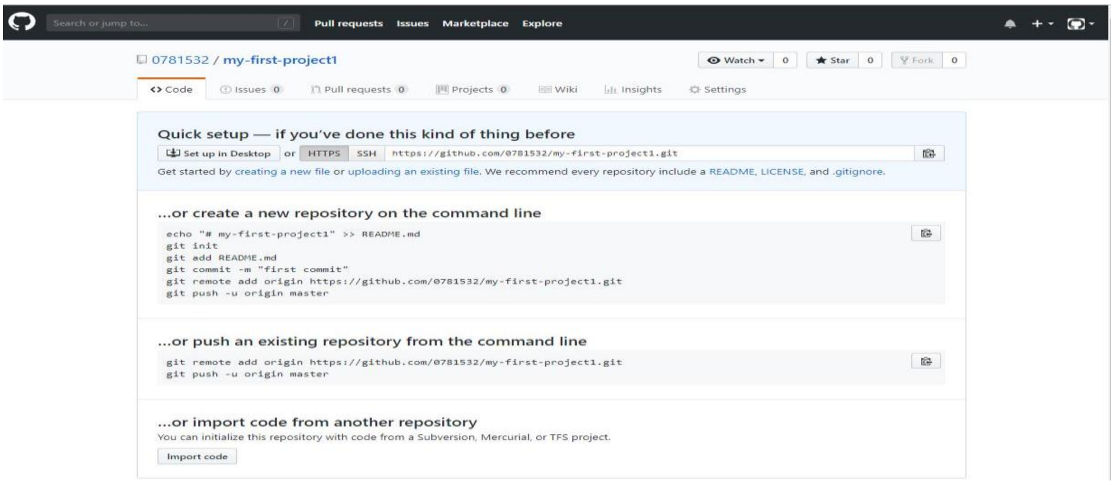
● Public
Anyone can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None

Step 6: Select Create repository, will get a remote repo to store the project



0781532 / my-first-project1

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/0781532/my-first-project1.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# my-first-project1" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/0781532/my-first-project1.git
git push -u origin master
```

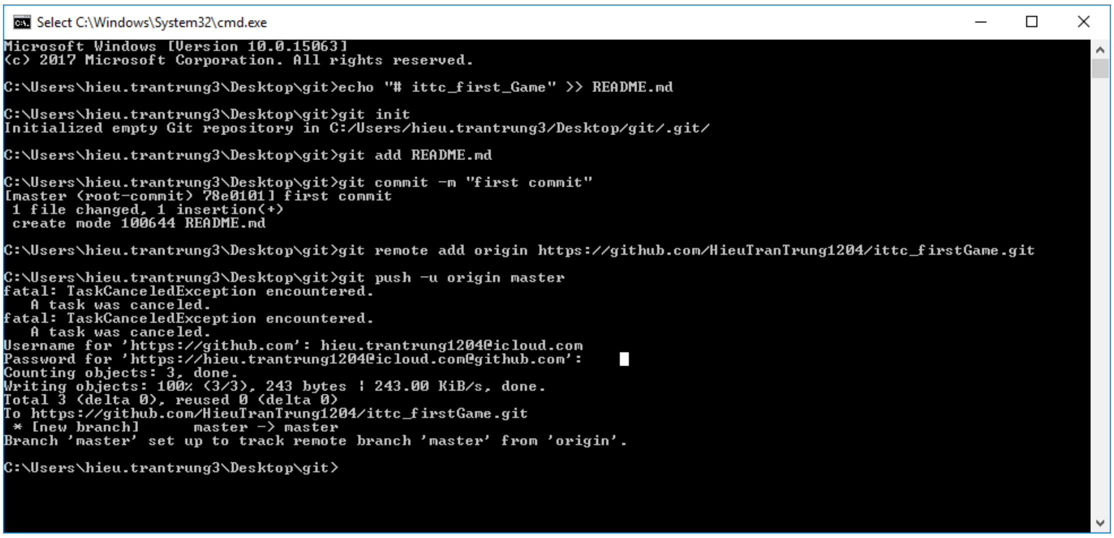
...or push an existing repository from the command line

```
git remote add origin https://github.com/0781532/my-first-project1.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

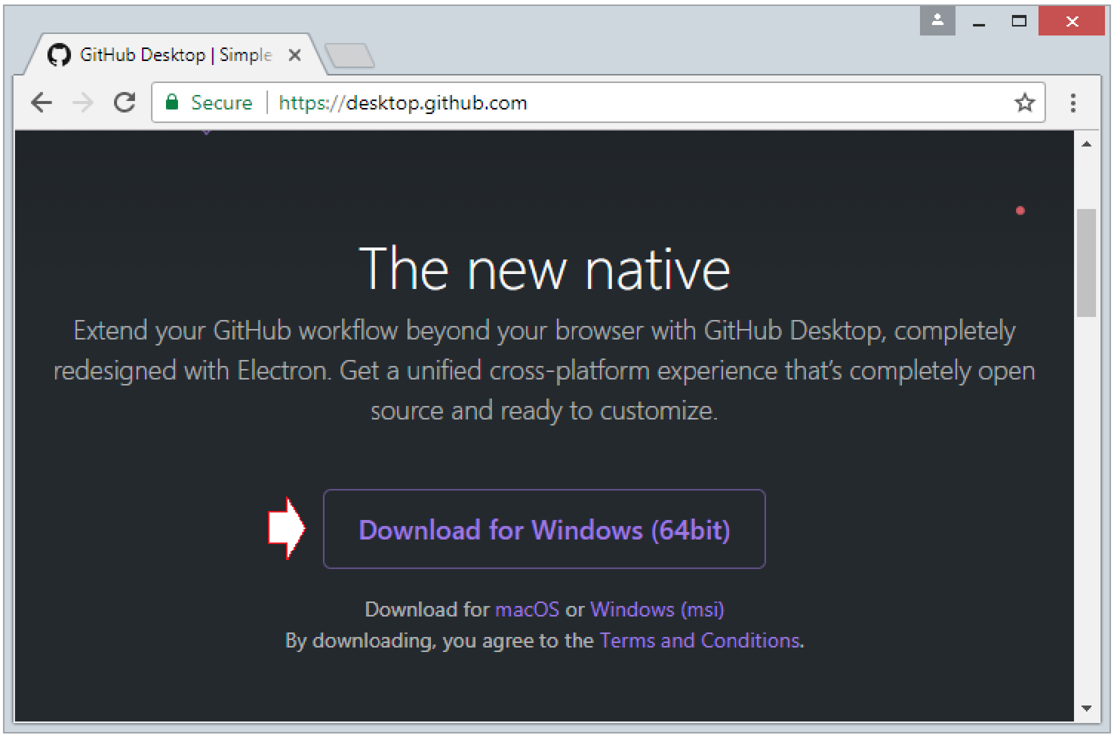
GIT already has some basic commands available as shown.



```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\hieu.trantrung3\Desktop\git>echo "# ittc_first_Game" >> README.md

C:\Users\hieu.trantrung3\Desktop\git>git init
Initialized empty Git repository in C:/Users/hieu.trantrung3/Desktop/git/.git/

C:\Users\hieu.trantrung3\Desktop\git>git add README.md

C:\Users\hieu.trantrung3\Desktop\git>git commit -m "first commit"
[master (root-commit) 78e0101] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

C:\Users\hieu.trantrung3\Desktop\git>git remote add origin https://github.com/HieuTranTrung1204/ittc_firstGame.git

C:\Users\hieu.trantrung3\Desktop\git>git push -u origin master
fatal: TaskCanceledException encountered.
   A task was canceled.
fatal: TaskCanceledException encountered.
   A task was canceled.
Username for 'https://github.com': hieu.trantrung1204@icloud.com
Password for 'https://hieu.trantrung1204@icloud.com@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 243 bytes | 243.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/HieuTranTrung1204/ittc_firstGame.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

C:\Users\hieu.trantrung3\Desktop\git>
```

Step 7: Download & Install **Github Desktop**

- GitHub Desktop is essentially a visual tool that allows you to manage Local Repository on your computer.
- Download **Github Desktop:** https://desktop.github.com

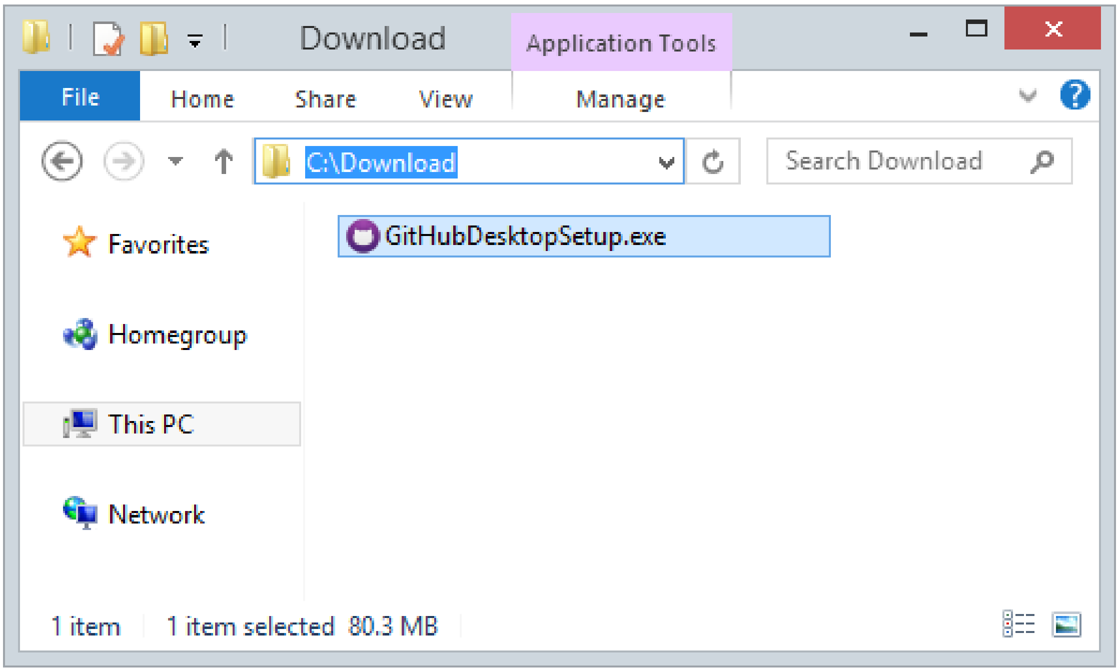


The new native

Extend your GitHub workflow beyond your browser with GitHub Desktop, completely redesigned with Electron. Get a unified cross-platform experience that's completely open source and ready to customize.

Download for macOS or Windows (msi)
By downloading, you agree to the Terms and Conditions.

Download results:

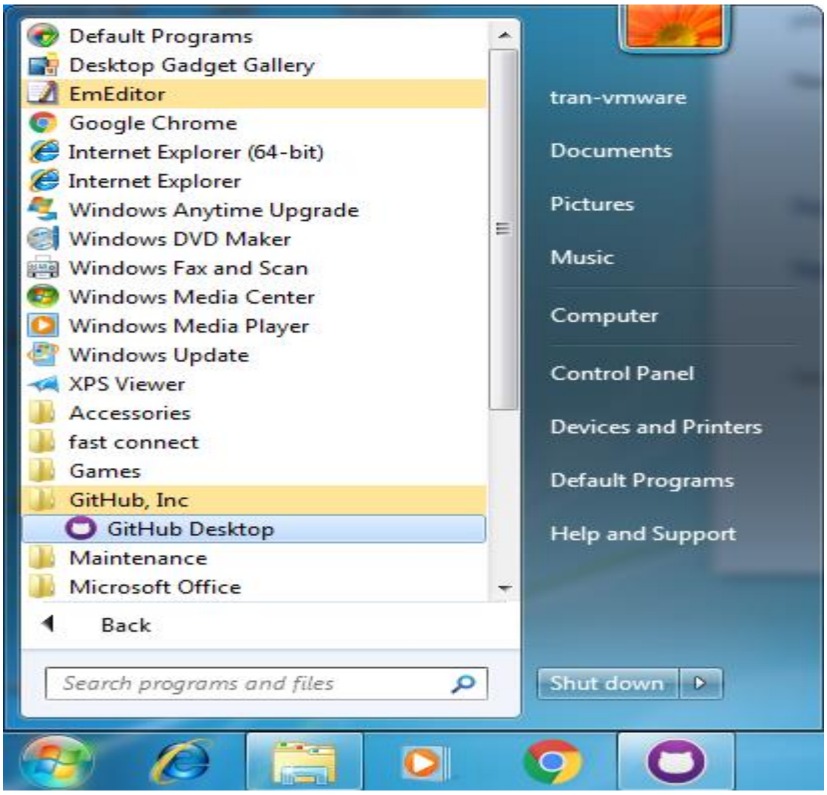


GitHubDesktopSetup.exe

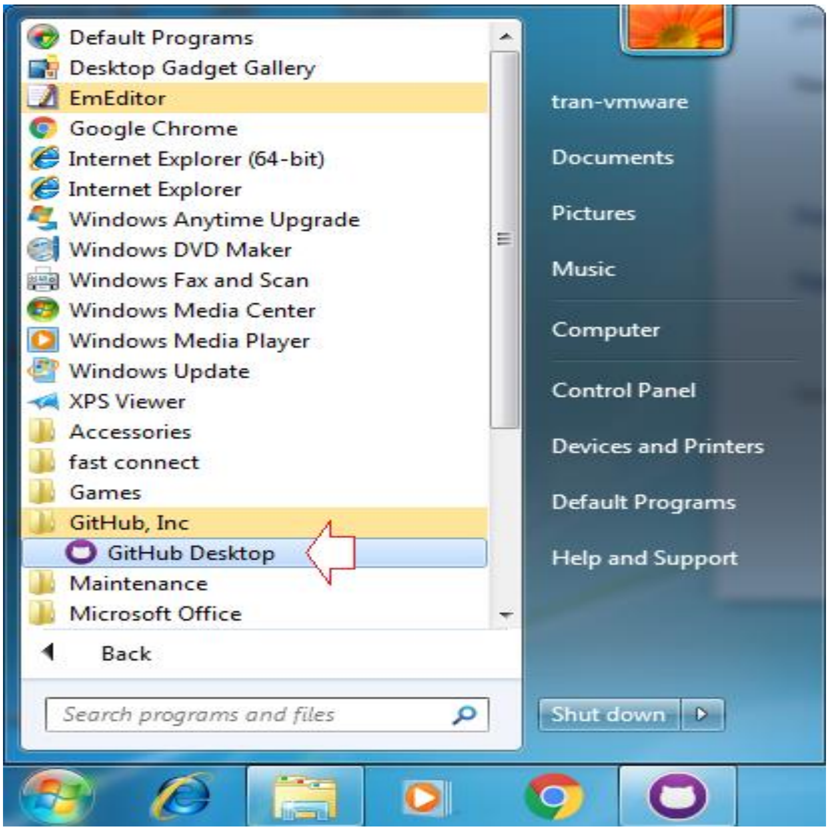
1 item   1 item selected 80.3 MB

- Install **Github Desktop**

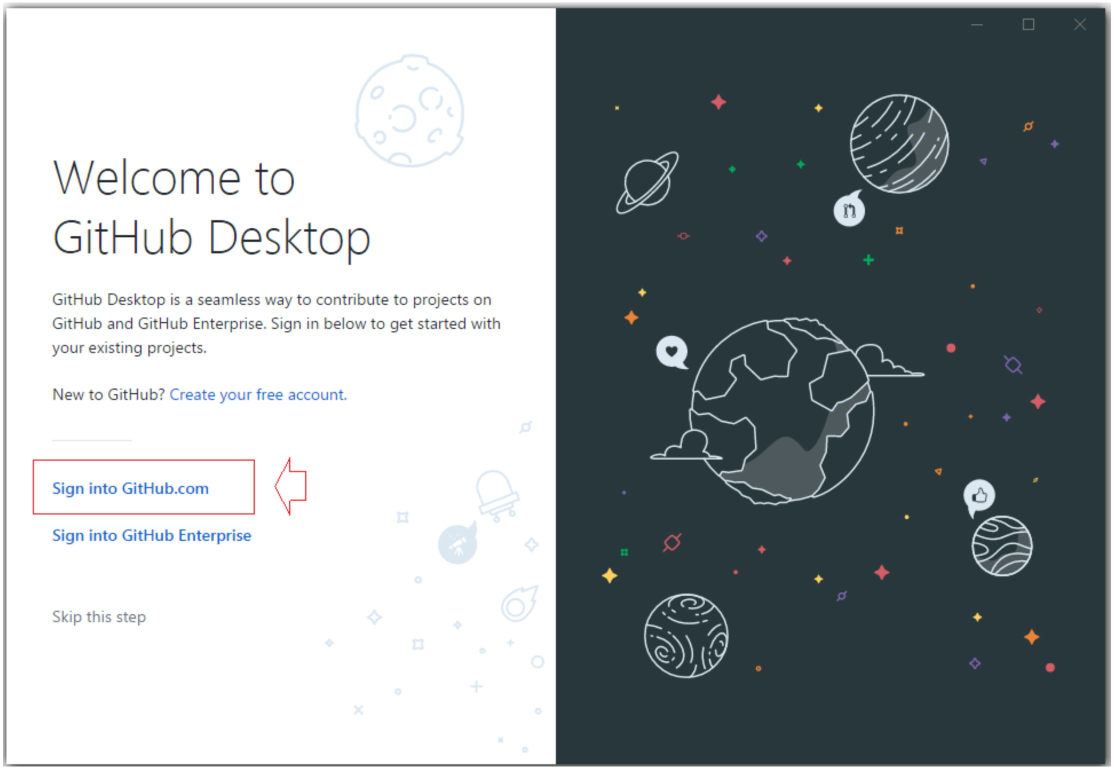
  Github Desktop has been successfully installed



- Run **Github Desktop**
-



- Log in on **Github Desktop** to connect to your **Github** account



Welcome to
GitHub Desktop

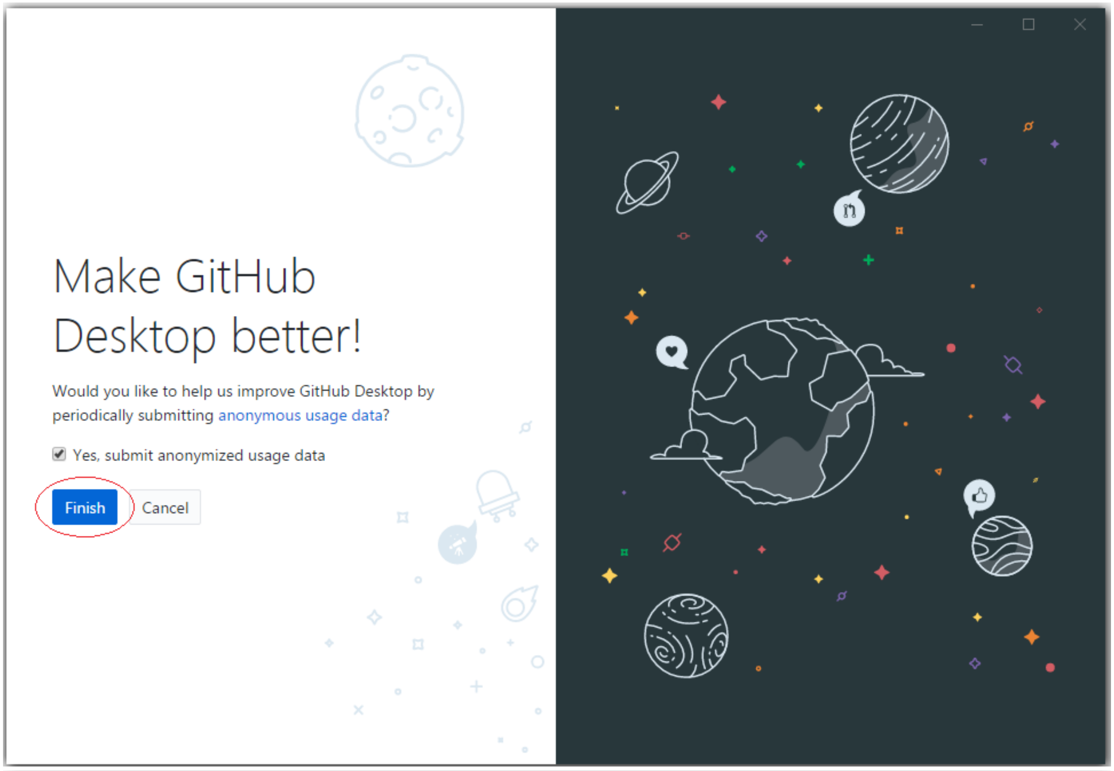
GitHub Desktop is a seamless way to contribute to projects on GitHub and GitHub Enterprise. Sign in below to get started with your existing projects.

New to GitHub? Create your free account.

Sign into GitHub.com

Sign into GitHub Enterprise

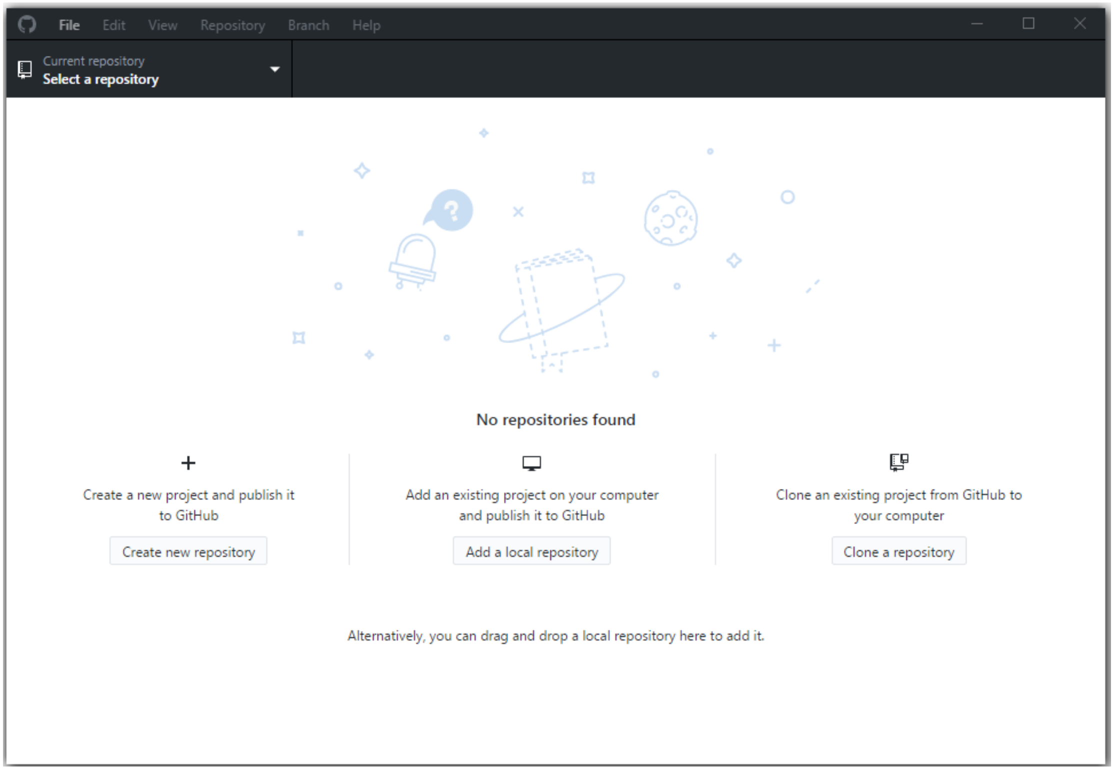
Skip this step



Make GitHub Desktop better!

Would you like to help us improve GitHub Desktop by periodically submitting anonymous usage data?

☑ Yes, submit anonymized usage data



No repositories found

Create a new project and publish it to GitHub

Add an existing project on your computer and publish it to GitHub

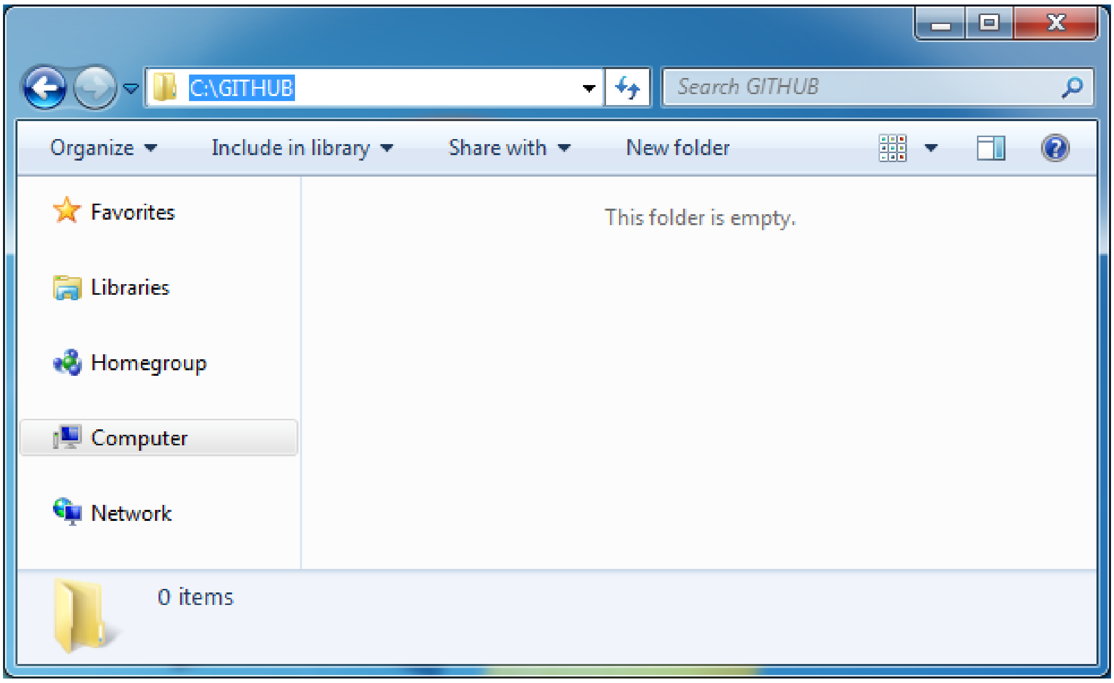
Clone an existing project from GitHub to your computer

Alternatively, you can drag and drop a local repository here to add it.

- Connect **GitHub** and **GitHub Desktop**

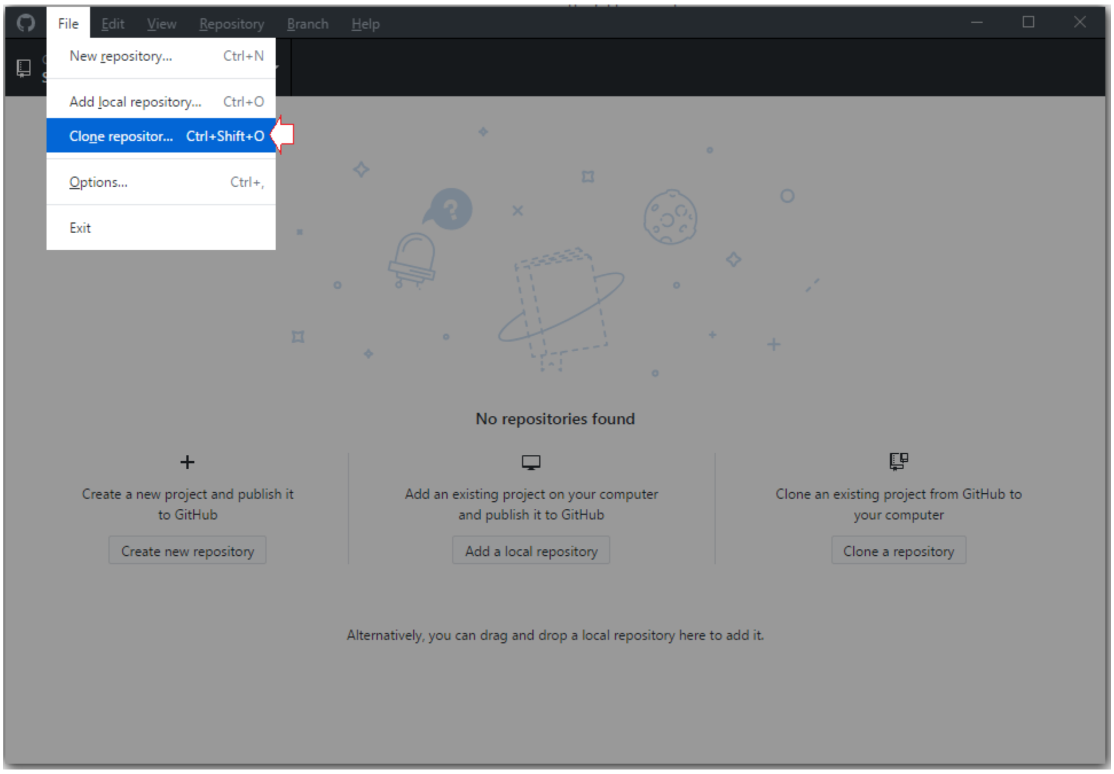
  First, select an empty directory as the location fot the local data

  ＊ C:/GITHUB



This folder is empty.

0 items



No repositories found

Create a new project and publish it to GitHub

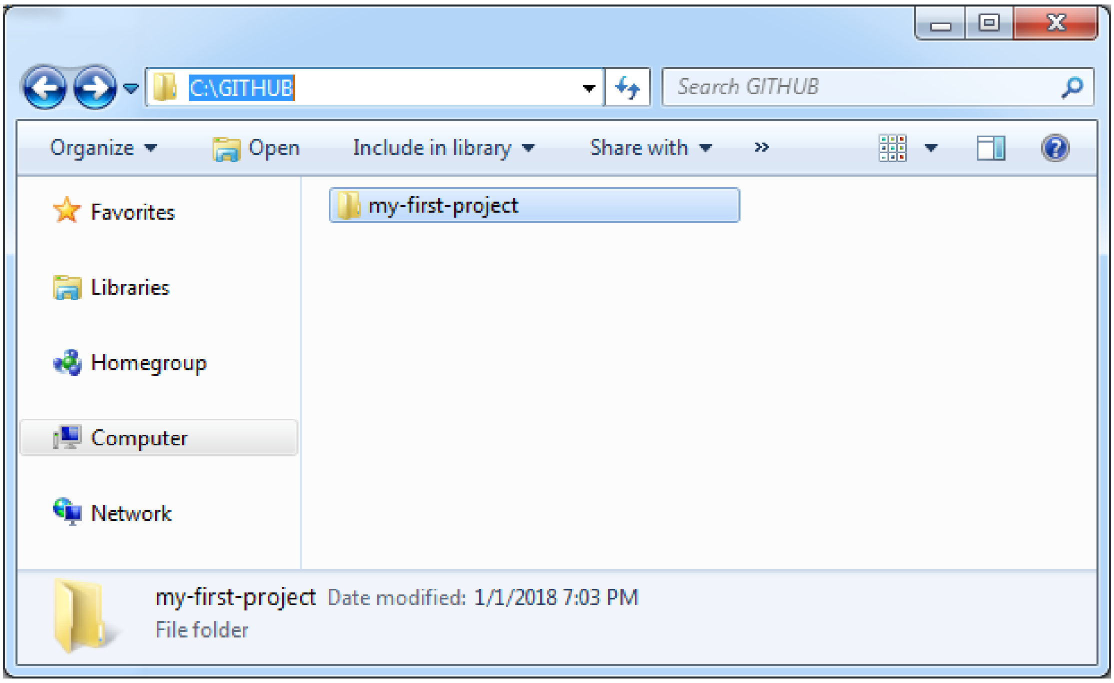
Add an existing project on your computer and publish it to GitHub

Clone an existing project from GitHub to your computer

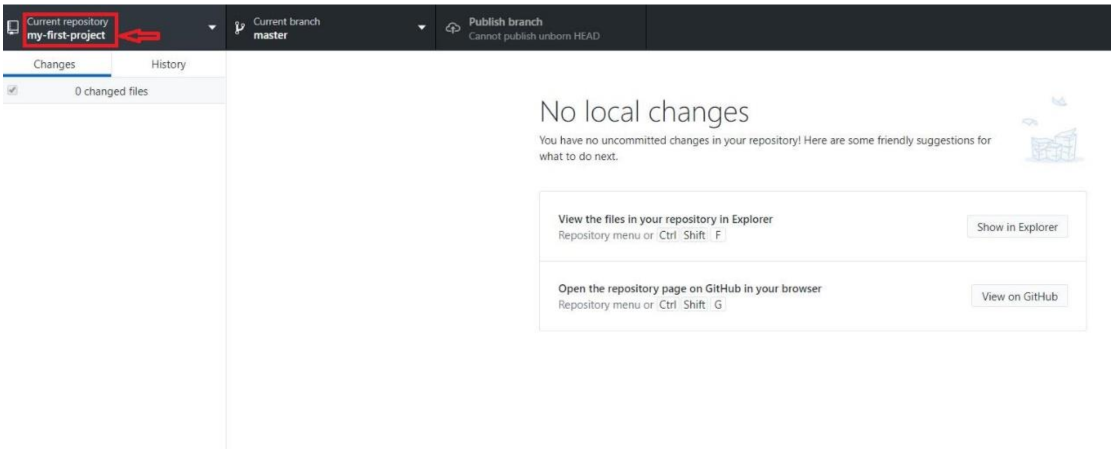
Alternatively, you can drag and drop a local repository here to add it.

✻ On GitHub Desktop, select a Repository yiu created on GitHub to clone (Create a copy) into a copy your local computer

✻ On GitHub Desktop you will see a local Repository hac been create.



my-first-project Date modified: 1/1/2018 7:03 PM
File folder



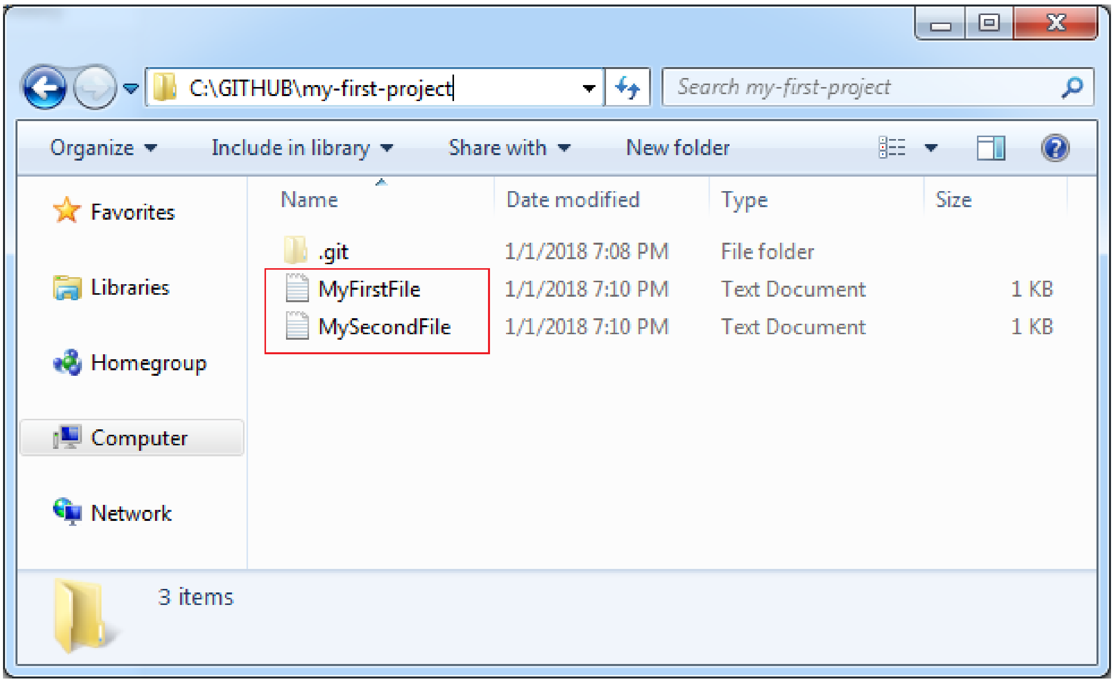
No local changes

You have no uncommitted changes in your repository! Here are some friendly suggestions for what to do next.

View the files in your repository in Explorer
Repository menu or Ctrl Shift F

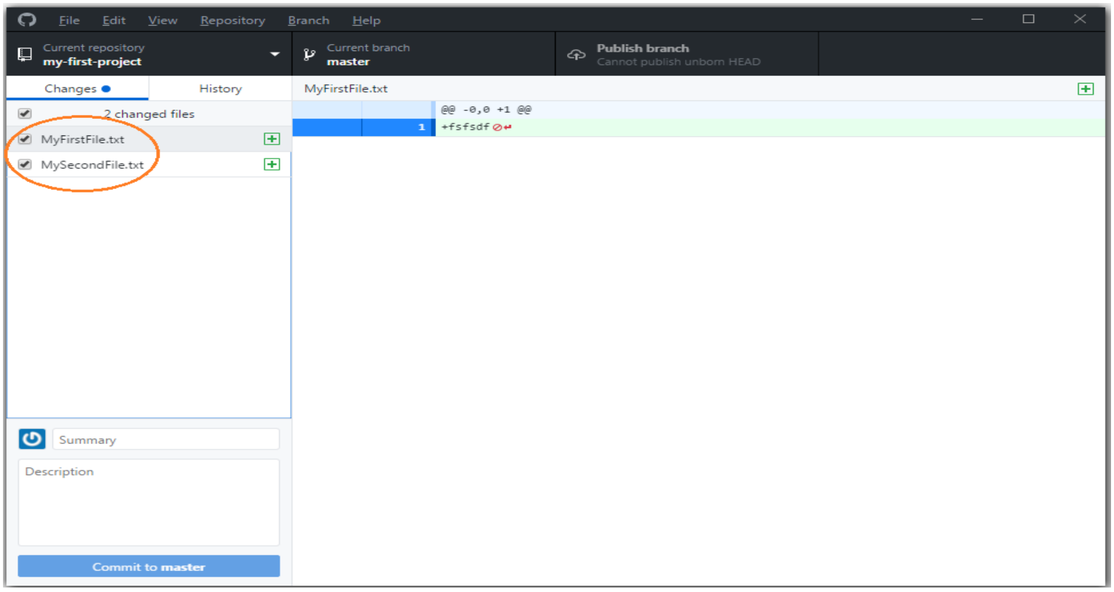
Open the repository page on GitHub in your browser
Repository menu or Ctrl Shift G

∗ Copy some data files into Local Repository



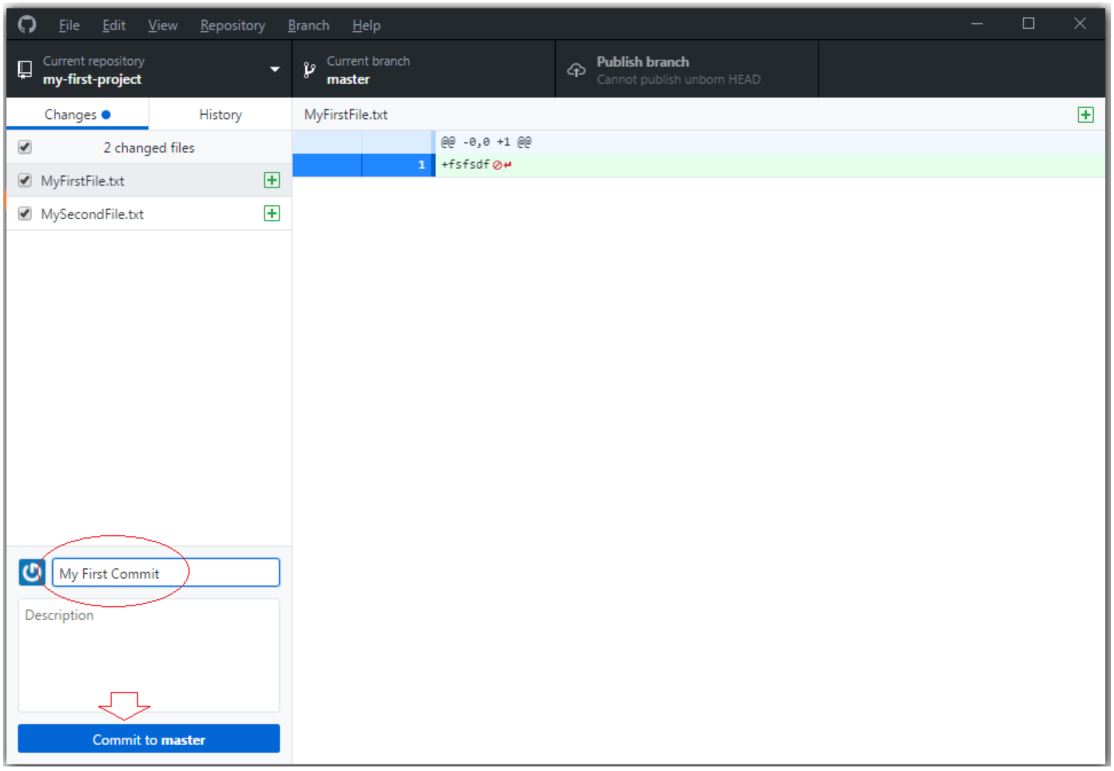
Name	Date modified	Type	Size
.git	1/1/2018 7:08 PM	File folder	
MyFirstFile	1/1/2018 7:10 PM	Text Document	1 KB
MySecondFile	1/1/2018 7:10 PM	Text Document	1 KB

＊ GitHub Desktop immediately recognizes the changes at Local Repository



```
@@ -0,0 +1 @@
1 +fsfsdf
```

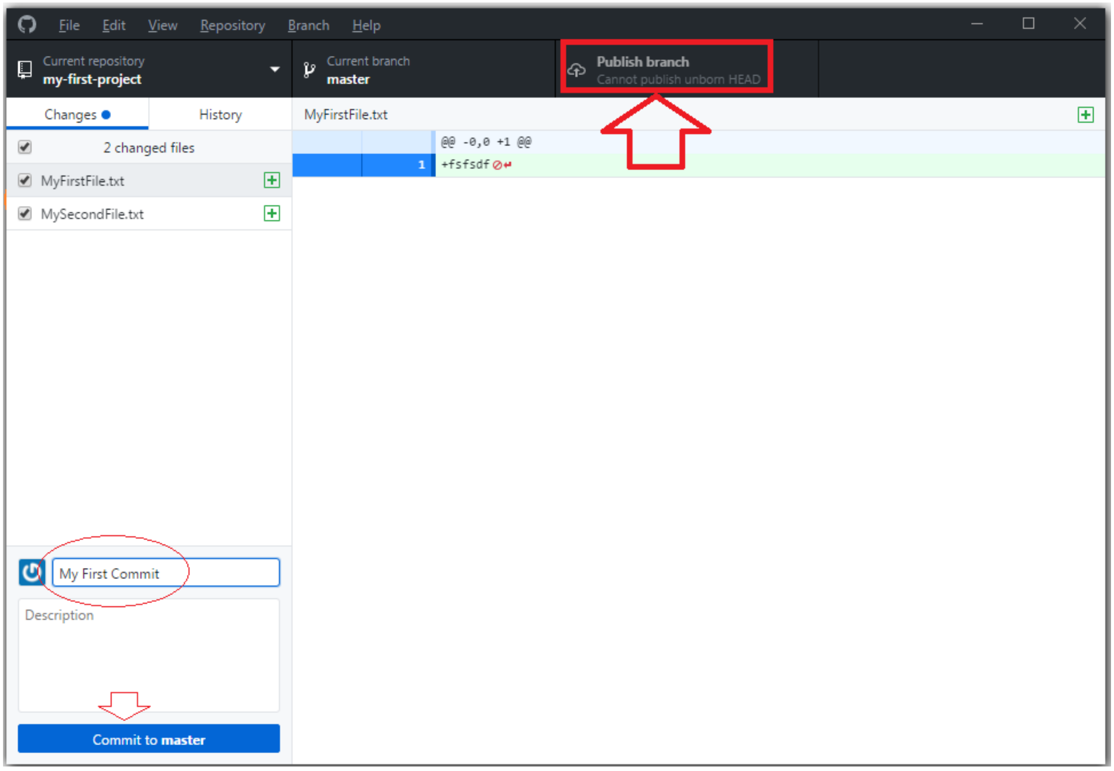
＊ Comment -> Commit



```
@@ -0,0 +1 @@
1 +fsfsdf
```

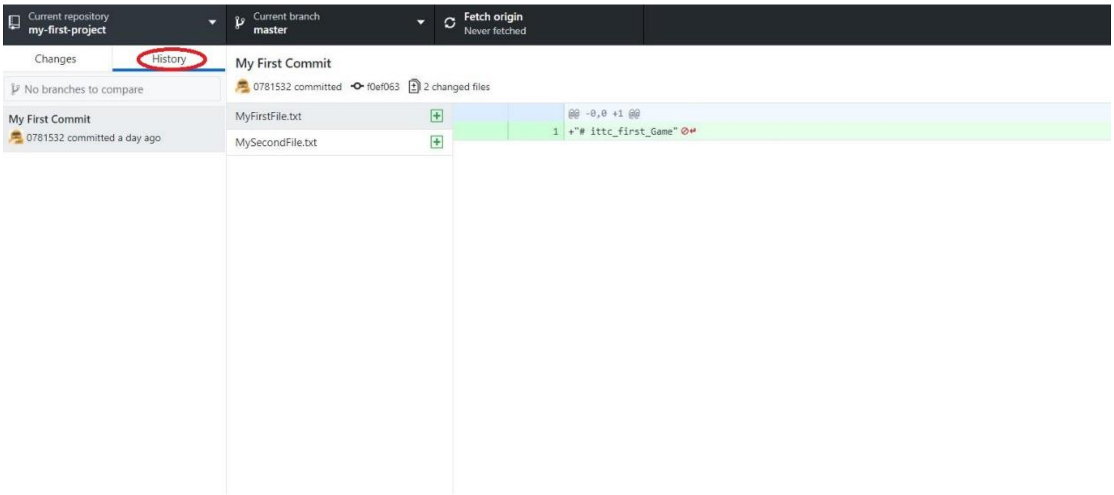
My First Commit

* **Publish branch** (Publish branch to GitHub)



```
@@ -0,0 +1 @@
1 +fsfsdf
```

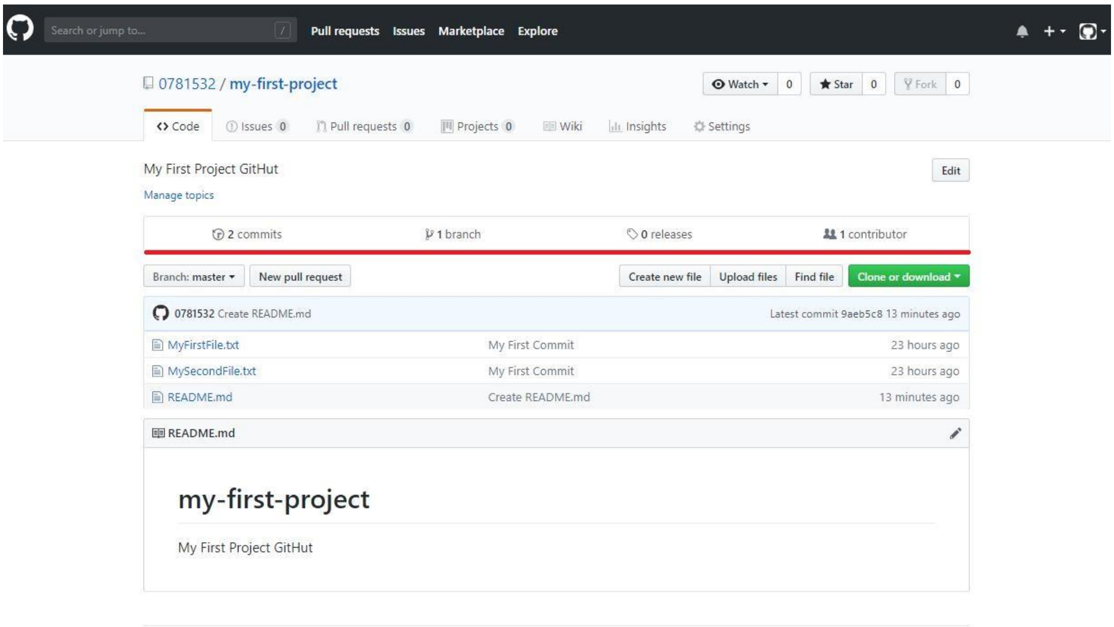
My First Commit



```
@@ -0,0 +1 @@
1 +"# ittc_first_Game"
```

Step 8: Use GIT for new folders >> git add README.md >>git remote add origin

- Connecting local server to online server (remote)
- Require to enter the username and password of GITHUB account:
- [new branch] master -> master
- Branch 'master' set up to track remote branch 'master' from 'origin'.



My First Project GitHut

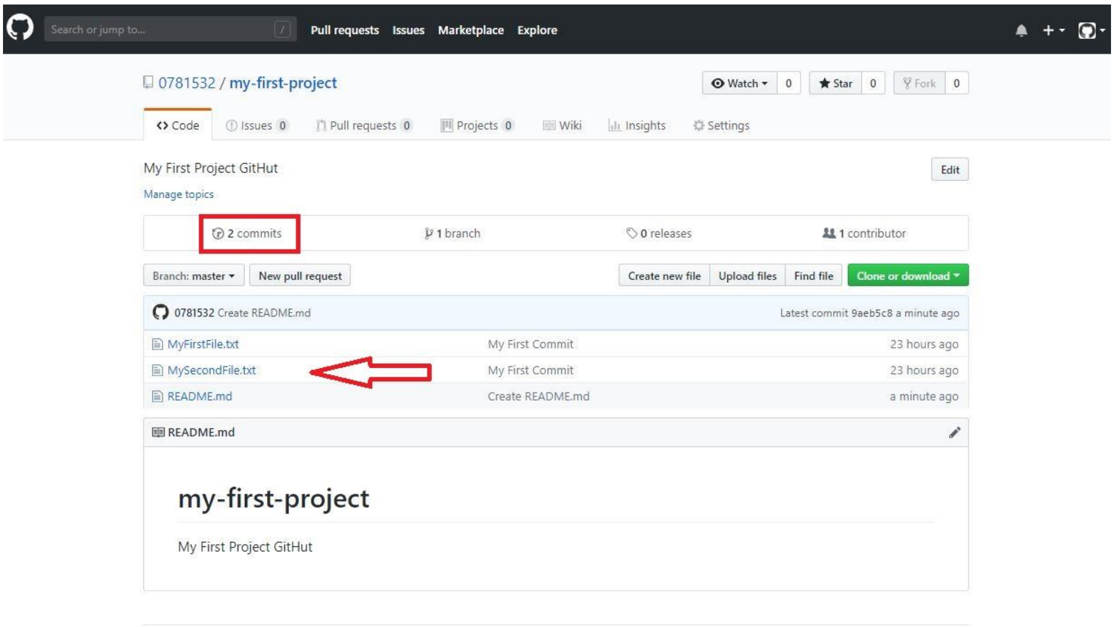
0781532 Create README.md — Latest commit 9aeb5c8 13 minutes ago

MyFirstFile.txt	My First Commit	23 hours ago
MySecondFile.txt	My First Commit	23 hours ago
README.md	Create README.md	13 minutes ago

my-first-project

My First Project GitHut

- **Check the log on the Github web site and the commit file**



My First Project GitHut

0781532 Create README.md — Latest commit 9aeb5c8 a minute ago

MyFirstFile.txt	My First Commit	23 hours ago
MySecondFile.txt	My First Commit	23 hours ago
README.md	Create README.md	a minute ago

README.md

my-first-project

My First Project GitHut

Step 9: Use GIT for an existing directory

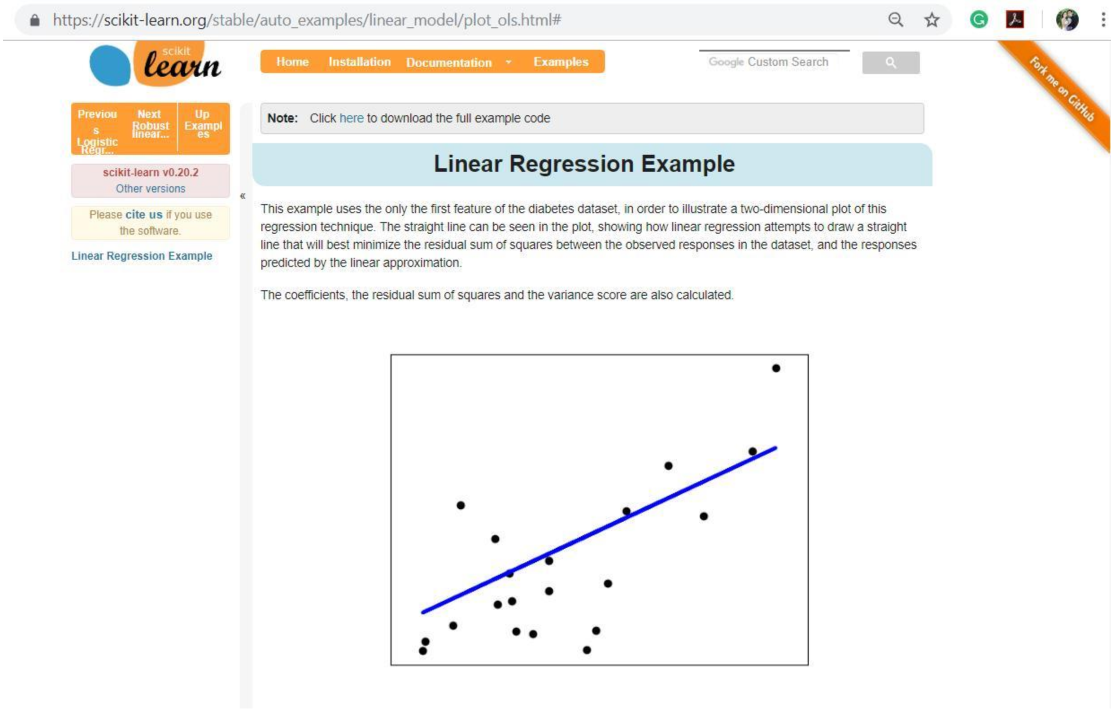
```
git remote add origin
https://github.com/0781532/my-first-project/tree/master
git push -u origin master
```

3. Download Linear Regression Example from scikit-learn and open it

https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html#

Step 1: click

https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html#


https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html#
scikit learn
Home
Installation
Documentation
Examples
Google Custom Search
Fork me on GitHub
Previous Logistic Regr...
Next Robust linear...
Up Examples
scikit-learn v0.20.2
Other versions
Please cite us if you use the software.
Linear Regression Example

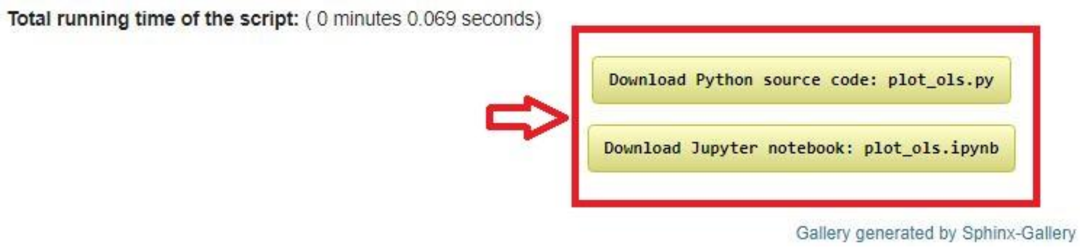
Note: Click here to download the full example code

Linear Regression Example

This example uses the only the first feature of the diabetes dataset, in order to illustrate a two-dimensional plot of this regression technique. The straight line can be seen in the plot, showing how linear regression attempts to draw a straight line that will best minimize the residual sum of squares between the observed responses in the dataset, and the responses predicted by the linear approximation.

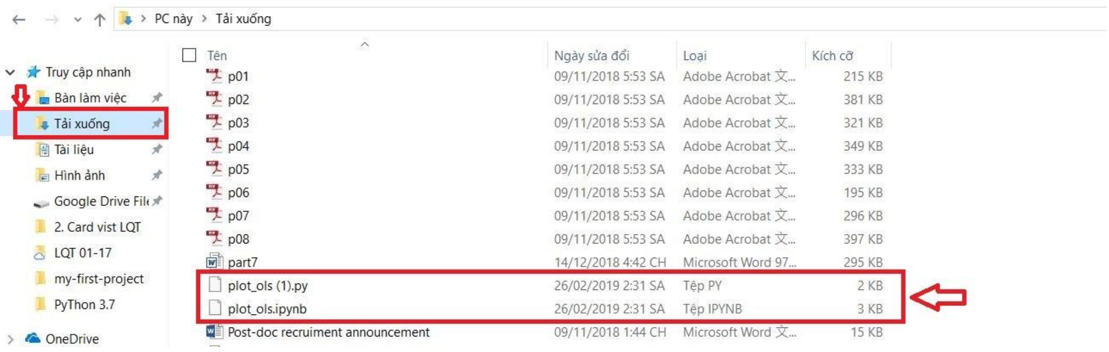
The coefficients, the residual sum of squares and the variance score are also calculated.

Step 2: Download



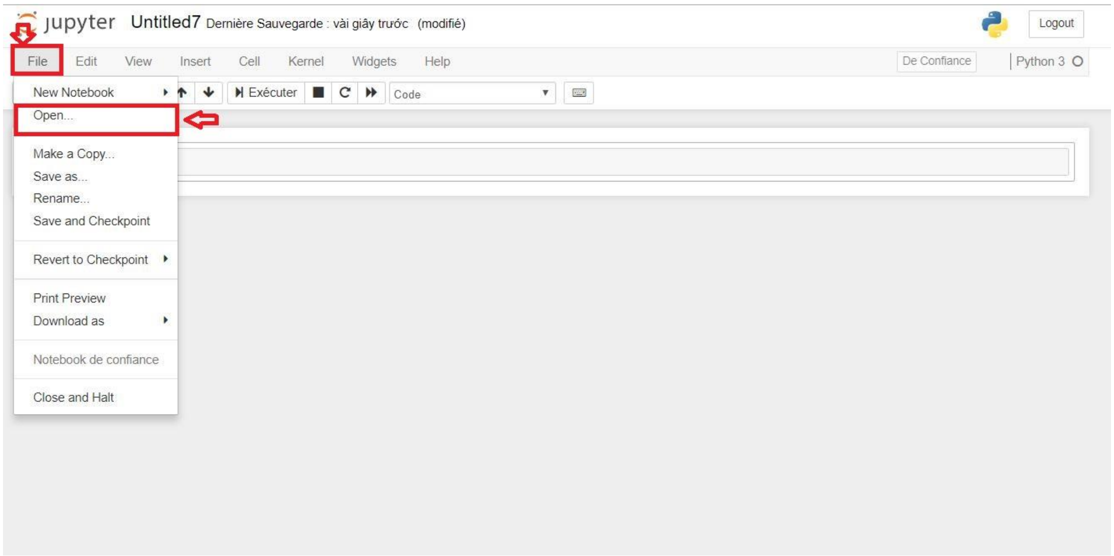
Total running time of the script: (0 minutes 0.069 seconds)

Download Python source code: plot_ols.py
Download Jupyter notebook: plot_ols.ipynb
Gallery generated by Sphinx-Gallery

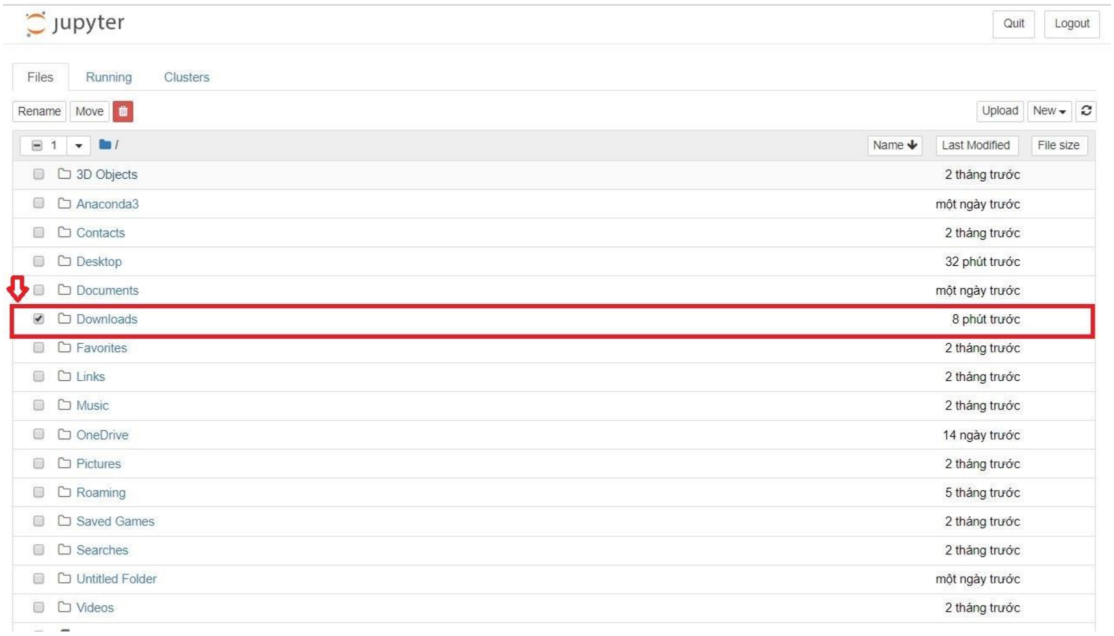


Tên	Ngày sửa đổi	Loại	Kích cỡ
p01	09/11/2018 5:53 SA	Adobe Acrobat 文...	215 KB
p02	09/11/2018 5:53 SA	Adobe Acrobat 文...	381 KB
p03	09/11/2018 5:53 SA	Adobe Acrobat 文...	321 KB
p04	09/11/2018 5:53 SA	Adobe Acrobat 文...	349 KB
p05	09/11/2018 5:53 SA	Adobe Acrobat 文...	333 KB
p06	09/11/2018 5:53 SA	Adobe Acrobat 文...	195 KB
p07	09/11/2018 5:53 SA	Adobe Acrobat 文...	296 KB
p08	09/11/2018 5:53 SA	Adobe Acrobat 文...	397 KB
part7	14/12/2018 4:42 CH	Microsoft Word 97...	295 KB
plot_ols (1).py	26/02/2019 2:31 SA	Tệp PY	2 KB
plot_ols.ipynb	26/02/2019 2:31 SA	Tệp IPYNB	3 KB
Post-doc recruiment announcement	09/11/2018 1:44 CH	Microsoft Word 文...	15 KB

Step 3: Click **Home** >> Launch **jupyter Notebook** >> choose **File** >> choose **Open**

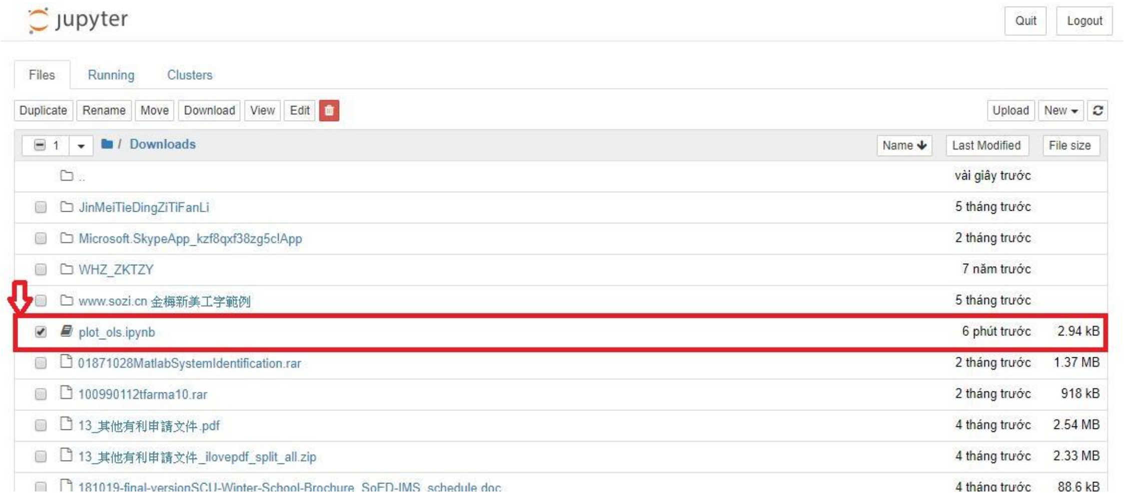


Choose **Download**



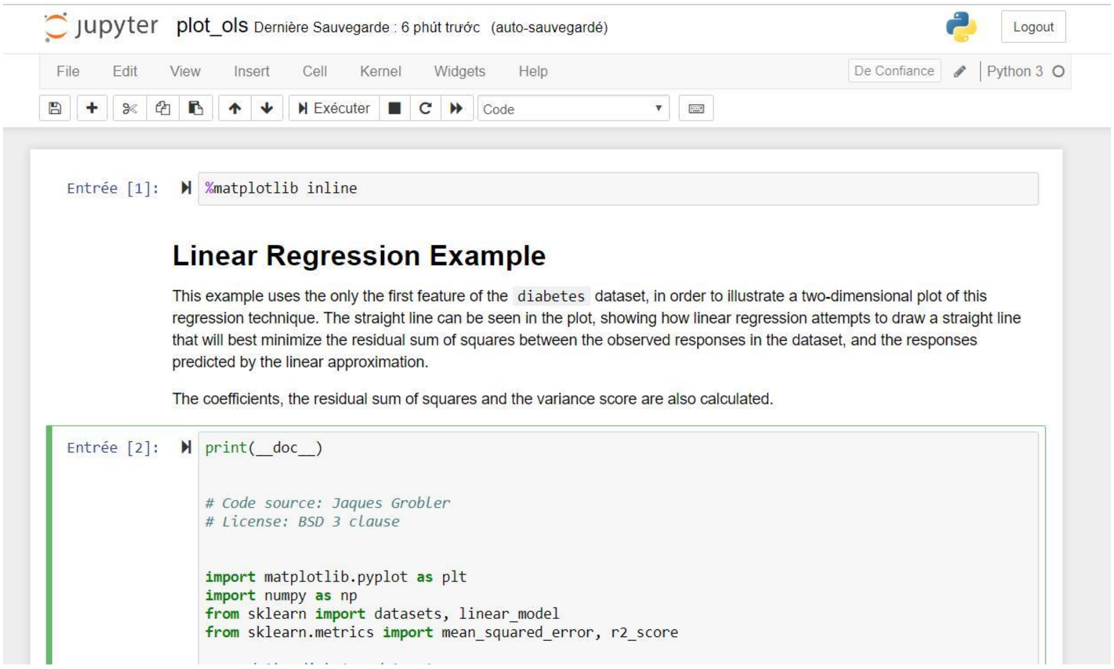
	Name	Last Modified	File size
□	3D Objects	2 tháng trước	
□	Anaconda3	một ngày trước	
□	Contacts	2 tháng trước	
□	Desktop	32 phút trước	
□	Documents	một ngày trước	
☑	Downloads	8 phút trước	
□	Favorites	2 tháng trước	
□	Links	2 tháng trước	
□	Music	2 tháng trước	
□	OneDrive	14 ngày trước	
□	Pictures	2 tháng trước	
□	Roaming	5 tháng trước	
□	Saved Games	2 tháng trước	
□	Searches	2 tháng trước	
□	Untitled Folder	một ngày trước	
□	Videos	2 tháng trước	

Choose file **plot_ols-checkpoint.ipynb**



	Name	Last Modified	File size
	..	vài giây trước	
□	JinMeiTieDingZiTiFanLi	5 tháng trước	
□	Microsoft.SkypeApp_kzf8qxf38zg5c!App	2 tháng trước	
□	WHZ_ZKTZY	7 năm trước	
□	www.sozi.cn 金梅新美工字範例	5 tháng trước	
☑	plot_ols.ipynb	6 phút trước	2.94 kB
□	01871028MatlabSystemIdentification.rar	2 tháng trước	1.37 MB
□	100990112tfarma10.rar	2 tháng trước	918 kB
□	13_其他有利申請文件.pdf	4 tháng trước	2.54 MB
□	13_其他有利申請文件_ilovepdf_split_all.zip	4 tháng trước	2.33 MB
□	181019-final-versionSCU-Winter-School-Brochure_SoED-IMS_schedule.doc	4 tháng trước	88.6 kB

Double click file **plot_ols-checkpoint.ipynb**



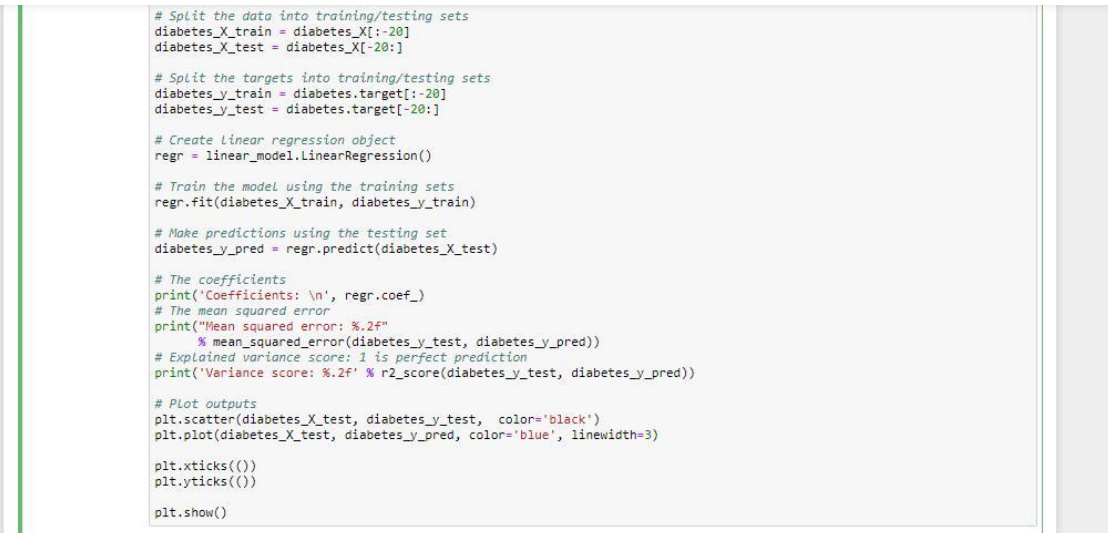
```
Entrée [1]:  %matplotlib inline
```

Linear Regression Example

This example uses the only the first feature of the `diabetes` dataset, in order to illustrate a two-dimensional plot of this regression technique. The straight line can be seen in the plot, showing how linear regression attempts to draw a straight line that will best minimize the residual sum of squares between the observed responses in the dataset, and the responses predicted by the linear approximation.

The coefficients, the residual sum of squares and the variance score are also calculated.

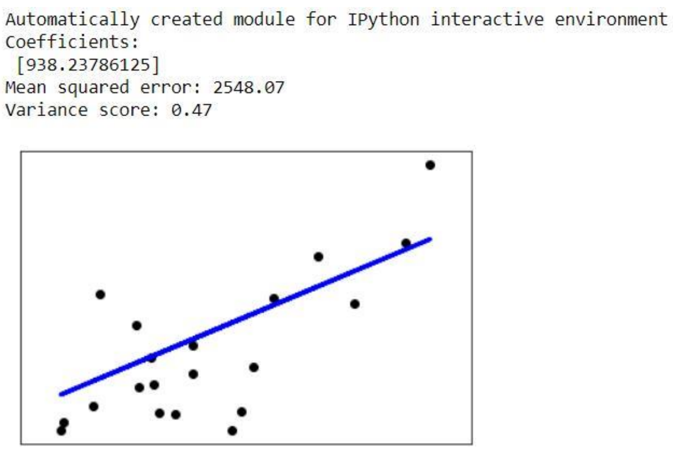
```
Entrée [2]:  print(__doc__)


             # Code source: Jaques Grobler
             # License: BSD 3 clause


             import matplotlib.pyplot as plt
             import numpy as np
             from sklearn import datasets, linear_model
             from sklearn.metrics import mean_squared_error, r2_score
```



```
# Split the data into training/testing sets
diabetes_X_train = diabetes_X[:-20]
diabetes_X_test = diabetes_X[-20:]

# Split the targets into training/testing sets
diabetes_y_train = diabetes.target[:-20]
diabetes_y_test = diabetes.target[-20:]

# Create linear regression object
regr = linear_model.LinearRegression()

# Train the model using the training sets
regr.fit(diabetes_X_train, diabetes_y_train)

# Make predictions using the testing set
diabetes_y_pred = regr.predict(diabetes_X_test)

# The coefficients
print('Coefficients: \n', regr.coef_)
# The mean squared error
print("Mean squared error: %.2f"
      % mean_squared_error(diabetes_y_test, diabetes_y_pred))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % r2_score(diabetes_y_test, diabetes_y_pred))

# Plot outputs
plt.scatter(diabetes_X_test, diabetes_y_test,  color='black')
plt.plot(diabetes_X_test, diabetes_y_pred, color='blue', linewidth=3)

plt.xticks(())
plt.yticks(())

plt.show()
```

Step 4: Run

- **Result**



```
Automatically created module for IPython interactive environment
Coefficients:
 [938.23786125]
Mean squared error: 2548.07
Variance score: 0.47
```