PYTHON CIRCLE (/)   ☰
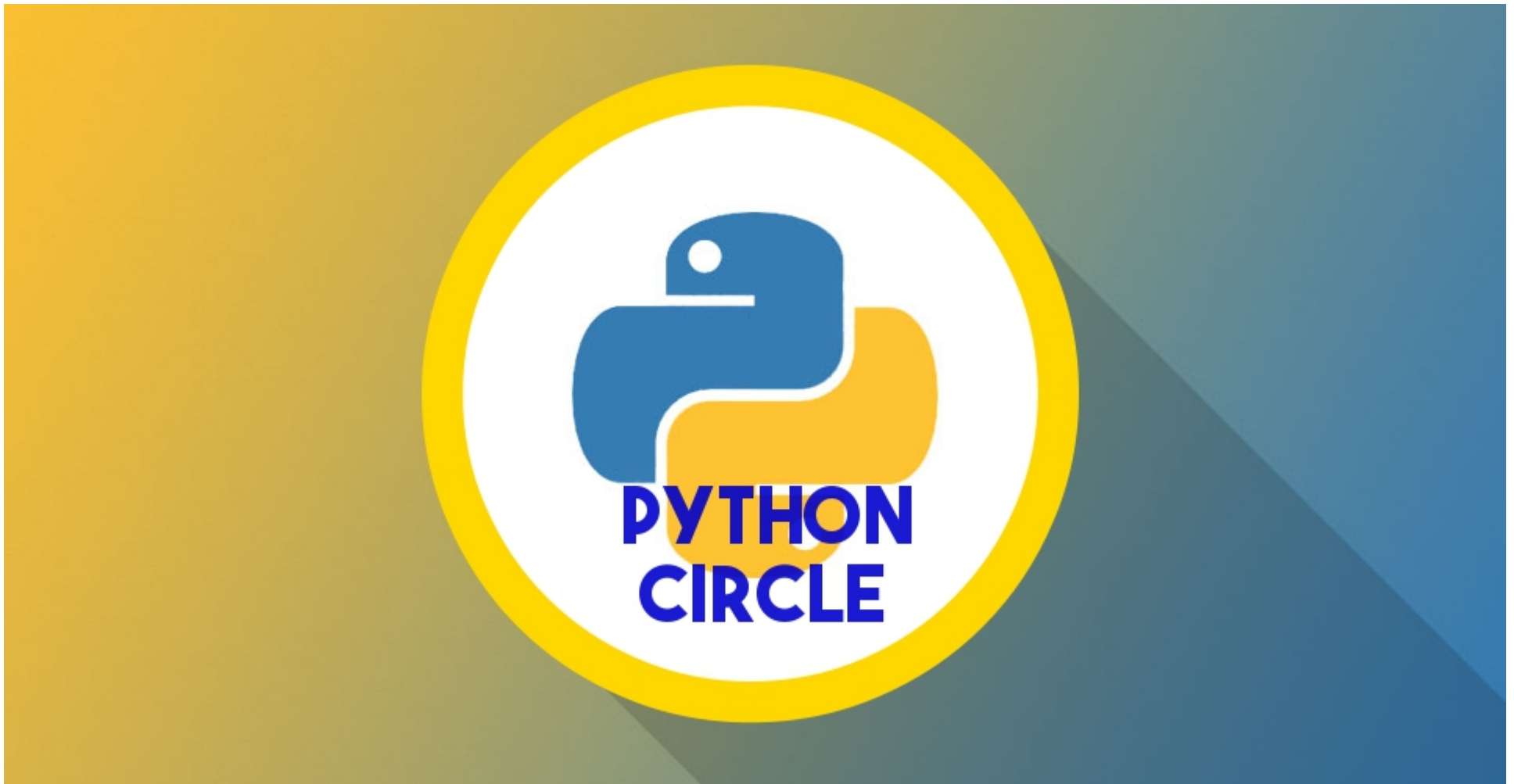
Virtual Environment in Python - A Pocket Guide (/post/404/virtual-environment-in-python-a-pocket-guide/)

🏷 virtual-env (/tag/virtual-env/)   💬 0   👁 2853

In almost every article, we recommended the use of virtual environment for developing any Python or Django project.

In this article, we will briefly cover the virtual environment in python, installation and usage.

What is a Virtual Environment:

Virtual environment is an isolated python environment which can be created using virtualenv python tool. This virtual environment contains all the packages that a python package would require.

Python project running in virtual environment does not use the system wide installed python package.

Installing Virtual Environment:

We can install virtual environment using pip.

```
pip install virtualenv
```

You can confirm the installation by running below command.

```
virtualenv --version
```

```
rana@Brahma ~$ virtualenv --version
15.1.0
```

Using virtual environment:

Create a virtual environment:

```
virtualenv project_1_venv
```

This will create a virtual environment with name `project_1_env` and install python binaries in this folder. This will use the system's default python' version.

To create virtual environment with specific python version use below command.

```
virtualenv -p /usr/bin/python2.7 venv2
```

or

```
virtualenv -p /usr/bin/python3 venv3
```

To work inside a virtual environment, we need to activate it.

```
source virtual_env_folder/bin/activate
```

Now you may install python packages inside this isolated environment using pip install command.

Once you are done working with your project, you may deactivate the virtual environment by `deactivate` command.

Deleting virtual environment:

To delete a virtual environment simply remove the directory.

```
rm -rf virtual_env_directory
```

Freezing the requirements:

When you deploy your project on any other server like PythonAnyWhere (https://www.pythoncircle.com/post/18/how-to-host-django-app-on-pythonanywhere-for-free/), you need to replicate the virtual environment.

You can not simply copy the folder as it will break the things. Also this is not a good idea as the size of the folder will be large.

We will first list all the packages installed in virtual environment and dump in a file.

```
pip freeze > requirement.txt
```

Now commit this file along with your code. On production server, clone the code repository and create a virtual environment. Once virtual environment is activated, install all the dependencies using the pip install command below.

```
pip install -r requirement.txt
```

This will create the exact copy of virtual environment you were working on development server.

Pipenv:

Pipenv is a dependency manager for python projects. It is similar to Symfony's composer and node.js's npm.

Its a higher lever tool than pip. We recommend to use pipenv for managing your virtual environments.

With pipenv, we need not to manage `requirement.txt` file separately.

As per pienv readthedocs:

> It automatically creates and manages a virtualenv for your projects, as well as adds/removes packages from your Pipfile as you install/uninstall packages. It also generates the ever–important Pipfile.lock, which is used to produce deterministic builds.

Install pipenv using pip.

```
pip install pipenv
```

Creating virtual environment using pipenv:

Go to your project directory. Install python packages using pipenv command.

```
rana@Brahma: ~$ cd  /tmp
rana@Brahma: tmp$ mkdir django_project_1
rana@Brahma: tmp$ cd django_project_1/
rana@Brahma: django_project_1$ pipenv install django requests
Creating a virtualenv for this project…
?Using base prefix '/usr'
New python executable in /home/rana/.local/share/virtualenvs/django_project_1-NloTdiFm/bin/python3
Also creating executable in /home/rana/.local/share/virtualenvs/django_project_1-NloTdiFm/bin/python
Installing setuptools, pip, wheel...done.

Virtualenv location: /home/rana/.local/share/virtualenvs/django_project_1-NloTdiFm
Creating a Pipfile for this project…
Installing django…
Collecting django
  Using cached Django-2.0-py3-none-any.whl
Collecting pytz (from django)
  Using cached pytz-2017.3-py2.py3-none-any.whl
Installing collected packages: pytz, django
Successfully installed django-2.0 pytz-2017.3

Adding django to Pipfile's [packages]…
Installing requests…
Collecting requests
  Using cached requests-2.18.4-py2.py3-none-any.whl
Collecting chardet<3.1.0,>=3.0.2 (from requests)
  Using cached chardet-3.0.4-py2.py3-none-any.whl
Collecting certifi>=2017.4.17 (from requests)
  Using cached certifi-2017.11.5-py2.py3-none-any.whl
Collecting idna<2.7,>=2.5 (from requests)
  Using cached idna-2.6-py2.py3-none-any.whl
Collecting urllib3<1.23,>=1.21.1 (from requests)
  Using cached urllib3-1.22-py2.py3-none-any.whl
Installing collected packages: chardet, certifi, idna, urllib3, requests
Successfully installed certifi-2017.11.5 chardet-3.0.4 idna-2.6 requests-2.18.4 urllib3-1.22
```
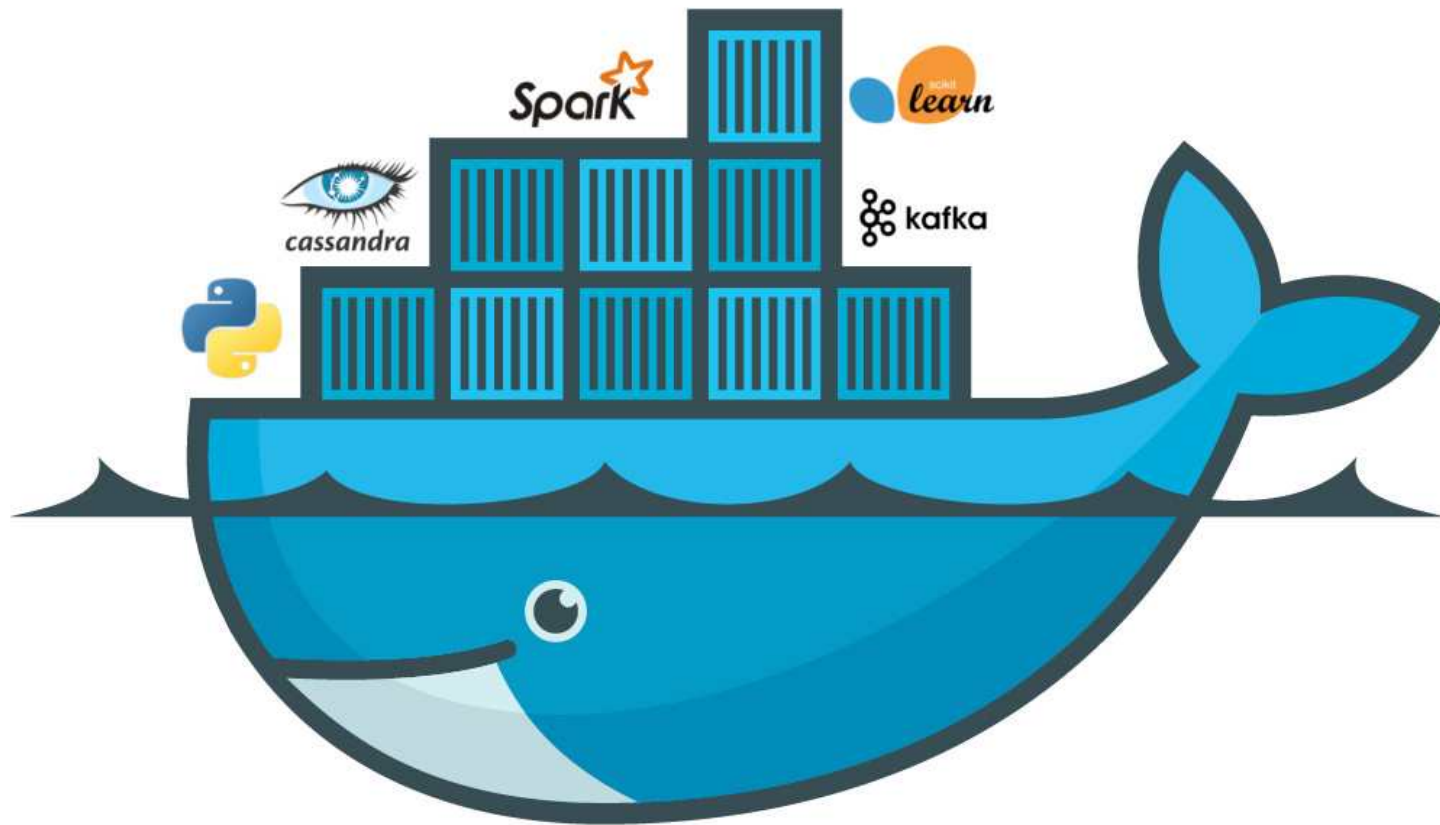
```
Adding requests to Pipfile's [packages]…
  PS: You have excellent taste! ?
```

🏷 virtual-env (/tag/virtual-env/)   💬 0   👁 2853

Related Articles:

Using Docker instead of Virtual Environment for Django app development (/post/488/using-docker-instead-of-virtual-environment-for-django-app-development/)

Using docker instead of virtual environment for Django app development, starting with Dockerization of your Django application. Django with Docker....

Read Full Article (/post/488/using-docker-instead-of-virtual-environment-for-django-app-development/)

---

0 thoughts on 'Virtual Environment In Python - A Pocket Guide'

Leave a comment:

Type your comment here. For code use pastebin link. Limit 500 chars.
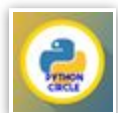
Your email please.

Your Name

Submit

*All Fields are mandatory. **Email Id will not be published publicly.

SUBSCRIBE

Please subscribe to get the latest articles in your mailbox.

Your Email ID Please

Subscribe

Python Circle
1,850 likes

Like Page   PYTHON CIRCLE   Learn More

Recent Posts:

- 5 lesser used Django template tags (/post/694/5-lesser-used-django-template-tags/)

- Solving Python Error- KeyError: 'key_name' (/post/693/solving-python-error-keyerror-key_name/)

- Intellectual property Law and Coding (/post/692/intellectual-property-law-and-coding/)

- Python Script 17: Setting bing image of the day as desktop wallpaper (/post/691/python-script-17-setting-bing-image-of-the-day-as-desktop-wallpaper/)

- Django Template Fiddle Launched !!!! (/post/690/django-template-fiddle-launched/)

- Python Script 16: Generating word cloud image of a text using python (/post/689/python-script-16-generating-word-cloud-image-of-a-text-using-python/)

- Preventing cross-site scripting attack on your Django website (/post/688/preventing-cross-site-scripting-attack-on-your-django-website/)

- How to generate ATOM/RSS feed for Django website (/post/687/how-to-generate-atomrss-feed-for-django-website/)

- How to create sitemap of Django website (/post/686/how-to-create-sitemap-of-django-website/)

- For loop in Django template (/post/685/for-loop-in-django-template/)

- How to add Favicon to Django websites (/post/684/how-to-add-favicon-to-django-websites/)

- Scraping data of 2019 Indian General Election using Python Request and BeautifulSoup and analyzing it (/post/683/scraping-data-of-2019-indian-general-election-using-python-request-and-beautifulsoup-and-analyzing-it/)

- How to display PDF in browser in Django instead of downloading it (/post/682/how-to-display-pdf-in-browser-in-django-instead-of-downloading-it/)

- Solving Python Error - UnboundLocalError: local variable 'x' referenced before assignment (/post/680/solving-python-error-unboundlocalerror-local-variable-x-referenced-before-assignment/)

- Python Script 15: Creating a port scanner in 8 lines of python (/post/679/python-script-15-creating-a-port-scanner-in-8-lines-of-python/)

- Python Script 14: Scraping news headlines using python beautifulsoup (/post/678/python-script-14-scraping-news-headlines-using-python-beautifulsoup/)

- How to download large csv files in Django (/post/677/how-to-download-large-csv-files-in-django/)

- Text based snake and ladder game in python (/post/676/text-based-snake-and-ladder-game-in-python/)

- Logging databases changes in Django Application (/post/675/logging-databases-changes-in-django-application/)

- Python Script 13: Generating ascii code from Image (/post/674/python-script-13-generating-ascii-code-from-image/)

- Python Frequently Asked Question 1: What can I do in Python? (/post/672/python-frequently-asked-question-1-what-can-i-do-in-python/)

- How to start with Python Programming - A beginner's guide (/post/671/how-to-start-with-python-programming-a-beginners-guide/)

- Python program to find whether a given year is leap year or not (/post/670/python-program-to-find-whether-a-given-year-is-leap-year-or-not/)

- Python Django Project Idea for beginners (/post/669/python-django-project-idea-for-beginners/)

- Uploading a file to FTP server using Python (/post/668/uploading-a-file-to-ftp-server-using-python/)

- Print statement in Python vs other programming languages (/post/667/print-statement-in-python-vs-other-programming-languages/)

- Automating Facebook page posts using python script (/post/666/automating-facebook-page-posts-using-python-script/)

- try .. except .. else .. in python with example (/post/664/try-except-else-in-python-with-example/)

- Python Script 12: Drawing Indian National Flag Tricolor using Python Turtle (/post/662/python-script-12-drawing-indian-national-flag-tricolor-using-python-turtle/)

- Python Script 11: Drawing Flag of United States of America using Python Turtle (/post/661/python-script-11-drawing-flag-of-united-states-of-america-using-python-turtle/)

- Solving Django Error: TemplateDoesNotExist at /app_name/ (/post/660/solving-django-error-templatedoesnotexist-at-app_name/)

- Adding Email Subscription Feature in Django Application (/post/657/adding-email-subscription-feature-in-django-application/)

- Using PostgreSQL Database with Python (/post/656/using-postgresql-database-with-python/)

- Programming On Raspberry Pi With Python: Activate LED and Buzzer on Motion Detection (/post/654/programming-on-raspberry-pi-with-python-activate-led-and-buzzer-on-motion-detection/)

- Programming on Raspberry Pi with Python: Controlling LED (/post/652/programming-on-raspberry-pi-with-python-controlling-led/)

- Programming on Raspberry Pi with Python: Sending IP address on Telegram channel on Raspberry Pi reboot (/post/651/programming-on-raspberry-pi-with-python-sending-ip-address-on-telegram-channel-on-raspberry-pi-reboot/)

- Programming on Raspberry Pi with Python: WIFI and SSH configuration (/post/650/programming-on-raspberry-pi-with-python-wifi-and-ssh-configuration/)

- Programming on Raspberry Pi with Python: Raspberry Pi Setup (/post/649/programming-on-raspberry-pi-with-python-raspberry-pi-setup/)

- Iterator and Generators in Python: Explained with example (/post/648/iterator-and-generators-in-python-explained-with-example/)

- Top 5 Python Books (/post/646/top-5-python-books/)

- Encryption-Decryption in Python Django (/post/641/encryption-decryption-in-python-django/)

- Sending Emails Using Python and Gmail (/post/628/sending-emails-using-python-and-gmail/)

- How to Track Email Opens Sent From Django App (/post/626/how-to-track-email-opens-sent-from-django-app/)

- Python Tip 1: Accessing localhost Django webserver over the Internet (/post/618/python-tip-1-accessing-localhost-django-webserver-over-the-internet/)

- 5 common mistakes made by beginner python programmers (/post/602/5-common-mistakes-made-by-beginner-python-programmers/)

- How to upload and process the Excel file in Django (/post/591/how-to-upload-and-process-the-excel-file-in-django/)

- Creating sitemap of Dynamic URLs in your Django Application (/post/584/creating-sitemap-of-dynamic-urls-in-your-django-application/)

- Adding Robots.txt file to Django Application (/post/578/adding-robotstxt-file-to-django-application/)

- Python Script 3: Validate, format and Beautify JSON string Using Python (/post/576/python-script-3-validate-format-and-beautify-json-string-using-python/)

- Displaying custom 404 error (page not found) page in Django 2.0 (/post/564/displaying-custom-404-error-page-not-found-page-in-django-20/)

Tags:

404 (/tag/404/)   500 (/tag/500/)   AJAX (/tag/ajax/)   API (/tag/api/)   AUTHENTICATION (/tag/authentication/)

AUTOMATION (/tag/automation/)   AWS (/tag/aws/)   BACKUP (/tag/backup/)   BEAUTIFULSOUP (/tag/beautifulsoup/)   BOOK (/tag/book/)

BREADBOARD (/tag/breadboard/)   CAPTCHA (/tag/captcha/)   CELERY (/tag/celery/)   COMMANDS (/tag/commands/)   CSV (/tag/csv/)

DATA (/tag/data/)   DATABASE (/tag/database/)   DECRYPTION (/tag/decryption/)   DJANGO (/tag/django/)

DJANGO APP (/tag/django%20app/)   DJANGO ERROR (/tag/django%20error/)   DOCKER (/tag/docker/)   DOWNLOAD (/tag/download/)

EC2 (/tag/ec2/)   ELASTIC SEARCH (/tag/elastic%20search/)   EMAIL (/tag/email/)   ENCRYPTION (/tag/encryption/)   ERROR (/tag/error/)

ERROR PAGE (/tag/error%20page/)   EXCEL (/tag/excel/)   EXCEPTION (/tag/exception/)   FACEBOOK (/tag/facebook/)   FAQ (/tag/faq/)

FEEDS (/tag/feeds/)   FOR LOOP (/tag/for%20loop/)   FORM (/tag/form/)   FREE EMAILS (/tag/free%20emails/)   FTP (/tag/ftp/)

GAME (/tag/game/)   GENERATOR (/tag/generator/)   GMAIL (/tag/gmail/)   GPIO (/tag/gpio/)   GRAPH API (/tag/graph%20api/)

HARDWARE (/tag/hardware/)   HOSTING (/tag/hosting/)   IMAGE (/tag/image/)   INSTAGRAM (/tag/instagram/)

IP ADDRESS (/tag/ip%20address/)   IP LAW (/tag/ip%20law/)   ITERATOR (/tag/iterator/)   JSON (/tag/json/)   KIBANA (/tag/kibana/)

LED (/tag/led/)   LOGGING (/tag/logging/)   MISTAKE (/tag/mistake/)   MONGODB (/tag/mongodb/)   PACKAGE (/tag/package/)

PAYMENT GATWAY (/tag/payment%20gatway/)　　PDF (/tag/pdf/)　　PITFALLS (/tag/pitfalls/)　　PROGRAM (/tag/program/)

PROJECT (/tag/project/)　　PROXY (/tag/proxy/)　　PYTHON BITES (/tag/python%20bites/)　　PYTHON TIPS (/tag/python%20tips/)

PYTHONANYWHERE (/tag/pythonanywhere/)　　RABBITMQ (/tag/rabbitmq/)　　RASPBERRYPI (/tag/raspberrypi/)　　REQUESTS (/tag/requests/)

RESPONSE (/tag/response/)　　SCRAPING (/tag/scraping/)　　SCRAPY (/tag/scrapy/)　　SCRIPT (/tag/script/)　　SECURITY (/tag/security/)

SENSOR (/tag/sensor/)　　SEO (/tag/seo/)　　SERVER (/tag/server/)　　SETUP (/tag/setup/)　　SIGNALS (/tag/signals/)

SITEMAP (/tag/sitemap/)　　SOCKET (/tag/socket/)　　SSH (/tag/ssh/)　　TAG (/tag/tag/)　　TELEGRAM (/tag/telegram/)

TEMPLATES (/tag/templates/)　　TIPS (/tag/tips/)　　TKINTER (/tag/tkinter/)　　TOR (/tag/tor/)　　TRACKING (/tag/tracking/)

TRY CATCH (/tag/try%20catch/)　　TURTLE (/tag/turtle/)　　UPLOAD (/tag/upload/)　　VIRTUAL ENV (/tag/virtual%20env/)　　WIFI (/tag/wifi/)

WORD CLOUD (/tag/word%20cloud/)