

---

# ColabHost Documentation

*Release 0.1.9*

**Puneetha Pai**

**Sep 15, 2020**



---

## Contents:

---

<b>1 ColabHost</b>	<b>1</b>
<b>2 Installation</b>	<b>3</b>
<b>3 Usage</b>	<b>5</b>
<b>4 colab_host</b>	<b>7</b>
<b>5 Contributing</b>	<b>9</b>
<b>6 Credits</b>	<b>13</b>
<b>7 History</b>	<b>15</b>
<b>8 ColabHost</b>	<b>17</b>
<b>9 Usage</b>	<b>19</b>
<b>10 Indices and tables</b>	<b>21</b>
<b>Python Module Index</b>	<b>23</b>
<b>Index</b>	<b>25</b>



Host any python application in colab or kaggle notebook environment. Inspired from [Abhishek Thakur](#), and his work on [colab code](#) to host VScode IDE in any notebook environment using pyngrok.

- Free software: MIT license
- Documentation: <https://colab-host.readthedocs.io>.

## 1.1 Features

Since colabcode already hosts VScode IDE this package extends the idea to host following IDE and python applications:

- Jupyter Notebook
- Jupyter Lab
- Flask and Gunicorn applications
- FastAPI and Uvicorn applications

## 1.2 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.



### 2.1 Stable release

To install ColabHost, run this command in your terminal:

```
$ pip install colab_host
```

This is the preferred method to install ColabHost, as it will always install the most recent stable release.

If you don't have [pip](#) installed, this [Python installation guide](#) can guide you through the process.

### 2.2 From sources

The sources for ColabHost can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/PuneethaPai/colab_host
```

Or download the [tarball](#):

```
$ curl -OJL https://github.com/PuneethaPai/colab_host/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```





## CHAPTER 3

---

### Usage

---

To start a simple HTTP File Server:

```
from colab_host import SimpleHttpServer  
  
SimpleHttpServer(port=1000)
```

To start Jupyter Notebook in colab:

```
from colab_host import JupyterNotebook  
  
JupyterNotebook(port=1000)
```

To start Jupyter Lab in colab:

```
from colab_host import JupyterLab  
  
JupyterLab(port=1000)
```

To start Flas Application in colab:

```
from colab_host import FlaskApp  
  
FlaskApp(  
    port=1000,  
    app="main:app",  
    git_url="https://github.com/PuneethaPai/colab_host_flask_demo.git",  
    requirements_file="requirements.txt"  
)
```

To start Uvicorn App in colab:

```
from colab_host import UvicornApp  
  
UvicornApp(  

```

(continues on next page)

(continued from previous page)

```
port=1000,  
app="main:app",  
git_url="https://github.com/PuneethaPai/colab_host_uvicorn_demo.git",  
requirements_file="requirements.txt"  
)
```

## 4.1 colab\_host package

### 4.1.1 Submodules

### 4.1.2 colab\_host.colab\_host module

```
class colab_host.colab_host.FlaskApp (port: int = 1000, app='main:app',
                                     git_url='https://github.com/PuneethaPai/colab_host_flask_demo',
                                     requirements_file: str = 'requirements.txt')
```

Bases: `colab_host.colab_host.Host`

Class to expose python Flask or Gunicorn application.

port : int, optional app : str, optional

Definition of your python gunicorn app. (Defaults to “*main:app*”).

**git\_url** [str, optional] Git URL to clone your repo containing application. (Defaults to “*https://github.com/PuneethaPai/colab\_host\_flask\_demo*”).

**requirements\_file: str, optional** Name of file in repo *git\_url* containing requirements for hosting the application. (Defaults to “*requirements.txt*”).

```
class colab_host.colab_host.Host (port: int = 1000, requirements: list = None, git_url: str =
                                     None)
```

Bases: `object`

Base class for hosting any python application.

Given *port* number it will expose the port to internet. Given *requirements* will install them using *pip install*. Given *git\_url* it will clone the repo for you.

port : int, optional requirements : List[str] or str, optional

List[str]: list of package requirements for hosting. str: requirements file path to install requirements from.

git\_url : str, optional

```
class colab_host.colab_host.JupyterLab (port: int = 1000, requirements: list = ['jupyter-  
lab'])
```

Bases: `colab_host.colab_host.Host`

Class to expose Jupyter Lab IDE on browser.

port : int, optional requirements : List[str], optional

Defaults to `["jupyterlab"]` and you can include other packages to include with this. For example notebook extension, theme, etc

```
class colab_host.colab_host.JupyterNotebook (port: int = 1000, requirements: list = ['note-  
book'])
```

Bases: `colab_host.colab_host.Host`

Class to expose Jupyter Notebook IDE on browser.

port : int, optional requirements : List[str], optional

Defaults to `["notebook"]` and you can include other packages to include with this. For example notebook extension, theme, etc

```
class colab_host.colab_host.SimpleHttpServer (port: int = 1000)
```

Bases: `colab_host.colab_host.Host`

Class to expose simple file server application.

port : int, optional

```
class colab_host.colab_host.UvicornApp (port: int = 1000, app='main:app',  
git_url='https://github.com/PuneethaPai/colab_host_flask_demo',  
requirements_file: str = 'requirements.txt')
```

Bases: `colab_host.colab_host.Host`

Class to expose python FastApi or Uvicorn application.

port : int, optional app : str, optional

Definition of your python gunicorn app. (Defaults to `"main:app"`).

**git\_url** [str, optional] Git URL to clone your repo containing application. (Defaults to `"https://github.com/PuneethaPai/colab_host_uvicorn_demo"`).

**requirements\_file**: str, optional Name of file in repo `git_url` containing requirements for hosting the application. (Defaults to `"requirements.txt"`).

### 4.1.3 Module contents

Top-level package for ColabHost.

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

## 5.1 Types of Contributions

### 5.1.1 Report Bugs

Report bugs at [https://github.com/PuneethaPai/colab\\_host/issues](https://github.com/PuneethaPai/colab_host/issues).

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

### 5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

### 5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

### 5.1.4 Write Documentation

ColabHost could always use more documentation, whether as part of the official ColabHost docs, in docstrings, or even on the web in blog posts, articles, and such.

### 5.1.5 Submit Feedback

The best way to send feedback is to file an issue at [https://github.com/PuneethaPai/colab\\_host/issues](https://github.com/PuneethaPai/colab_host/issues).

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## 5.2 Get Started!

Ready to contribute? Here's how to set up *colab\_host* for local development.

1. Fork the *colab\_host* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/colab_host.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv colab_host
$ cd colab_host/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 colab_host tests
$ python setup.py test or pytest
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

## 5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 3.5, 3.6, 3.7 and 3.8, and for PyPy. Check [https://travis-ci.com/PuneethaPai/colab\\_host/pull\\_requests](https://travis-ci.com/PuneethaPai/colab_host/pull_requests) and make sure that the tests pass for all supported Python versions.

## 5.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_colab_host
```

## 5.5 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bump2version patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.





### 6.1 Development Lead

- Puneetha Pai <puneethapai29@gmail.com>

### 6.2 Contributors

None yet. Why not be the first?



#### 7.1 0.1.0 (2020-09-10)

- First release on PyPI.



Host any python application in colab or kaggle notebook environment. Inspired from [Abhishek Thakur](#), and his work on [colab code](#) to host VScode IDE in any notebook environment using pyngrok.

- Free software: MIT license
- Documentation: <https://colab-host.readthedocs.io>.

## 8.1 Features

Since colabcode already hosts VScode IDE this package extends the idea to host following IDE and python applications:

- Jupyter Notebook
- Jupyter Lab
- Flask and Gunicorn applications
- FastAPI and Uvicorn applications

## 8.2 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.



## CHAPTER 9

---

### Usage

---

To start a simple HTTP File Server:

```
from colab_host import SimpleHttpServer  
  
SimpleHttpServer(port=1000)
```

To start Jupyter Notebook in colab:

```
from colab_host import JupyterNotebook  
  
JupyterNotebook(port=1000)
```

To start Jupyter Lab in colab:

```
from colab_host import JupyterLab  
  
JupyterLab(port=1000)
```

To start Flas Application in colab:

```
from colab_host import FlaskApp  
  
FlaskApp(  
    port=1000,  
    app="main:app",  
    git_url="https://github.com/PuneethaPai/colab_host_flask_demo.git",  
    requirements_file="requirements.txt"  
)
```

To start Uvicorn App in colab:

```
from colab_host import UvicornApp  
  
UvicornApp(  

```

(continues on next page)

(continued from previous page)

```
port=1000,  
app="main:app",  
git_url="https://github.com/PuneethaPai/colab_host_uvicorn_demo.git",  
requirements_file="requirements.txt"  
)
```



## CHAPTER 10

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



### C

`colab_host`, [8](#)

`colab_host.colab_host`, [7](#)



## C

`colab_host` (*module*), 8

`colab_host.colab_host` (*module*), 7

## F

`FlaskApp` (*class in colab\_host.colab\_host*), 7

## H

`Host` (*class in colab\_host.colab\_host*), 7

## J

`JupyterLab` (*class in colab\_host.colab\_host*), 8

`JupyterNotebook` (*class in colab\_host.colab\_host*),  
8

## S

`SimpleHttpServer` (*class in colab\_host.colab\_host*),  
8

## U

`UvicornApp` (*class in colab\_host.colab\_host*), 8