Python Object-Oriented Program with Libraries

Unit 6: Flask Web App Development

CHAPTER 1: OVERVIEW – BRING UP FLASK

DR. ERIC CHOU

IEEE SENIOR MEMBER

Objectives

- Understand what is Flask
- Create Web Applications with Flask for both localhost testing version and deployed web-site version.

Overview of Flask and Server-side Programming

LECTURE 1

What is Flask?

- Lightweight WSGI web application framework
  - WSGI: Web Server Gateway Interface — specification for server-app communications
- A Web framework is a collection of packages or modules to write Web applications or services
- without having to handle such low-level details as protocols, sockets or process/thread management. Popular frameworks: Django, web2py, Flask, Bottle, CherryPy, Raspbian includes Flask, don’t need to install it.

First Hello Flask App

LECTURE 2

Step 1: create a site

Demo Program: app.py

```
from flask import Flask
app = Flask(__name__)    # create Flask as app

@app.route("/")    # root directory of the site
def home():         # return HTML template
    return "<h1>Hello Flask</h1>"

if __name__ == "__main__":  # run the app
    app.run()
```

Testing on Local Host

LECTURE 2

Installing IIS

To install IIS:

1.In Windows, access the Control Panel and click **Add or Remove Programs**.

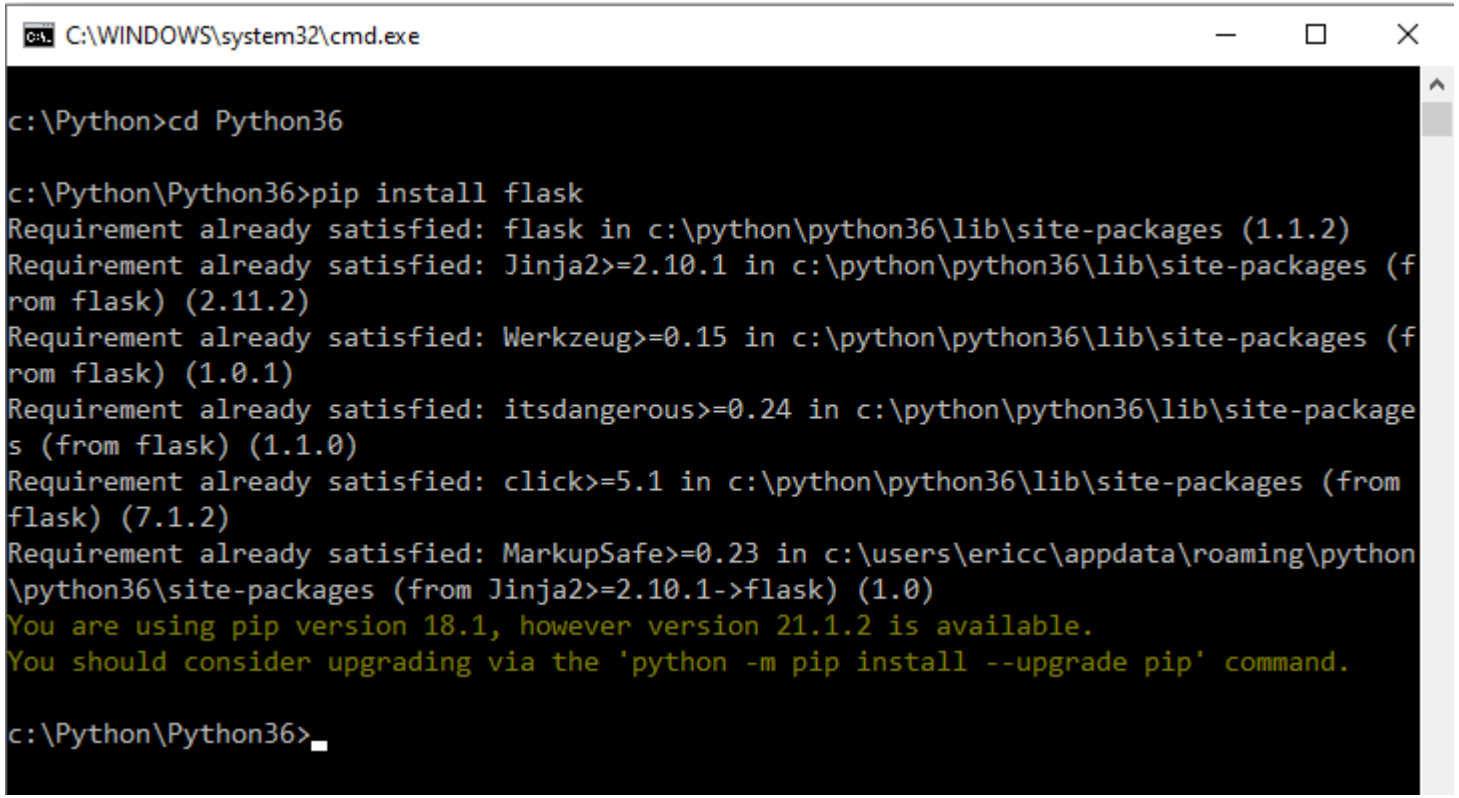
2.In the Add or Remove Programs window, click **Add/Remove Windows Components**.

3.Select the **Internet Information Services (IIS)** check box, click **Next**, then click **Finish**.

To learn how to use IIS, you can view the documentation at http://localhost/iishelp/iis/misc/default.asp.

Watch Video:
https://youtu.be/bJrOASXslwU

pip install flask



Step 1: create a site

Demo Program: app.py as the root program

```
from flask import Flask
app = Flask(__name__)    # create Flask as app

@app.route("/")    # root directory of the site
def home():         # return HTML template
    return "<h1>Hello Flask</h1>"

if __name__ == "__main__":
    app.run()
```

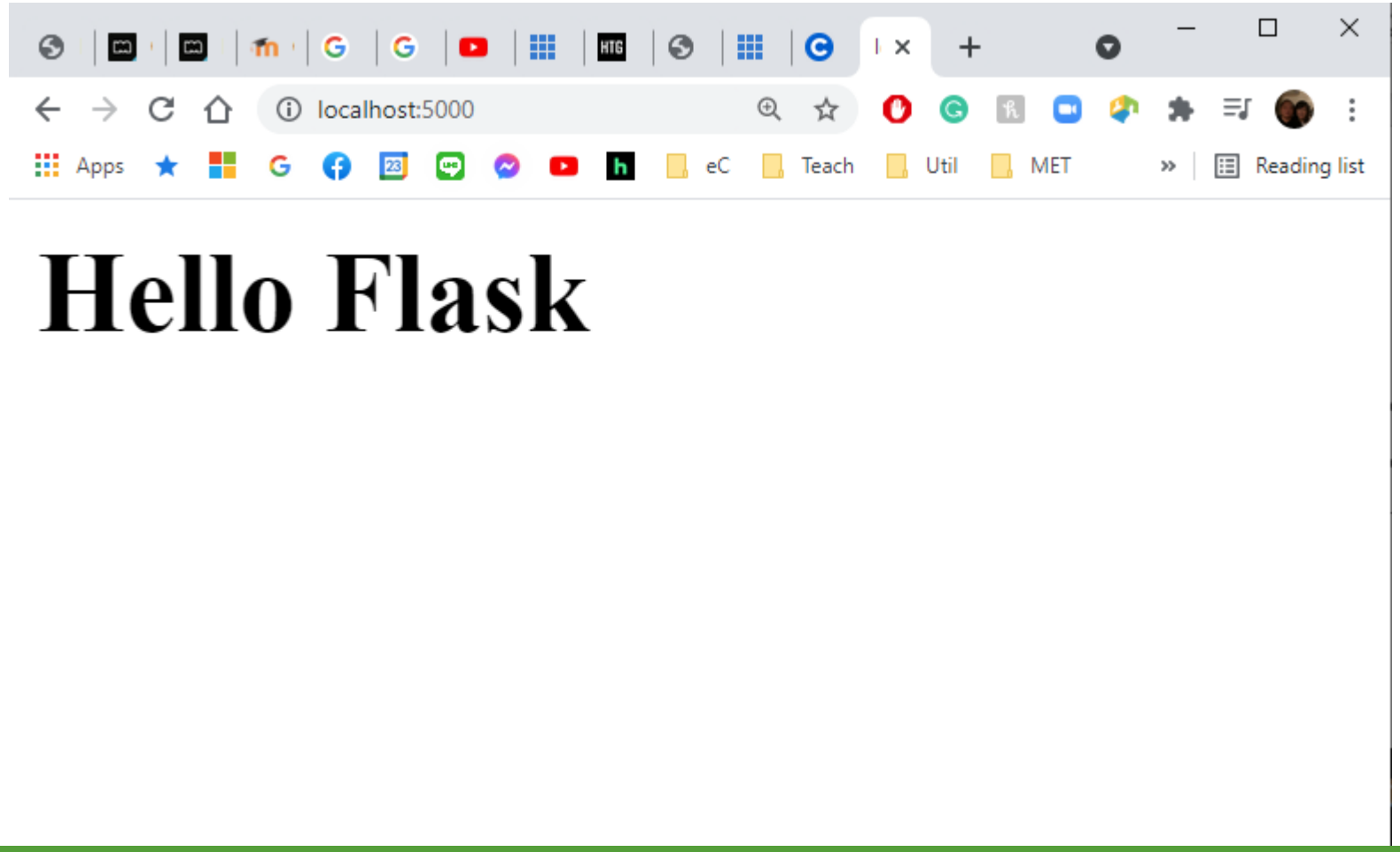
Step 2: Run the server program

- Under command line.

```
C:\Eric_Chou\Python Course\Python Object-Oriented Programming with Libraries\PyDev\U6 Fl
ask\Site1>python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Step 3: Try localhost:5000

127.0.0.1:5000

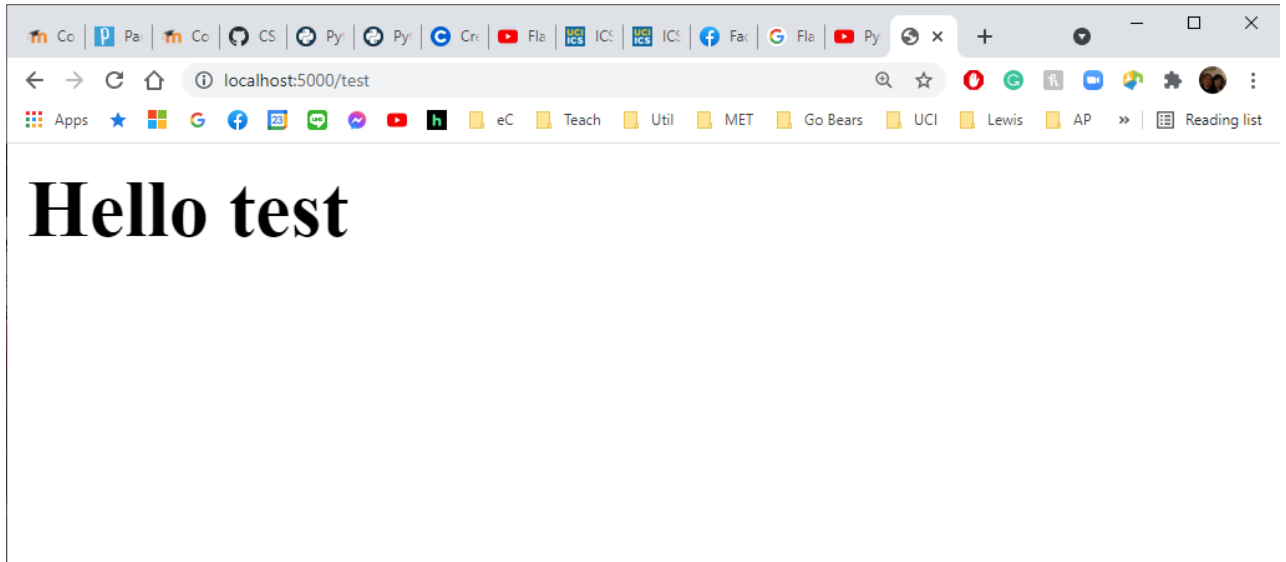


Demo Program: app2.py

```
from flask import Flask
app = Flask(__name__)    # create Flask as app

@app.route("/")    # root directory of the site
def home():         # return HTML template
    return "<h1>Hello Flask</h1>"

@app.route("/<name>") # sub-direcotry as variable pass to
def user(name):          # return HTML template
    return f"<h1>Hello {name}</h1>"    # fill in as name

if __name__ == "__main__":
    app.run()
```

Demo Program: app2.py

Local Client



Local Server

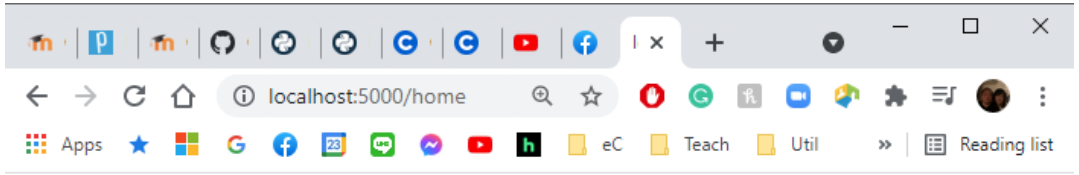
```
C:\Eric_Chou\Python Course\Python Object-Oriented Programming with Libraries\PyDev\U6 Fl
ask\Site1>python app2.py
 * Serving Flask app "app2" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [25/Jun/2021 02:27:46] "←[37mGET /test HTTP/1.1←[0m" 200 -
```

Turn on debug mode

```
from flask import Flask
app = Flask(__name__)    # create Flask as app

@app.route("/")    # root directory of the site
def home():         # return HTML template
    return "<h1>Hello Flask</h1>"

@app.route("/<name>") # sub-direcotry as variable pass to
def user(name):          # return HTML template
    return f"<h1>Hello {name}</h1>"    # fill in as name

if __name__ == "__main__":
    app.run(debug=True)
```

Turn on debug mode

Local Client



Hello home

Local Server

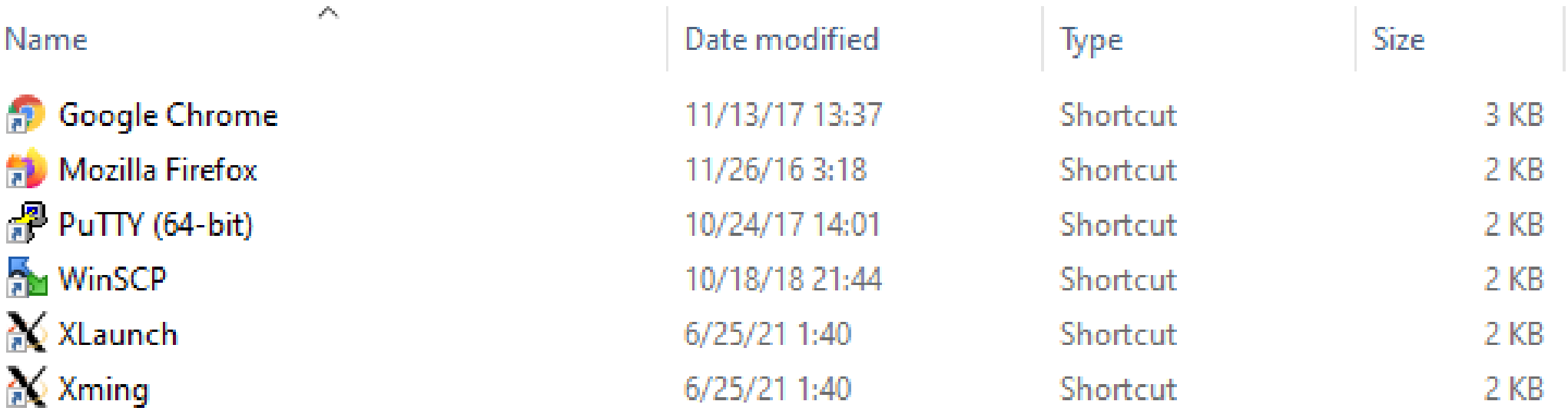
```
C:\Eric_Chou\Python Course\Python Object-Oriented Programming with Libraries\PyDev\U6 Flask\Site1>python app3.py
 * Serving Flask app "app3" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 119-386-998
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [25/Jun/2021 03:06:39] "←[37mGET /home HTTP/1.1←[0m" 200 -
```

You may see the HTTPrequest and HTTPrespond

Deploy the first App to a CPanel Shared Host Web-site

LECTURE 3

Use PuTTy



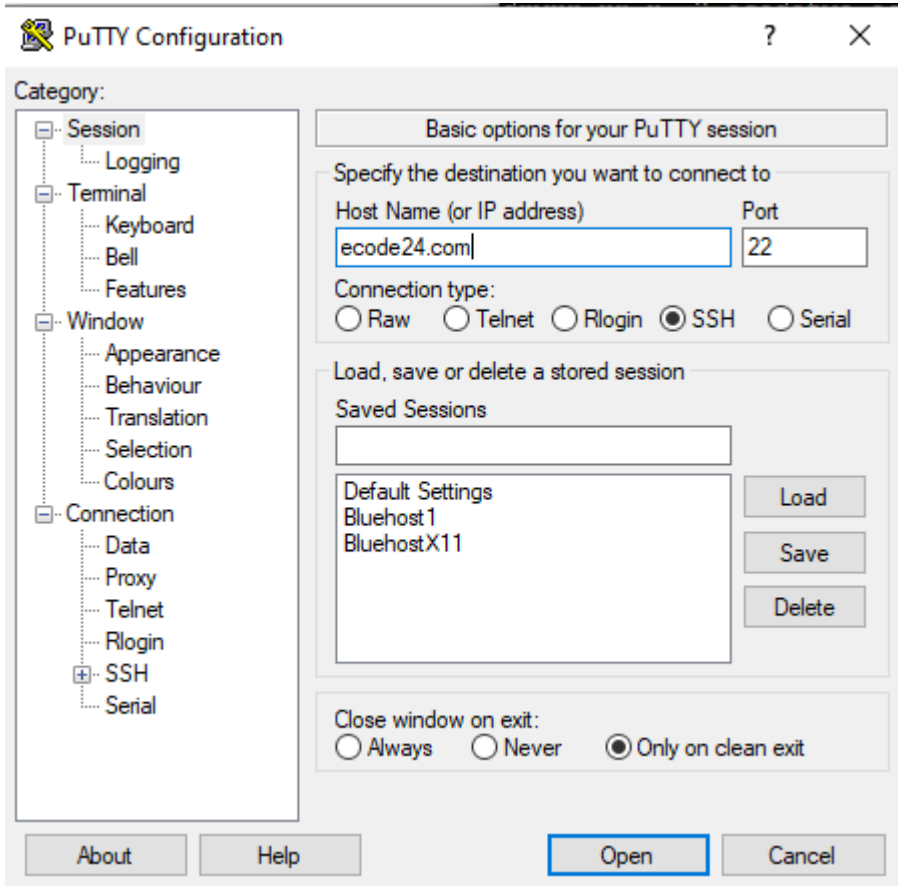
Name	Date modified	Type	Size
Google Chrome	11/13/17 13:37	Shortcut	3 KB
Mozilla Firefox	11/26/16 3:18	Shortcut	2 KB
PuTTY (64-bit)	10/24/17 14:01	Shortcut	2 KB
WinSCP	10/18/18 21:44	Shortcut	2 KB
XLaunch	6/25/21 1:40	Shortcut	2 KB
Xming	6/25/21 1:40	Shortcut	2 KB

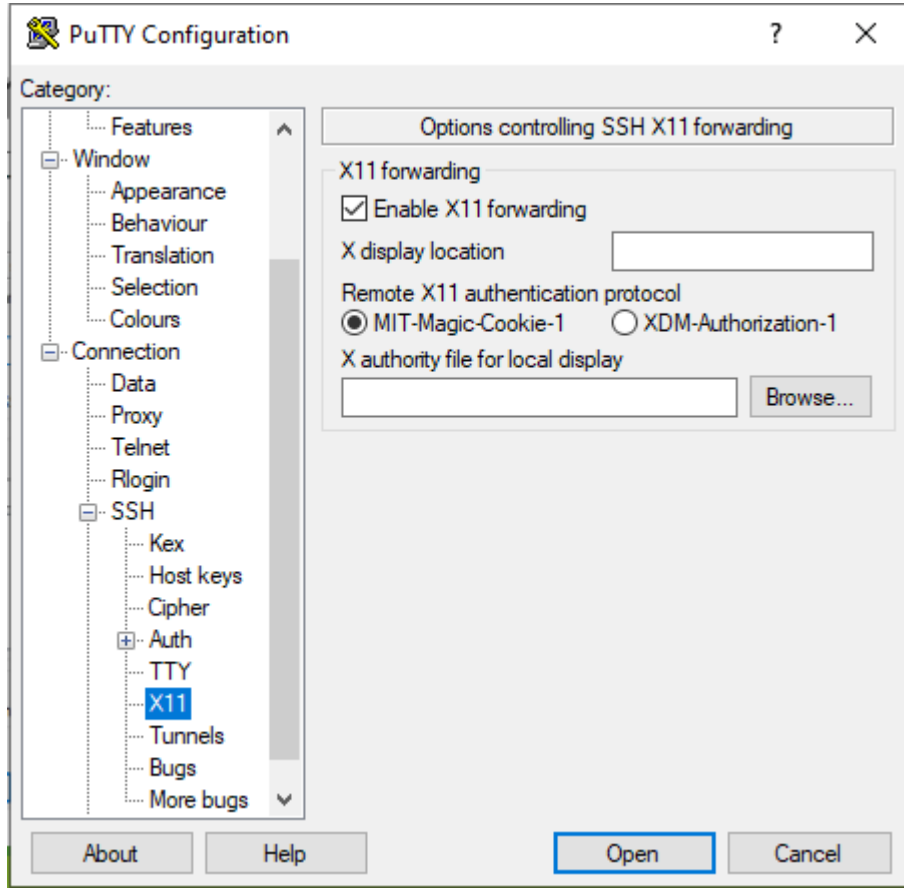
https://www.bluehost.com/help/article/using-ssh-on-windows-putty

Install Xming

http://www.geo.mtu.edu/geoschem/docs/putty_install.html

Login as admin





Install Python

Download Python

```
mkdir ~/python

cd ~/python

wget http://www.python.org/ftp/python/3.9.2/Python-3.9.2.tgz

tar zxfv Python-3.9.2.tgz

find ~/python -type d | xargs chmod 0755

cd Python-3.9.2
```

https://www.bluehost.com/help/article/python-installation

Install Python

Install Python

```
./configure --prefix=$HOME/python

make

make install
```

https://www.bluehost.com/help/article/python-installation

Modify Shell Script

Wrong Version

```
ecodetwo@ecode24.com [~/python/Python-3.9.2]# python
Python 2.7.5 (default, Nov 16 2020, 22:23:17)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-44)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
>>> exit()
ecodetwo@ecode24.com [~/python/Python-3.9.2]#
```

https://www.bluehost.com/help/article/python-installation

Modify Shell Script

Install Python

For your local version of python to load, you will need to add it to the .bashrc file.

```
vim ~/.bashrc
```

Press i

Enter:

```
export PATH=$HOME/python/Python-3.9.2/:$PATH
```

Write the changes (press ESC) and close vim:

```
:wq
```

Press Enter

```
source ~/.bashrc
```

https://www.bluehost.com/help/article/python-installation

Modify Shell Script

Correct Version

```
ecodetwo@ecode24.com [~/python/Python-3.9.2]# python
Python 2.7.5 (default, Nov 16 2020, 22:23:17)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-44)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
>>> exit()
ecodetwo@ecode24.com [~/python/Python-3.9.2]# vim ~/.bashrc
ecodetwo@ecode24.com [~/python/Python-3.9.2]# vim ~/.bashrc
ecodetwo@ecode24.com [~/python/Python-3.9.2]# source ~/.bashrc
ecodetwo@ecode24.com [~/python/Python-3.9.2]#
ecodetwo@ecode24.com [~/python/Python-3.9.2]#
ecodetwo@ecode24.com [~/python/Python-3.9.2]# python
Python 3.9.2 (default, Jun 25 2021, 11:18:58)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-44)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Check python/bin

```
ecodetwo@ecode24.com [~/python/bin]# ls -al
total 17516
drwxr-xr-x 2 ecodetwo ecodetwo     4096 Jun 25 11:49 ./
drwxr-xr-x 7 ecodetwo ecodetwo     4096 Jun 25 11:22 ../
lrwxrwxrwx 1 ecodetwo ecodetwo        8 Jun 25 11:22 2to3 -> 2to3-3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo        0 Jun 25 11:57 2to3-3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo      250 Jun 25 11:22 easy_install-3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo      236 Jun 25 11:47 f2py*
-rwxr-xr-x 1 ecodetwo ecodetwo      236 Jun 25 11:47 f2py3*
-rwxr-xr-x 1 ecodetwo ecodetwo      236 Jun 25 11:47 f2py3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo      228 Jun 25 11:49 flask*
lrwxrwxrwx 1 ecodetwo ecodetwo        7 Jun 25 11:22 idle3 -> idle3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo        0 Jun 25 11:57 idle3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo      241 Jun 25 11:22 pip3*
-rwxr-xr-x 1 ecodetwo ecodetwo      241 Jun 25 11:22 pip3.9*
lrwxrwxrwx 1 ecodetwo ecodetwo        8 Jun 25 11:22 pydoc3 -> pydoc3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo        0 Jun 25 11:57 pydoc3.9*
lrwxrwxrwx 1 ecodetwo ecodetwo        9 Jun 25 11:22 python3 -> python3.9*
lrwxrwxrwx 1 ecodetwo ecodetwo       16 Jun 25 11:22 python3-config -> python3.9-config*
-rwxr-xr-x 1 ecodetwo ecodetwo 17896592 Jun 25 11:22 python3.9*
-rwxr-xr-x 1 ecodetwo ecodetwo        0 Jun 25 11:57 python3.9-config*
```

Add python/bin to path

ecode24.com - PuTTY

```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions
export PATH=$HOME/python/Python-3.9.2/:$PATH
export PATH=$HOME/python/bin/:$PATH
~
~
```

pip3.9 install

pip3.9 install flask

mkdir PyDev under public_html

```
ecodetwo@ecode24.com [~/public_html]# mdkir PyDev
-jailshell: mdkir: command not found
ecodetwo@ecode24.com [~/public_html]# mkdir PyDev
ecodetwo@ecode24.com [~/public_html]# dir
./                           PyDev/                      moodle/              wp-config-sample.php  wp-login.php
../                          Students/                   readme.html          wp-config.php         wp-mail.php
.ftpquota                    cgi-bin/                    wp-activate.php      wp-content/           wp-settings.php
.htaccess                    error_log                   wp-admin/            wp-cron.php           wp-signup.php
.htaccess.phpupgrader.255e5095  googleea64356b25118633.html  wp-blog-header.php   wp-includes/          wp-trackback.php
.htaccess.phpupgrader.initial   index.php                wp-cli.yml           wp-links-opml.php     xmlrpc.php
.well-known/                 license.txt                 wp-comments-post.php  wp-load.php
ecodetwo@ecode24.com [~/public_html]# cd PyDev
ecodetwo@ecode24.com [~/public_html/PyDev]# mkdir sl
ecodetwo@ecode24.com [~/public_html/PyDev]# cd sl
ecodetwo@ecode24.com [~/public_html/PyDev/sl]# python --version
Python 3.9.2
ecodetwo@ecode24.com [~/public_html/PyDev/sl]#
```

ecode24.com - PuTTY

app.py

```
from flask import Flask, render_template

app = Flask(__name__)

@app.route("/")
def home():
    return render_template('home.html')

if __name__ == "__main__":
    app.debug = True
    app.run(host='127.0.0.1', port = 5000)
```

Running on localhost of ecode24.com

home.html

```
<!DOCTYPE html>
<html>
   <head>
      <title>Home Page</title>
   </head>
   <body>
      <h1>Hello Flask</h1>
      <p>It's Working!!!</p>
   </body>
</html>
```

Application Configuration

Show/Hide Help

Application Name

Enter a display name for the deployed application.

Site1

Deployment Domain

Select the domain or subdomain to deploy the application to.

ecode24.com

Base Application URL

Enter the application's base URL. After you register the application, you can use this URL to access it.

ecode24.com /PyDev/s1

Application Path

Enter the path to your application's source code directory, relative to the home directory.

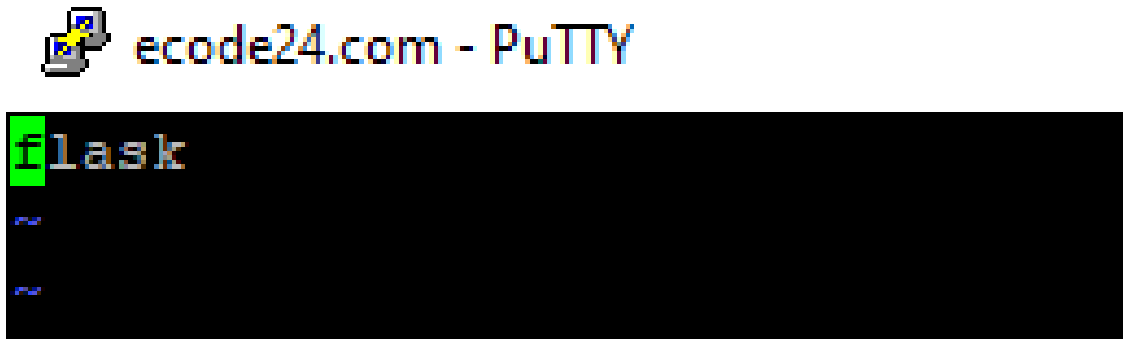
/ PyDev/s1

Deployment Environment

Select the server environment in which your application will run.

- (•) Development
- ( ) Production

requirements.txt



Application Manager

This feature allows you to register, manage, and deploy your custom applications using Phusion Passenger.

Search by name

+ Register Application

Page Size 10 << < > >>

Displaying 1 to 1 out of 1 item

Name ▲	Domain	Path	Status	Actions
Site1	ecode24.com	/PyDev/s1	Enabled	Edit Unregister