

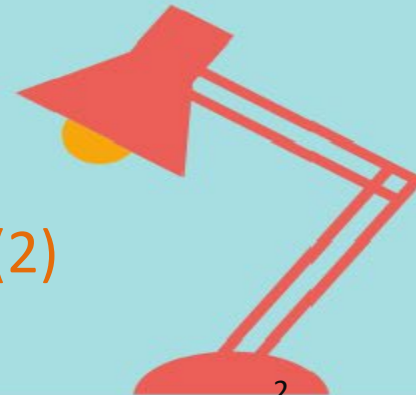
A stylized illustration of a row of books on a shelf. The books are in various colors (white, red, blue, yellow) and some have decorative patterns like stripes or a diamond. They are arranged in a slightly staggered manner.

# 2023-Spring Advanced Computer Programming (Week 12)

CSIE, Asia Univ.

# Course schedule

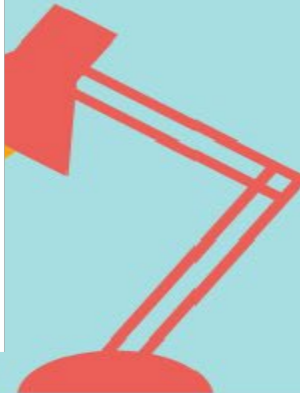
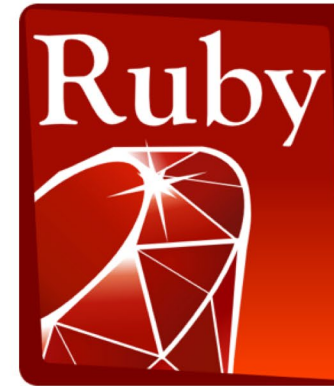
- W1-Introduction
- W2-Python libraries
- W3-BeautifulSoup(1)
- W4-BeautifulSoup(2)
- W5-
- W6-Scrapy(1)
- W7-Scrapy(2)
- W8-Storing Data
- W9-Midterm project
- W10-Web & HTTP
- W11-Flask
- W12-Flask Routes
- W13-Jinja template
- W14-Flask-form
- W15-Flask-mail
- W16-REST API
- W17-Project development(2)
- W18-Final presentation



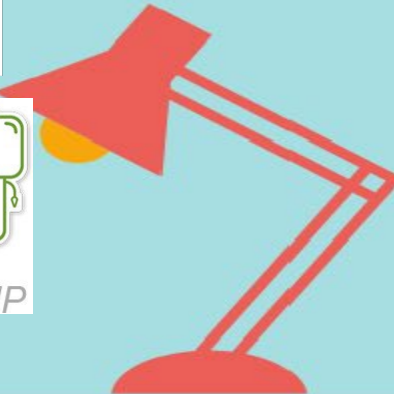
# (Big) Frameworks



python



# (Micro $\mu$ ) Frameworks



# PythonAnywhere – Free online Python execution environment

- Free account is limited as follows:
  - Only one App (Application) can be created
  - Off-net access to the Internet is limited
  - CPU and storage are limited (100 seconds of CPU time a day, 512MB of storage)
  - Does not provide Jupyter (but does have IPython)
  - There can only be two Consoles (Bash and Python)





python



Flask

web development,  
one drop at a time

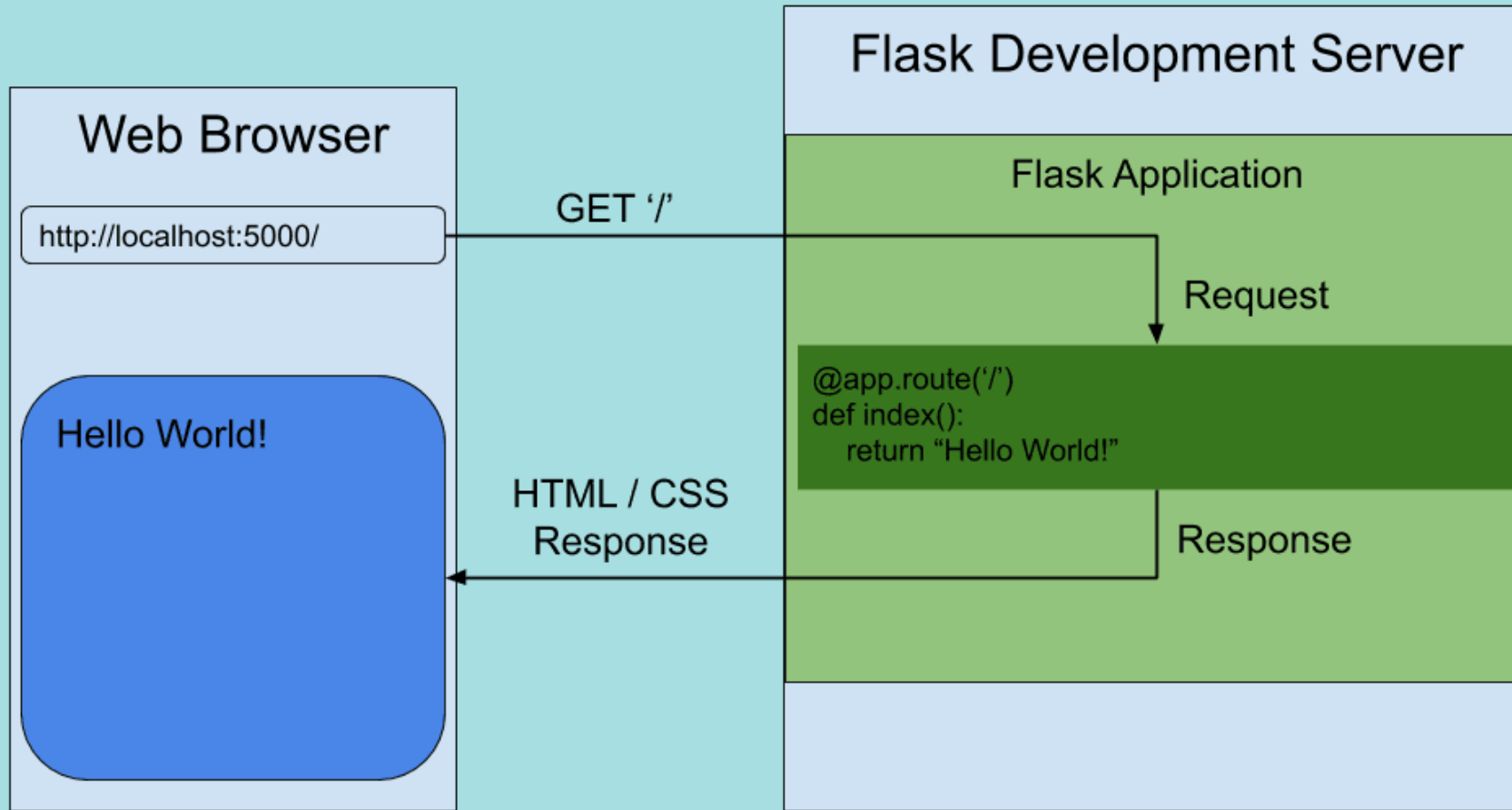


# Basic Flask Application Structure

- Routes and View Functions
- Dynamic Routes
- Application and Request Contexts
- Flask context globals
- Flask request object
- Flask response object

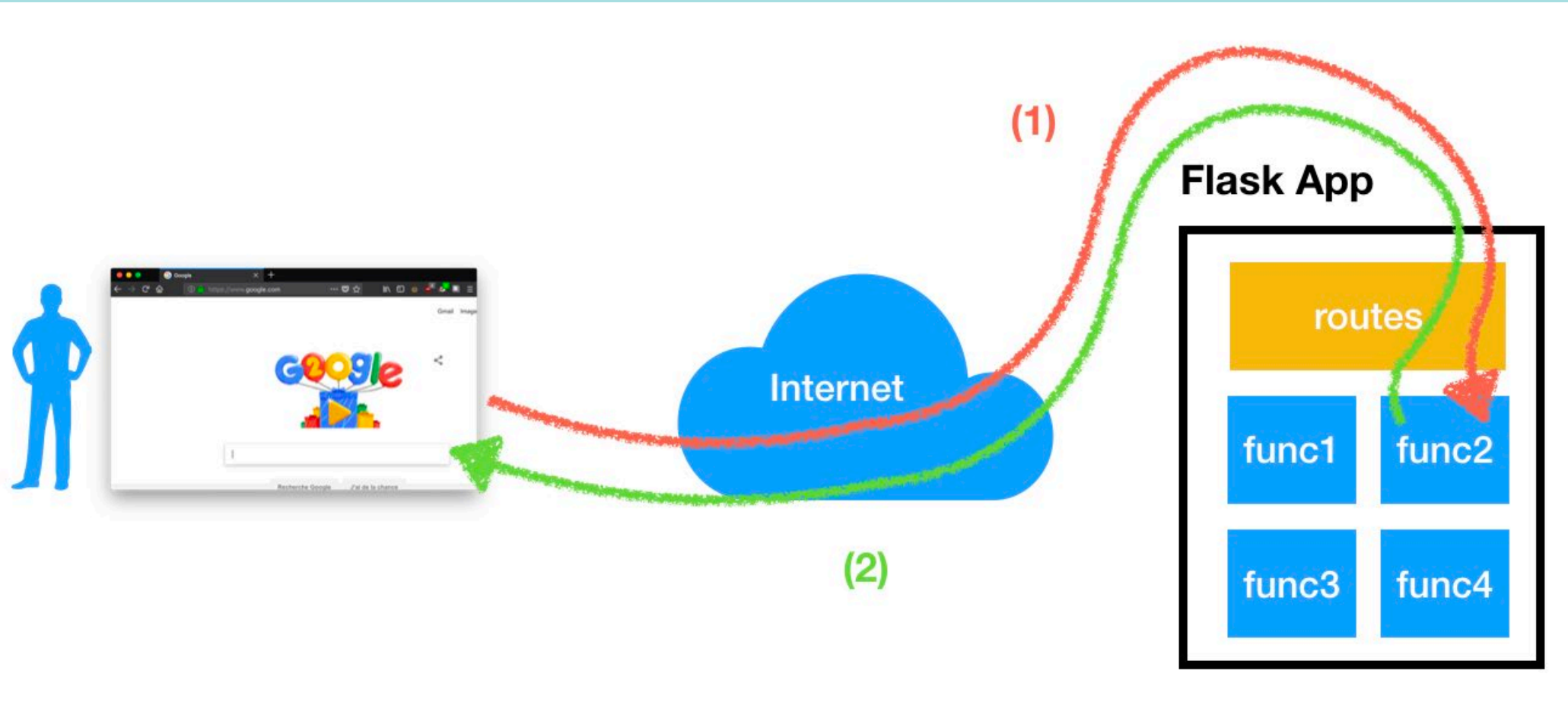


# Routes and View Functions





# View functions



# Dynamic Routes

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def index():
    return '<h1>Hello World!</h1>'

@app.route('/user/<name>')
def user(name):
    return '<h1>Hello, {}!</h1>'.format(name)
```



# Application and Request Contexts

```
from flask import request

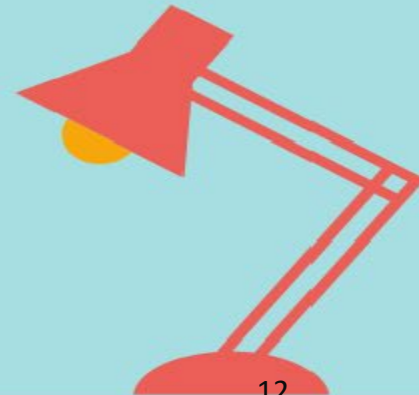
@app.route('/')
def index():
    user_agent = request.headers.get('User-Agent')
    return '<p>Your browser is {}</p>'.format(user_agent)
```

*Table 2-1. Flask context globals*

Variable name	Context	Description
<code>current_app</code>	Application context	The application instance for the active application.
<code>g</code>	Application context	An object that the application can use for temporary storage during the handling of a request. This variable is reset with each request.
<code>request</code>	Request context	The request object, which encapsulates the contents of an HTTP request sent by the client.
<code>session</code>	Request context	The user session, a dictionary that the application can use to store values that are “remembered” between requests.

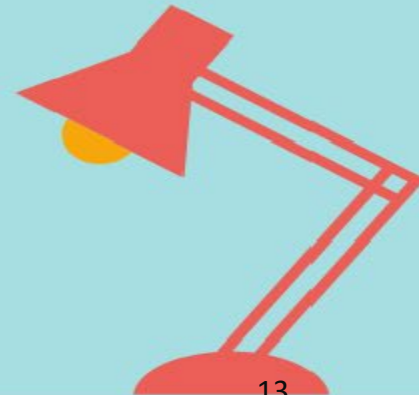
# Flask context globals

form	A dictionary with all the form fields submitted with the request.
args	A dictionary with all the arguments passed in the query string of the URL.
values	A dictionary that combines the values in form and args.
cookies	A dictionary with all the cookies included in the request.
headers	A dictionary with all the HTTP headers included in the request.



# Flask request object

<code>status_code</code>	The numeric HTTP status code
<code>headers</code>	A dictionary-like object with all the headers that will be sent with the response
<code>set_cookie()</code>	Adds a cookie to the response
<code>delete_cookie()</code>	Removes a cookie
<code>content_length</code>	The length of the response body
<code>content_type</code>	The media type of the response body
<code>set_data()</code>	Sets the response body as a string or bytes value
<code>get_data()</code>	Gets the response body

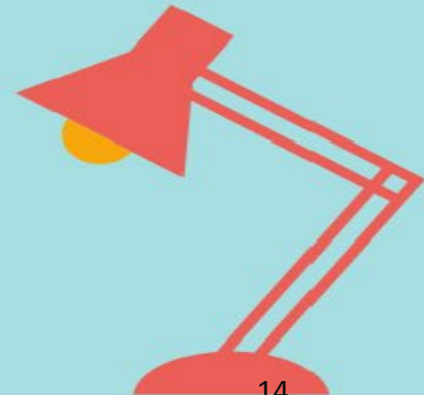


# Flask response object

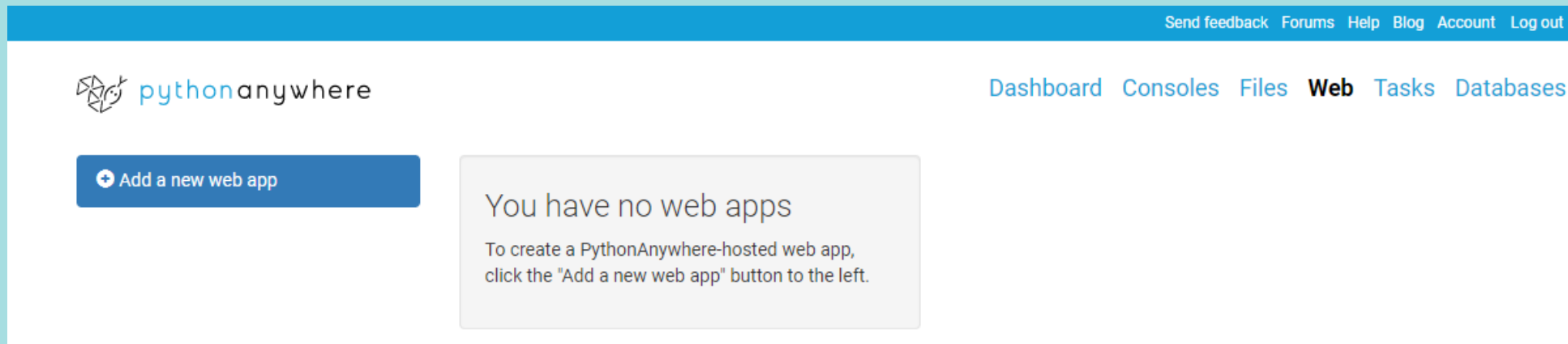
<code>status_code</code>	The numeric HTTP status code
<code>headers</code>	A dictionary-like object with all the headers that will be sent with the response
<code>set_cookie()</code>	Adds a cookie to the response
<code>delete_cookie()</code>	Removes a cookie
<code>content_length</code>	The length of the response body
<code>content_type</code>	The media type of the response body
<code>set_data()</code>	Sets the response body as a string or bytes value
<code>get_data()</code>	Gets the response body

```
from flask import redirect

@app.route('/')
def index():
    return redirect('http://www.example.com')
```



# Step 1: Go to Web tab



The screenshot shows the PythonAnywhere dashboard interface. At the top, a blue navigation bar contains links for "Send feedback", "Forums", "Help", "Blog", "Account", and "Log out". Below this, the main header area includes the PythonAnywhere logo on the left and a set of navigation tabs on the right: "Dashboard", "Consoles", "Files", "Web" (which is bolded and highlighted), "Tasks", and "Databases". On the left side of the dashboard, there is a blue button with a plus icon and the text "Add a new web app". To the right of this button, a light gray box contains the text "You have no web apps" followed by instructions: "To create a PythonAnywhere-hosted web app, click the 'Add a new web app' button to the left."

Send feedback Forums Help Blog Account Log out

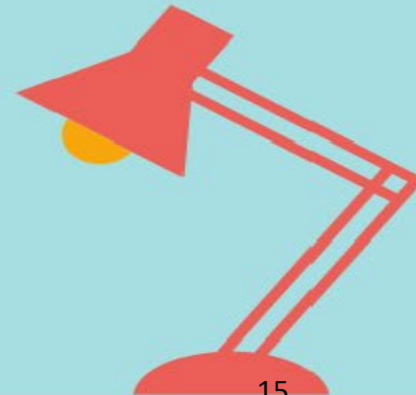
pythonanywhere

Dashboard Consoles Files **Web** Tasks Databases

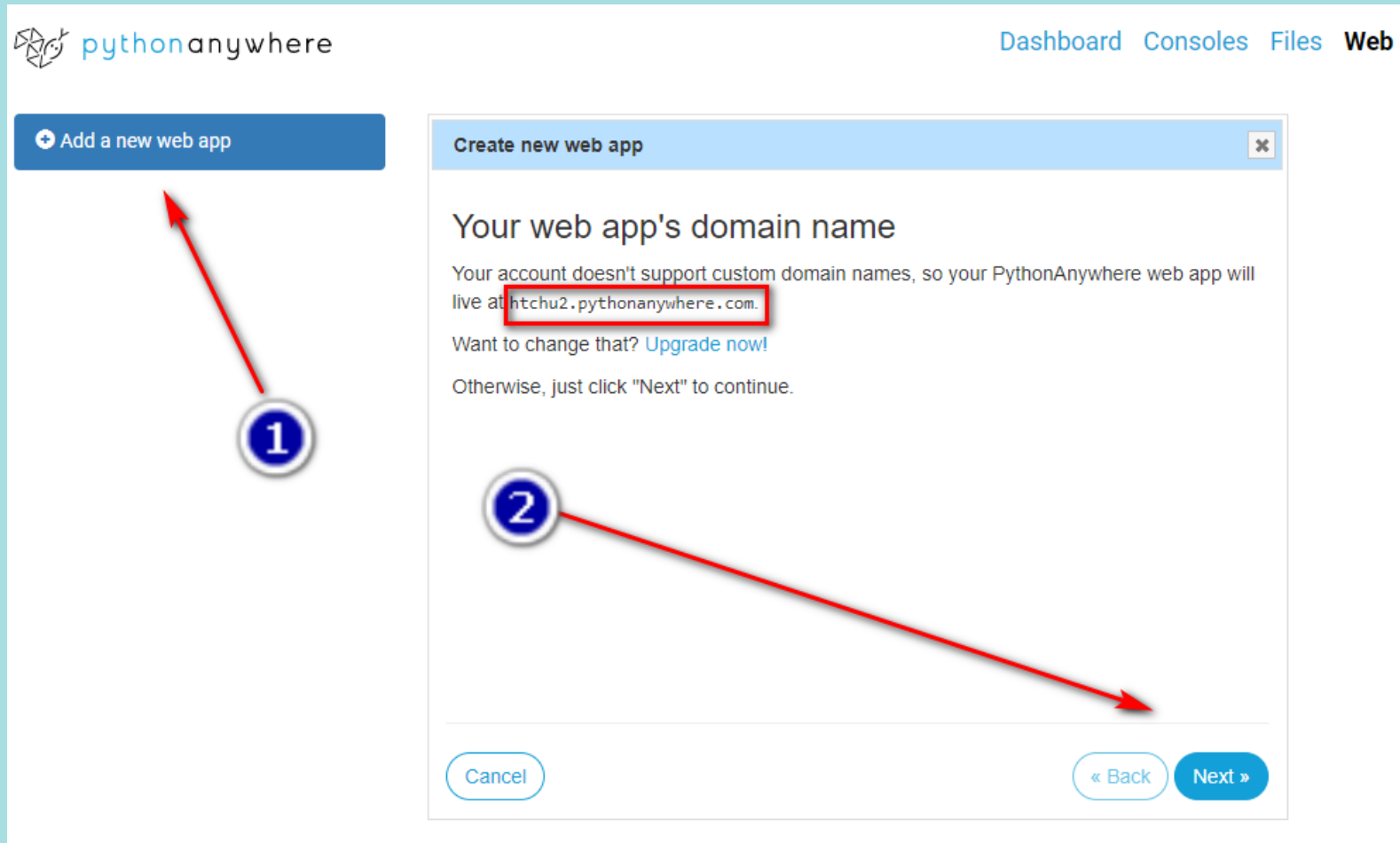
+ Add a new web app

You have no web apps

To create a PythonAnywhere-hosted web app, click the "Add a new web app" button to the left.



# Step 2: Add a new web app



The screenshot shows the PythonAnywhere dashboard. At the top left is the PythonAnywhere logo. At the top right are navigation links: Dashboard, Consoles, Files, and Web. On the left sidebar, there is a blue button labeled '+ Add a new web app'. A red arrow points from a blue circle with the number '1' to this button. A modal window titled 'Create new web app' is open. It contains the text 'Your web app's domain name' followed by 'Your account doesn't support custom domain names, so your PythonAnywhere web app will live at `htchu2.pythonanywhere.com`'. The domain name is highlighted with a red box. Below this, it says 'Want to change that? [Upgrade now!](#)' and 'Otherwise, just click "Next" to continue.' A red arrow points from a blue circle with the number '2' to the 'Next »' button at the bottom right of the modal. The modal also has a 'Cancel' button on the bottom left and a close button (X) in the top right corner.

pythonanywhere

Dashboard Consoles Files Web

+ Add a new web app

Create new web app

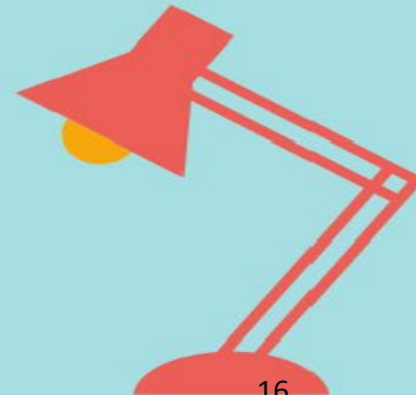
Your web app's domain name

Your account doesn't support custom domain names, so your PythonAnywhere web app will live at `htchu2.pythonanywhere.com`.

Want to change that? [Upgrade now!](#)

Otherwise, just click "Next" to continue.

Cancel « Back Next »





# Step 3: Select a Python Web framework and a Python version

## Select a Python Web framework

...or select "Manual configuration" if you want detailed control.

- » Django
- » web2py
- » Flask
- » Bottle
- » **Manual configuration** (including virtualenvs)

What other frameworks should we have here? Send us some feedback using the link at the top of the page!

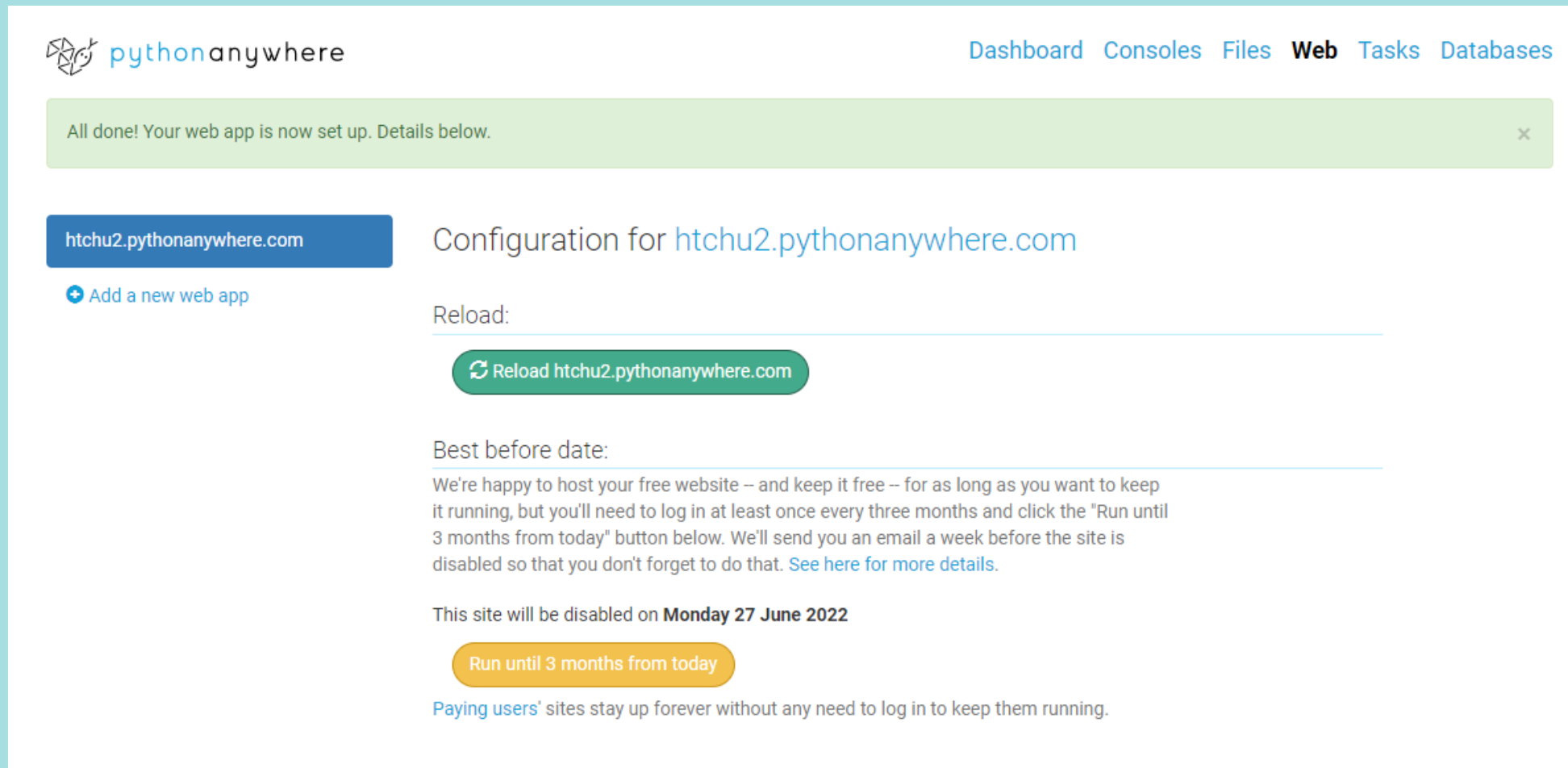
## Select a Python version

- » **Python 3.6** (Flask 2.0.0)
- » **Python 3.7** (Flask 2.0.0)
- » **Python 3.8** (Flask 2.0.0)
- » **Python 3.9** (Flask 2.0.0)

**Note:** If you'd like to use a different version of Flask to the default version, you can use a virtualenv for your web app. There are [instructions here](#).



# Step 4: Quick start new Flask project



The screenshot shows the PythonAnywhere web interface. At the top left is the PythonAnywhere logo. To the right are navigation links: Dashboard, Consoles, Files, Web (highlighted), Tasks, and Databases. A green notification bar at the top states: "All done! Your web app is now set up. Details below." Below this, on the left, is a blue button labeled "htchu2.pythonanywhere.com" and a link "+ Add a new web app". The main content area is titled "Configuration for htchu2.pythonanywhere.com". It contains a "Reload:" section with a green button "Reload htchu2.pythonanywhere.com". Below that is a "Best before date:" section with a paragraph explaining the free hosting policy and a yellow button "Run until 3 months from today". At the bottom, it states "This site will be disabled on Monday 27 June 2022" and includes a note about paying users.

pythonanywhere

Dashboard Consoles Files **Web** Tasks Databases

All done! Your web app is now set up. Details below. ×

htchu2.pythonanywhere.com

+ Add a new web app

### Configuration for htchu2.pythonanywhere.com

Reload:

Reload htchu2.pythonanywhere.com

Best before date:

We're happy to host your free website – and keep it free – for as long as you want to keep it running, but you'll need to log in at least once every three months and click the "Run until 3 months from today" button below. We'll send you an email a week before the site is disabled so that you don't forget to do that. [See here for more details.](#)

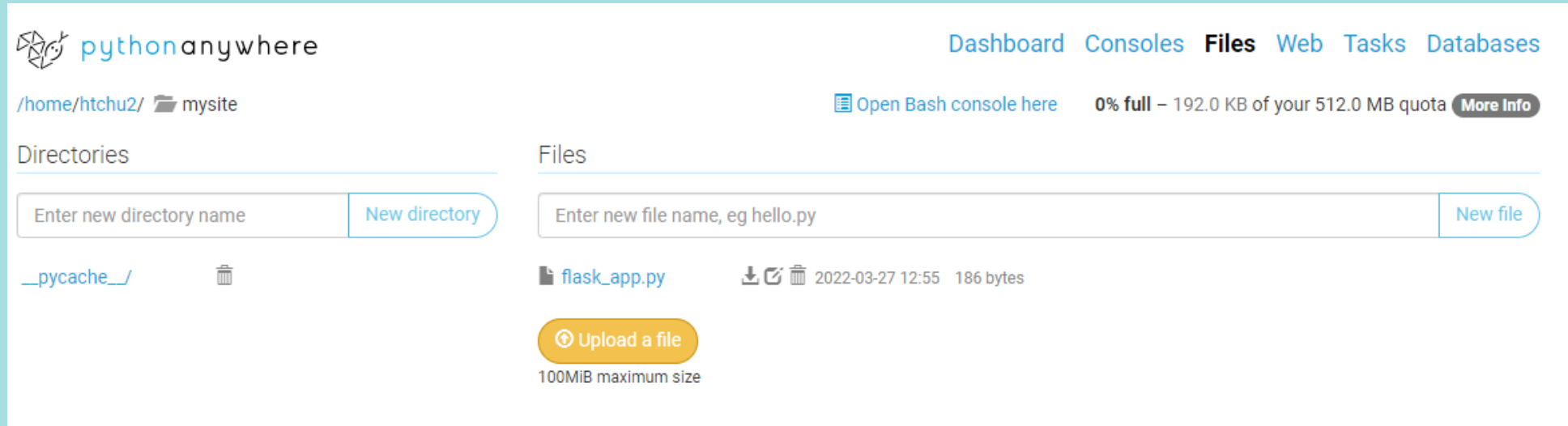
This site will be disabled on **Monday 27 June 2022**

Run until 3 months from today

[Paying users'](#) sites stay up forever without any need to log in to keep them running.



# Step 5: check the files

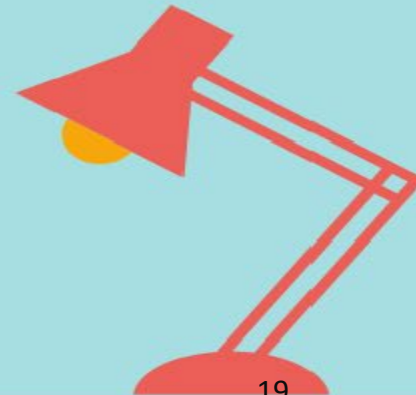


The screenshot shows the PythonAnywhere web interface. At the top left is the PythonAnywhere logo. To its right are navigation links: Dashboard, Consoles, **Files**, Web, Tasks, and Databases. Below the logo, the current path is shown as `/home/htchu2/` with a folder icon and the name `mysite`. On the right side of the header, there is a link to "Open Bash console here" and a storage status indicator: "0% full – 192.0 KB of your 512.0 MB quota" with a "More Info" button.


The main content area is divided into two sections: "Directories" on the left and "Files" on the right.

**Directories:** It features a text input field "Enter new directory name" and a "New directory" button. Below this, a directory named `__pycache__` is listed with a trash icon next to it.

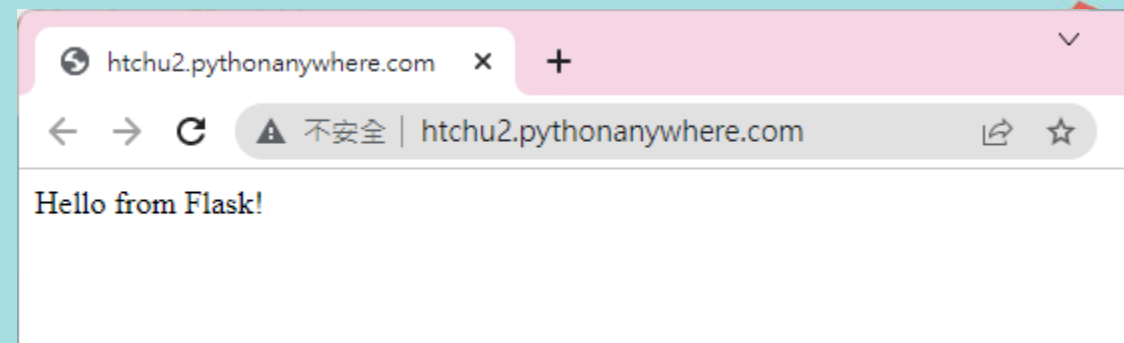
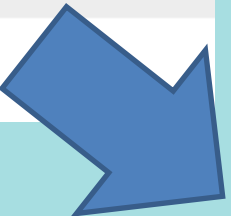
**Files:** It features a text input field "Enter new file name, eg hello.py" and a "New file" button. Below this, a file named `flask_app.py` is listed. To the left of the filename is a file icon. To the right are icons for download, copy, and delete, followed by the date and time "2022-03-27 12:55" and the size "186 bytes". Below the file list is an orange "Upload a file" button with a circular arrow icon, and the text "100MiB maximum size" below it.



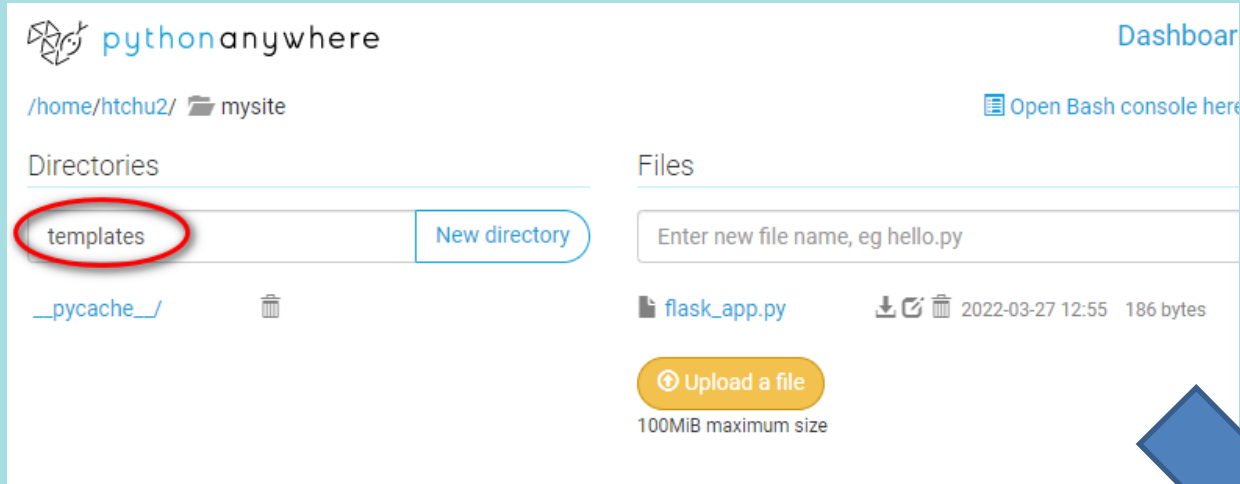
# Step 6: check the program and the web app

 /home/htchu2/mysite/flask\_app.py

```
1
2 # A very simple Flask Hello World app for you to get started with...
3
4 from flask import Flask
5
6 app = Flask(__name__)
7
8 @app.route('/')
9 def hello_world():
10     return 'Hello from Flask!'
11
12
```



# Step 7: add html templates



pythonanywhere Dashboard

/home/htchu2/ mysite

Open Bash console here

Directories

- templates
- \_\_pycache\_\_

New directory

Files

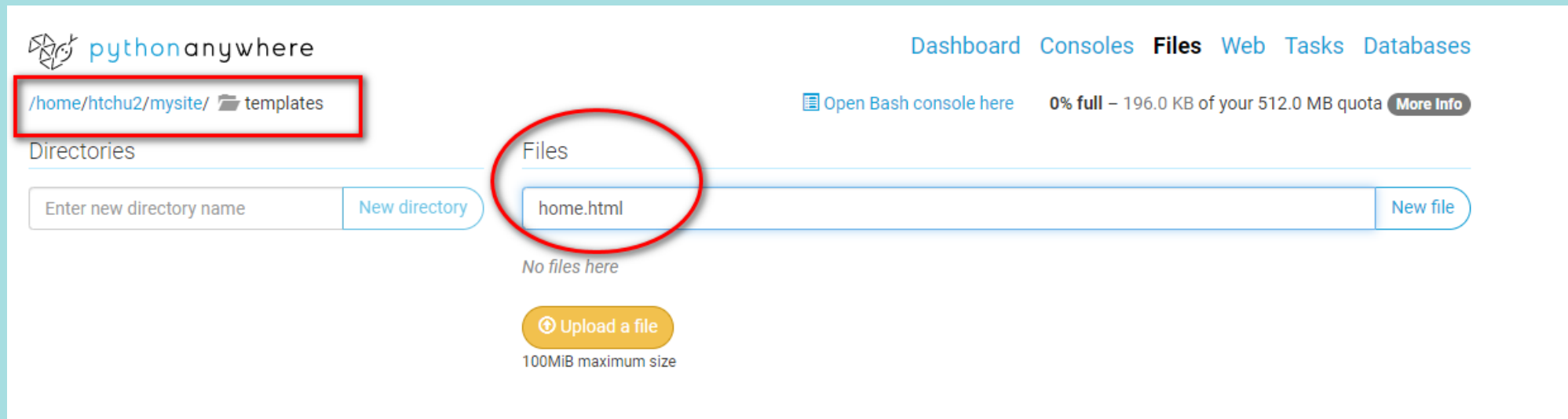
Enter new file name, eg hello.py

flask\_app.py 2022-03-27 12:55 186 bytes

Upload a file

100MiB maximum size

A large blue arrow points from this screenshot to the one below.



pythonanywhere Dashboard Consoles **Files** Web Tasks Databases

/home/htchu2/mysite/ templates

Open Bash console here 0% full – 196.0 KB of your 512.0 MB quota More Info

Directories

Enter new directory name New directory

Files

home.html New file

No files here

Upload a file

100MiB maximum size


# Step 8: edit home.html



- `<h1>Top 10 HTML tags</h1>`  
`<br> <br> <br>`In this article we will explain our list of top 10 HTML tags.`<br> <br>` `<h2>First tag is`  
`&lt;b>&lt;/b>``</h2>` Bold is all about making words more important.




# Step 9: edit flask\_app.py

 /home/htchu2/mysite/flask\_app.py

```
1
2 # A very simple Flask Hello World app for you to get started with...
3
4 from flask import Flask, render_template
5
6 app = Flask(__name__)
7
8 @app.route('/')
9 def home():
10     return render_template("home.html")
11
```



# Step 10: Reload web

 pythonanywhere

Dashboard Consoles Files Web

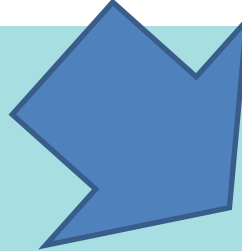
htchu2.pythonanywhere.com

+ Add a new web app

Configuration for htchu2.pythonanywhere.com

Reload:

Reload htchu2.pythonanywhere.com



htchu2.pythonanywhere.com x +

← → ↻ ⚠ 不安全 | htchu2.pythonanywhere.com

My First Web Page

Headings Are Great Fun

This is my first **paragraph** in my new *webpage*. This is going to be great. I am so excited I can hardly contain **myself**. Don't you just love paragraphs? I find them very useful.

Web Pages Are Exciting Too

Yes, that's right - web pages can be a lot of fun. Learning how to create web pages is easy and **entertaining**. This is my second *paragraph*. I hope you like it.


[Link to Google](#)

- Apples
- Bananas
- Pears
- Oranges
- Grapes

"A designer knows he has achieved perfection not when there is nothing left to add, but when there is nothing left to take away." - Saint Exupéry

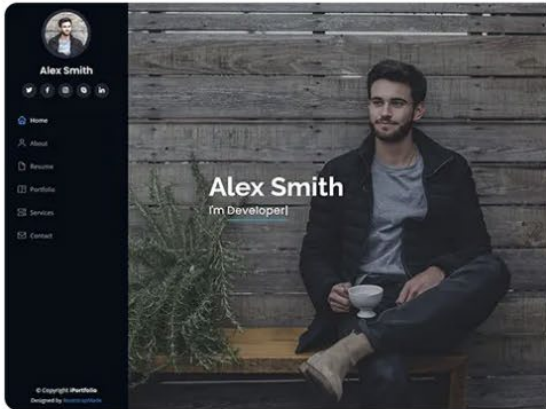


# Step 11: Bootstrap Personal Templates Website

 [Home](#) [Templates ▾](#) [License](#) [Contact](#) [Members](#)

## Bootstrap Personal Templates

Free and high quality Bootstrap personal website templates

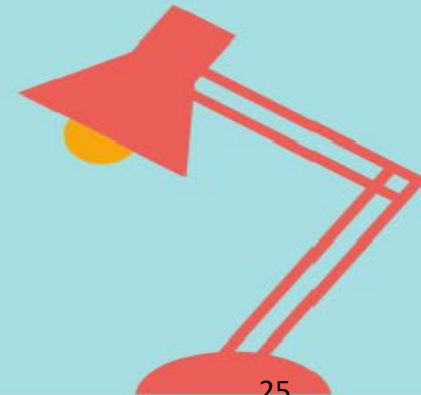


**Alex Smith**  
[m Developer]


### iPortfolio – Bootstrap Portfolio Websites Template

iPortfolio is a modern personal CV and portfolio HTML template. It's creative, minimal and clean design. iPortfolio can be used for many purposes starting from minimal portfolios, freelancers, graphic designers, illustrators, photographers and ...


[Live Demo](#) [Free Download](#)



# Step 12: Downloading the iPortfolio Template

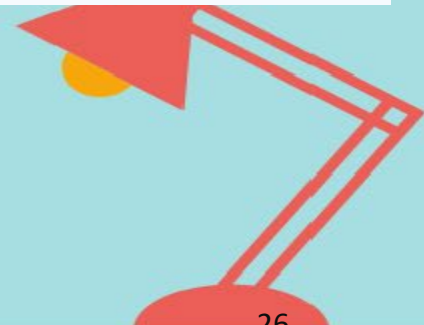
 [Home](#) [Templates ▾](#) [License](#) [Contact](#) [Members](#)

## iPortfolio - Bootstrap Portfolio Websites Template




Downloading the **iPortfolio Template**

**Click here** if the download didn't start automatically!




# Step 13: Upload iPortfolio.zip to mysite

 pythonanywhere  
by ANACONDA.

Dashboard Consoles **Files** Web Tasks Databases

**Warning** You have not confirmed your email address yet. This means that you will not be able to reset your password if you lose it. If you cannot find your confirmation email anymore, send yourself a new one [here](#).




/home/htchu44/  mysite

[Open Bash console here](#) **41% full** – 208.0 MB of your 512.0 MB quota [More Info](#)

Directories





Enter new directory name [New directory](#)





[\\_\\_pycache\\_\\_/](#)  
[static/](#)  
[templates/](#)

Files

Enter new file name, eg hello.py [New file](#)

 [flask\\_app.py](#)    2023-05-01 15:39 361 bytes

 **iPortfolio.zip**    2023-05-01 15:13 4.5 MB

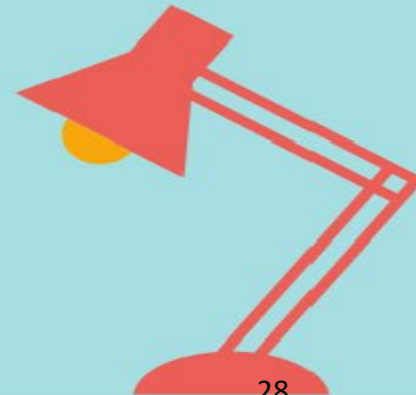
[Upload a file](#)

# Step 14: Unzip iPortfolio.zip



Bash console 28503198


```
/home/htchu44
15:14 ~ $ cd mysite/
15:14 ~/mysite $ unzip iPortfolio.zip
Archive:  iPortfolio.zip
  inflating: iPortfolio/assets/css/style.css
  extracting: iPortfolio/assets/img/apple-touch-icon.png
  extracting: iPortfolio/assets/img/favicon.png
  extracting: iPortfolio/assets/img/hero-bg.jpg
  extracting: iPortfolio/assets/img/portfolio/portfolio-1.jpg
  extracting: iPortfolio/assets/img/portfolio/portfolio-2.jpg
  extracting: iPortfolio/assets/img/portfolio/portfolio-3.jpg
```



# Step 15: move mvfiles

```
~/mysite $ mv iPortfolio/*.html templates/.  
~/mysite $ mv iPortfolio static
```

mv iPortfolio/\*.html templates/.  
mv iPortfolio static

[/home/htchu44/](#)  mysite [Open Bash console h](#)




Directories

[New directory](#)


[\\_\\_pycache\\_\\_/](#)




[static/](#)


[templates/](#)





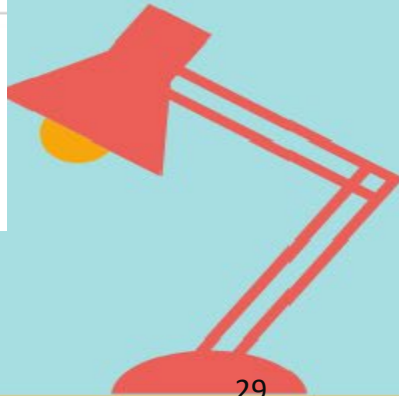
Files

 flask\_app.py

   2023-05-01 15:39 361 bytes

 iPortfolio.zip

  2023-05-01 15:13 4.5 MB



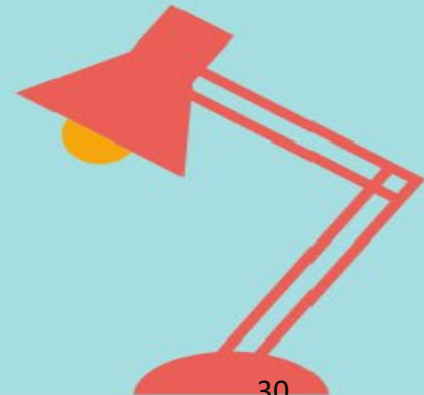
# Step 16: edit flask\_app.py

```
from flask import Flask, render_template
from flask import send_from_directory

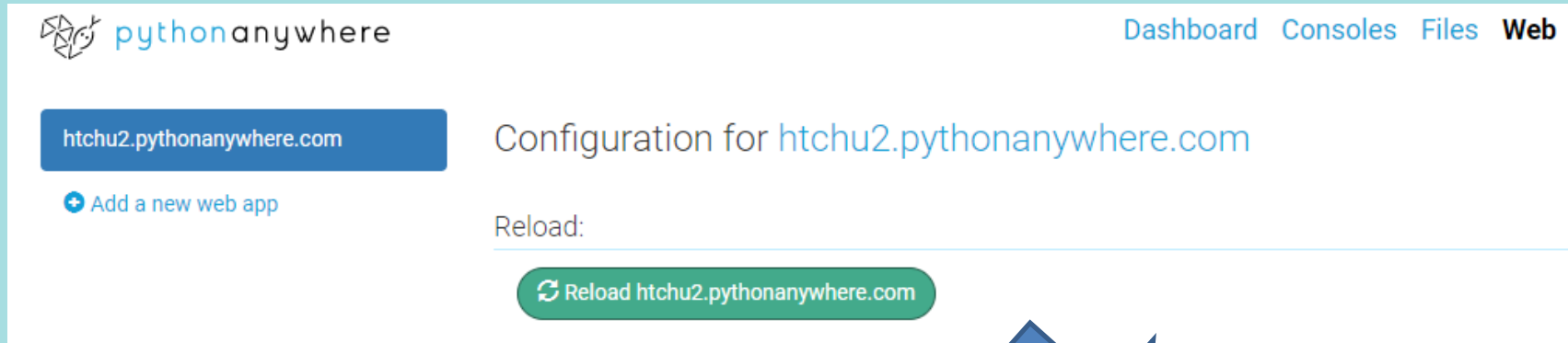
app = Flask(__name__)

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

@app.route('/assets/<path:path>')
def send_report(path):
    return send_from_directory('static/assets', path)
```



# Step 17: Reload web

 pythonanywhere


Dashboard Consoles Files **Web**

htchu2.pythonanywhere.com

+ Add a new web app

Configuration for htchu2.pythonanywhere.com

Reload:

 Reload htchu2.pythonanywhere.com



Not secure | htchu44.pythonanywhere.com



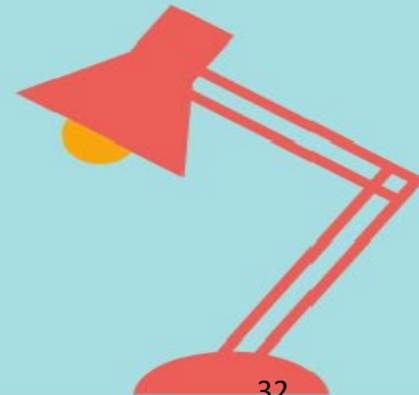
Alex Smith

I'm Designer!

Home About Resume Portfolio Services

# Assignment 3

- Choose one of Bootstrap Personal Templates from <https://bootstrapmade.com/bootstrap-personal-templates/>
- Replace the personal data and photos on the webpage with your own information.
- Refer to the step-by-step tutorial for constructing the pythonanywhere website
- submit the url of your pythonanywhere website







Thanks!

Q&A

