

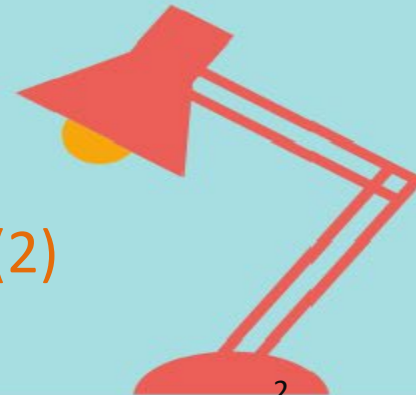
A stylized illustration of a row of books in various colors (white, red, blue, yellow) with different patterns and sizes, standing on a shelf.

2023-Spring Advanced Computer Programming (Week 13)

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





python



Flask

web development,
one drop at a time

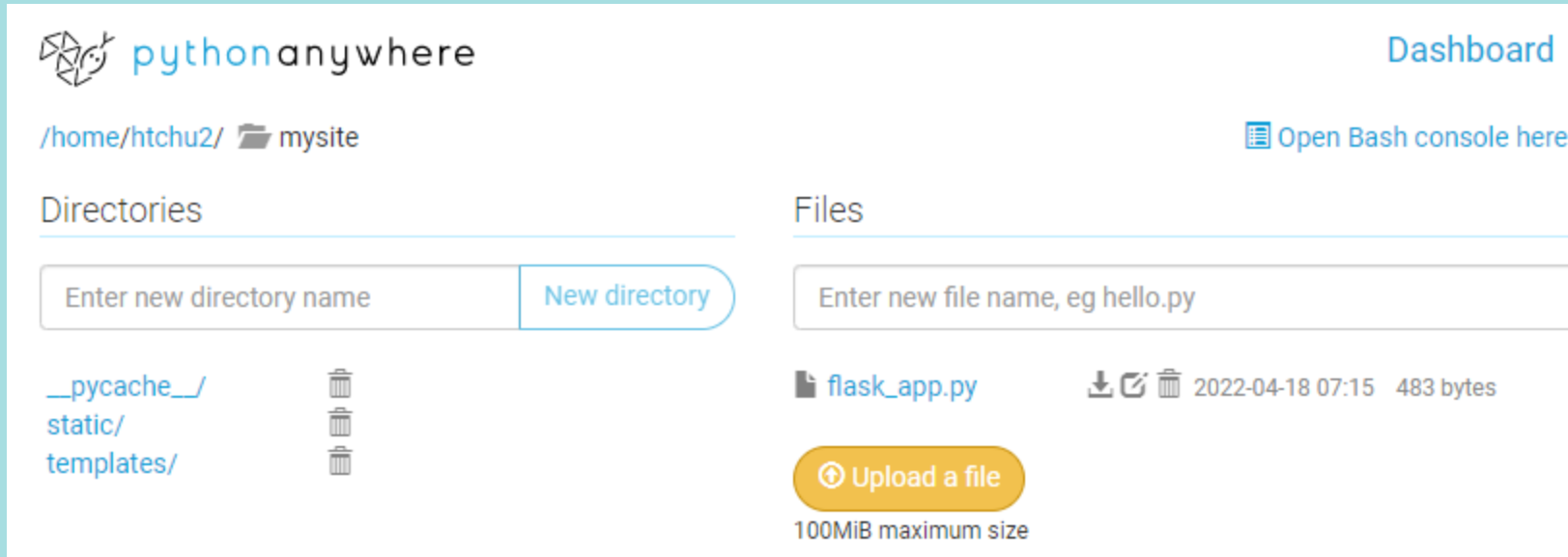


Assignment 3

- 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




1. Flask framework



The screenshot displays the PythonAnywhere web interface. At the top left is the PythonAnywhere logo and name. To the right is a 'Dashboard' link. Below the logo, the current path is shown as `/home/htchu2/` with a folder icon and the name 'mysite'. A link 'Open Bash console here' is located to the right of the path. The interface is divided into two main sections: 'Directories' on the left and 'Files' on the right. The 'Directories' section has a text input 'Enter new directory name' and a 'New directory' button. Below this, three directories are listed: `__pycache__`, `static`, and `templates`, each with a trash icon. The 'Files' section has a text input 'Enter new file name, eg hello.py'. Below this, a file named `flask_app.py` is listed with icons for download, share, and delete, along with the date '2022-04-18 07:15' and size '483 bytes'. An 'Upload a file' button is present, with a note '100MiB maximum size' below it.




pythonanywhere

Dashboard

`/home/htchu2/`  mysite [Open Bash console here](#)





Directories

Enter new directory name [New directory](#)

`__pycache__` 
`static` 
`templates` 

Files

Enter new file name, eg hello.py

 `flask_app.py`    2022-04-18 07:15 483 bytes

[Upload a file](#)

100MiB maximum size



Basic Flask framework

```
# main.py

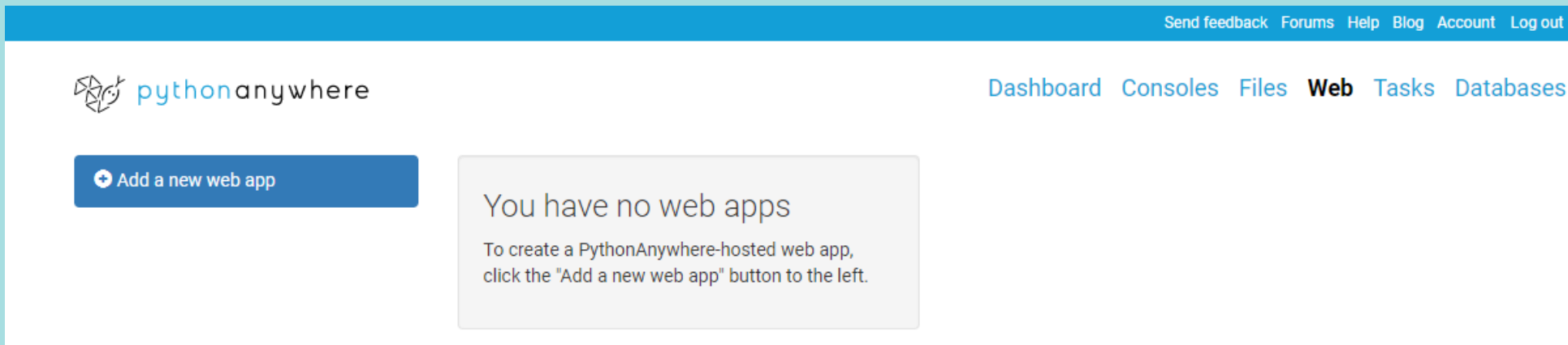
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'
```



- (1) The line `app = Flask(__name__)`, `__name__` is used to locate the location of the currently loaded folder, which is used to determine the location of the `template` folder or `static` folder.
- (2) `@app.route('/')` A decorator that is used to register a view function for a given URL rule. This does the same thing as `add_url_rule()` but is intended for decorator usage

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 PythonAnywhere logo is on the left, and a horizontal menu on the right includes "Dashboard", "Consoles", "Files", "Web" (which is bolded and underlined), "Tasks", and "Databases". On the left side of the main content area, there is a blue button with a plus icon and the text "Add a new web app". To the right of this button is a light gray box with the heading "You have no web apps" and the text "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

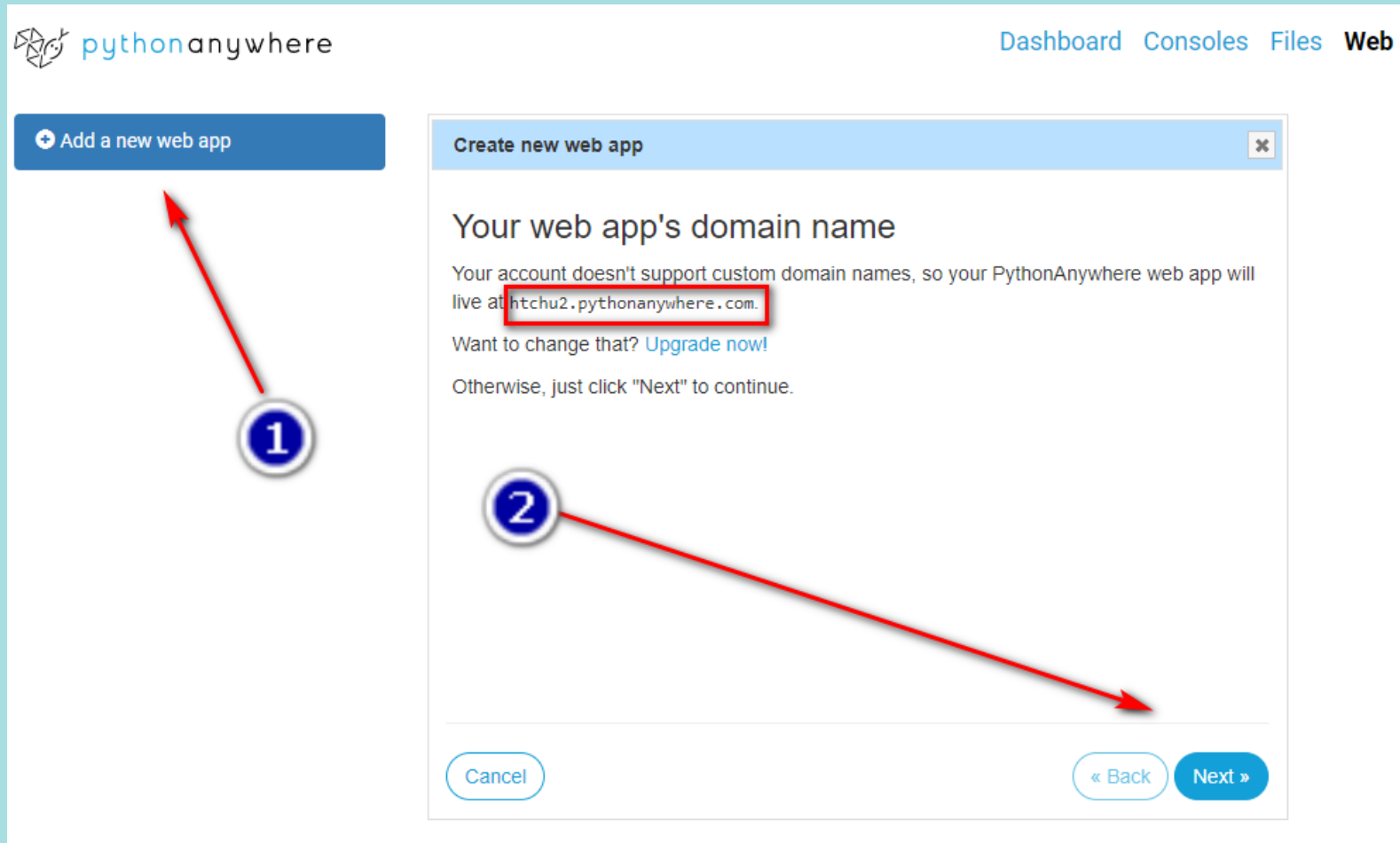
+ Add a new web app

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 side, 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. On the right side, a modal window titled 'Create new web app' is open. Inside the modal, the text reads: '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!](#)', 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.



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!

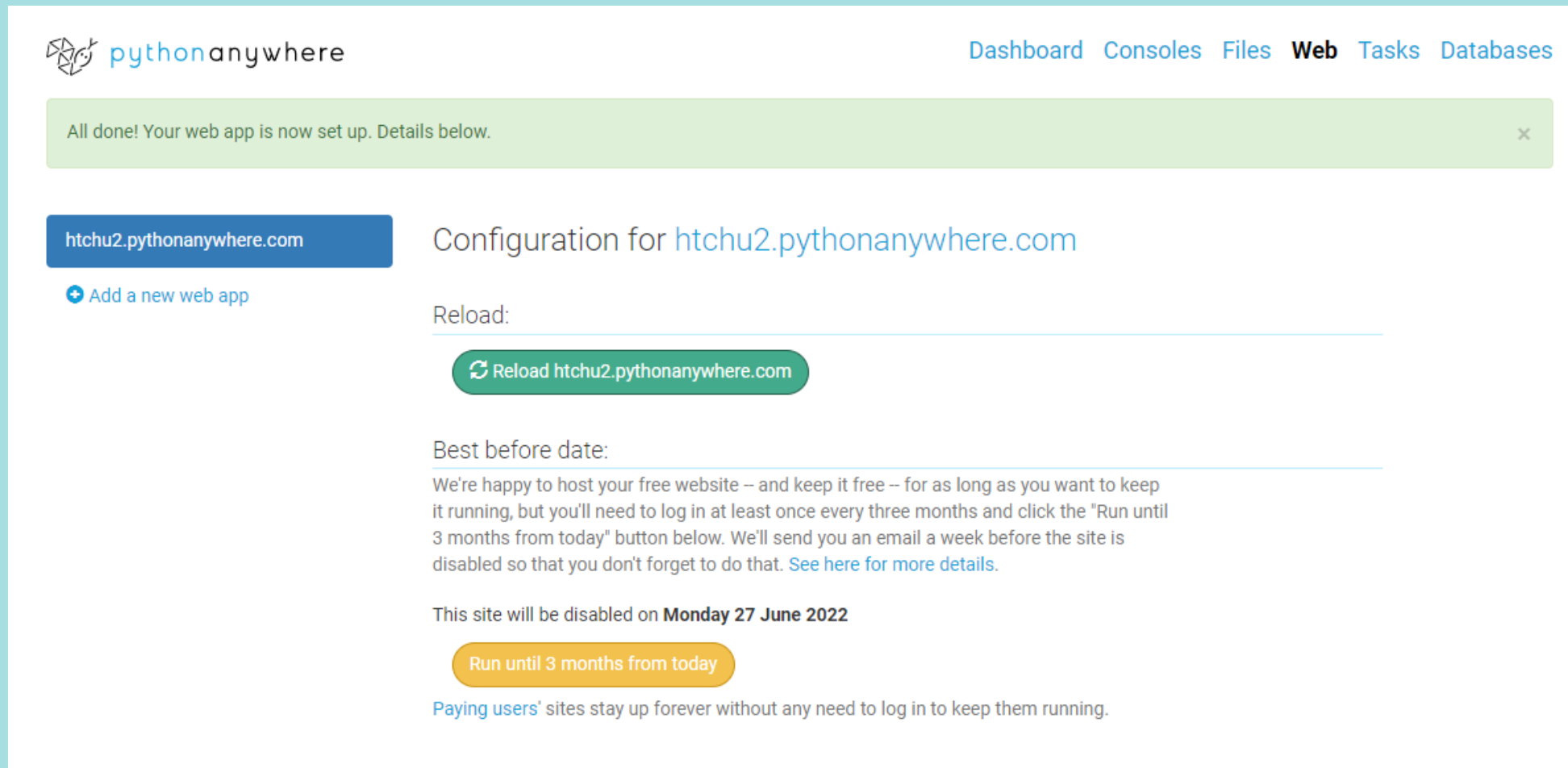
Select a Python version

- » 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

 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.](#)

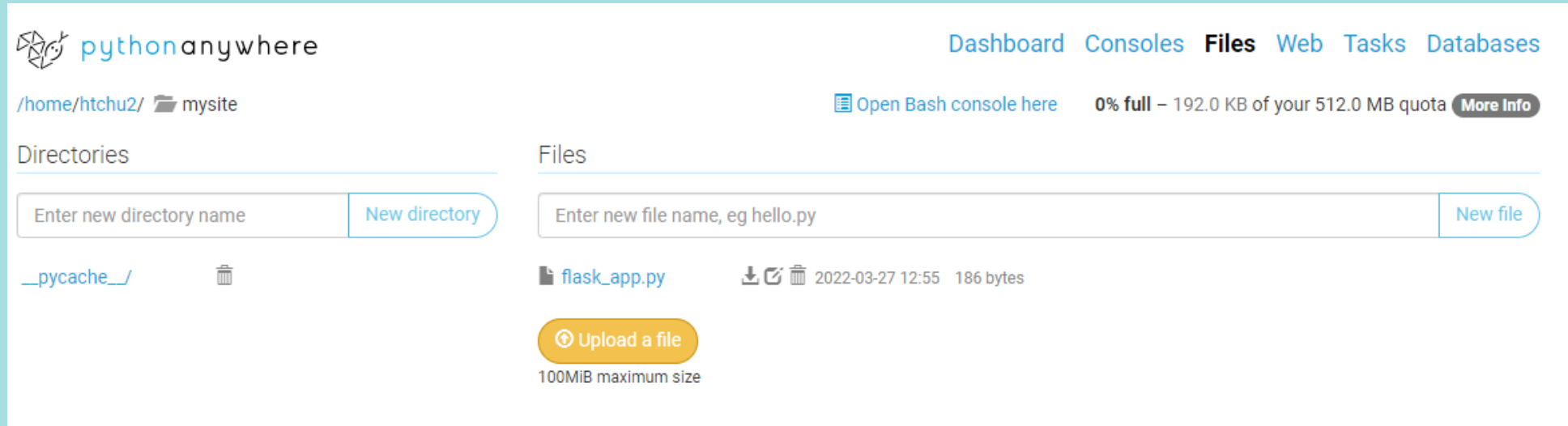
This site will be disabled on **Monday 27 June 2022**

 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



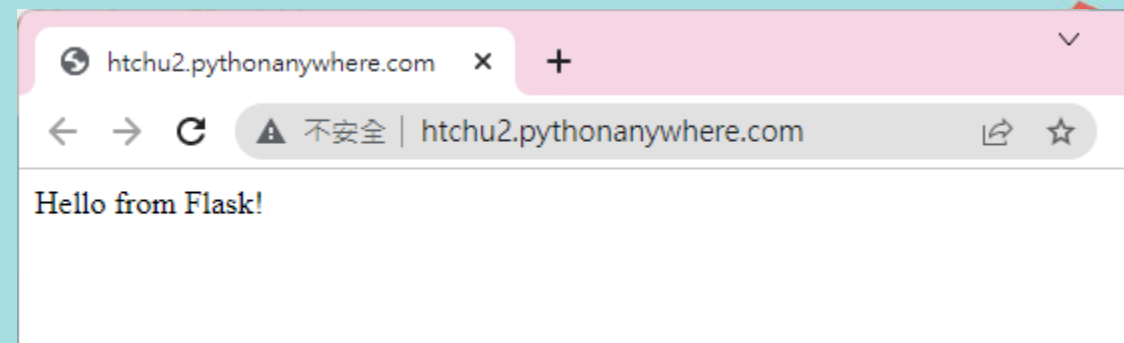
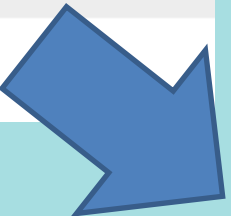
The screenshot displays the PythonAnywhere web interface. At the top left is the PythonAnywhere logo. To its right are navigation links: Dashboard, Consoles, Files (highlighted), Web, Tasks, and Databases. Below the logo, the current path is shown as `/home/htchu2/` with a folder icon and the name `mysite`. 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", followed by a "More Info" button. The main content area is divided into two panels. The left panel, titled "Directories", contains a text input field for "Enter new directory name" and a "New directory" button. Below this, a directory named `__pycache__` is listed with a trash icon. The right panel, titled "Files", contains a text input field for "Enter new file name, eg hello.py" and a "New file" button. Below this, a file named `flask_app.py` is listed with download, edit, and delete icons, along with its timestamp "2022-03-27 12:55" and size "186 bytes". At the bottom of the Files panel is an "Upload a file" button and the text "100MiB maximum size".



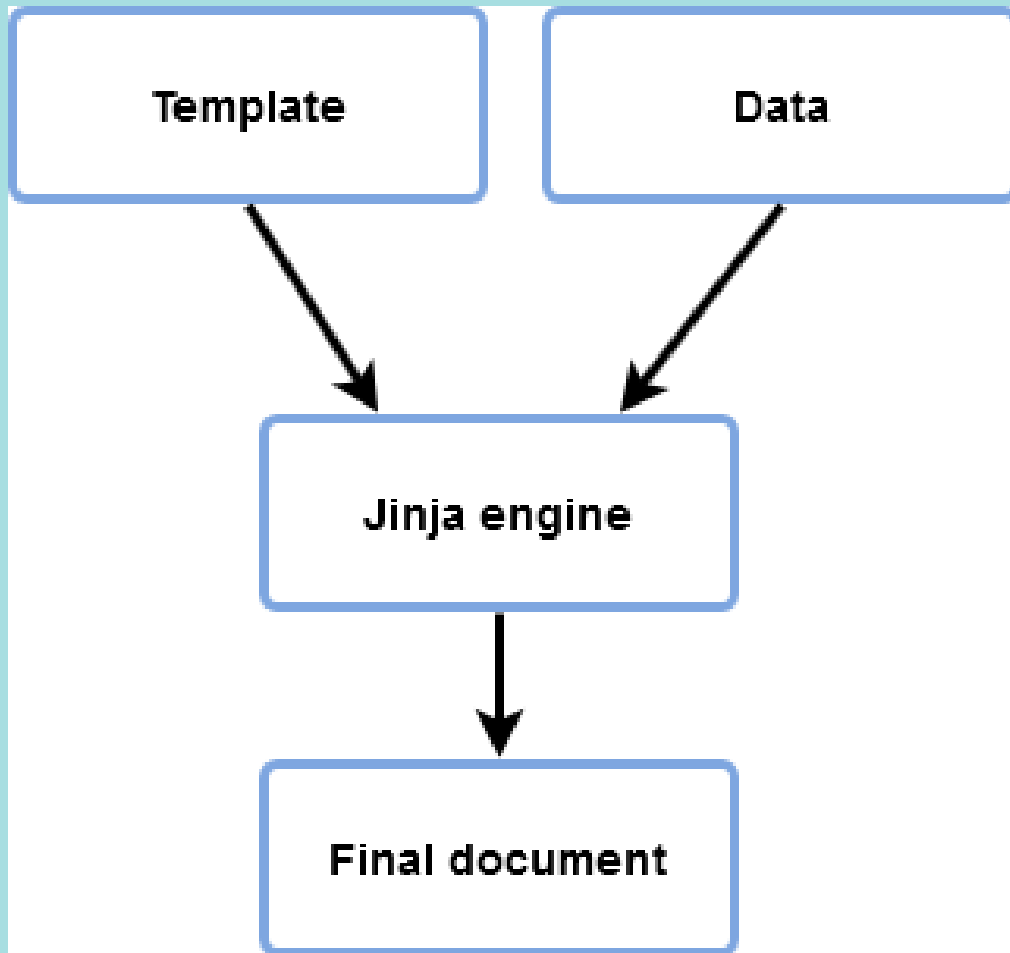
Step 6: check the program and the web app

 /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
```



2. Jinja template engine

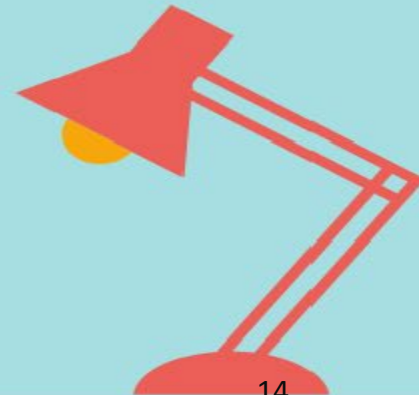


- Jinja2 essentially needs two source ingredients, template and data that will be used to render the final document.



Jinja-template engine

- **delimiters**
 - `{% ... %}` for Statements
 - `{{ ... }}` for Expressions to print to the template output
 - `{# ... #}` for Comments not included in the template output
 - `# ... ##` for Line Statements
- **extensions**
 - `{% extends "filename" %}`
- **blocks**
 - `{% block blockname %}`
 - `{% endblock %}`



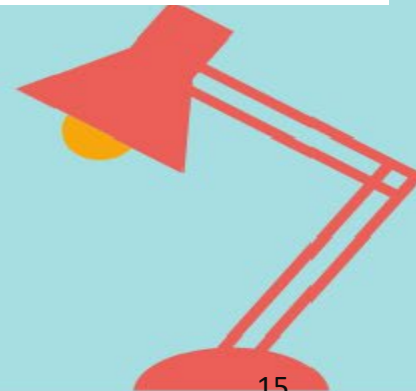
A template

```
<h1>{{ username }}'s profile</h1>
{% if bio %}
    <p>{{ bio }}</p> {# The indentation here is for readability only, not
necessary #}
{% else %}
    <p>The self-introduction is empty. </p>
{% endif %} {# Most Jinja statements need to declare closing #}
```

{{ ... }} is used to mark variables.

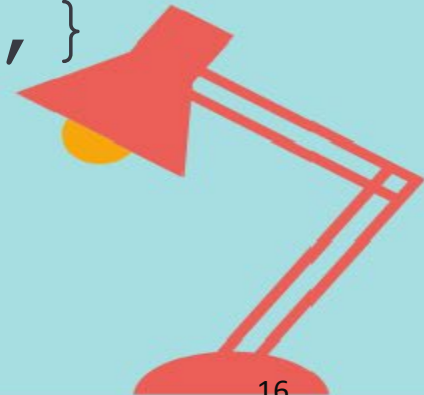
{% ... %} is used to mark statements, such as if statements, for statements, etc.

{# ... #} is used to write comments.

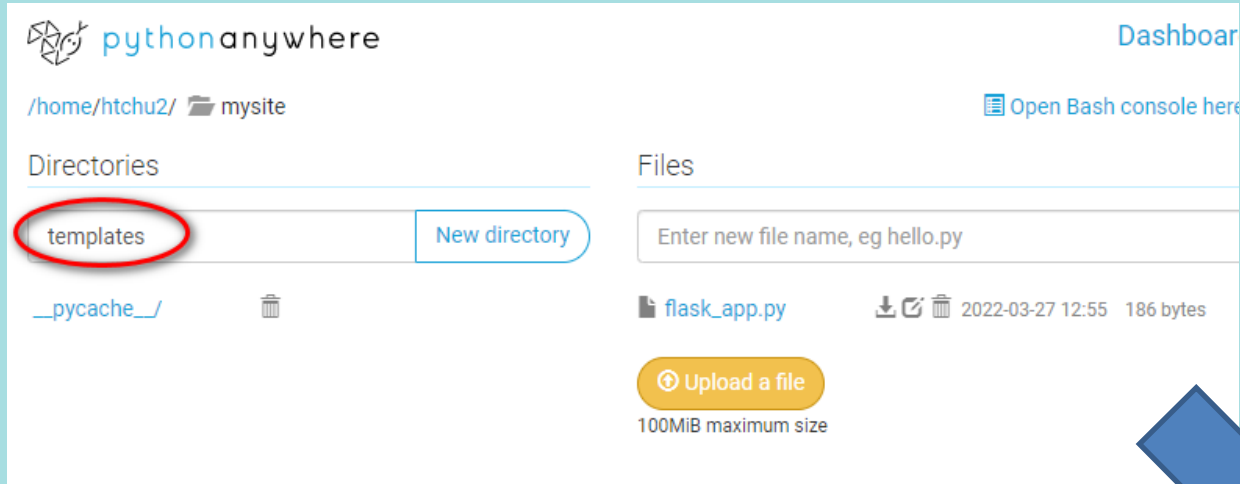


Data

- `data = {`
- `"name": "Hsueh-Ting Chu",`
- `"expertise": "University teacher",`
- `"location": "Taichung, Taiwan",`
- `"web": "http://htchu.pythonanywhere.com/",`
- `"github": "https://github.com/htchu/", }`



Step 7: add html templates



pythonanywhere Dashboard

/home/htchu2/ mysite

Open Bash console here

Directories

- templates
- __pycache__

New directory

Files

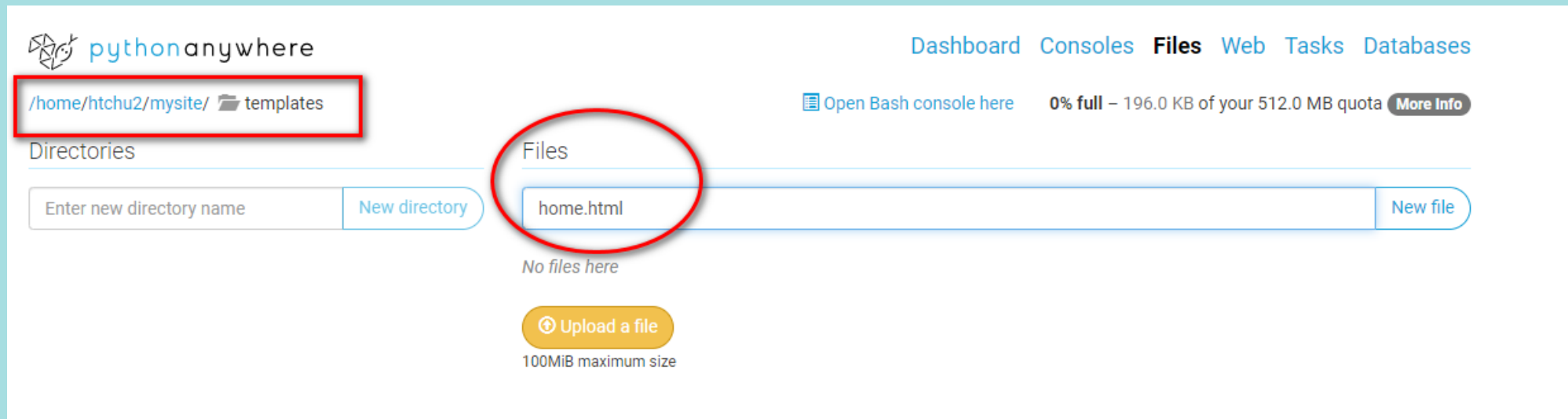
Enter new file name, eg hello.py

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

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <title>Bootstrap-Flask Demo Application</title>
  <!-- Latest compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/css/bootstrap.min.css"
integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">
  <link rel=stylesheet type=text/css href="{{ url_for('static',
filename='style.css') }}">
</head>
<body>
```

Step 9: create a static folder and add a css file

pythonanywhere

Dashboard Consoles **Files** Web Tasks Databases

/home/htchu/mysite/ **static** 1

Open Bash console here 15% full – 78.2 MB of your 512.0 MB quota [More Info](#)

Directories

Enter new directory name [New directory](#)

Files

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

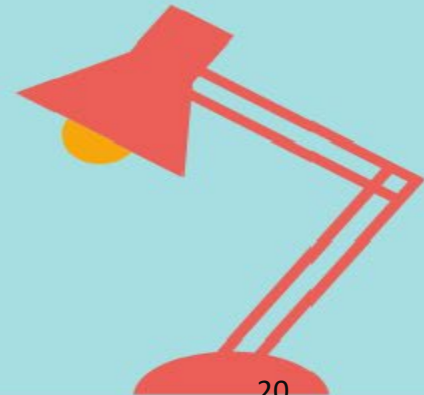
style.css 2 2022-04-19 00:31 952 bytes

[Upload a file](#)


100MiB maximum size

Step 10: edit style.css


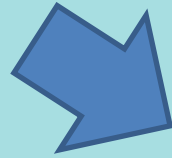
```
body{  
    margin-top:20px;  
    color: #1a202c;  
    text-align: left;  
    background-color: #e2e8f0;  
}  
.main-body {  
    padding: 15px;  
}
```



Step 11: 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  
5  
6 app = Flask(__name__)  
7  
8 @app.route('/')  
9 def hello_world():  
10     return 'Hello from Flask!'  
11  
12
```

 /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 12: Reload web

 pythonanywhere

Dashboard Consoles Files **Web**

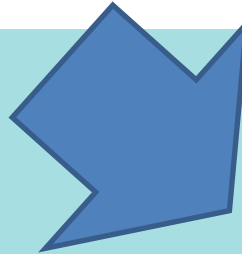
htchu2.pythonanywhere.com

+ Add a new web app

Configuration for htchu2.pythonanywhere.com

Reload:

↻ Reload htchu2.pythonanywhere.com



The Jinja2 Template Engine

```
{% extends "layout.html" %}
{% block body %}
    <ul>
        {% for user in users %}
            <li><a href="{{ user.url }}">{{ user.username }}</a></li>
        {% endfor %}
    </ul>
{% endblock %}
```

{% ... %} for Statements

{{ ... }} for Expressions to print to the template output

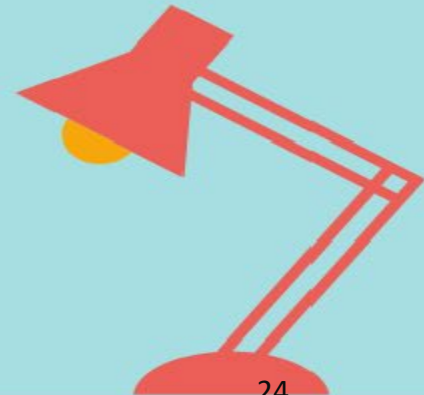
{# ... #} for Comments not included in the template output



Variables and Tests

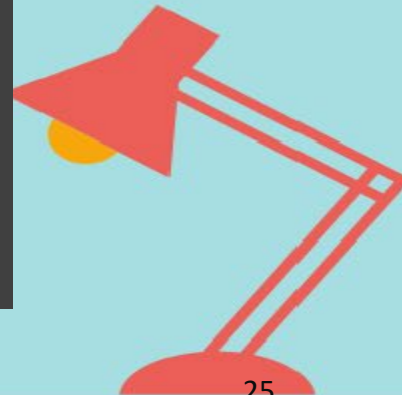
```
{{ foo.bar }}  
{{ foo['bar'] }}
```

```
{% if loop.index is divisibleby 3 %}  
{% if loop.index is divisibleby(3) %}
```



Template and Block

```
{% extends "base.html" %}
{% block title %}Index{% endblock %}
{% block head %}
    {{ super() }}
    <style type="text/css">
        .important { color: #336699; }
    </style>
{% endblock %}
{% block content %}
    <h1>Index</h1>
    <p class="important">
        Welcome to my awesome homepage.
    </p>
{% endblock %}
```





Thanks!

Q&A

