

A row of stylized books in various colors (white, red, blue, yellow) with some having decorative patterns like stripes or dots, arranged on a shelf.

2023-Spring Advanced Computer Programming (Week 15)

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-Databases
- W16-SQLAlchemy
- W17-Project development
- W18-Final report and test





python



Flask

web development,
one drop at a time



免費的線上 Python 執行環境

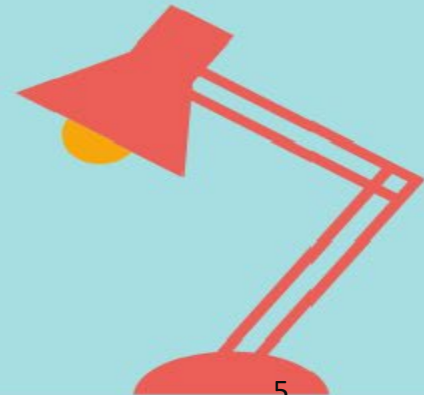
PythonAnywhere

- 免費帳號的功能受到如下限制：
 - 只能建立一個 App (應用程式)
 - 網外存取 Internet 有限制
 - CPU 與儲存有限制 (一天 100 秒 CPU 時間, 512MB 儲存)
 - 不提供 Jupyter (但有 IPython)
 - 只能有兩個 Console (Bash 與 Python)



Flask framework

- Required
 - **Jinja**-template engine
 - **Werkzeug**-WSGI toolkit
- Optional
 - **sqlalchemy**-SQL toolkit
 - marshmallow: simplified object serialization
 - Celery-task queue



Flask extensions

- **Flask-Bootstrap:** Bootstrap
- **Flask-WTF:** WTForms including CSRF, file upload, and reCAPTCHA
- **Flask-Moment:** Localization of Dates and Times
- **Flask-Babel:** Internationalization and localization support
- **Flask-DebugToolbar:** In-browser debugging tools
- **Flask-Assets:** Integration of CSS and JavaScript assets
- **Flask-Session:** implementation of user sessions with server-side storage
- **Flask-SocketIO:** Socket.IO server implementation with support for WebSocket and long-polling



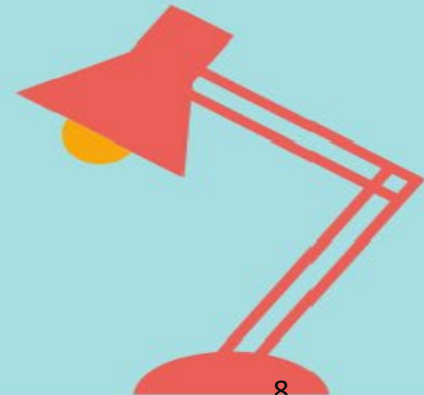
Flask

- Flask is a class with
 - run() function
 - route() functions
- Flask is a command
 - flask run
 - flask routes
 - flask shell



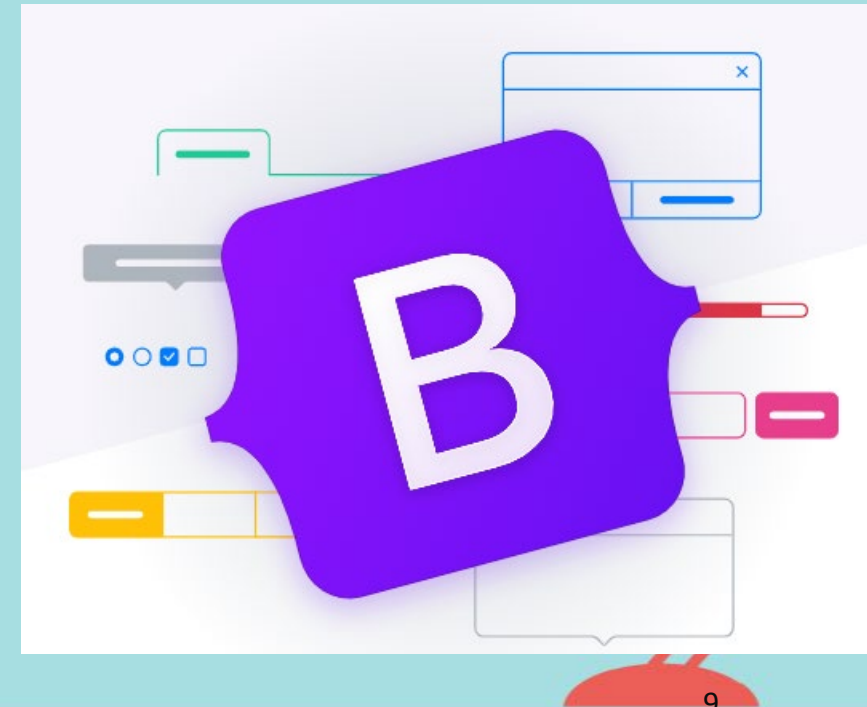
Flask environment

- Bash (Linux/Mac)
 - export FLASK_APP= appname
 - export FLASK_ENV=development
 - flask run
- Windows command
 - set FLASK_APP=appname
 - set FLASK_ENV=development
 - flask run



Bootstrap

- The most popular front-end toolkit in the world.
- Quickly design and customize responsive mobile-first sites
- Currently v5.1.3



Bootstrap-Flask

Bootstrap-Flask

license MIT pypi v2.0.2 build passing coverage 92%

Bootstrap-Flask is a collection of Jinja macros for Bootstrap 4 & 5 and Flask. It helps you to render Flask-related data and objects to Bootstrap markup HTML more easily:

- Render Flask-WTF/WTForms form object to Bootstrap Form.
- Render data objects (dict or class objects) to Bootstrap Table.
- Render Flask-SQLAlchemy `Pagination` object to Bootstrap Pagination.
- etc.

Installation

```
$ pip install -U bootstrap-flask
```

Example

Register the extension:

```
from flask import Flask
# To follow the naming rule of Flask extension, although
# this project's name is Bootstrap-Flask, the actual package
# installed is named `flask_bootstrap`.
from flask_bootstrap import Bootstrap5

app = Flask(__name__)
bootstrap = Bootstrap5(app)
```

Installation

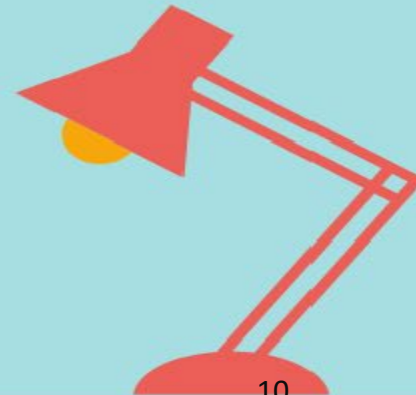
```
$ pip install -U bootstrap-flask
```

Example

Register the extension:

```
from flask import Flask
from flask_bootstrap import Bootstrap5
```

```
app = Flask(__name__)
bootstrap = Bootstrap5(app)
```



Jinja-the template language

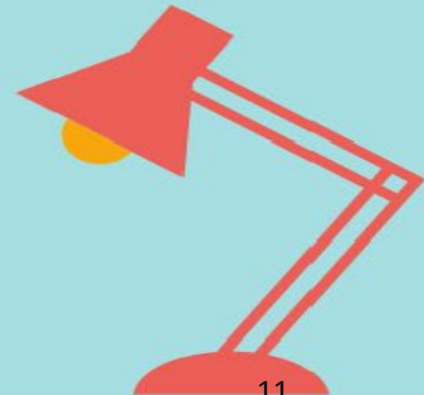
```
from jinja2 import Environment, PackageLoader, select_autoescape
env = Environment(
    loader=PackageLoader("yourapp"),
    autoescape=select_autoescape()
)
```

To load a template from this environment, call the `get_template()` method, which returns the loaded Template.

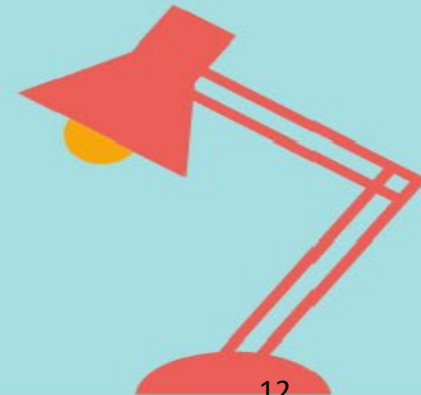
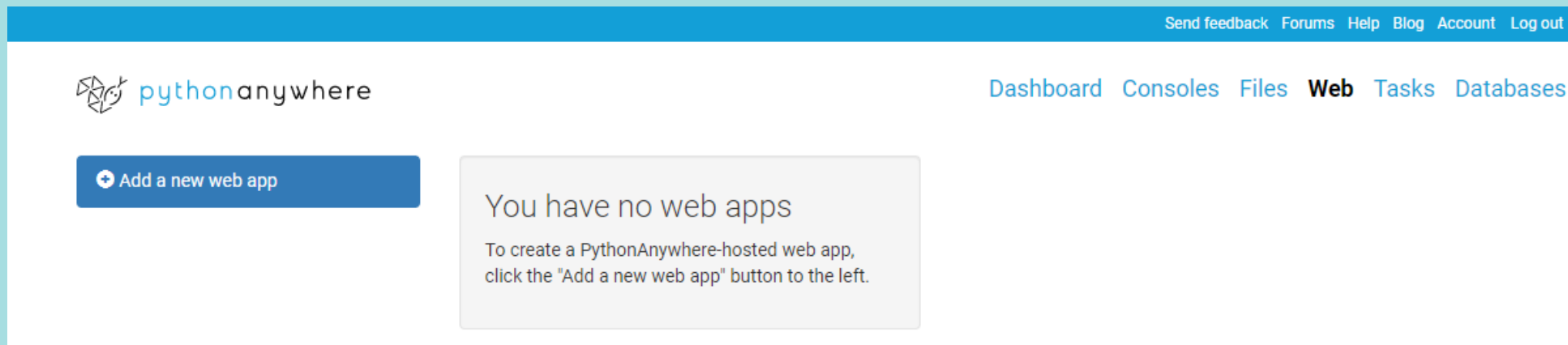
```
template = env.get_template("mytemplate.html")
```

To render it with some variables, call the `render()` method.

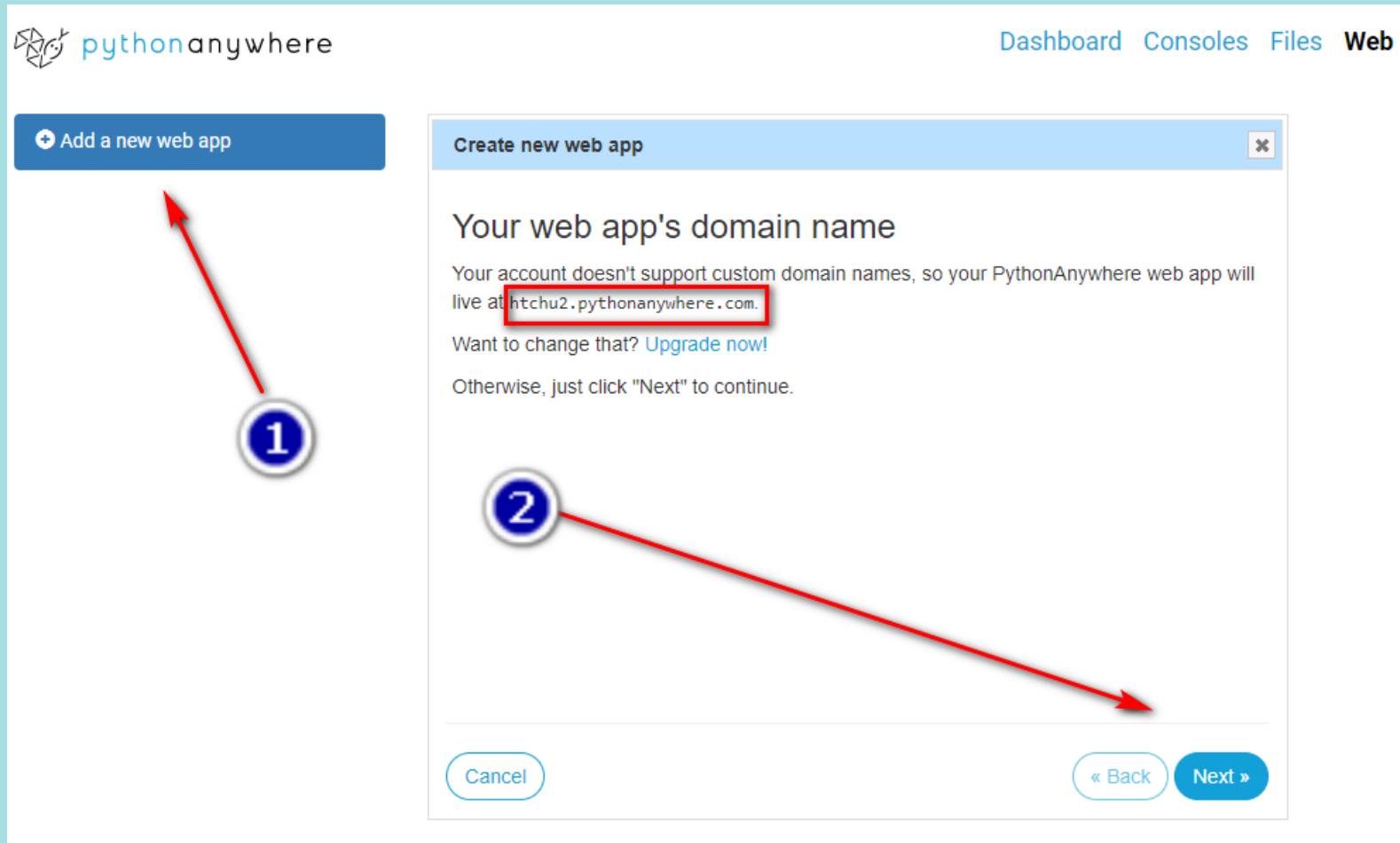
```
print(template.render(the="variables", go="here"))
```



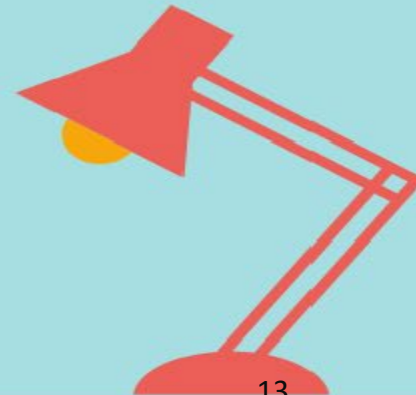
Step 1: Go to Web tab



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.

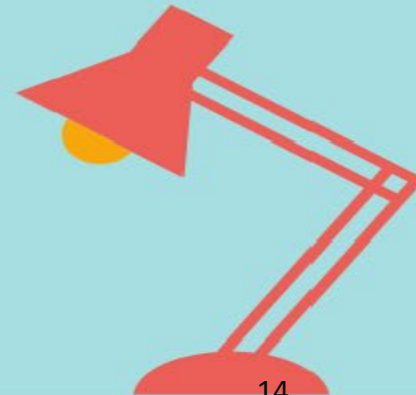
- » Django
- » web2py
- » Flask
- » Bottle
- » **Manual configuration** (including virtualenvs)

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

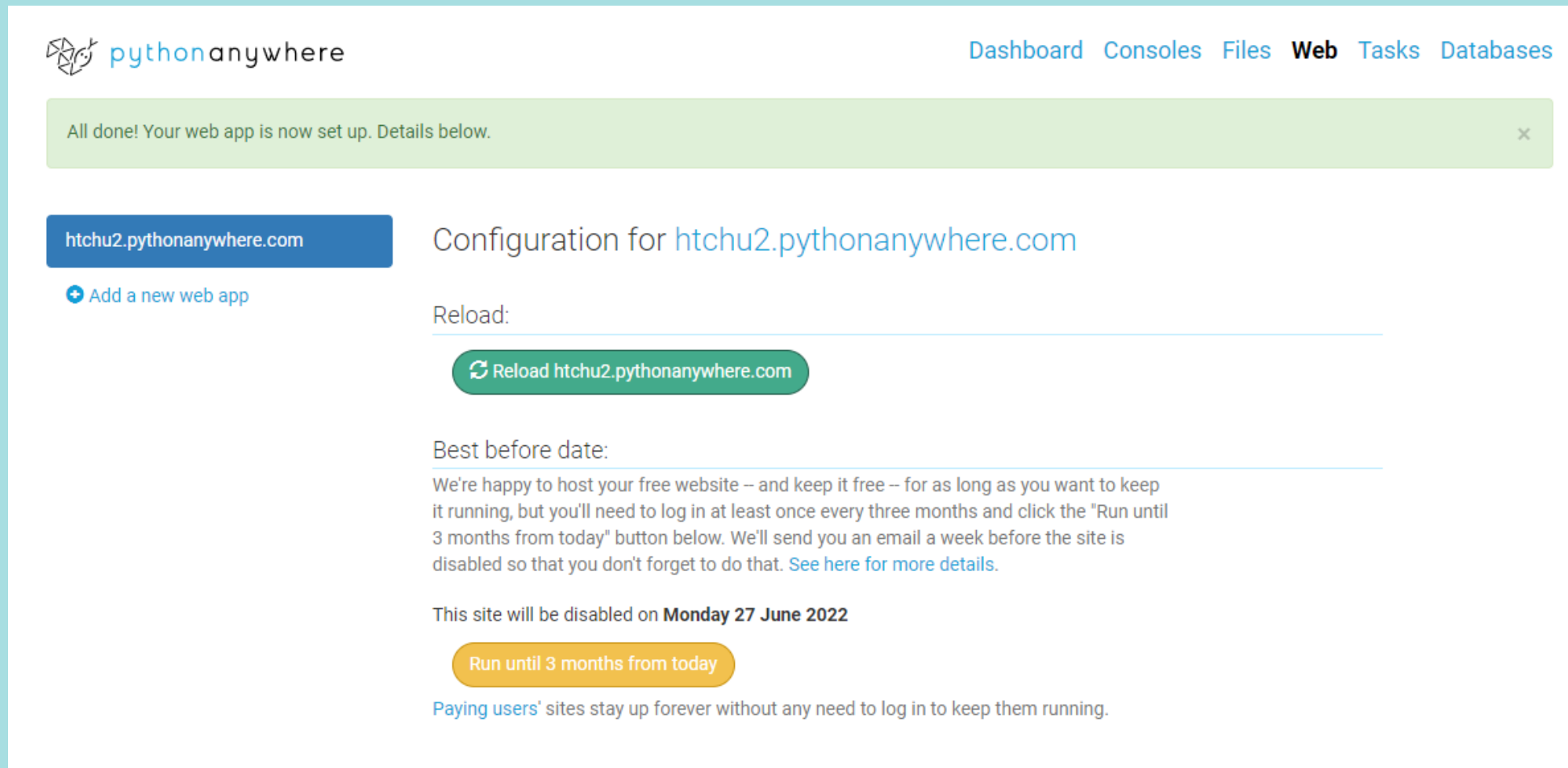
Select a Python version

- » 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



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.](#)

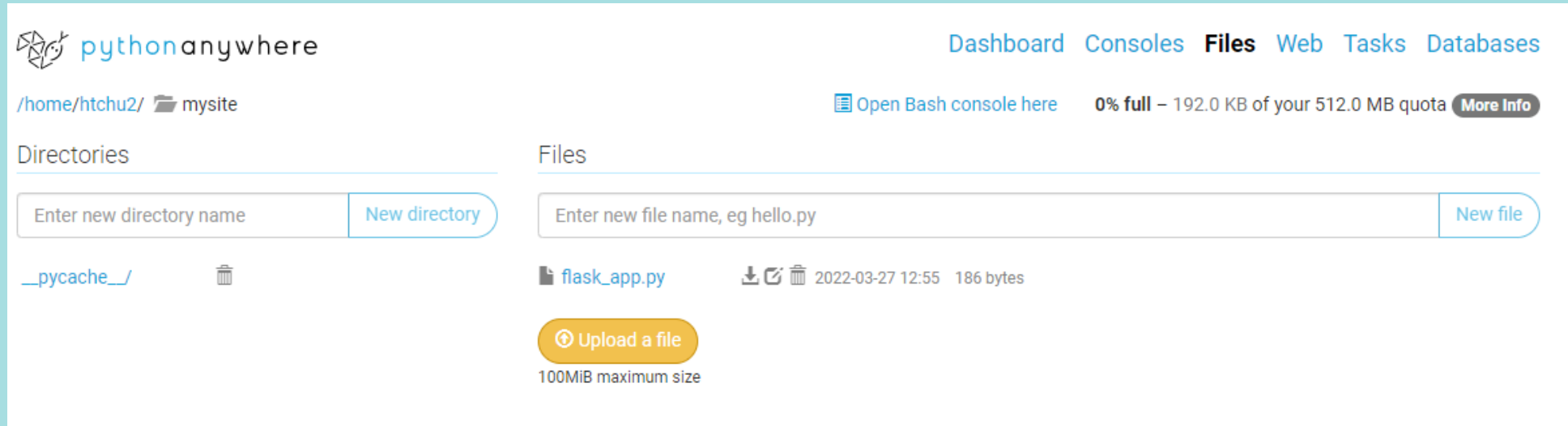
This site will be disabled on **Monday 27 June 2022**

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

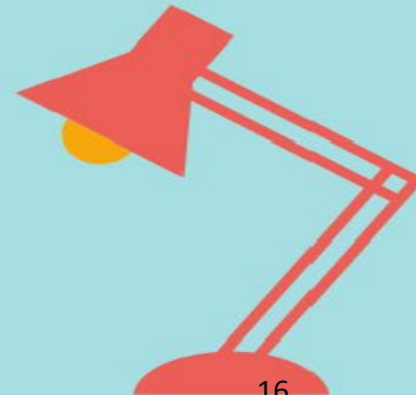


The screenshot shows the PythonAnywhere web interface. At the top left is the PythonAnywhere logo. To its right are navigation links: Dashboard, Consoles, **Files**, Web, Tasks, and Databases. Below the logo, the current path is shown as `/home/htchu2/` with a folder icon and the name `mysite`. On the 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 section: It features a text input field labeled "Enter new directory name" and a "New directory" button. Below this, a directory named `__pycache__` is listed with a trash icon next to it.

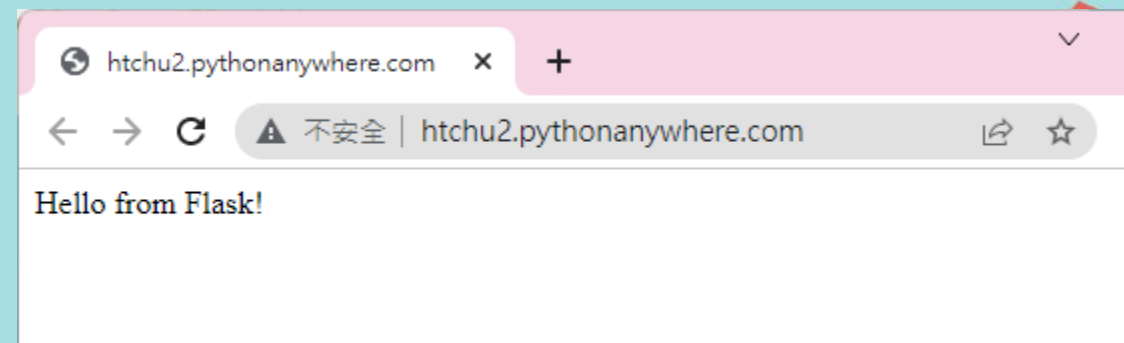
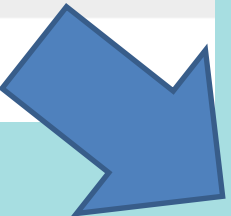
Files section: It features a text input field labeled "Enter new file name, eg hello.py" and a "New file" button. Below this, a file named `flask_app.py` is listed. To the left of the filename is a file icon. To the right are icons for download, copy, and delete, followed by the file's metadata: "2022-03-27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



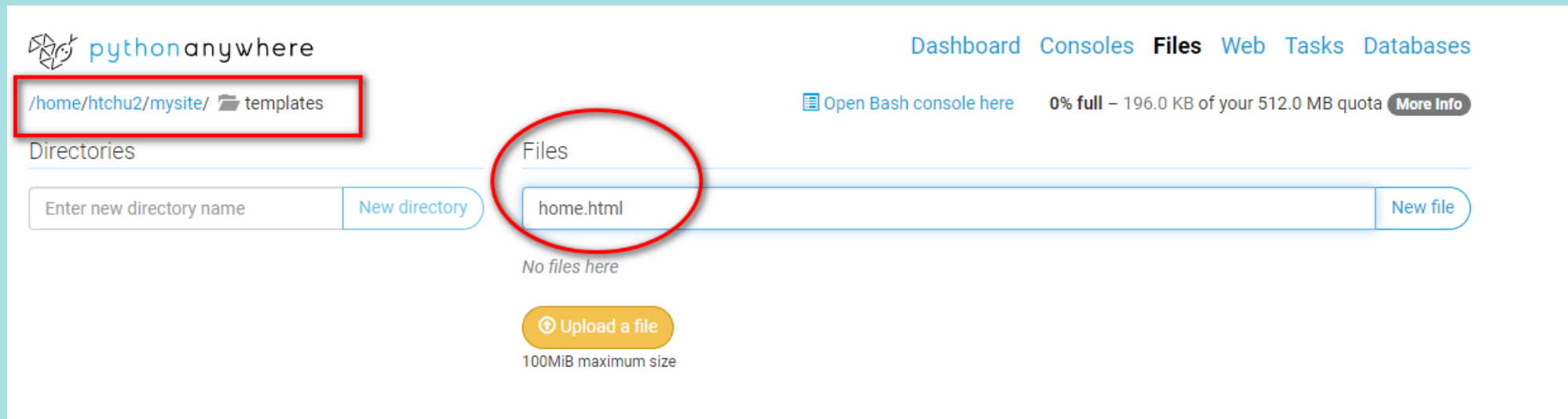
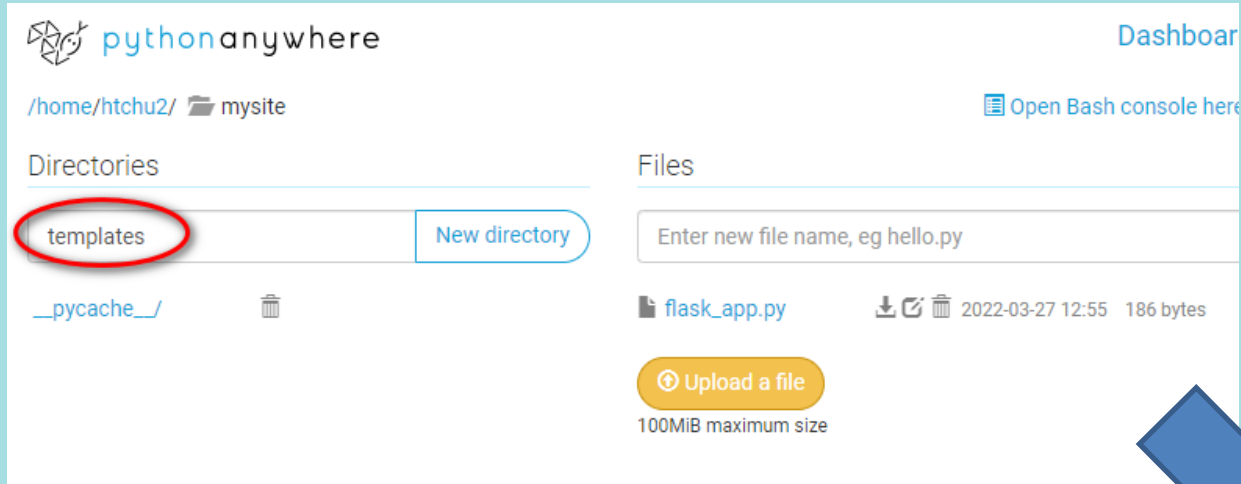
Step 6: check the program and the web app

 /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



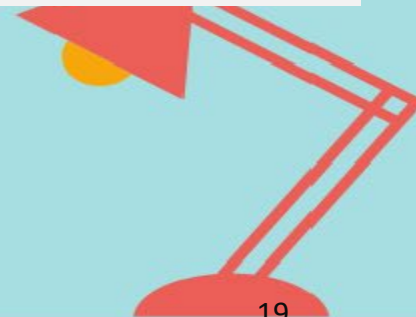
Step 8: edit home.html




- `<h1>Top 10 HTML tags</h1>`
`

`In this article we will explain our list of top 10 HTML tags.`

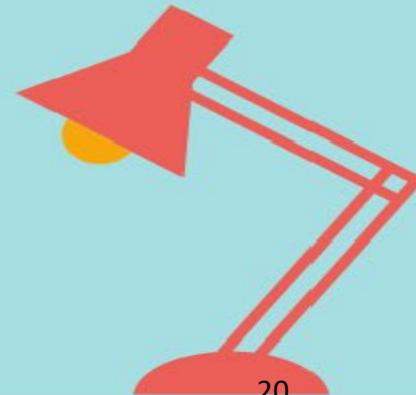
` `<h2>First tag is`
```</h2>` **Bold** is all about making words more important.




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

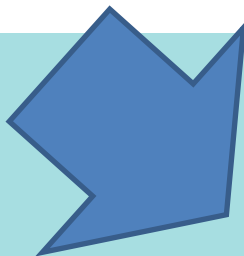
htchu2.pythonanywhere.com

+ 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

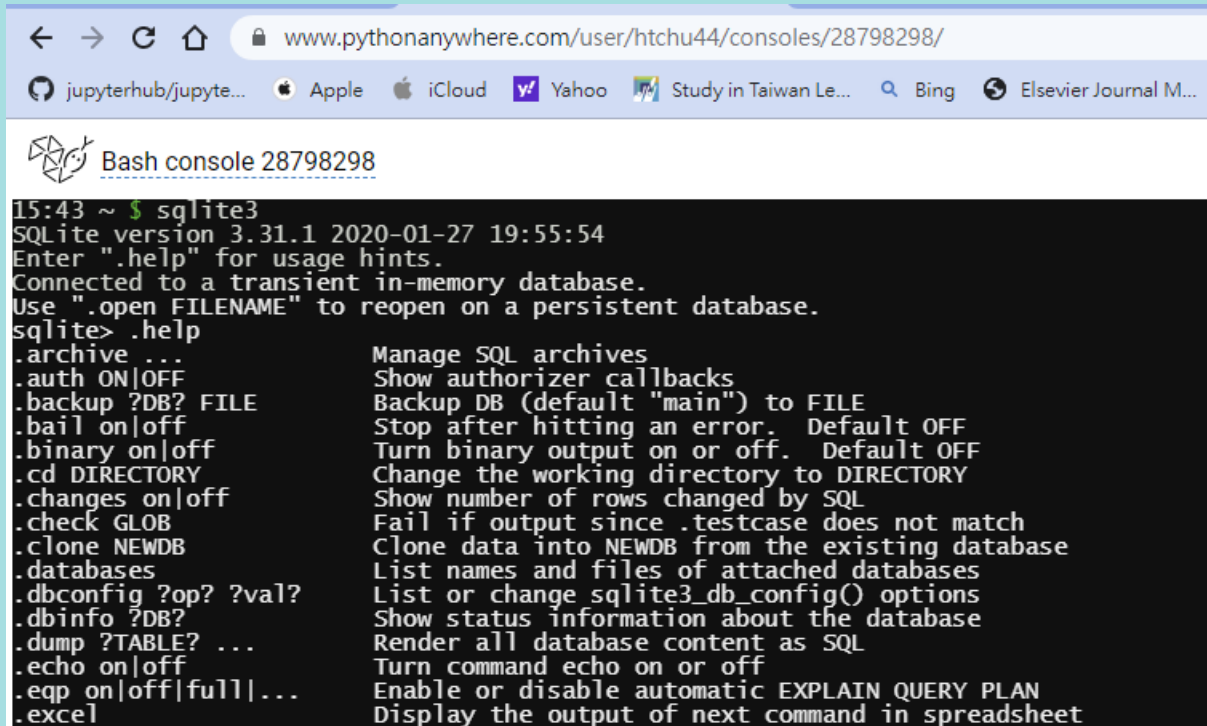
"A designer knows he has achieved perfection not when there is nothing left to add, but when there is nothing left to take away." - Saint Exupéry

SQLite

- SQLite是小型關聯式數據庫管理系統，它包含在一個相對小的C程式庫中。與許多其它數據庫管理系統不同，SQLite不是一個客戶端/伺服器結構的數據庫引擎，而是被整合在用戶程式中。
- SQLite實現了大多數SQL標準。它使用動態的、弱類型的SQL語法。它作為嵌入式數據庫，是應用程式，如網頁瀏覽器，在本地/客戶端儲存資料的常見選擇。它可能是最廣泛部署的數據庫引擎，因為它正在被一些流行的瀏覽器、作業系統、嵌入式系統所使用。
- SQLite是D. Richard Hipp建立的。

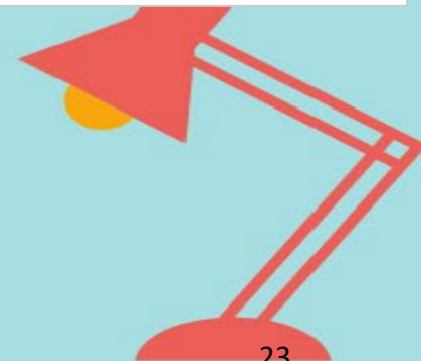


Meta Commands



```
15:43 ~ $ sqlite3
SQLite version 3.31.1 2020-01-27 19:55:54
Enter ".help" for usage hints.
Connected to a transient in-memory database.
Use ".open FILENAME" to reopen on a persistent database.
sqlite> .help
.archive ...      Manage SQL archives
.auth ON|OFF      Show authorizer callbacks
.backup ?DB? FILE Backup DB (default "main") to FILE
.bail on|off      Stop after hitting an error. Default OFF
.binary on|off    Turn binary output on or off. Default OFF
.cd DIRECTORY     Change the working directory to DIRECTORY
.changes on|off   Show number of rows changed by SQL
.check GLOB       Fail if output since .testcase does not match
.clone NEWDB      Clone data into NEWDB from the existing database
.databases        List names and files of attached databases
.dbconfig ?op? ?val? List or change sqlite3_db_config() options
.dbinfo ?DB?     Show status information about the database
.dump ?TABLE? ... Render all database content as SQL
.echo on|off      Turn command echo on or off
.eqp on|off|full|... Enable or disable automatic EXPLAIN QUERY PLAN
.excel           Display the output of next command in spreadsheet
```

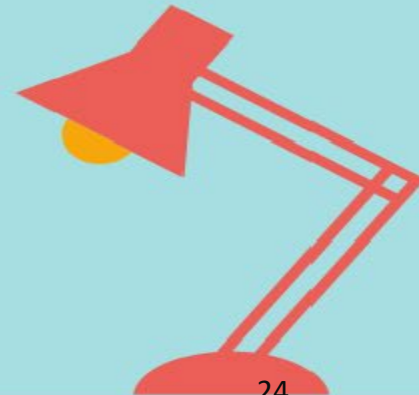
Command	Description
.show	Displays current settings
.databases	Provides database names and files
.quit	Quit sqlite3 program
.tables	Show current tables
.schema	Display schema of table
.header	Display the output table header
.mode	Select mode for the output table
.dump	Dump database in SQL text format



SQL(Structured Query Language) commands

case insensitive

- DDL-Data Definition Language
 - CREATE TABLE
 - DROP TABLE
 - ALTER TABLE
- DML-Data Manipulation Language
 - INSERT
 - UPDATE
 - DELETE
- DQL-Data Query Language
 - SELECT



SQL Tutorial

SQL Tutorial

SQL HOME

SQL Intro

SQL Syntax

SQL Select

SQL Select Distinct

SQL Where

SQL And, Or, Not

SQL Order By

SQL Insert Into

SQL Null Values

SQL Update

SQL Delete

SQL Select Top

SQL Min and Max

SQL Count, Avg, Sum

SQL Like

SQL Wildcards

SQL In

SQL Between

SQL Aliases

SQL Joins

SQL Inner Join

SQL Left Join

SQL Tutorial

[← Home](#)

SQL is a standard language for storing, manipulating and retrieving data in databases.

Our SQL tutorial will teach you how to use SQL in: MySQL, SQL Server, MS Access, Oracle, Sybase, Informatica, etc.

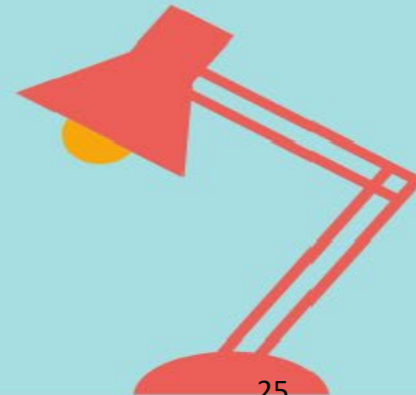
[Start learning SQL now »](#)

Examples in Each Chapter

With our online SQL editor, you can edit the SQL statements, and click on a button to view the result.

Example

```
SELECT * FROM Customers;
```



SQL Examples

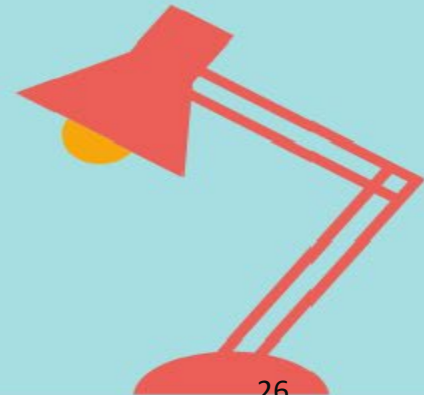
```
CREATE TABLE news (  
    news_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,  
    news_caption TEXT NOT NULL,  
    news_time DATETIME DEFAULT CURRENT_TIMESTAMP,  
    news_url TEXT NULL);
```

```
ALTER TABLE news ADD news_txt TEXT NULL;
```

```
INSERT INTO news (news_caption, news_url)  
VALUES ('N1', 'https://aaa.com/xxx111.aspx');
```

```
INSERT INTO news (news_caption, news_url)  
VALUES ('N1', 'https://aaa.com/xxx111.aspx');
```

```
SELECT * FROM news;
```



Python sqlite3 -SQLite DB-API 2.0

```
import sqlite3

con = sqlite3.connect('example.db')

cur = con.cursor()

# Create table

cur.execute('CREATE TABLE stocks (date text, trans text, symbol text, qty real, price real)')

# Insert a row of data

cur.execute("INSERT INTO stocks VALUES ('2006-01-05','BUY','RHAT',100,35.14)")

# Save (commit) the changes

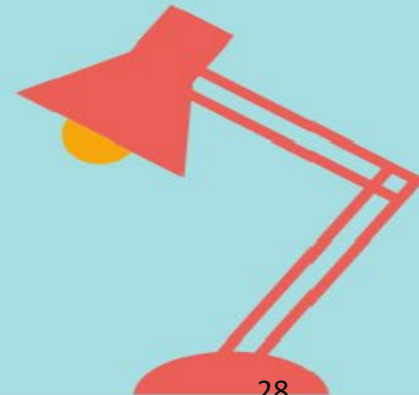
con.commit()

# We can also close the connection if we are done with it.
# Just be sure any changes have been committed or they will be lost.

con.close()
```

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

