

National Institute of Allergy and Infectious Diseases

NIH Hour of Code: Python Programming

NIAID



National Institute of
Allergy and
Infectious Diseases

R. Burke Squires

Bioinformatics and Computational Biosciences Branch (BCBB)

BCBB / OCICB / OSMO / OD / NIAID / NIH

Humans

Computers

Learn in a variety of ways



to-do list:

1. finish eng lit h/w
2. tidy room
3. call tim
4. etc

Can be forgetful

Can think creatively and independently



Can manage lots of different tasks simultaneously



Retain large amounts of data precisely

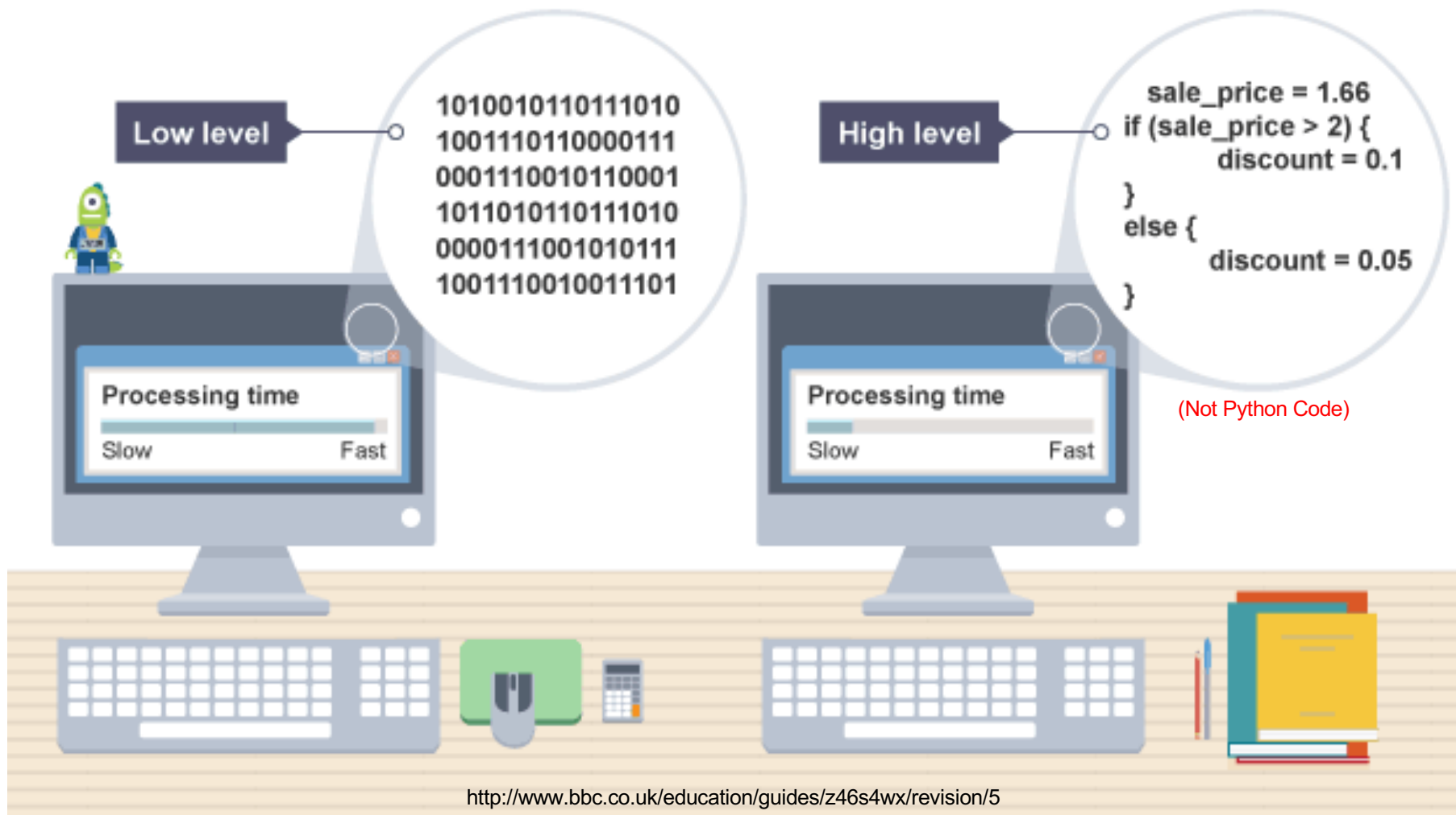


Need to be programmed and follow algorithms

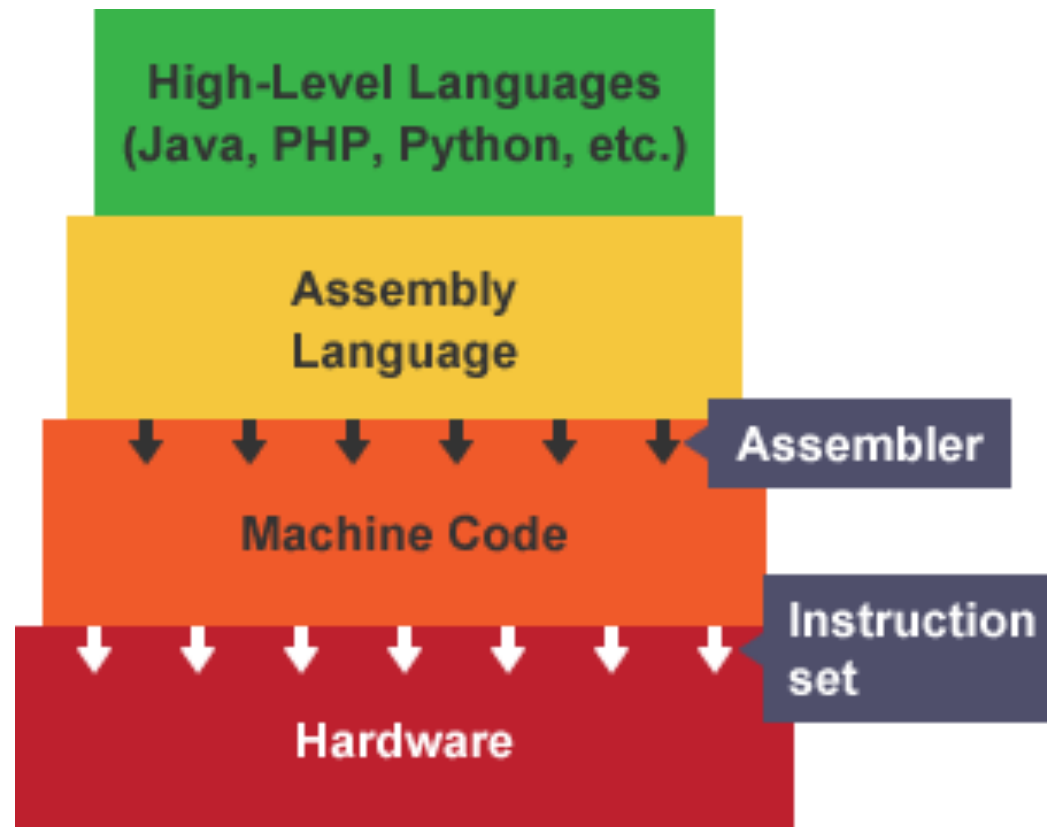


<http://www.bbc.co.uk/education/guides/z46s4wx/revision/5>

Why Do We Need To Program?



Why Do We Need To Program?



Why Python?

Why Python?

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Why Python?

Advantages of Python

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

- Source code is freely downloadable – python.org
 - You can fix, enhance, contribute
- Started in 1991
 - Guido van Rossum
 - Borrowed the best parts of many languages
- Available on (almost) all platforms – Mac, Windows, linux, raspberry pi, etc

Why Python?

Advantages of Python

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

- Design philosophy emphasizes code readability
 - Code is read many more times than it is written!
 - Indentation is required; enhances readability
- Very similar to English language; pseudocode

Printing “Hello World!” in Four Programming Languages

C++

```
#include <iostream>
int main()
{
    std::cout << "Hello World" <<
std::endl;
    return 0;
}
```

Python

```
print("Hello World!")
```

Java

```
public class HelloWorld {
    public static void main (String[]
args) {
        System.out.println("Hello
World!");
    }
}
```

R

```
print("Hello World!", quote = FALSE)
```

Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

- Supports multiple programming paradigms:
 - Object-oriented
 - Imperative
 - Functional programming
 - Procedural styles.
- Dynamic type system
- Automatic memory management
- Can write prototype script very fast
- Some packages are written in or enable you to write in C, very fast

Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

- Python has a large and comprehensive standard library
 - No need to write code yourself
 - No need to install packages

Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

- Python's standard library
 - data types
 - strings
 - networking
 - threads
 - operating
 - compression
 - GUI
 - arguments
 - system
 - complex
 - CGI
- FTP
- cryptography
- numbers
- testing
- multimedia
- databases
- CSV
- files
- calendar
- email
- XML
- serialization

Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

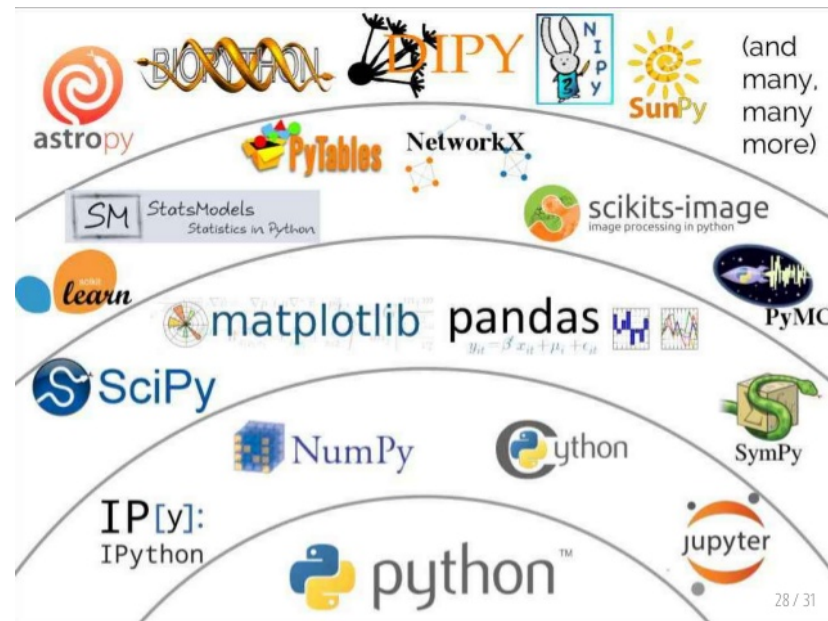
- PyPI – Python Package Index
 - 92,000+ packages 11/2016
 - Easily installable
- Great documentation
- Free video training - pyvideo.org
- Large support community on stackoverflow, etc
- Great conferences: SciPy, PyCon, PyData

Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically



Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically

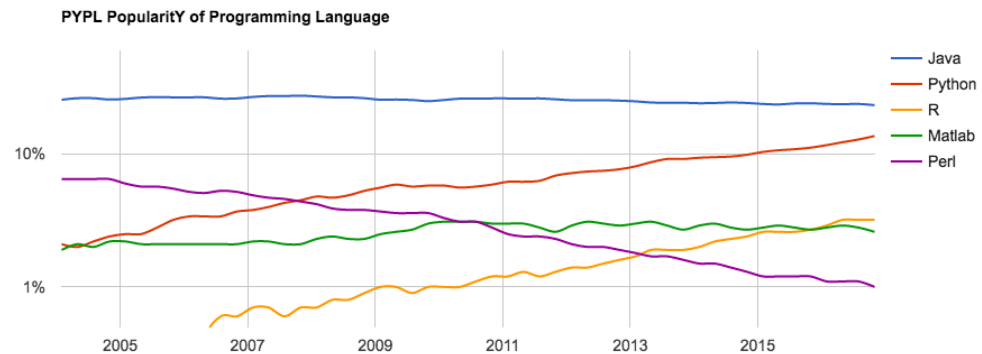
- Some of the companies using python
 - Google (Youtube)
 - Facebook (Tornado)
 - Dropbox
 - Yahoo
 - NASA
 - IBM
 - Mozilla
 - Quora
 - Instagram
 - Reddit
 - IML (Movies)

Why Python?

Advantages

- Open source
- Easy to learn
- Powerful, fast, flexible
- “Batteries included”
- Large, thriving community
- Powerful, mature scientific python stack
- Used by many companies
- Most popular language for...

Specifically



Zen of Python (by Tim Peters)

- Beautiful is better than ugly.
- Explicit is better than implicit.
- Simple is better than complex.
- Complex is better than complicated.
- Flat is better than nested.
- Sparse is better than dense.
- Readability counts.
- Special cases aren't special enough to break the rules.
- Although practicality beats purity.
- Errors should never pass silently.
- Unless explicitly silenced.
- In the face of ambiguity, refuse the temptation to guess.
- There should be one--and preferably only one--obvious way to do it.
- Although that way may not be obvious at first unless you're Dutch.
- Now is better than never.
- Although never is often better than *right* now.
- If the implementation is hard to explain, it's a bad idea.
- If the implementation is easy to explain, it may be a good idea.
- Namespaces are one honking great idea -- let's do more of those!

What Can You Do With Python?

- Quickly create (prototype) applications
- Bioinformatics workflows
- Image analysis
- Web scrapers
- Web Applications (Django, Pylons)
- Games (Eve Online – MMORPG, pygame)
- Software Development (Trac for Project Management)
- Object Databases (ZODB / Durus)
- Network Programming (Bittorent)
- Mobile applications
- Data science

Let's Code!

Programming in Python

- Open account at PythonAnywhere.org
- Create BASH console
 - Install biopython module
 - `pip3.5 install --user biopython`
- Go back to Dashboard
- Create a new file
 - Type in code
 - Run
- (Advanced) Update code to save to a file

Open Account at PythonAnywhere.org

The screenshot shows the PythonAnywhere website. The main heading is "Host, run, and code Python in the cloud!". Below this, it says "Get started for free. Our basic plan gives you access to machines with a full Python environment already installed. You can develop and host your website or any other code directly from your browser without having to install software or manage your own server." It also mentions "Need more power? Upgraded plans start at \$5/month." A red circle highlights a green button that says "Start running Python online in less than a minute!". Below this is a button that says "Watch our one-minute video". At the bottom, there are four sections: "Start hosting quickly", "Develop anywhere", "Teach and learn", and "Amazing support". Each section has a brief description and a "More" link.

pythonanywhere

Send feedback Forums Help Blog Pricing & signup Log in

Host, run, and code Python in the cloud!

Get started for free. Our basic plan gives you access to machines with a full Python environment already installed. You can develop and host your website or any other code directly from your browser without having to install software or manage your own server.

Need more power? Upgraded plans start at \$5/month.

Start running Python online in less than a minute!

Watch our one-minute video »

Not convinced? [Read what our users are saying!](#)

pythonanywhere

Consoles Files Web Schedule Databases

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

- ✓ [www.tdd-django-tutorial.com](#)
- ✓ [harry.pythonanywhere.com](#)
- ✓ [www.testster.com](#)
- ✓ [www.obeythetestinggoat.com](#)
- ✓ [dev.obeythetestinggoat.com](#)

[Reload www.bla.com](#)

You can see your web app at <http://www.bla.com/>

Details

The files for your web app are in `/home/harry/sweet_django_site`

It is configured via a WSGI file stored at: `/var/www/www_bla_com_wsgi.py`

It's running Python version 3.3

Start hosting quickly

Just write your application. No need to configure or maintain a web server — everything is set up and ready to go.

[More »](#)

Develop anywhere

Take your development environment with you! If you have a browser and an Internet connection, you've got everything you need.

[More »](#)

Teach and learn

PythonAnywhere is a fully-fledged Python environment, ready to go, for students and teachers — concentrate on teaching, not on installation hassles.

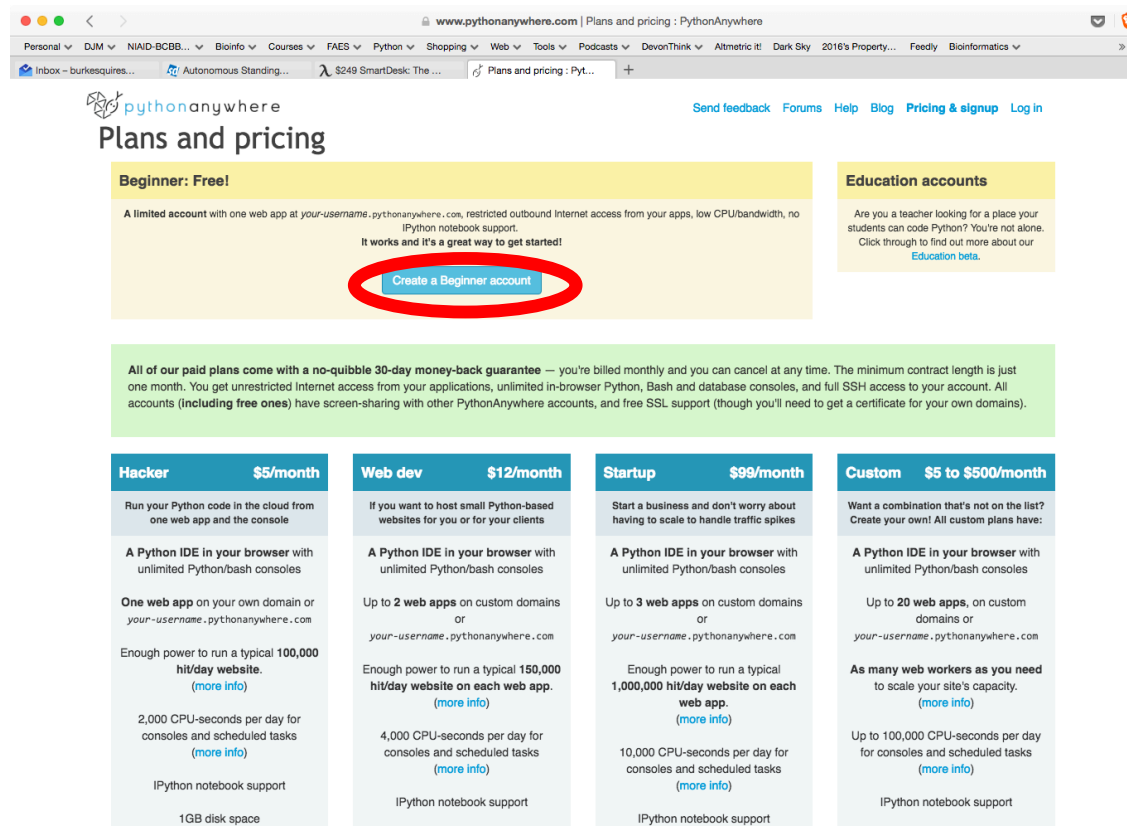
[More »](#)

Amazing support

Need help with PythonAnywhere? If you get in touch, you can talk directly with the development team. Help for developers, from developers.

[More »](#)

Open Account at PythonAnywhere.org



The screenshot shows the PythonAnywhere website's 'Plans and pricing' page. The 'Beginner: Free!' plan is highlighted with a red circle around the 'Create a Beginner account' button. The page also features a table of paid plans: Hacker (\$5/month), Web dev (\$12/month), Startup (\$99/month), and Custom (\$5 to \$500/month).

Beginner: Free!

A limited account with one web app at `your-username.pythonanywhere.com`, restricted outbound Internet access from your apps, low CPU/bandwidth, no IPython notebook support.

It works and it's a great way to get started!

[Create a Beginner account](#)

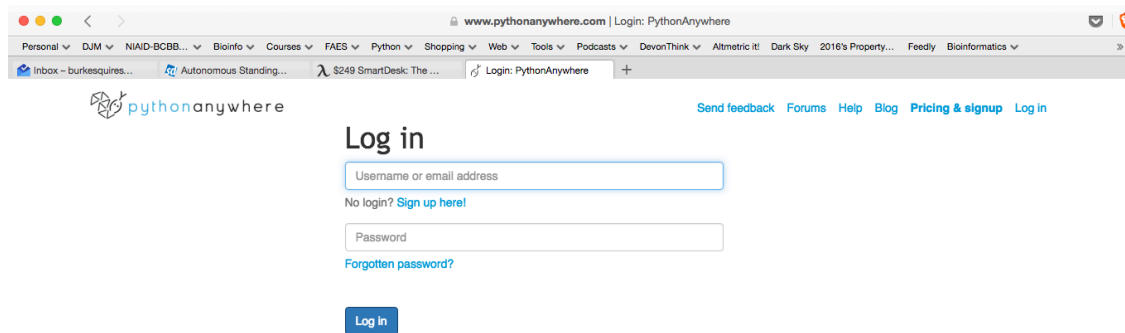
Education accounts

Are you a teacher looking for a place your students can code Python? You're not alone. Click through to find out more about our [Education beta](#).

All of our paid plans come with a no-quibble 30-day money-back guarantee — you're billed monthly and you can cancel at any time. The minimum contract length is just one month. You get unrestricted Internet access from your applications, unlimited in-browser Python, Bash and database consoles, and full SSH access to your account. All accounts (including free ones) have screen-sharing with other PythonAnywhere accounts, and free SSL support (though you'll need to get a certificate for your own domains).

Hacker	\$5/month	Web dev	\$12/month	Startup	\$99/month	Custom	\$5 to \$500/month
Run your Python code in the cloud from one web app and the console		If you want to host small Python-based websites for you or for your clients		Start a business and don't worry about having to scale to handle traffic spikes		Want a combination that's not on the list? Create your own! All custom plans have:	
A Python IDE in your browser with unlimited Python/bash consoles		A Python IDE in your browser with unlimited Python/bash consoles		A Python IDE in your browser with unlimited Python/bash consoles		A Python IDE in your browser with unlimited Python/bash consoles	
One web app on your own domain or <code>your-username.pythonanywhere.com</code>		Up to 2 web apps on custom domains or <code>your-username.pythonanywhere.com</code>		Up to 3 web apps on custom domains or <code>your-username.pythonanywhere.com</code>		Up to 20 web apps, on custom domains or <code>your-username.pythonanywhere.com</code>	
Enough power to run a typical 100,000 hit/day website. (more info)		Enough power to run a typical 150,000 hit/day website on each web app. (more info)		Enough power to run a typical 1,000,000 hit/day website on each web app. (more info)		As many web workers as you need to scale your site's capacity. (more info)	
2,000 CPU-seconds per day for consoles and scheduled tasks (more info)		4,000 CPU-seconds per day for consoles and scheduled tasks (more info)		10,000 CPU-seconds per day for consoles and scheduled tasks (more info)		Up to 100,000 CPU-seconds per day for consoles and scheduled tasks (more info)	
IPython notebook support		IPython notebook support		IPython notebook support		IPython notebook support	
1GB disk space							

Log into PythonAnywhere.org



A screenshot of a web browser showing the PythonAnywhere login page. The browser's address bar displays 'www.pythonanywhere.com | Login: PythonAnywhere'. The page features a navigation menu at the top with links like 'Personal', 'DJM', 'NIAID-BCBB...', 'Bioinfo', 'Courses', 'FAES', 'Python', 'Shopping', 'Web', 'Tools', 'Podcasts', 'DevonThink', 'Altmetric it!', 'Dark Sky', '2016's Property...', 'Feedly', and 'Bioinformatics'. Below the menu, the PythonAnywhere logo is on the left, and a row of links including 'Send feedback', 'Forums', 'Help', 'Blog', 'Pricing & signup', and 'Log in' is on the right. The main heading 'Log in' is centered. Below it are two input fields: 'Username or email address' and 'Password'. A link 'No login? Sign up here!' is positioned between the fields. Below the password field is a link 'Forgotten password?'. At the bottom of the form is a blue 'Log in' button.

www.pythonanywhere.com | Login: PythonAnywhere

Personal DJM NIAID-BCBB... Bioinfo Courses FAES Python Shopping Web Tools Podcasts DevonThink Altmetric it! Dark Sky 2016's Property... Feedly Bioinformatics

Inbox - burkesquires... Autonomous Standing... \$249 SmartDesk: The ... Login: PythonAnywhere

pythonanywhere

Send feedback Forums Help Blog Pricing & signup Log in

Log in

Username or email address

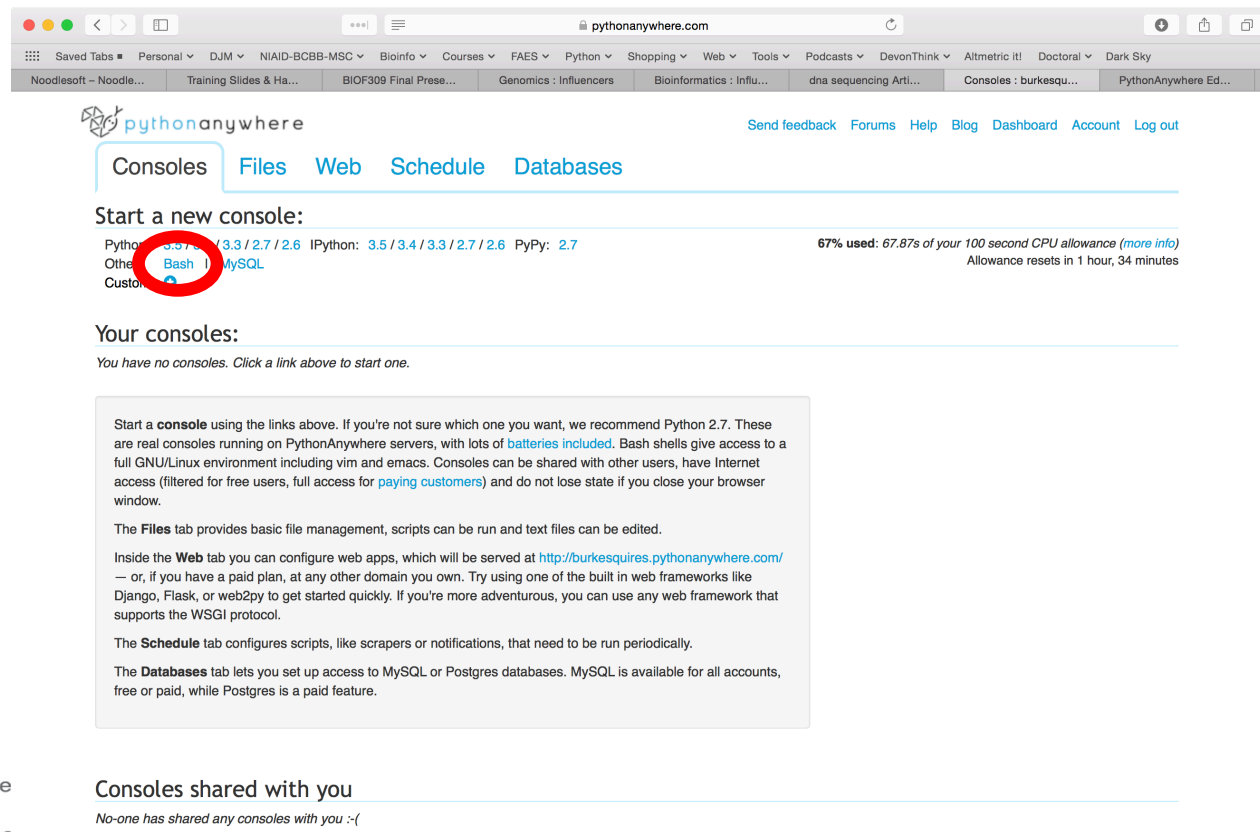
No login? [Sign up here!](#)

Password

[Forgotten password?](#)

Log in

Create BASH Console



pythonanywhere.com

Send feedback Forums Help Blog Dashboard Account Log out

Consoles Files Web Schedule Databases

Start a new console:

Python: [3.5/3.6/3.7/2.7/2.6](#) IPython: [3.5/3.4/3.3/2.7/2.6](#) PyPy: [2.7](#) 67% used: 67.87s of your 100 second CPU allowance ([more info](#))
Other: [Bash](#) [MySQL](#) Allowance resets in 1 hour, 34 minutes

Your consoles:

You have no consoles. Click a link above to start one.

Start a **console** using the links above. If you're not sure which one you want, we recommend Python 2.7. These are real consoles running on PythonAnywhere servers, with lots of [batteries included](#). Bash shells give access to a full GNU/Linux environment including vim and emacs. Consoles can be shared with other users, have Internet access (filtered for free users, full access for [paying customers](#)) and do not lose state if you close your browser window.

The **Files** tab provides basic file management, scripts can be run and text files can be edited.

Inside the **Web** tab you can configure web apps, which will be served at <http://burkesquires.pythonanywhere.com/> — or, if you have a paid plan, at any other domain you own. Try using one of the built in web frameworks like Django, Flask, or web2py to get started quickly. If you're more adventurous, you can use any web framework that supports the WSGI protocol.

The **Schedule** tab configures scripts, like scrapers or notifications, that need to be run periodically.

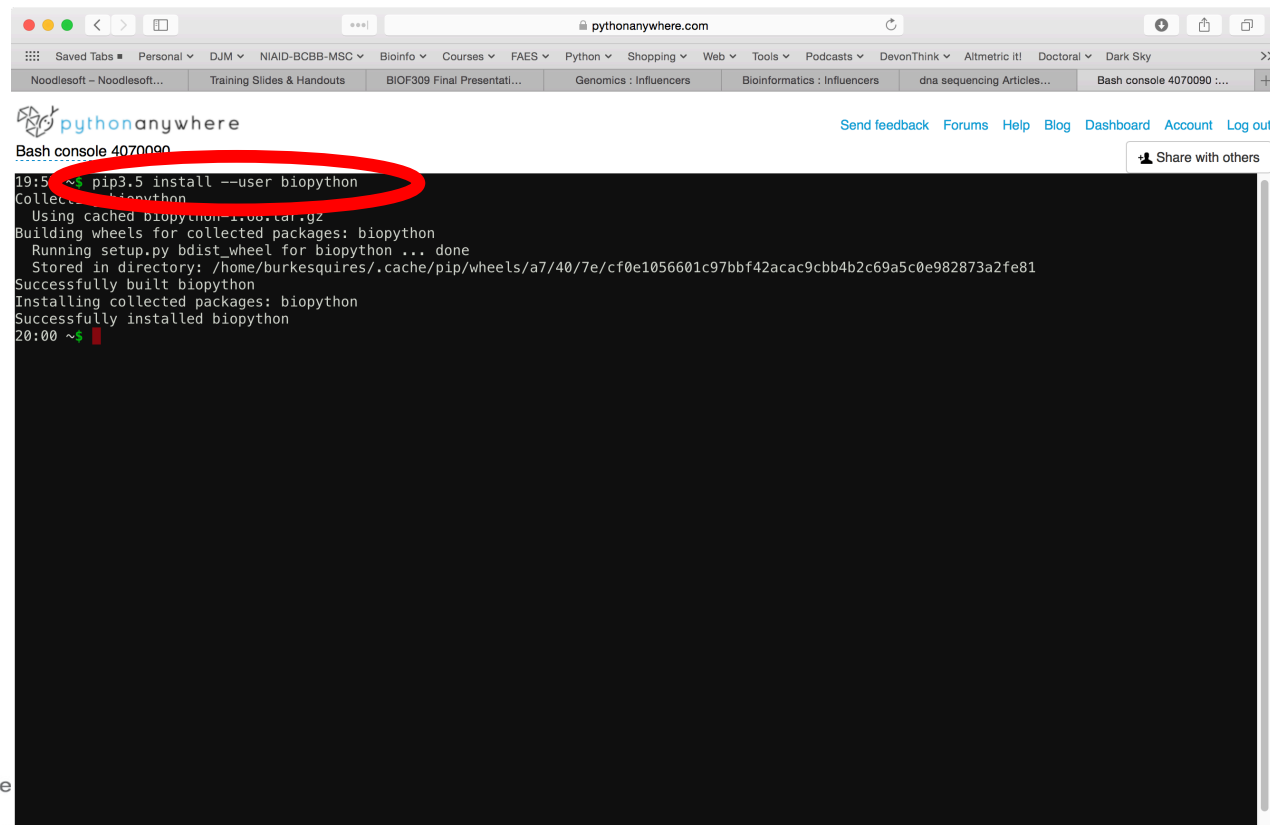
The **Databases** tab lets you set up access to MySQL or Postgres databases. MySQL is available for all accounts, free or paid, while Postgres is a paid feature.

Consoles shared with you

No-one has shared any consoles with you :-(

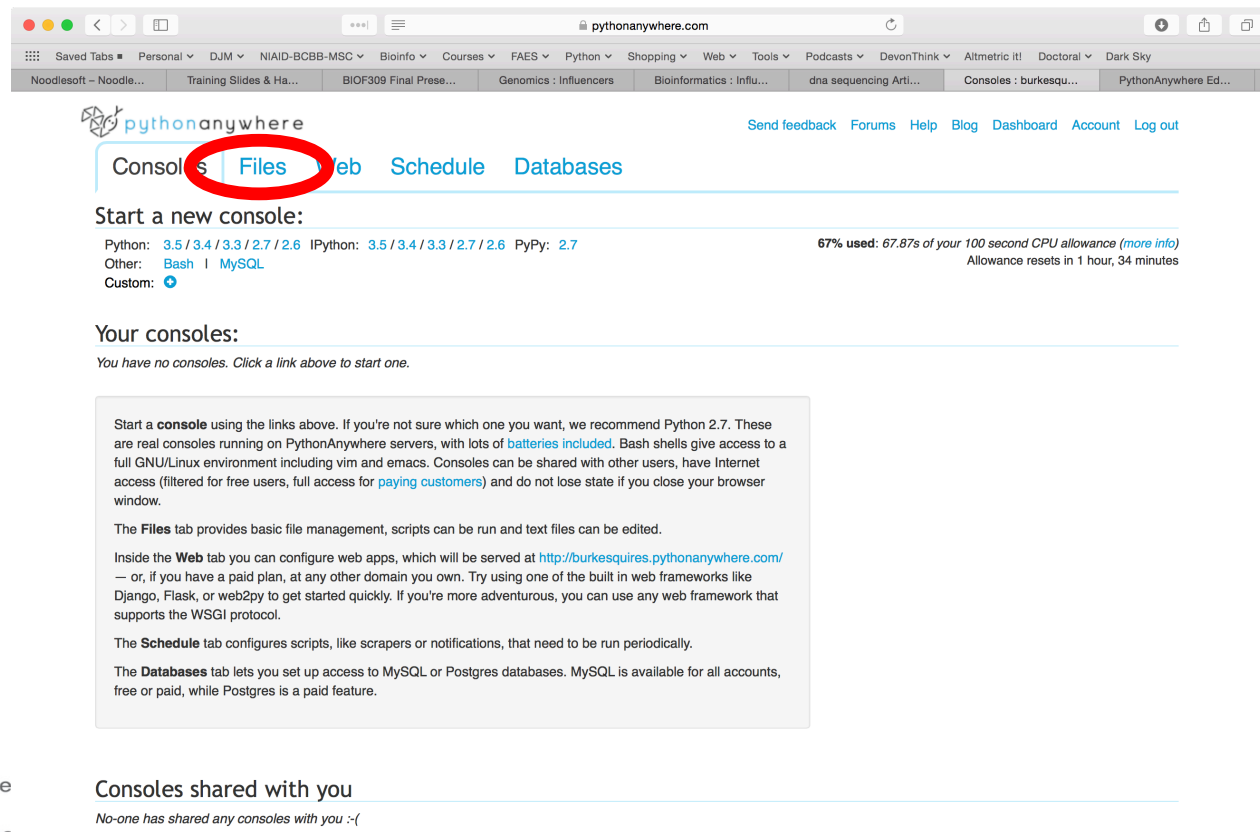
Install Biopython Module:

```
pip3.5 install --user biopython
```



```
pythonanywhere.com
Send feedback Forums Help Blog Dashboard Account Log out
Share with others
Bash console 4070090
19:58 ~$ pip3.5 install --user biopython
Collecting biopython
Using cached biopython-1.68.tar.gz
Building wheels for collected packages: biopython
Running setup.py bdist_wheel for biopython ... done
Stored in directory: /home/burkesquires/.cache/pip/wheels/a7/40/7e/cf0e1056601c97bbf42acac9cbb4b2c69a5c0e982873a2fe81
Successfully built biopython
Installing collected packages: biopython
Successfully installed biopython
20:00 ~$
```

Go Back to Dashboard



pythonanywhere.com

Send feedback Forums Help Blog Dashboard Account Log out

Consoles **Files** Web Schedule Databases

Start a new console:

Python: [3.5](#) / [3.4](#) / [3.3](#) / [2.7](#) / [2.6](#) IPython: [3.5](#) / [3.4](#) / [3.3](#) / [2.7](#) / [2.6](#) PyPy: [2.7](#)

Other: [Bash](#) | [MySQL](#)

Custom: [+](#)

67% used: 67.87s of your 100 second CPU allowance ([more info](#))
Allowance resets in 1 hour, 34 minutes

Your consoles:

You have no consoles. Click a link above to start one.

Start a **console** using the links above. If you're not sure which one you want, we recommend Python 2.7. These are real consoles running on PythonAnywhere servers, with lots of [batteries included](#). Bash shells give access to a full GNU/Linux environment including vim and emacs. Consoles can be shared with other users, have Internet access (filtered for free users, full access for [paying customers](#)) and do not lose state if you close your browser window.

The **Files** tab provides basic file management, scripts can be run and text files can be edited.

Inside the **Web** tab you can configure web apps, which will be served at <http://burkesquires.pythonanywhere.com/> — or, if you have a paid plan, at any other domain you own. Try using one of the built in web frameworks like Django, Flask, or web2py to get started quickly. If you're more adventurous, you can use any web framework that supports the WSGI protocol.

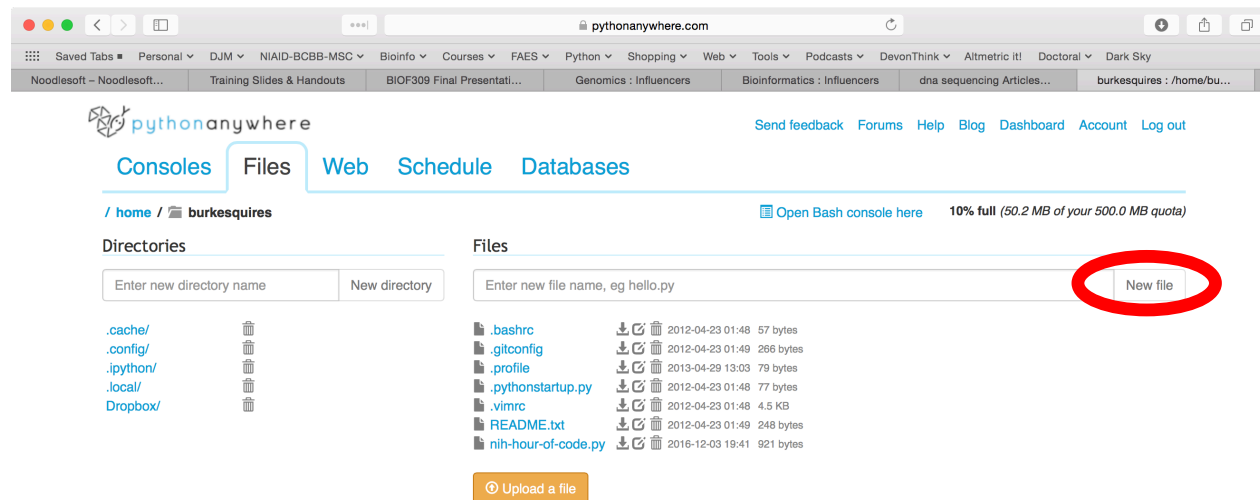
The **Schedule** tab configures scripts, like scrapers or notifications, that need to be run periodically.

The **Databases** tab lets you set up access to MySQL or Postgres databases. MySQL is available for all accounts, free or paid, while Postgres is a paid feature.

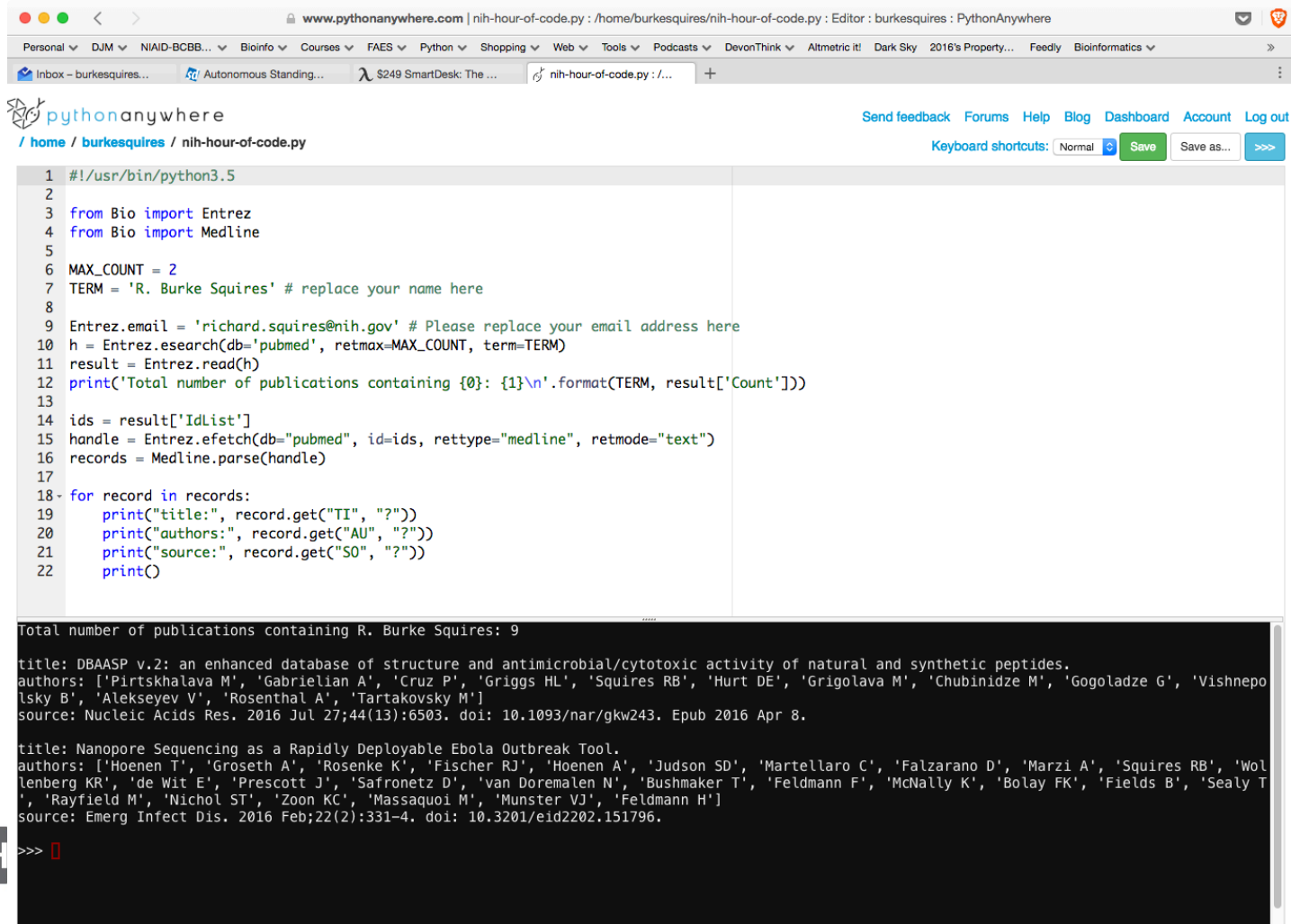
Consoles shared with you

No-one has shared any consoles with you :-)

Create a new file



Let's Write Some Python Code!



The screenshot shows a web browser window with the URL `www.pythonanywhere.com | nih-hour-of-code.py : /home/burkesquires/nih-hour-of-code.py`. The browser's address bar and tabs are visible. The page header includes the PythonAnywhere logo and navigation links: [Send feedback](#), [Forums](#), [Help](#), [Blog](#), [Dashboard](#), [Account](#), and [Log out](#). Below the header, there are keyboard shortcuts: `Normal`, `Save`, `Save as...`, and `>>>`.

The main content area displays a Python script in a text editor. The script is as follows:

```
1 #!/usr/bin/python3.5
2
3 from Bio import Entrez
4 from Bio import Medline
5
6 MAX_COUNT = 2
7 TERM = 'R. Burke Squires' # replace your name here
8
9 Entrez.email = 'richard.squires@nih.gov' # Please replace your email address here
10 h = Entrez.esearch(db='pubmed', retmax=MAX_COUNT, term=TERM)
11 result = Entrez.read(h)
12 print('Total number of publications containing {0}: {1}\n'.format(TERM, result['Count']))
13
14 ids = result['IdList']
15 handle = Entrez.efetch(db='pubmed', id=ids, rettype="medline", retmode="text")
16 records = Medline.parse(handle)
17
18 for record in records:
19     print("title:", record.get("TI", "?"))
20     print("authors:", record.get("AU", "?"))
21     print("source:", record.get("SO", "?"))
22     print()
```

Below the script, the output of the program is shown in a terminal window. The output is as follows:

```
Total number of publications containing R. Burke Squires: 9

title: DBAASP v.2: an enhanced database of structure and antimicrobial/cytotoxic activity of natural and synthetic peptides.
authors: ['Pirts Khalava M', 'Gabrielian A', 'Cruz P', 'Griggs HL', 'Squires RB', 'Hurt DE', 'Grigolava M', 'Chubinidze M', 'Gogoladze G', 'Vishnepo
lsky B', 'Alekseyev V', 'Rosenthal A', 'Tartakovsky M']
source: Nucleic Acids Res. 2016 Jul 27;44(13):6503. doi: 10.1093/nar/gkw243. Epub 2016 Apr 8.

title: Nanopore Sequencing as a Rapidly Deployable Ebola Outbreak Tool.
authors: ['Hoenen T', 'Groseth A', 'Rosenke K', 'Fischer RJ', 'Hoenen A', 'Judson SD', 'Martellaro C', 'Falzarano D', 'Marzi A', 'Squires RB', 'Wol
lenberg KR', 'de Wit E', 'Prescott J', 'Safronetz D', 'van Doremalen N', 'Bushmaker T', 'Feldmann F', 'McNally K', 'Bolay FK', 'Fields B', 'Sealy T
', 'Rayfield M', 'Nichol ST', 'Zoon KC', 'Massaquoi M', 'Munster VJ', 'Feldmann H']
source: Emerg Infect Dis. 2016 Feb;22(2):331-4. doi: 10.3201/eid2202.151796.
```

The terminal window shows the output of the script, which is a list of publications containing the term 'R. Burke Squires'. The output is formatted as a list of records, each containing a title, authors, and source. The first record is 'DBAASP v.2: an enhanced database of structure and antimicrobial/cytotoxic activity of natural and synthetic peptides.' and the second record is 'Nanopore Sequencing as a Rapidly Deployable Ebola Outbreak Tool.'

Python Code to Retrieve Your Publications

```
#!/usr/bin/python3.5

from Bio import Entrez
from Bio import Medline

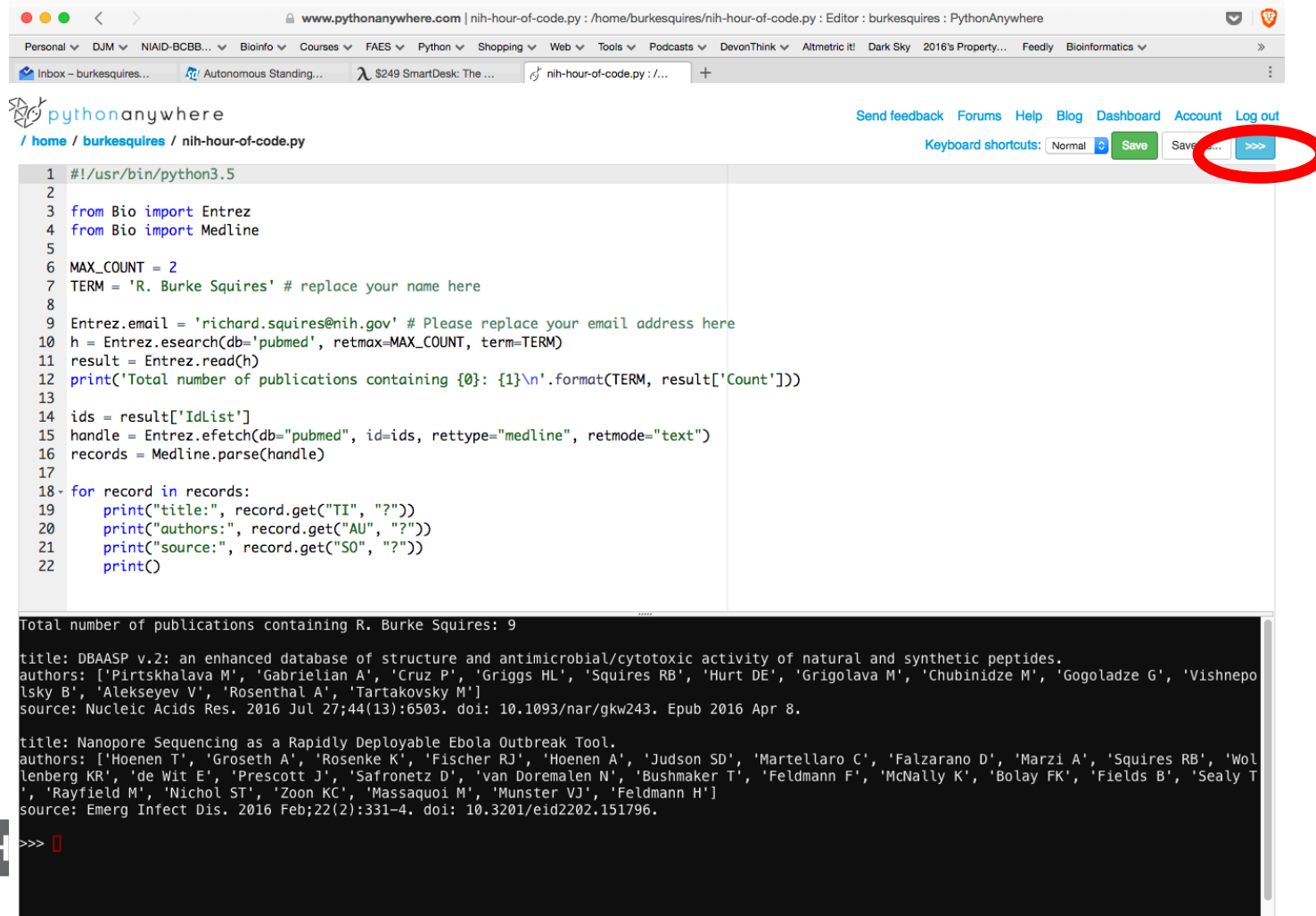
MAX_COUNT = 2
TERM = 'R. Burke Squires' # replace your name here

Entrez.email = 'richard.squires@nih.gov' # Please replace your email address here
h = Entrez.esearch(db='pubmed', retmax=MAX_COUNT, term=TERM)
result = Entrez.read(h)
print('Total number of publications containing {0}: {1}\n'.format(TERM, result['Count']))

ids = result['IdList']
handle = Entrez.efetch(db="pubmed", id=ids, rettype="medline", retmode="text")
records = Medline.parse(handle)

for record in records:
    print("title:", record.get("TI", "?"))
    print("authors:", record.get("AU", "?"))
    print("source:", record.get("SO", "?"))
    print()
```

Run Your Code!



www.pythonanywhere.com | nih-hour-of-code.py : /home/burkesquires/nih-hour-of-code.py : Editor : burkesquires : PythonAnywhere

Personal ▾ DJM ▾ NIAID-BCBB... ▾ Bioinfo ▾ Courses ▾ FAES ▾ Python ▾ Shopping ▾ Web ▾ Tools ▾ Podcasts ▾ DevonThink ▾ Altmetric It! ▾ Dark Sky ▾ 2016's Property... ▾ Feedly ▾ Bioinformatics ▾

Inbox - burkesquires... Autonomous Standing... \$249 SmartDesk: The ... nih-hour-of-code.py : /... +

pythonanywhere

/ home / burkesquires / nih-hour-of-code.py

Send feedback Forums Help Blog Dashboard Account Log out

Keyboard shortcuts: Normal Save Save >>>

```
1 #!/usr/bin/python3.5
2
3 from Bio import Entrez
4 from Bio import Medline
5
6 MAX_COUNT = 2
7 TERM = 'R. Burke Squires' # replace your name here
8
9 Entrez.email = 'richard.squires@nih.gov' # Please replace your email address here
10 h = Entrez.esearch(db='pubmed', retmax=MAX_COUNT, term=TERM)
11 result = Entrez.read(h)
12 print('Total number of publications containing {0}: {1}\n'.format(TERM, result['Count']))
13
14 ids = result['IdList']
15 handle = Entrez.efetch(db='pubmed', id=ids, rettype="medline", retmode="text")
16 records = Medline.parse(handle)
17
18 for record in records:
19     print("title:", record.get("TI", "?"))
20     print("authors:", record.get("AU", "?"))
21     print("source:", record.get("SO", "?"))
22     print()
```

Total number of publications containing R. Burke Squires: 9

title: DBAASP v.2: an enhanced database of structure and antimicrobial/cytotoxic activity of natural and synthetic peptides.
authors: ['Pirts Khalava M', 'Gabrielian A', 'Cruz P', 'Griggs HL', 'Squires RB', 'Hurt DE', 'Grigolava M', 'Chubinidze M', 'Gogoladze G', 'Vishnepo
lsky B', 'Alekseyev V', 'Rosenthal A', 'Tartakovsky M']
source: Nucleic Acids Res. 2016 Jul 27;44(13):6503. doi: 10.1093/nar/gkw243. Epub 2016 Apr 8.

title: Nanopore Sequencing as a Rapidly Deployable Ebola Outbreak Tool.
authors: ['Hoenen T', 'Groseth A', 'Rosenke K', 'Fischer RJ', 'Hoenen A', 'Judson SD', 'Martellaro C', 'Falzarano D', 'Marzi A', 'Squires RB', 'Wol
lenberg KR', 'de Wit E', 'Prescott J', 'Safronetz D', 'van Doremalen N', 'Bushmaker T', 'Feldmann F', 'McNally K', 'Bolay FK', 'Fields B', 'Sealy T
, 'Rayfield M', 'Nichol ST', 'Zoon KC', 'Massaquoi M', 'Munster VJ', 'Feldmann H']
source: Emerg Infect Dis. 2016 Feb;22(2):331-4. doi: 10.3201/eid2202.151796.

>>> █

How To Install Python On Your Computer?

- Install Python:
 - Anaconda distribution
 - Enthought distribution
- Installs:
 - Python
 - pip
 - jupyter
 - numpy
 - pandas,
 - etc



Where to Go From Here?

- Python Seminars
 - CIT (No cost)
- FAES
 - Python Programming courses
- Tutorials on PyVideo.org
- Practice, practice, practice

Resources To Learn More About Python

- Free electronic books:
 - Automate the boring stuff
 - <https://automatetheboringstuff.com>
 - Python for everyone
 - <https://wiki.python.org/moin/IntroductoryBooks>
 - Python books
 - <http://pythonbooks.revolunet.com>
- Free Python Videos
 - <http://pyvideo.org>