Hyperion Development

The blaze, the splendor, and the symmetry.

Free Python Training introduction

python™

Hello, and welcome to your Python training!

We are always excited to have a new student join our team, no matter what background or knowledge you have – you are most welcome and we hope to have a long relationship working with you.

You are about to embark on training the powerful programming language. Python, which is used around the world for millions of tasks. This training is unique because it will expose you to the applications of Python and computer programming to new and exciting fields such as Artificial Intelligence, Machine Learning, Natural Language Processing and Bioinformatics while you learn Python. Real problems in these fields will be introduced as early as Task 3 and you will write Python code to solve these problems. I will take every effort to also explain the relevant science behind these problems for example:

- Why computers have trouble understanding human speech and how computer programs advance to try solve this, especially Python and the Natural Language Processing Toolkit.
- How we can try to make "decision making robots", such as chess AI, and determining what their limitations are.
- 'How we can use Bioinformatics to understand how HIV evolves to survive the drugs we use to combat it'.

Our curriculum is built from a wide variety of sources. We have taken programming exercises, notes and tasks from some of the most advanced and cutting edge Computer Science programs in the world - **such as the University of Edinburgh AI department** - the most advanced AI research department in the world.

How does it work?

A teacher has setup your **Dropbox** folder. In this folder, you should be able to see another folder called Task 1. Open this folder and read the file named 'Instructions' to find out what you need to do for this task. Task 1 is a bare bones foundation to Python and many of the concepts in Task 1 are applicable to all programming languages such as Java, Haskell etc. Please keep all files and work inside the relevant Task folder located inside your Dropbox.

Please make sure you have Python and Notepad++ installed before continuing. Visit **http://rmoola.com/pythonlessons.html** on instructions on how to set these up. **You should open all .txt and .py documents in Notepad++.**

How Dropbox works:

Every single file you change in this Dropbox folder will **immediately** be seen by our tutors. That means as soon as you create a new file in this folder, we'll be able to see it. If you edit that file, as soon as you save it it will also be updated on our tutors computers. Our tutors can also access your files in this folder and make changes and then save them - you'll be able to see their changes right away and you'll get a popup notification in the bottom right of your screen everytime this

happens. Go ahead and make a new textfile in this folder called **'Comments'** and inside the document write your name and save the file - our tutors will use it right away!

We highly recommend you visit www.dropbox.com/events and sign in. This website will give you a list of all changes in your Dropbox folder, **even while you were offline**. So if while you were sleeping we checked your work, in the morning you may want to visit that website to get a full detailed list of what changed rather than having to open your Dropbox folder and check each file individually!

The structure of each Task:

Each task folder will generally contain:

- An "example.py" Python file with example code and extensive commenting to help you learn new concepts.
- Optional programming exercises can be found throughout the comments in "example.py" in the 'Playing around with Python' sections.
- One compulsory exercise is given at the end of "examples.py". You must complete this exercise correctly in order to be given the next Task.
- Bonus optional exercises are given. Optional extra reading on the subject material of the task which can range from simple Python notes to scientific papers/external university resources on interesting and relevant topics.

An outline of the Python syllabus:

- **Task 1** – A strong introduction to the fundamental concepts of Python including control and data structures, variables and running and writing Python programs.
- **Task 2** – An introduction to more advanced concepts such as file I/O, creating functions and using other Python modules.
- **Task 3: Bioinformatics** – An introduction to Bioinformatics, DNA sequences, and using Python to solve problems in these fields. Bioinformatics in South Africa and abroad.
- **Task 4: Natural Language Processing** – An introduction to Natural Language Processing, with slides, notes and exercises from the University of Edinburgh in Scotland. Programming using the Python Natural Language Toolkit.
- **Task 5: Python on the web** – An introduction to writing Python scripts on web servers and an example from previous South African Medical Research Council work for the means of displaying Javascript graphs.
- **Task 6: Artificial Intelligence** – An introduction to some concepts in Artificial Intelligence including using Python for constraint satisfaction problems. Slides and examples from the University of Pennsylvania and the University of Edinburgh courses with programming exercises from the same.

- **Task 9: Cloud Computing** – Introducing the concept of Cloud Computing, using the Amazon Cloud for free and setting up and working with Ubuntu (Linux) servers to understand how we can take advantage of cloud computing.

More tasks in development. Please submit any suggestions you have to students@hyperiondev.com. As you can see, complete more tasks, and the programming exercises and problems will become more interesting and advanced – but you will get to work at your own pace.

Getting help:

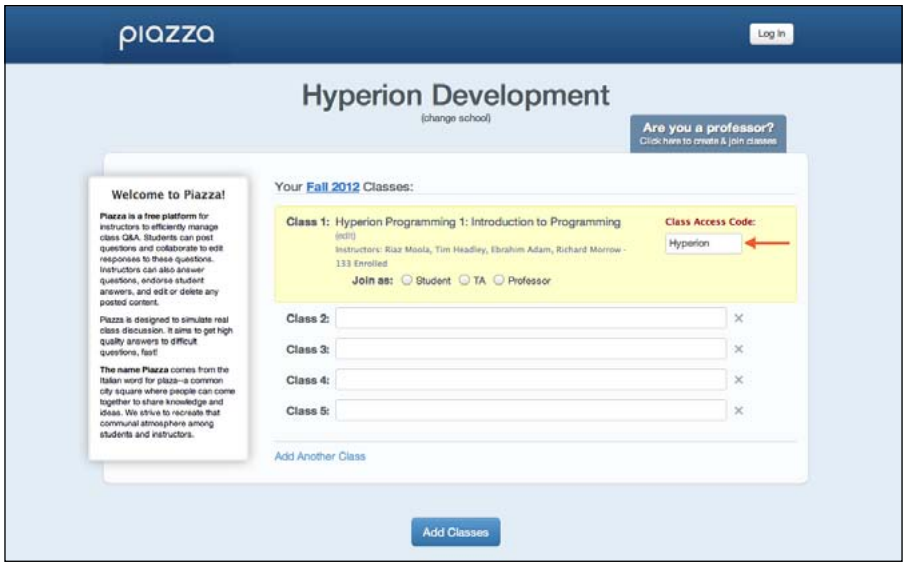
If you have absolutely any problems, queries or requests, please email us on students@hyperiondev.com and a teacher will get back to you immediately. You can also add the above address to Google Chat or add hyperion.development on Skype to get one on one video call help.

Make sure to check your Dropbox regularly and to also visit http://rmoola.com/advice.html for some advice and http://rmoola.com/founders.html to see some frequently asked questions.

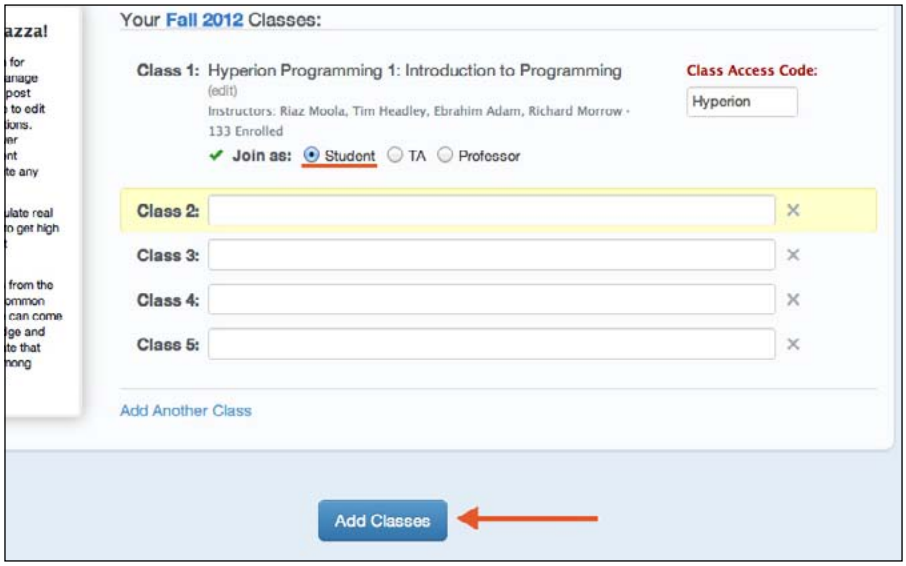
Getting help on Piazza

You will need internet access on any device (even a phone) to complete this step.

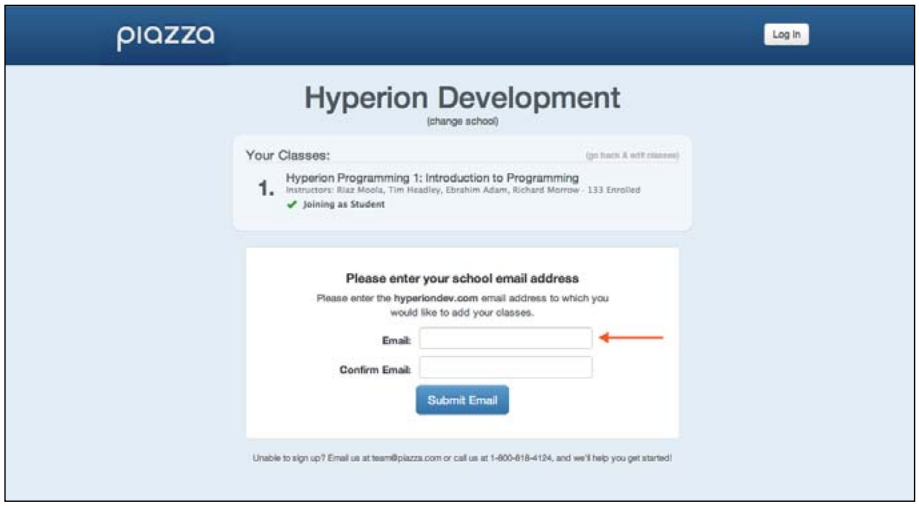
1. Visit the following web address: www.tinyurl.com/hyperionpiazza



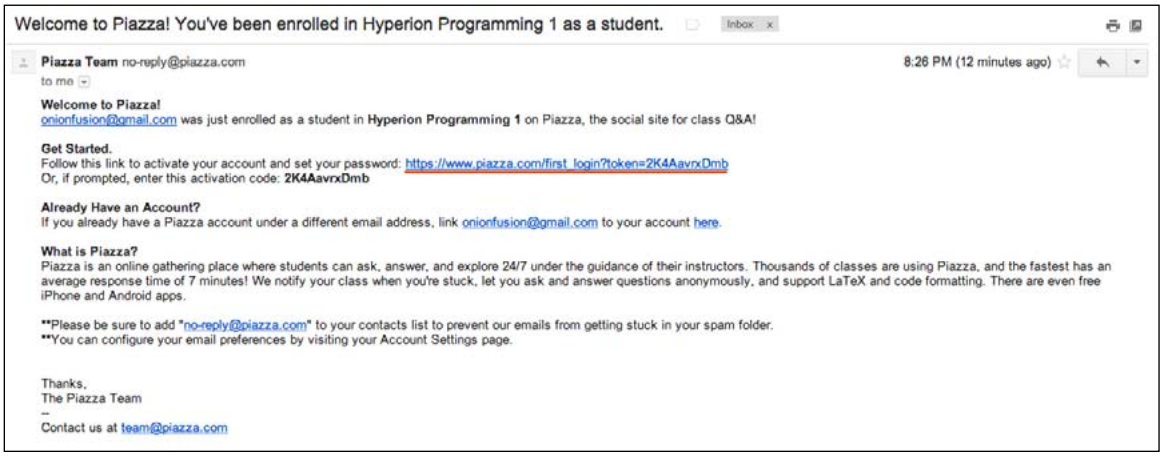
2. The above should appear. Enter "**Hyperion**" in the "Class Access Code" field.



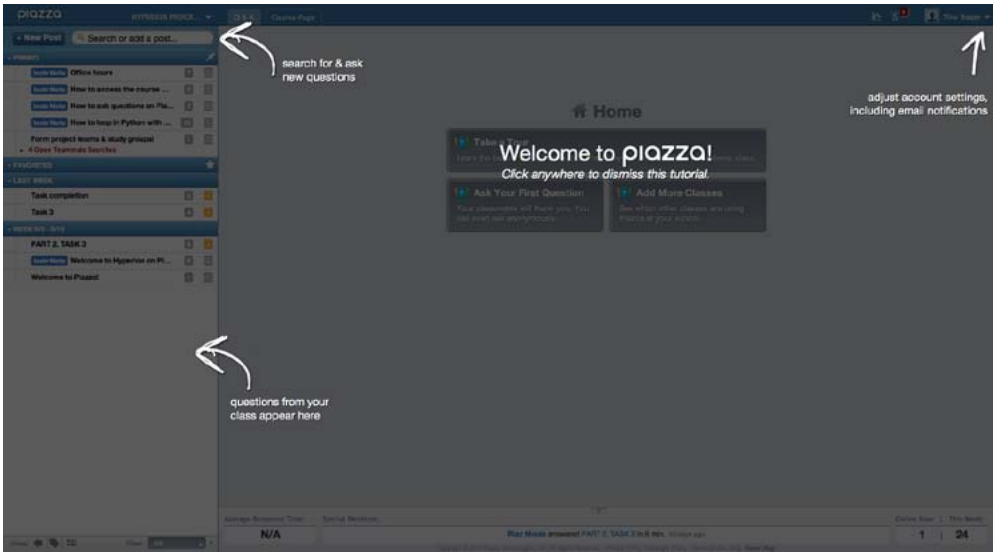
3. Select "**Student**" by clicking the circle to its left. Then click "**Add Classes**".



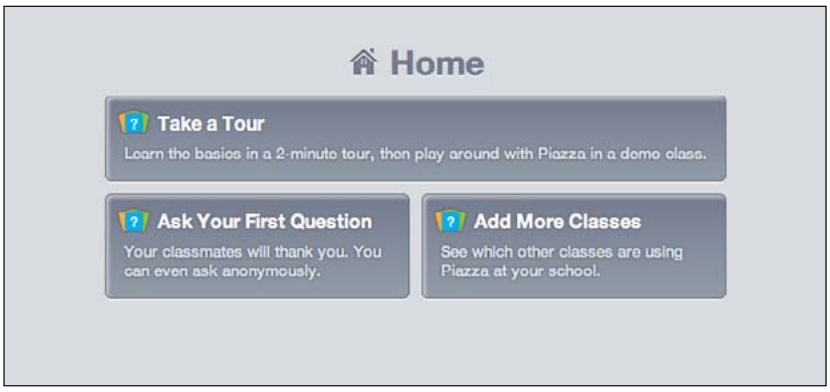
4. Proceed to enter your **email address** and confirm to enrol for our programming course.



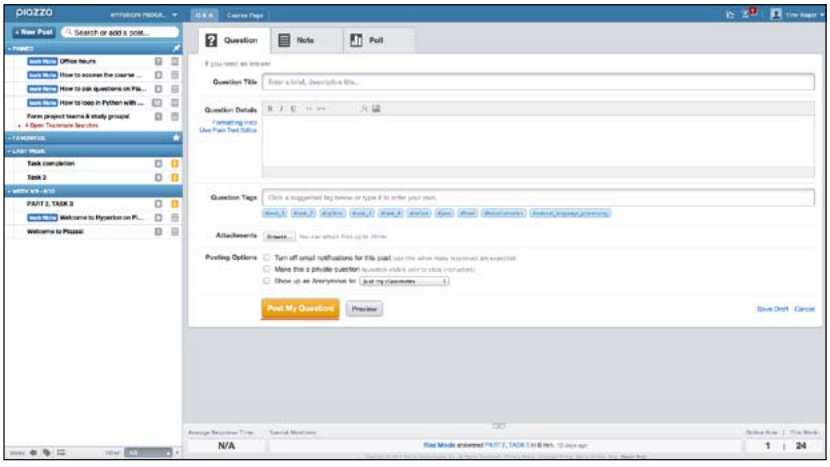
5. You should soon receive an email like above, guiding you on how to **confirm** and begin your learning with Hyperion.



6. Once you log in to your Piazza home page, **click anywhere to dismiss** the tutorial.



7. We recommend that you **"Take a Tour"** before continuing. Once you've completed the tour, click **"New Post"** top-left to post a question for our teachers to assist you with.



8. Here you can choose a **Question Title**, type out your **Question Details**, add **Question Tags** that make your question easier to recognise and add **attachments**. Our teachers will be notified once you click "**Post My Question!**" and assist you.

Getting help on Facebook

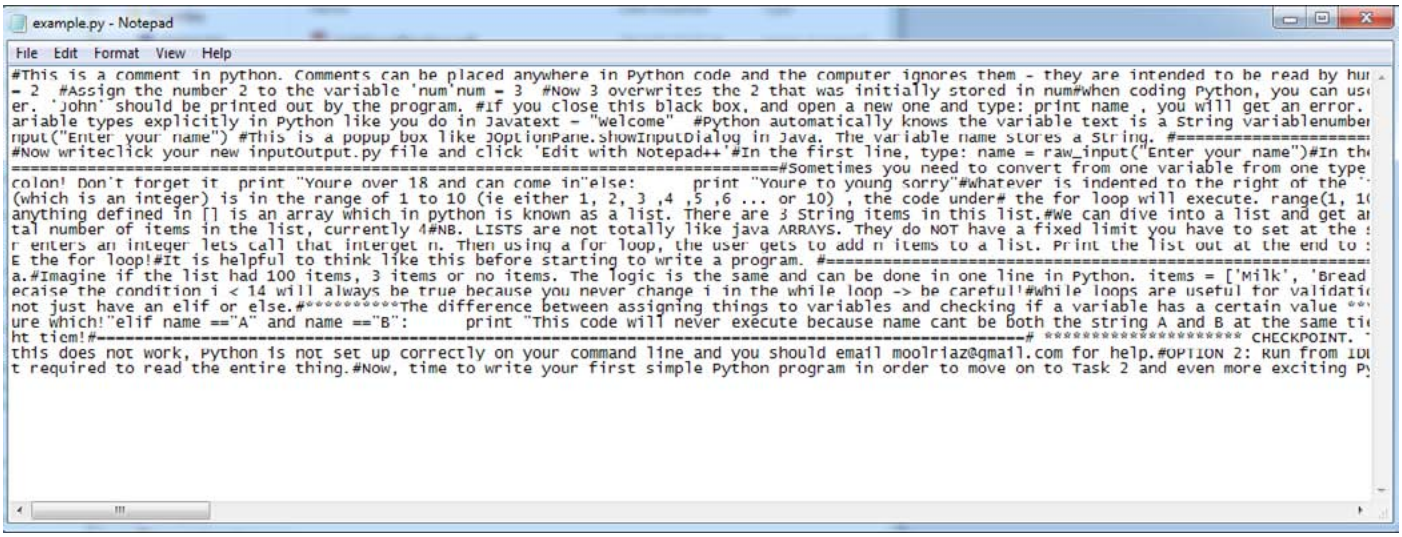
1. Find us at www.facebook.com/hyperiondev, where you can "like" *Hyperion Development* and (only if you cannot get assistance through Piazza) post any questions or any requests for help on our wall.

Paid projects:

After you complete the first two tasks, you will be eligible for paid and unpaid projects with our clients. Please contact students@hyperiondev.com to express interest in this, or in the case of fast progress you will be approached! Title your email as 'Interest in paid project'.

I'm having trouble reading the documents:

When you open instructions.txt it should ***not*** look like this:



Rather, it should like this:

```
#This is a comment in python. Comments can be placed anywhere in Python code and the computer ignores them - t


#Please open this file with Notepad++! It will make it much easier to read. Right click this file --> Edit wit
#Once in Notepad++, click 'View' on the top toolbar and select 'Word Wrap'. Things should be much easier to re
#Comments in Python appear in GREEN if you have the file opened in notepad++. Keywords in BLUE.

#This is an example program to help you learn Python. Let's get started. If you already know Java or another p

#Python is VERY similar to Java and most programming languages have the same structure so if you learn one - y

#========== Variables ===========
#A variable is a placeholder
num = 2  #Assign the number 2 to the variable 'num'
num = 3  #Now 3 overwrites the 2 that was initially stored in num

#When coding Python, you can use call variables anything you want, but try using descriptive variable names.
myName = "Tom"       # A good variable name
variableOne = "Tom" #Not a good variable name, doesnt give a description of what would be stored inside it

#======================== Play around with Python a bit (OPTIONAL) ================================
#At this point, why not play around with creating variables? Press the windows start button, in the 'Search fo
#If this does not happen, your Python isn't installed on your system yet. Please download and install Python 2
#In the black box that appeared, type name = "John"
#then press enter. Nothing happens but this Python program has remembered what you set the variable 'name' to.
#To prove this type: print name
#and then hit enter. 'John' should be printed out by the program.
#If you close this black box, and open a new one and type: print name , you will get an error. This is because
#We write python in text editors like Notepad++ or the IDLE Python GUI so that all our variable definitions an
#Keep the black box open and try out some commands as you read through this file -> you are actually writing P
#=============================================================================================

#You get different types of variables : Strings, Ints and Floats
#Ints store integers (ie whole numbers).
#String store text and you always use " " to denote Strings. do not use ' ' . Strings can have spaces in them.
#Floats store real numbers (ie numbers with decimal points) so numbers such as 1.234
```

The image on the previous page is code viewed in Notepad. The above image is code viewed in a new program called **Notepad++** that you need to **download**. It will be almost impossible to read instructions and code if you're not using Notepad++!

Who wrote these learning materials?

Riaz Moola, a former UKZN student currently studying Artificial Intelligence and Computer Science at the University of Pennsylvania and the University of Edinburgh wrote all these documents and designed these notes. Riaz Moola reserves full rights on all the materials you have been provided with.

More information: **www.rmoola.com**