

**Item Frequency Report** [Filter Results](#)

**Introduction to Python Programming - 2015-01-12 to 2015-01-16**

1. Please check the box for each question that represents your opinion (1 to 5 where 5 is the highest rating):

The presentation was informative.

		Response Percent	Response Total
1		0%	0
2		0%	0
3		0%	0
4		30%	3
5		70%	7
<b>Total Responses</b>		<b>10</b>	

The instructor was knowledgeable of the materials.

1		0%	0
2		0%	0
3		0%	0
4		10%	1
5		90%	9
<b>Total Responses</b>		<b>10</b>	

The instructor was effective in delivering the materials.




1		0%	0
2		0%	0
3		0%	0
4		20%	2
5		80%	8
<b>Total Responses</b>		<b>10</b>	

I will be able to use the skills learned in my current job.

1		0%	0
2		0%	0
3		20%	2
4		40%	4
5		40%	4
<b>Total Responses</b>		<b>10</b>	

2. The length of the class was:

		Response Percent	Response Total
<b>Too Short</b>		0%	0
<b>Just Right</b>		90%	9

<b>Too Long</b>		<b>10%</b>	<b>1</b>
		<b>Total Responses</b>	<b>10</b>
4. Would you recommend this class to others?		<b>Response Percent</b>	<b>Response Total</b>
<b>Yes</b>		<b>100%</b>	<b>10</b>
<b>No</b>		<b>0%</b>	<b>0</b>
		<b>Total Responses</b>	<b>10</b>
3. How could the course be improved for future offerings?			
Please make it clearer that the slides are always available on the web site,. It was not clear that the text the instructor types on the screen can also be studied later on my own.The exercises of uploading code to the Sphere resource was an excellent idea - perhaps one more example could be included. The pace of the class was engaging and time really flew by. Also, if there were a bit more time it would be nice to append a half-hour on reading/writing to data files			
1.			
2.	Adding some smaller exercises in the first few days would help increase facility with the basic language constructs. Pointing out notable differences between Python 2.x and 3.x would be useful, as much code at JPL still uses Python 2.x.		
3.	Maybe spend slightly less time on all the idiosyncrasies of Python specifically and a little more time on how some of the common things that are done in other languages are done in Python. Like maybe half a day of going through how Python reads in files or data, manipulates numbers, text, or binary data, and outputs the results to a user or to a file. Also maybe an hour or so at the end of summarizing important or often-used modules, libraries, and popularly used outside additions or expansions (that would be useful to JPL'ers) such as SciPy, NumPy, graphics, plotting, mapping, Internet interfacing, web stuff--with a list of links.		
4.	It is probably a subject unto itself, but I would have liked to have seen some material on building GUIs with Python as I have a strong interest in doing instrument control.		
5.	This course was superb. The problem sets could have been more oriented to scientific software engineering problems.		
6.	As an "intro" to the language, the course covered a great deal of material in a short time. Still, for me, it would have been helpful to have some additional simple, practical examples using some commonly-used packages such as SciPy, NumPy, and Matplotlib.		
7.	Maybe for some of the more difficult/abstract concepts (generators, closures, etc), it might be nice to have more comments in the provided code with some basic definitions.		
8.	It would help me personally to spend a little less time on the computer science fundamentals aspects of Python. Although important to build the foundation for programming in Python, I (and probably others) are interested in applying Python to engineering tasks (parsing data from files, running numerical analyses, making engineering plots, etc.) and understanding why Python is a better tool for this than others (like MATLAB, IDL).		
9.	It is good already. The Socratic method is a great way to get everyone's participation. It gives everyone the one-on-one attention.		
10.	More regex. Perhaps a couple exercises. I know that isn't the intent of the class, but it would have been the most valuable portion for me.		
5. General Comments:			
1.	This was an excellent class! The instructor made an effort to engage EVERYONE and gave us a good heads-up on the power (and pitfalls) of Python. Thanks for the frequent (hourly) breaks.		
2.	The style of the instructor is to approach concepts via examples, which usually works quite well. However, for more obscure/complex ideas, such as closure and decorators, it might be helpful to add a succinct summary of the concept (even if it is somewhat more theoretical) in addition to the examples. Please continue to take breaks every hour - it was very helpful to maintain concentration. Occasionally, the instructor would ask a student for an explanation for a line of code but then either interrupt or cut off the student's response with the correct answer. While this did keep the class		

moving, it was a little self-defeating to the Socratic method.

3. Great class! And instructor was very good at engaging the students and making us actually program tricky stuff that taught us the new principles for that section.

I really enjoyed Glen's use of the Socratic method and the fact that he took a gentle approach pulling us along toward the answer with little hints on those occasion when we were starring back at him like "deer in the headlights". I also enjoyed the way Glenn combines his enthusiasm and knowledge of the subject with sprinkles of humor and an occasional salty

4. word. Also, it might not seem like much, but the fact that Glenn gave us a 10 minute break every hour was extremely helpful, especially when you consider the amount of concentration needed to keep up with his fast-paced lecture style. I have to say that Glenn Downing is one of the best instructors I've ever had the privilege of taking a course under. I felt like there was never a dull moment from start to finish.

Dr Downing used the Socratic method of instruction during this class. His execution was nearly perfect, however, sometimes he would answer the questions he asked for the student instead of waiting for the response from the student.

5. I'm not sure if this was purposeful or whether it was a result of Dr Downings obvious enthusiasm for the material and teaching.

6. I don't really like to speak up in class, but I thought the socratic method of teaching was very effective. Glenn is an enthusiastic instructor and really made the class interesting and enjoyable. Thank you so much Glenn!

7. The instructor is good at communicating and conveying the fundamental principles of the programming language. Engages students well.

I was really hoping to learn techniques and available functions that would help me improve the quality and efficiency of my

8. Python scripts. However I don't think I really learned much that will change my current practices. I do however feel like I now understand the language a lot better and have a better grasp on how everything works.

