Item Frequency Report Filter Results

Introduction to Python Programming - 2016-03-14 to 2016-03-17

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

The presentation was informative.

	Response Percent	Response Total
1	0%	**0**
2	0%	**0**
3	0%	**0**
4	18%	**3**
5	82%	**14**
Total Responses	**17**	

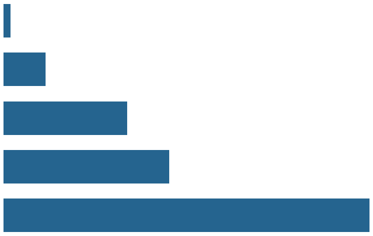
The instructor was knowledgeable of the materials.

	Response Percent	Response Total
1	0%	**0**
2	0%	**0**
3	0%	**0**
4	18%	**3**
5	82%	**14**
Total Responses	**17**	

The instructor was effective in delivering the materials.

	Response Percent	Response Total
1	0%	**0**
2	0%	**0**
3	0%	**0**
4	12%	**2**
5	88%	**15**
Total Responses	**17**	

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



	Response Percent	Response Total
1	0%	**0**
2	6%	**1**
3	18%	**3**
4	24%	**4**
5	53%	**9**
Total Responses	**17**	

2. The length of the class was:	Response Percent	Response Total
Too Short	0%	**0**
Just Right	94%	**16**

Too Long	**6%**	**1**
Total Responses	**17**	

4. Would you recommend this class to others?	**Response Percent**	**Response Total**
Yes	**100%**	**17**
No	**0%**	**0**
Total Responses	**17**	

3. How could the course be improved for future offerings?

1. Less iteration....I don't want to hear the words "iter", "iterable", "iterate over" or "iterator" ever again.

2. I would like to see more of the object-oriented programming

3. Update the course with new output formatting methods. https://docs.python.org/3/tutorial/inputoutput.html section 7.1 versus 7.1.1 which is taught in this course

4. recommend a couple of textbooks.

5. It could be broken into stages so that we could practice what we learned and thus retain more information.

6. Another Python course for intermediate or above level

7. Even more examples.

8. Literally my only suggestion is to keep the class better informed of example .py file version updates when they occur during class time.

9. gear it towards scientist/engineers than programmers

10. Go into numpy with more detail. Devote a whole day+ to it. Maybe even have some more complex example programs with numpy. Use some kind of graphing library like matplotlib. Maybe this is covered in the intermediate class?

11. The instructors notes should be easier to follow, the combination of Google Doc and github, little hard to follow.

12. I enjoy the class, but a lot of the exercise are interesting to the language but not necessary useful in practice. If adding at least one of the file processing, data conversion and implementation of a algorithm would help.

13. More comments in the .py files would be really helpful. Even a small summary (in a pdf format?) would be great.

14. The detailed approach taken to help us understand the principles of Python, by replicating its built in functions, is an excellent way to establish a deep understanding of the language. But this is best done in an academic environment, in which one can follow this conceptual approach to its end. For my purposes, however, I was left without a broader understanding of how to work with Python generally. Without the ability to follow up with the rest of the language in equal depth (although I certainly intend on taking the next class), I feel the depth of information presented exceeds my needs, while the breadth of information fails to meet them. Don't mistake my constructive criticism for dissatisfaction, however. I would take the class again knowing what I know now. It's probably true as well that my needs are not representative of the rest of the class's needs.

15. There's a lot of material to cover, thus it's important to keep things going at a proper pace. At times the students would debate finer points of a small matters for an extended period of time. One example is the for/else loop debate which lasted for about 15 minutes. In my opinion this was 14 minutes longer than it deserved. I think the instructor could have reigned this in a little more.

16. - Add instruction on "good Python programming style" -- Error handling -- Code documentation -- Required/suggested elements needed for defining classes, iterators, etc - Add examples of practical applications of Python to scientific problems -- More on file I/O and command line I/O interaction -- String/data parsing and handling (ASCII and binary) -- Date/time operations and conversions - Provide a quick overview (or reference) for common Python libraries and what is in them

This course was an excellent introduction to Python for application developers. But as a scientist interested in using

17. Python for analyzing data and modelling, I would have liked more of a focus on getting up and running to do productive work. For example, in almost all of the computing I do, I need to input a data file, perform some computations and analysis of the data, then display a plot of some result that allows me to visualize the data/model. I believe the package SciPy includes most of this needed capability and if just one of the days had been spent covering getting started producing products that could be used in papers, presentations, reports (i.e. analysis and plots), the course would have better served most of the attendees.

5. General Comments:

1. Glenn is very good at keeping students involved and making sure everyone keeps up with the material. Great instructor!

2. The instructor is very knowledgeable and engaged in the entire class to lively discussions. The style of his teaching is very effective by picking on individuals to answer various questions. He dives in the hard core of python straightly, however it is not hard to understand even for someone like me who has not have any python experience before. Great classes.

3. The instructor was very patient and made sure the entire class kept up with the subject matter.

4. Great class!

5. Great class!

6. I am new to OO programming and still learned a ton, so this course is worth it even for those new to OO.

7. It was a fun class to attend. I'd like to attend a more advanced Python class in the future.

8. Great Class

9. Before the beginning of the class, it'd be useful to have instruction on how to setup or run python3 environment. Which is not usually present in most OS.

10. Glenn is very patient and knowledgeable. I think this is something that makes his classes very interesting and beneficial to the users.

11. Great instructor - thanks for the class!

12. Great course, it covers a lot of important basics, and the exercises are excellent. Instructor was very helpful and knowledgeable. There were many things I was confused about before this course that I feel way more comfortable with after. I'm glad to have taken it and will recommend it to others.

13. Good introduction to the language and how it works. Instructor is very passionate and dynamic, and engages class constantly, which is good. Covered a wide range of material quickly. I enjoyed the class, but wish there was one more day of examples of scientific applications of Python to typical JPL engineering problems. Would also like a quick overview of existing Python libraries of functions (and general description of what you can find in them) so that we have a better roadmap of what built-in capabilities already exist. Perhaps this is already in follow-on classes at the Intermediate and Advanced levels. Would love to see a short 1-2 day follow-on offering to show case practical applications of Python, including good Python programming practices so that we structure our code in a standard way.

14. The course ended with an encouragement to participate in a online activity to solve example programming problems. I took the course so that I could begin to use Python to solve the myriad of problems that I already have waiting to be solved in my everyday work. I don't need more problems - I need solutions. The idea is to get us to practice and use Python, but I suggest that the best way to accomplish that goal is to demonstrate at least one example where the problem is to input a realistically complicated and significantly large file of data, use Python to do some analysis beyond what could be easily accomplished with excel, and produce a product, e.g. an output file or plot, that we would recognize as an accomplishment that we might need to do on the job. Then we hopefully could begin using Python in our everyday work activities immediately upon completing this class and be encouraged to get practice and familiarize us with the utility and beauty of Python as it applies to our needs. The understanding of the Python language given in this course is very useful in terms of understanding using it. Perhaps my needs would be better served if I could augment this class with a shorter "Scientific Python" course more focused on using the language plus the SciPy package, etc. to cut to the chase and use Python to better do my work.