

[Return to Classroom](#)

Explore US Bikeshare Data

REVIEW	CODE REVIEW	HISTORY
--------	-------------	---------

Meets Specifications

Thank you so much for your submission!

The script covers all the requirements in a neat manner. Congrats for your great job!
For future projects in your career as data scientist / analyst, please keep always in mind some pillars which you started seeing here but which will make you outstand vs your peers:

- Code readability + docstrings. Proper code documentation is key for the future yourself who will have to come back to an old code, but also to make sure you are always ready to proudly share your code with your peers, managers or clients.
- Encapsulation and code reusability. Keep using functions as you did in this exercise to make sure you can reuse your own code in the future.

Please, keep up the good work and happy learning ahead! Also, remember that your comments and feedback are more than welcome!

Code Quality

✔	<p>All code cells can be run without error.</p> <p><i>Tips:</i> Implement safeguards against invalid user inputs that can potentially break the codes. Please refer to the “Solicit and handle raw user input” rubric item for further details.</p>
✔	<p>Appropriate data types (e.g. strings, floats) and data structures (e.g. lists, dictionaries) are chosen to carry out the required analysis tasks.</p> <p>Beyond what you have already used in this course, I strongly encourage you to continue digging into this area. Having an in-depth knowledge of data structures and algorithms will help you think differently about how to design a solution for the requests that you will be requested with. I know you have a lot on your plate, but if you have some idle time after finishing this course, go and continue learning about this topic.</p> <p>Here are some free resources about this topic you can use to complement the course materials.</p> <p>1) http://greenteapress.com/thinkpython/html/index.html 2) https://runestone.academy/runestone/books/published/pythonds/index.html 3) https://docs.google.com/viewer?a=v&pid=sites&srcid=dnVrbWFsYmFzYS5jb218d3d3fGd4OjU3NWZjOWU1MTM4ZTI4OQ 4) this one is my favourite, it is about solving hard mathematical problems via programming languages. You really need to know how to play with data types : https://projecteuler.net</p>
✔	<p>Loops and conditional statements are used to process the data correctly.</p>
✔	<p>Packages are used to carry out advanced tasks.</p> <p>Pandas is an amazingly handy library widely used in the industry. Said so, data wrangling is just the first part of what is expected from an excellent Data Scientist / Data Analyst. I strongly encourage you to spend some time in something we don't cover in this particular assignment: data visualisation.</p> <p>There are amazing packages out there that complements Pandas. Even though they are not discussed in the course, I'm sure you will greatly benefit from learning a bit about them..</p> <p>I suggest you following visualization libraries, which are quite trendy right now: 1) plotly (https://plotly.com/python/) 2) streamlit (https://www.streamlit.io)</p>
✔	<p>Functions are used to reduce repetitive code.</p> <p>Functions are a key component of a good code. Organising the code in functions with a clearly delimited scope and interface helps writing clean, readable and reusable code. Every time you face a coding problem, please spend some time thinking on what tasks is your code performing and if some of them could be encapsulated in functions to improve the code. Let me just insist: improving the code may mean making it work better or adding functionalities, but something vital which is often ignored is its readability! Someone else may need to continue working with your code in a while or even for your "future yourself" will need to revisit it: make sure you make their lives easy!</p> <p>The next step in terms of code clarity, readability and encapsulation are "classes". Take a look into this concept if you are not familiar with it. Its use is not always necessary and often just using functions a code can be ok. Said so, specially when projects get big the use of classes can really be a material improvement.</p> <p>If you want to to take a quick look, I suggest you this quick tutorial about classes: https://www.learnpython.org/en/Classes_and_Objects</p>
✔	<p>Docstrings, comments, and variable names enable the readability of the code.</p> <p><i>Tips:</i> Please refer to the Python's documentation PEP 257 -- Docstring Conventions. Example of docstring conventions:</p> <pre>def function(a, b): """Do X and return a list."""</pre> <p>Docstrings are very important as they help document your program and make it much easier to understand, especially at a later date. So always remember to include them whenever you write a function in your code. You can visit Python Docstrings for more information on docstrings.</p> <p>It's quite common that we (yes, we all do so :)) ignore writing comments, or documenting the functions. This is fine in the short term, but in the long run the code maintainability suffers a lot. I can't emphasize this more: if you want to deliver a superior code, pay attention to documenting / commenting your code. Even if you don't perceive it while you write it, I can guarantee you this is for sure improving the quality of your code ad your colleagues / employers will appreciate it.</p>

Script and Questions

✔	<p>Raw input is solicited and handled correctly to guide the interactive question-answering experience; no errors are thrown when unexpected input is entered.</p> <p>User inputs should be made case insensitive, which means the input should accept the string of "Chicago" and its case variants, such as "chicago", "CHICAGO", or "chicAgo".</p> <p>You should also implement error handlings so your program does not throw any errors due to invalid inputs. For example, if the user enters "Los Angeles" for the city, the error handling should reject the user input and avoid breaking the codes.</p> <p>Your input is brilliantly handled guaranteeing that it's handled in a case insensitive way. Input handling is one of the most challenging tasks since there are many corner cases and unexpected circumstances than can mess things up. I can just say: Good job!</p>
✔	<p>Descriptive statistics are correctly computed and used to answer the questions posed about the data.</p> <p>Raw data is displayed upon request by the user in the following manner:</p> <ul style="list-style-type: none">• Your script should prompt the user if they want to see 5 lines of raw data,• Display that data if the answer is 'yes',• Continue iterating these prompts and displaying the next 5 lines of raw data at each iteration,• Stop the program when the user says 'no' or there is no more raw data to display. <p>Tips: you can implement the <code>while</code> loop and track the row index in order to display the continuous raw data.</p>

[Download Project](#)

Rate this review

[Return to Path](#)

[Start](#)