## **Self-Reflection: Creating a changelog**

1.

Question 1



In this activity, you’ll reflect on the importance of record keeping as a junior data analyst, and consider how a changelog helps with that process.

In previous activities, you've reviewed the different types of questions to ask before exploring data, the importance of pre-cleaning data, the basic functions of SQL, how to clean data with spreadsheets, and more. As a junior data analyst, most of your projects will consist of these activities. Moreover, as you have experienced, each of these project tasks follows a complicated process. Therefore, consistent and accurate record-keeping is essential.

### Changelogs



A changelog is a document used to record the notable changes made to a project over its lifetime across all of its tasks. It is typically curated, and the changes it records are listed chronologically across all the different versions of the project.

The major benefit to using changelogs is that contributors and users connected with the project get a specific list of what important alterations have been made, when they were made, and sometimes, what version they were released for. It is an invaluable tool for communicating how the project has evolved over time to coworkers, management, and stakeholders.

A changelog for a personal project may take any form desired. However, in a professional setting, particularly when collaborating with others, readability is important. In light of this, you can follow the guiding principles that make the interpretation of the changelog accessible to others:

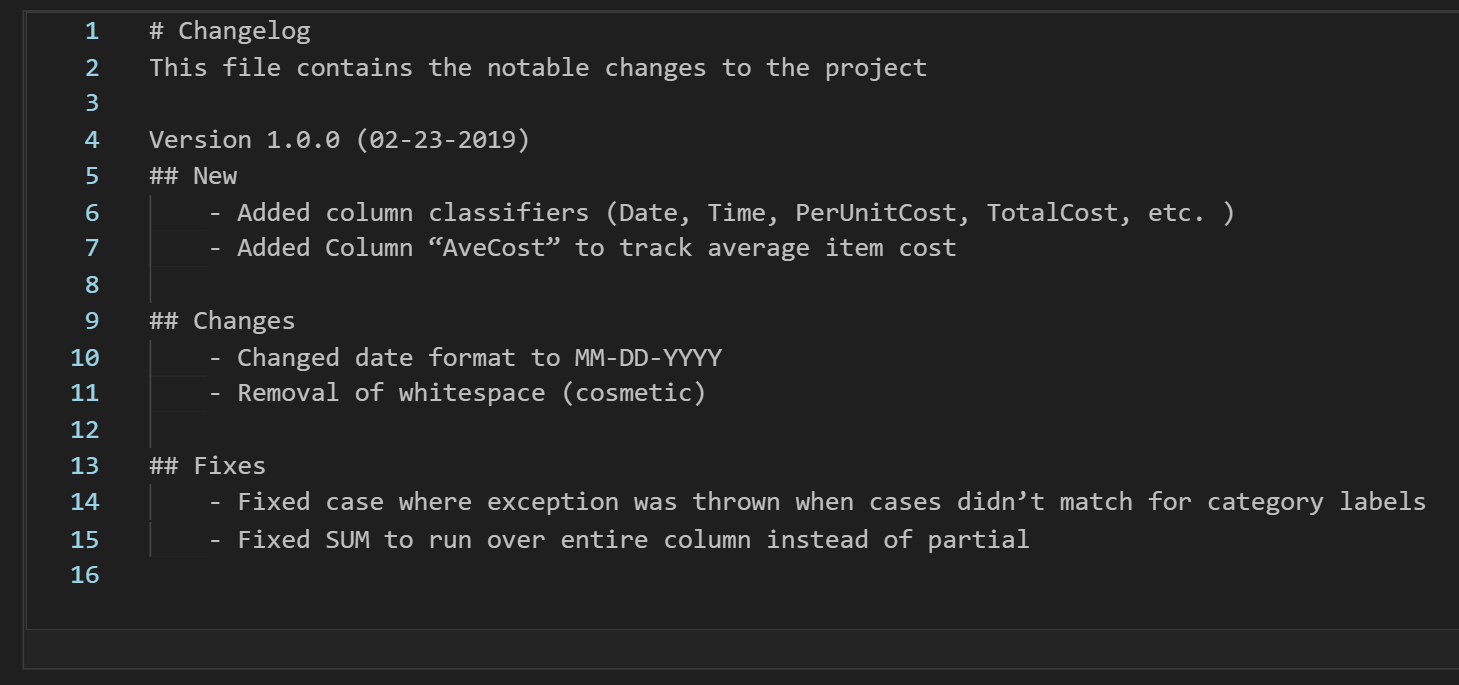
* Changelogs are for humans, not machines, so write legibly
* Every version should have its own entry
* Group the same types of changes. For example, "Fixed" should be grouped separately from "Added"
* Each change gets its own line
* Versions should be ordered chronologically starting with the latest
* Release date of each version should be noted

Types of changes usually fall into one of the following categories. All the changes for each category should be grouped together.

* Added: new features introduced
* Changed: changes in existing functionality
* Deprecated: features about to be removed
* Removed: features that have been removed
* Fixed: bug fixes
* Security: lowering vulnerabilities



Examine the figure below for an example of a changelog. Note that the following example is written in [Markdown](https://docs.github.com/en/free-pro-team@latest/github/writing-on-github/basic-writing-and-formatting-syntax), as it is common to keep changelogs as a readme file in a code repository.





In the final course in this certification program, you’ll have the opportunity to complete a capstone project. This is a great chance to demonstrate your ability to organize a project like a professional data analyst by keeping your own changelog. You can do this using a simple companion text file that can be included with the project write-up.

In the log, you will want to record the various changes, additions, fixes, etc. that were discussed above. Arrange them using bullets or similar with one change per bullet. Group similar changes together with a label describing the change immediately above them.

Use different version numbers for each milestone reached in your project. Within each version, place the logged changes that were made since the previous version (milestone). Dates are not generally necessary for each change, but they are recommended for each version.

Keep this in mind when you reach the capstone project in a later course, and don’t be afraid to revisit this lesson if you have questions!

Use this knowledge to create and maintain a changelog for your data cleaning capstone project. It will help you stay organized and make collaborating with others much easier.

## Reflection



In the text box below, write a 5 -7 sentence (100 - 150 words) response addressing the following questions:

* What makes for a good changelog?
* How do you decide if a change is significant enough to include in the changelog?

**1 / 1 point**

A good changelog which includes all the meaningful changes that happened to the project. And it should have a record of this. If it's a small or big change I think it's important to put it in the changelog file.

**Correct**

Thank you for your response! A good changelog should indicate the notable changes to a project. If you perform any of the following changes to the dataset while cleaning, you should capture those changes in the changelog:

* Treated missing data
* Changed formatting
* Changed values or cases for data

Chances are that you have made some of these changes while cleaning the data. Also, any other substantial changes should be included. As a rule-of-thumb, when in doubt about the significance of a change, you should enter it into the changelog.