

How do I Complete this Project?

This project is connected to the [Data Visualization](#) course, but depending on your background knowledge of data visualization, [dimple.js](#), and [d3.js](#) you may not need to take the whole course to complete this project.

After completing Lesson 2 and Problem Set 2 of the course, you will be able to complete this project since you will have learned about [dimple.js](#).

If you want to become more technical and expand your skill set, you can continue to Lesson 3 and Lesson 4, in which you will learn more about narrative structures and how to create graphics using [d3.js](#). The [d3.js](#) library has a steeper learning curve, and we encourage you to take on the challenge if you desire.

The process for evaluating your project is not affected by your choice of using [dimple.js](#) or [d3.js](#).

Introduction

For the final project, you will create an **explanatory** data visualization from a data set that communicates a clear finding or that highlights relationships or patterns in a data set. Your work should be a reflection of the theory and practice of data visualization, and you must use either [dimple.js](#) or [d3.js](#).

We will provide some options of data sets to explore; however, you may choose to explore an entirely different data set. You should be aware that finding your own data set and cleaning it using Python, R, or some other language can take considerable time and effort. This can add as much as a day, a week, or even months to your project so embark on the adventure to find and clean a data set if you are truly prepared with programming and data wrangling skills.

You have three options for this project. You should pick an option based on your prior experience with data munging and exploratory data analysis. The option you choose will not affect the evaluation of the project.

- **Option 1**

Select one of the beginner data sets, which already has a summary of findings, from the [Data Set Options](#) document. Then, create a visualization that communicates the findings.

- **Option 2**

Select one of the intermediate data sets from the [Data Set Options](#) document. You will investigate the data set to share a story or message about the data and then create a suitable visualization.

- **Option 3**

Find a data set, investigate it, and share your findings in a visualization. Your final graphic should primarily be explanatory, but it may also contain exploratory components. You can find a list of recommended websites to find data sets in the [Data Set Options](#) document. You should be aware that finding your own data set, cleaning the data set, and analyzing it (using R, iPython Notebook, or another tool) can take considerable time and effort. This can lengthen the time you spend on your project by days, weeks, or even months. Choose the option only if you feel prepared for a challenge!

Now, on to the details!

Step One - Choose a Data Set

First, you will choose a data set from the [Data Set Options](#) document or find a data set to explore and visualize. You should choose a data set based on your prior experiences in programming and working with data. The data set you choose will not increase or decrease your chances of passing this project.

Step Two - Get Organized

Eventually you'll want to submit your project and share it. If you are familiar with [GitHub](#), we encourage you to create a public repository or a public [Gist](#) for your project to track changes. Otherwise, you need to create the following files.

- an **index.html** file containing the code to create your visualization (you may include the JavaScript and CSS in this file or separate them in other files)
- a **README.md** file that includes four sections...
 - **Summary** - in no more than 4 sentences, briefly introduce your data visualization and add any context that can help readers understand it
 - **Design** - explain any design choices you made including changes to the visualization after collecting feedback
 - **Feedback** - include all feedback you received from others on your visualization from the first sketch to the final visualization
 - **Resources** - list any sources you consulted to create your visualization
- **data files**
 - the final data set used to create the visualization (usually .csv, .tsv, or .json file)
 - a codebook or other files related to the data set (description, readme, license)
- **OPTIONAL FOLDERS IF YOU USE [GITHUB](#)**
 - **data** folder to include all the data related files
 - **js** folder to include .js files (not needed if javascript is in the index.html file)
 - **css** folder to include .css files (not needed if CSS is in the index.html file)

Step Three - Find a Data Story

Explore your data set and craft a message or story around your data! Think about the overall message you want to convey and think about the comparison(s) or relationship(s) you want your readers to see.

Step Four - Create Your Visualization

First, sketch ideas for your visualization. Once you settle on a sketch, explain any design choices in that sketch, such as chart type, visual encodings, and layout, in the **Design** section of the **README.md** file. Then, write code to create your visualization using either [dimple.js](#) or [d3.js](#). The visualization must include animation, interaction, or both. See the [Project Rubric](#) for more information.

Step Five - Get Feedback

Share your visualization with **at least 3 other people** and document their feedback. There are many ways to get feedback, and more feedback is generally better! Here are some options. - Share your visualization with others in person and have them think aloud as they read and explore the graphic so you can document what

stands out to them and how they interpret the graphic. - Share a link to your repository in the discussions and ask others to share constructive criticisms. Be sure to offer advice to others who are seeking feedback too!

- Create and share a [Gist](#), which contains an **index.html** file, data file, and any .js or .css files). Directions for creating and sharing a Gist can be found at <http://bl.ocks.org/>.
 - **Box Plots Gist EXAMPLE:**
 - <https://gist.github.com/mbostock/4061502>
 - <http://bl.ocks.org/mbostock/4061502>

You might need to ask specific questions to prompt the reader. Here are some questions to help you. You can, of course, ask others.

- What do you notice in the visualization?
- What questions do you have about the data?
- What relationships do you notice?
- What do you think is the main takeaway from this visualization?
- Is there something you don't understand in the graphic?

Step Six - Document Feedback and Improve the Visualization

For each person that gives you feedback, add the person's feedback to your **README.md** file in the *Feedback* section. As you improve and iterate on your visualization, update your code **AND** describe any changes in the *Design* section of the **README.md** file.

You should save multiple versions of your data visualization after you make changes to it. You can do this using GitHub or a Gist by making commits to your project, or you can simply save multiple version of you data visualization such as index1.html, index2.html, ... , index_final.html. Remember to save related files with similar numbers...

- main1.js, main2.js, ... , main_final.js (if you separate your Javascript from the HTML file)
- style1.css, style2.css, ... , style_final.css (if you separate your styling from the HTML file)

When should you save your files? You should save your files whenever you have a working version of your data visualization. If you get feedback and make changes, then wait to save the file until you have the data visualization working as you want it. Your goal is to build evidence that you have shared your visualization, received feedback, and responded to that feedback. You will need to submit the different versions of your visualization. **At a minimum, you need to submit an initial version of your data visualization (either as a sketch or as code) and the final index.html file and related files.**

Step Seven - Review

Use the [Project Rubric](#) to review your project. If you are happy with your submission, then you're ready to submit your project. If you see room for improvement, keep working to improve your project!