

[Return to Classroom](#)

# Communicate Data Findings

REVIEW

CODE REVIEW

HISTORY

## Meets Specifications

Hi Udacity Learner,  
Mostafa Abobakr  
Congratulations on completing the project! 🎉

You have done outstanding work on this project.

ADDITIONAL LINKS 🔗 TO READ IN FREE TIME

- 7 Fundamental Steps to Complete a Data Analytics Project
- A Comprehensive Guide to Data Exploration

Sure you have learned a lot and we encourage you to keep up with this hard work. Have a nice day and good luck forward. 🍀

What issues did you face in the project?  
How long you took to complete this project?  
Any suggestions or ideas you may have on the project.

- Don't forget to rate my work as a project reviewer! Your detailed feedback is very helpful and appreciated - thank you!
- I'll look forward to reading from you. Thanks a lot! 🙏

## Code Quality



All code is functional (i.e. no errors are thrown by the code). Warnings are okay, as long as they are not a result of poor coding practices.



All the code runs without any error. Good job.

ADDITIONAL LINKS

- If you want to learn more about python programming you can [follow this link](#)
- 15 Python tips and tricks to master Data Science and Machine Learning
- The ultimate guide to writing better Python code



The project uses functions and loops where possible to reduce repetitive code. Comments and docstrings are used as needed to document code functionality.

- great job performing some wrangling tasks before the actual exploration.
- You have documented your code functionality with the help of comments and docstrings.
- Functions are used to avoid code repetition.

```
# ArrDelay violinsplots per Months for flights with 15+ minutes arrival delay
#colors = (color for color in sns.color_palette()[0:4])
def by_stolkinp_size_kwsrgs):
    #color=sns.color_palette()
    sns.violinplot(y=y, #color=color, #,orient='v')
    sns.set(font_size=15,style='whitegrid')
    g = sns.FacetGrid(data=df, row='Month',
                    col_wrap=4,
                    size=3.5,
                    gridspec_kw={ 'grid':1, 'style':'whitegrid',}) #style='whitegrid',
    #sns.set(style='white')
    g.fig.suptitle('Arrival Delay (15+ minutes) violin plots per Months')
    g.map(violin, 'ArrDelay', #,binom=20,)
    g.set_titles(col_name)
    #g.set_ylabel('Freq')
    #g.set_xlabel('Freq')
```

ADDITIONAL LINKS WHICH YOU CAN READ IN YOUR FREE TIME.

- Ten Good Coding Practices for Data Scientists
- Six steps to more professional data science code
- Good coding practices - Describing your code

## Exploratory Data Analysis



The project appropriately uses univariate, bivariate, and multivariate plots to explore many relationships in the data set. Reasoning is used to justify the flow of the exploration.

- Great job defining three different sections in your exploration report for univariate, bivariate, and multivariate plots. The flow of your exploration was very smooth as you started from univariate and gradually moved on to bivariate and multivariate plots. Also, proper reasoning was used to justify the flow of the exploration.



Questions and observations are placed regularly throughout the report, after each plot or set of related plots.

- great job placing the question in between your exploration. it really helps in going through the flow of the exploration. I was able to follow along, and understand what was going on in your head while you were trying to answer those questions!
- You have added observation after the plots which really helped.

Of the features you investigated, were there any unusual distributions? Did you perform any operations on the data to tidy, adjust, or change the form of the data? If so, why did you do this?

'Carrier' delays had a wide spread range with a bit surprising delays-outliers values

I decided to move to work on DB Browser for SQLite to work more rapidly with this huge data points number of over 7 Million, and to get needed columns for my investigation, with the code. I reduced dataset from 29 to 10 columns, and I made a left join with carriers.csv data there to get carrier names instead of there codes. I exported the columns to 2008\_flights.csv later, then I came back again to Jupyter notebook to complete the work.

I extracted the columns I thought as helpful for this investigation, they were ['Month', 'Carrier', 'ArrDelay', 'Cancelled', 'CancellationCode', 'CarrierDelay', 'WeatherDelay', 'NASDelay', 'SecurityDelay'] into 'df\_inv' dataframe.

Great job framing question and answering them based on findings

From 'df\_inv', I derived:

```
1 'airline_cancelled' dataframe for data points of cancelled flights data
2 'on_time' dataframe for flights with less than 15 min. arrival delay and not null, or data points with arrival delays less than 15 minutes and not null
3 'df_inv_15' for flights with arrival delays that equal 15 minutes or more, then I sampled 'df_inv_15_samp' from them. I melted the delay-causes columns within 'cause' column, and I put their values in 'Minutes' column. Next, I removed 'Minutes' with zero, less or null values, producing 'melt_15' and 'melt_samp_15' dataframes for using new created columns from melting in investigating relationships among delay-causes and other features like month, carrier, and arrival delay.
```

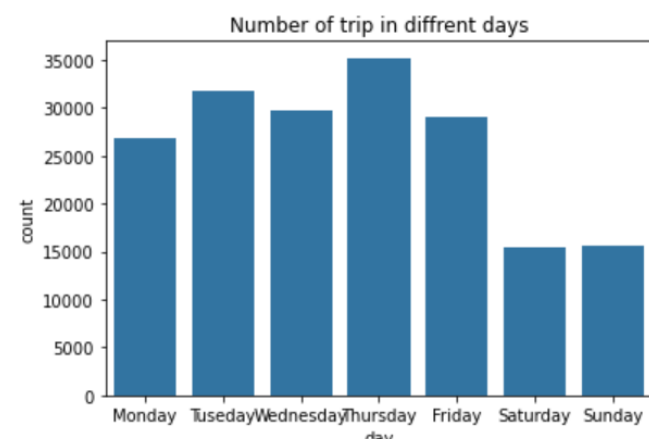
## Additional Notes

- Though you did this in some way, I wanted to explicitly talk about the **Question-Visualization-Observations framework**
- This framework involves asking a question from the data, creating a visualization to find answers and, then recording your observations.
- If you do this for each visualization, you'll be successfully narrating your exploration.

### Example

#### How many trips in a week?

```
In [119]: base_color = sns.color_palette()[0]
sns.countplot(data=df, x='day', order=["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"], color=base_color);
plt.title('Number of trip in different days');
```



In the barchart above we have an overview of the days where the trips found place. Further we want to analyse if these results change when we compared to different costumer types and in gender comparison.

In this figure we can see that first a question is placed and then that question is explained with the help of plot and observations.

## HOW TO ASK BETTER QUESTIONS IN THE EXPLORATION PROCESS?

Explore your data - Before performing any analysis on your data, it is important to explore the available data carefully

Determine the type of problems to be addressed using the data - the problem to be solved could fall under one of the following categories: descriptive analytics, predictive analytics, or prescriptive analytics.



Visualizations made in the project depict the data in an appropriate manner that allows plots to be readily interpreted. This includes choice of appropriate plot type, data encodings, transformations, and labels as needed.

- You have included a nice selection of univariate, bivariate, and multivariate plots.
- Your aesthetic and labeling choices have made the plots readily interpretable.

## Additional Notes

IMPORTANCE OF HAVING PROPER VISUALIZATION.

- It helps in making better decisions
- Data Visualization amplifies the messaging.
- Data visualization provides a clearer understanding.

ADDITIONAL LINK

- Fundamentals of Data Visualization
- How to avoid overplotting with python

## Explanatory Data Analysis



A section in the submitted materials includes a summary of main findings that reflects on the steps taken during the data exploration. The section also describes the key insights that are conveyed by the explanatory presentation.

- Great job summarizing your project in the readme file.
- You have included a summary of the main findings and key insights.

ADDITIONAL LINKS

- You can go through this link to know answers to the questions.

Why is a README File necessary?  
What is a README File?  
How to write a good README File?  
How to make your README file more interesting?



A slideshow is provided, with at least three visualizations used in the presentation to convey key insights. These key insights match those documented in the summary. Each visualization is associated with comments that accurately depict their purpose.



Great job, submitting the slideshow. The format of your slideshow is correct and it meets all the rubrics requirements.



All plots in the presentation have an appropriate title with labeled axes and legends. Labels include units as needed. Plot type, encodings, and transformations are all appropriate.



All the plots in the presentation are well polished that is they have an appropriate title with labeled axes and legends.

[Download Project](#)[RETURN TO PATH](#)

Rate this review

[START](#)