**Exercise- Altair: Declarative Visualization in Python**

**Student Name: Student Id:**

**Date:**

Please use the screenshots ONLY as a reference. The written instructions have to be followed AS written.

### Objective:

The objective of this exercise is to develop skills on how to use Altair for Visualization using Jupyter notebook in Python.

**Prerequisite:** Install Anaconda 2019.03 Installer, Python 3.7 version

Since we already have Anaconda installed in our systems from Assignment 3 on Data Cleansing on Jupyter Notebook, go ahead and launch **Jupyter Notebook** in the anaconda Navigator.

**A screenshot of a cell phone

Description automatically generated**

* In the Jupyter notebook ‘Files’ tab, Navigate to data folder and create new Python 3 notebook.

**A screenshot of a cell phone

Description automatically generated**

* Rename the file in given format by clicking on “Untitled” (top left corner) in the newly created notebook.

A screenshot of a cell phone

Description automatically generated

**Instructions:**

**Step 1: Import and Installing Required libraries and packages**

Write the following code in the next cell, this will install the required packages we need for this assignment.

* Remember to check the output of the written code block we need click on RUN.



After you run this, write the following to import all the required Libraries.

A screenshot of a cell phone

Description automatically generated

**Question:**

1. **Take a screen shot of your page, showing page Title (your name) and page background.**

**Step 2: Setting Directory and Reading File**

Add the following code.

* Add the path where your csv file(‘JHU.csv’) provided along with the assignment is stored in your local system.



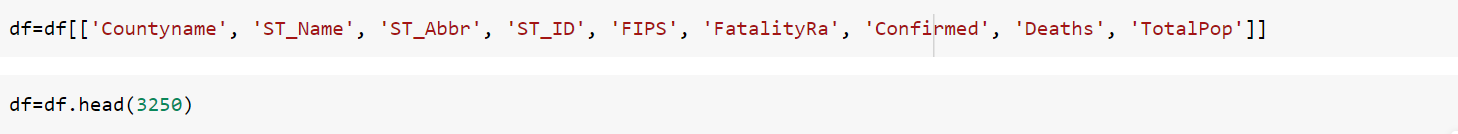
* Now, to look at the first five instances of the dataset, we use the head() function.

**A screenshot of a cell phone

Description automatically generated**

**# Subsetting the Data Set.**

Since we have a lot of unnecessary columns in the dataset, we subset our data to include only those columns necessary for our analysis. As for the second line of the code, all entries from row number 3251 are nulls, so we remove them from our analysis and subset the data to only include particular columns and rows.

****

**Question:**

1. **Paste a screen shot of your updated dataset displaying the first five instances.**

**Step 3: Creating a Bar Chart Visualization in Altair**

*A bar chart Visualization showing Top 10 states of COVID-19 cases:*

1. Type in the following code in the code cell to get the Top 10 states in US according to the sum of confirmed cases:

**A screenshot of a social media post

Description automatically generated**

**Explanation:**

In the code above, we group the data by ST\_Name(which is States in the US), use the rank function of altair and sort the data by the count of sum of confirmed cases in the United States to display the bar chart we get from the above code.

**Question:**

1. **Paste a screen shot of your visualization above.**

**Question:**

1. **List the top 5 states according to the bar chart(we will be needing these for our next visualization).**

**Step 4: Creating an Interactive Visualization in Altair**

*An interactive visualization of confirmed cases and deaths across top 5 states(calculated above) in the US*

1. Type in the following code in the code cell to get an interactive visualization comparing the Confirmed cases and Deaths across the top 5 states:

A screenshot of a cell phone

Description automatically generated

**Explanation:**

For the code above, you have to put the Top 5 State Names(just like they appear in the data) in inverted commas in the bracket which is hidden by a black rectangular box.

One of the unique features of Altair, inherited from Vega-Lite, is a declarative grammar of not just visualization, but *interaction*. There are three core concepts of this grammar:

* the [**selection()**](https://altair-viz.github.io/user_guide/generated/api/altair.selection.html#altair.selection) object which captures interactions from the mouse or through other inputs to effect the chart. Inputs can either be events like mouse clicks or drags. Inputs can also be elements like a drop-down, radio button or slider. Selections can be used alone but if you want to change any element of your chart you will need to connect them to a *condition*.
* the [**condition()**](https://altair-viz.github.io/user_guide/generated/api/altair.condition.html#altair.condition) function takes the selection input and changes an element of the chart based on that input.
* the bind property of a selection which establishes a two-way binding between the selection and an input element of your chart.

Bindings and input elements can also be used to filter data on the client side. Reducing noise in the chart and allowing the user to see just certain selected elements(Just like we did in our example above by adding a dropdown to choose any one state keeping the other states data grey). We have also added an element of interactivity to the chart which can be accessed by zooming in or out of the scatter plot to look at individual points more clearly.

Since we see over plotting of points towards the bottom left side of the chart, we have also added an element of interactivity which will help us zoom in and out of the chart to see the data points more clearly and a little more spread out.

**Question:**

1. **Paste a screen shot of the above visualization as a whole.**

**Question:**

1. **Paste a screen shot of the above visualization by selecting any one of the 5 states from the dropdown list.**

**Question:**

1. **Paste a screen shot by zooming the map to show data points more clearly for 0-4000 in the x-axis.**

**Step 5: Creating a geographical choropleth map at the county level of the Fatality Rates due to COVID 19 across US.**

*A choropleth map describing the spread of corona virus fatality rates at the county level*

1. Type in the following code in the code cell to get a choropleth map describing the count of confirmed cases at the county level through different hues of the same color:

*A screenshot of a social media post

Description automatically generated*

**Explanation:**

In order to visualize Geopositioned data in Altair, we need to join our dataset with the inbuilt Altair dataset. Since, here we are planning to visualize county level data, we will use the “Counties” specification from the inbuilt dataset.

As can be seen in the above code, we have chosen the mark as geoshape which means Geographical visualization. Since, we want to differentiate between different counties based on the Fatality Rate due to Covid-19, we choose color encoding for it.

In order to join our Covid dataset with the inbuilt dataset, we use Altair’s lookup function and join the dataset on a common column which in our case is ‘FIPS’ and ‘id’. The FIPS maps with the county IDs. **FIPS codes** are numbers which uniquely identify geographic areas. State-level **FIPS codes** have two digits, county-level **FIPS codes** have five digits of which the first two are the **FIPS code** of the state to which the county belongs.

**Question:**

**i. Paste a screen shot of your above visualization.**

**Instructions:**

1. Submit the assignment document in Microsoft word.
2. Submit .ipynb file created

Make sure you submit all the files.