Dash Fundamentals Assignment

The following versions of Dash and Plotly are used for this assignment:

- Dash 2.13.0

There is no need to submit assignments anywhere. This is for your own practice.

Do your best to solve these assignments before looking at the solutions. Even if you don't solve all the exercises correctly, the process of trying is where you learn the most.

The solutions can be found at this Google Doc.

Reminder:

> Please make sure to open an account on the Plotly Dash forum: https://community.plotly.com/
>
> Once your account is created, please **accept this invite** to join the Dash Online Course group. https://community.plotly.com/invites/Kq5BnFpMh1
>
> Then, go to the dash-online-course tag, and click the bell icon on the top right corner of the page to select your notifications settings.

Exercise A: incorporate the dataset `shades.csv` into your app. And create the following layout in one app file:

1. A Dropdown that uses the column `brand` as the dropdown options. Make sure the brand names are unique (do not repeat themselves). Then, assign "Revlon" as the initial value.

2. A RadioItems component in which the values from the column named `group` are assigned to the `options` property. The options should be unique and sorted from 0 to 7.

3. Update the `options` property of the RadioItems component so that the `values` (of the options) represent numbers from 0 to 7, but the `labels` are their respective strings (see Readme-shades for the strings).

Exercise B: using the same `shades.csv` create another app that incorporates Dash AG Grid into the layout:

1. The Dash AG Grid should represent the complete dataset with all its columns.

2. Using Pagination, add automatic pagination to Dash AG Grid and make sure all columns fit into the screen with no horizontal scroll bar (using the `columnSize` property).

Exercise C: using the same `shades.csv` create a new app, where the layout has two new Dash Core Components that you haven't used so far.

There will be no posted solution to exercise C. The goal is to choose whichever components you prefer to practice with.

Exercise D: using the following scatter plot example, add a scatter plot to your app that displays `V` (value/brightness) on the x-axis and `S` (saturation) on the y-axis.

Clue: to display the plot in the layout, remember to assign your plot to the `figure` property of the dcc.Graph, for example: `dcc.Graph(figure=my_scatter_plot)`

Exercise E: start thinking of an app that you would like to build with Dash by the end of course