Reporte – Taller 3

Primer tablero en Dash

i) Incluya el código con el que se crea el objeto app.



ii) Use un atributo o método del dataframe para imprimir en consola los nombres de las columnas que componen el conjunto de datos.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza baja

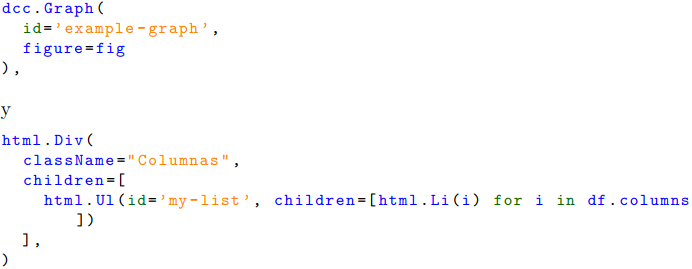
iii)Incluya un pantallazo donde aparezca el código en ejecución y la página cargada.



iv) Describa qué hace el comando “px.bar”.

El comando crea una gráfica de barras utilizando la librería “plotly”. Dicha función recibe como parámetro los datos que se desean gráficas, que en el contexto actual corresponden al dataframe “df”. Además, se deben especificar los ejes (X y Y), así como el modo de graficar la información (barmode = ”group”). Dado que se tienen 2 grupos de datos: diagnóstico positivo y diagnóstico negativo, se utiliza el parámetro “color” para indicar que cada grupo tendrá un color que lo identifique.

v) Describa qué hacen las siguientes líneas.



El primer comando “dcc.Graph” renderiza la gráfica de barras definida anteriormente, además le asigna un id para identificarla. Luego, “html.Div” permite utilizar la etiquieta <div> de HTML. El parámetro “className” permite identificar el elemento y personalizar su visualización, como el tamaño de la letra, el color de fondo, el color de la letra, etc. Asimismo, el atribute “children” corresponde al contenido del objeto “Columnas”.

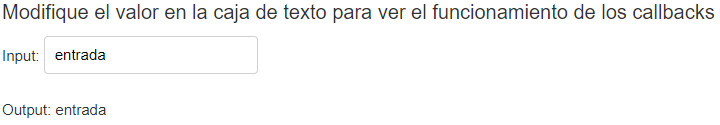
vi) Realice cambios en la gráfica.

Gráfico

Descripción generada automáticamente

Callbacks

i) Describa el layout de la aplicación.



La aplicación esta conformada por 3 elementos. El primero de ellos es el título, el cual se genero por medio del comando “html.H6” que asimila la etiqueta <H6> de html. Luego, se encuentra un bloque de contenido con el texto “Input” y una caja de texto con id “my-input” con valor inicial “entrada” renderizada por medio de dcc. Después de un espacio en blanco (blank row) se encuentra un bloque de contenido con el ide “my-output”. Dicho bloque imprime el texto que se ingrese en el bloque “my-input”.

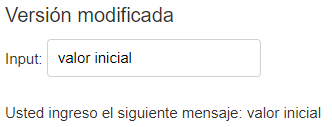
ii) Describa qué hace la función update\_output-div.

La función retorna un mensaje. Recibe un parámetro denotado como “input\_value” y genera un mensaje con su contenido. El mensaje tiene el formato: Output: input\_value, es por ello que utiliza a función “format”.

iii) Describa el decorador.

El decorador es el encargado de actualizar el mensaje en la página. El decorador tiene como parámetros “Input” y “Output”. El primero es aquel componente cuyo estado se monitorea, en caso de presentar algún cambio se utiliza la función update\_output\_div para actualizar el estado del componente “Output”. Así pues, cada vez que se ingresa algún carácter en la caja de texto del bloque “my-input” se realiza un llamado a la función update\_output\_div, la cual recibe como parámetro el nuevo contenido del bloque. La función retorna un mensaje, el cual será el nuevo valor o “component\_property” del compontente “my-output”.

iv) Realice cambios en la función update\_output\_div.



Visualizaciones e interacciones

i) Describa el layout de la aplicación.

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

La aplicación esta conformada por una gráfica de dispersión y un slider. El slider se utiliza para indicar el año del cual se desea mostrar la gráfica de dispersión. Así pues, la gráfica se actualiza según varia la posición del slider.

ii) Describa qué hace la función update\_figure.

La función genera una gráfica de dispersión a partir del año del cual se desea dicha ilustración. En particular, se recibe el parámetro selected\_year, el cual se utiliza para filtrar la información del año que se desea. Luego, se utiliza la función scatter de plotly para generar la gráfica. Por último, se retorna la nueva gráfica.

iii) Describa el decorador.

El decorador se encarga de llamar la función ante cada cambio en el deslizador. De modo que el Input es el componente “year-slider” y el Output es el bloque de contenido con el id “graph-with-slider”, cuyo nuevo valor es de tipo “figure”. Es decir, el usuario mueve el slider, se detecta el cambio en el bloque de contenido, se ejecuta la función, y su salida se utiliza para actualizar el contenido del componente “graph-with-slider”.

iv) Explore los datos empleando métodos y atributos de la clase dataframe.

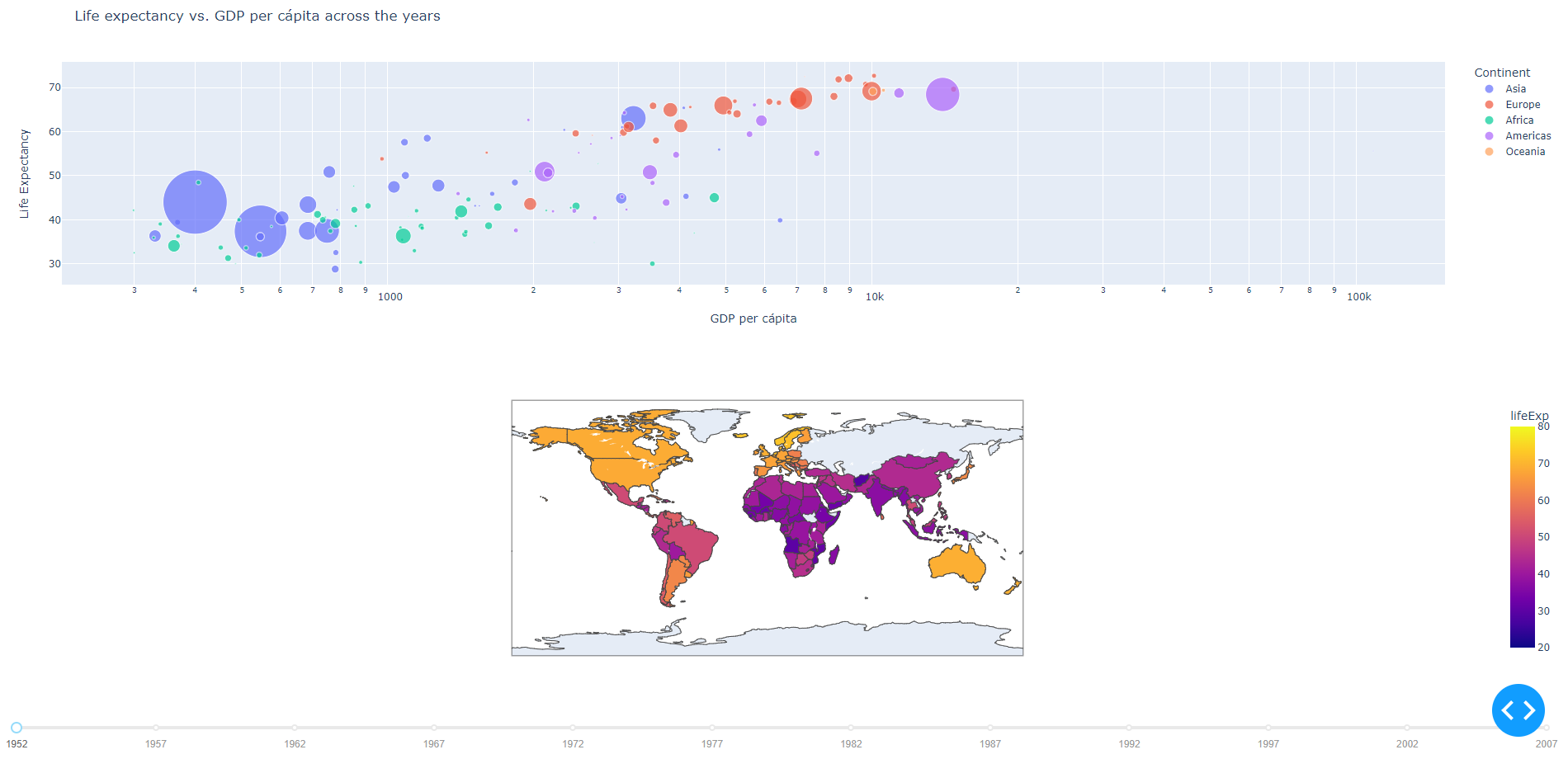
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Con el fin de entender los datos empleados se utilizaron métodos de la clase dataframe. En la imagen anterior se encuentra la salida resultante de aplicar la función describe.

v) Realice una nueva visualización de su interés.

A la visualización se agregó un Choropleth map con el fin de comparar los datos con respecto a regiones, como sur américa, la península ibérica, Asia central, etc.



**Nota:**

App1\_mod.py: Contiene la modificación en la aplicación del archivo app1.py.

App2\_mod.py: Contiene la modificación en la aplicación del archivo app2.py.

App3\_mod.py: Contiene la modificación en la aplicación del archivo app3.py.

App3\_see\_df.ipynb: Contiene la exploración de los datos usados por la aplicación app3.py.