Terrorism Analysis with Insights

1. **Introduction**:

This project comprises both Map tool and Chart tool which consists of World data and also India data.

The Python script can be run from the command line which will open a new page in the browser with a decent and usable interface.

This web page can be closed by an option which kills/stops the script from running in the background.

The User Interface has a Map Tool to generate a Scatter Geo Map with markers for highlighting the latitude/longitude where the incident is based on combinations of Month, Day, Attack Type, Region, Country, State, and City.

We can select any Day and Month based on that Day and Month.

We can filter the 9 types of attacks from the dropdown and another filter would be for Region, Country, State, and City.

The Hovering and clicking of the mouse display some information.

There is a range slider for selecting the range of years.

Another User Interface has a Chart Tool to show Stacked Line Chart images of the frequency of terrorist incidents each year.

We would be able to Group first by Country Attacked, Region, Target, Nationality, Target Type, Type of Attack, Weapon Type, Terrorist Organisation.

We would be able to search any group based on the search box.

The range slider is also included in this for selecting a range of years.

The Hovering of the mouse displays the year and counts the data shown.

2. **Related Work:**

The Project related to the python library called Dash. In this Dash there are lots of classes in it. It includes like dash core components that the there are lots of components which includes Inputs, Text Areas, Checkboxes, Graph etc. and there are dash html components which consists of Div, Header, Area etc.. So basically i worked with Plotly along with the dash.

3. **My Work**:

In this Project Terrorist Analysis with Insights I basically worked with the plotly.express to display the maps and charts and also the ui consist of lots of dash_core_components like dropdowns, Range Slider which shows the year of 1970 to 2018. I also used some Inputs from dash core components and i have used lots of callbacks to meet our expectation. The call back is the one where the ui can change my clicking some functions in my ui. Then, I have used some dash_html_components like H1 where it gives the heading of my ui and Br, Hr etc. I have researched a lot and worked hard and finally did the project .So, that's all about my work.

4. **Implementations**:

The screenshots of the implemented project are given below; there are a total of 14 screenshots

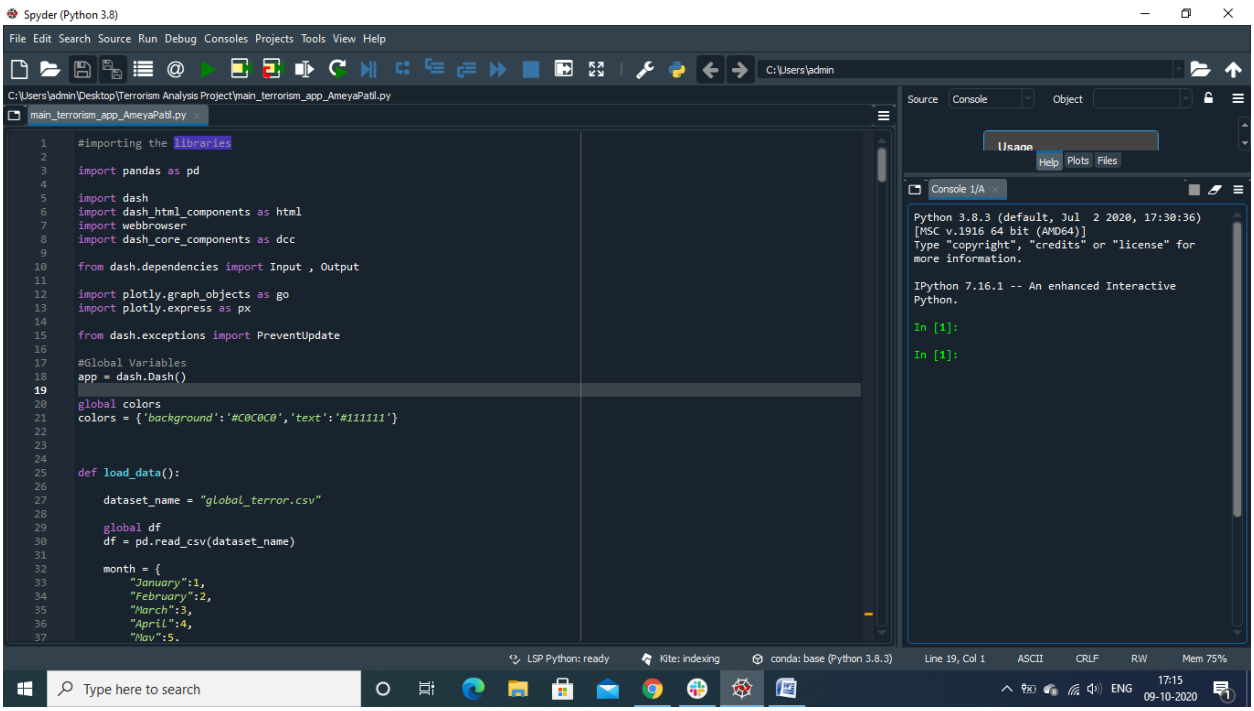
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py
#importing the libraries
import pandas as pd
import dash
import dash_html_components as html
import webbrowser
import dash_core_components as dcc
from dash.dependencies import Input , Output
import plotly.graph_objects as go
import plotly.express as px
from dash.exceptions import PreventUpdate
#Global Variables
app = dash.Dash()
global colors
colors = {'background':'#C0C0C0','text':'#111111'}
def load_data():
dataset_name = "global_terror.csv"
global df
df = pd.read_csv(dataset_name)
month = {
"January":1,
"February":2,
"March":3,
"April":4,
"May":5.
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: indexing
conda: base (Python 3.8.3)
Line 19, Col 1
ASCII
CRLF
RW
Mem 75%
Type here to search
ENG
17:15
09-10-2020

Fig.1

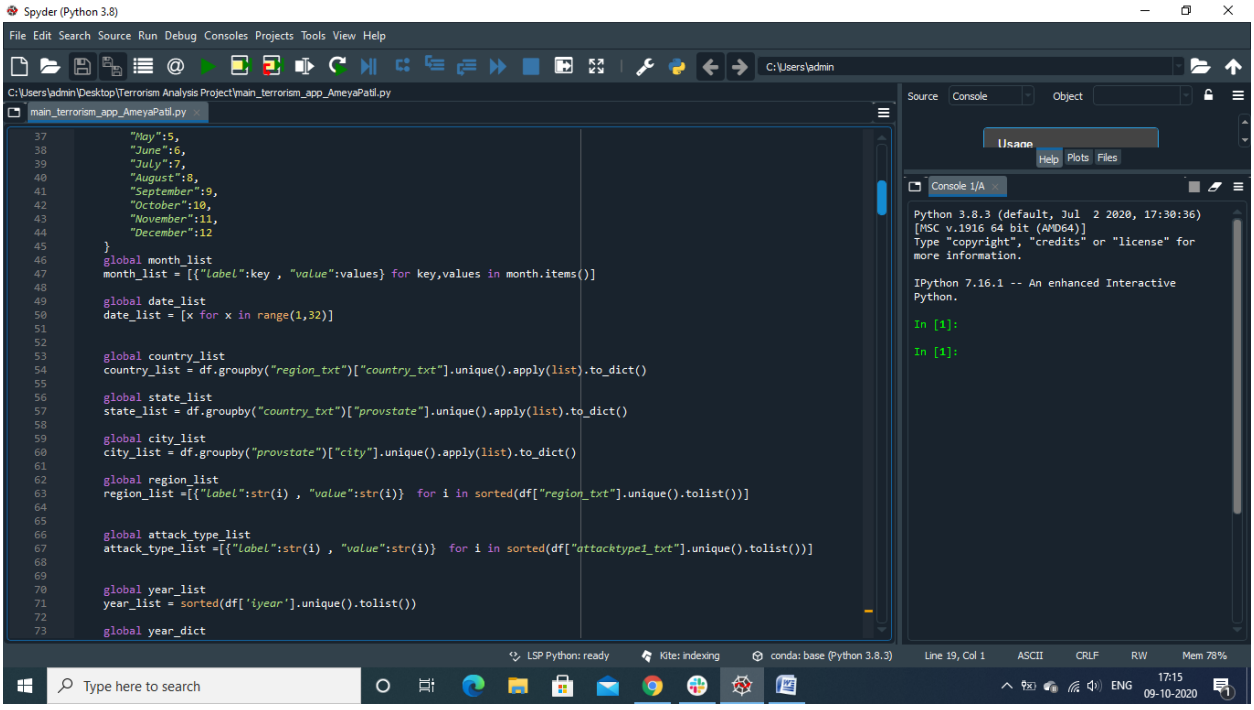
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py
"May":5,
"June":6,
"July":7,
"August":8,
"September":9,
"October":10,
"November":11,
"December":12
}
global month_list
month_list = [{"label":key , "value":values} for key,values in month.items()]
global date_list
date_list = [x for x in range(1,32)]
global country_list
country_list = df.groupby("region_txt")["country_txt"].unique().apply(list).to_dict()
global state_list
state_list = df.groupby("country_txt")["provstate"].unique().apply(list).to_dict()
global city_list
city_list = df.groupby("provstate")["city"].unique().apply(list).to_dict()
global region_list
region_list =[{"label":str(i) , "value":str(i)} for i in sorted(df["region_txt"].unique().tolist())]
global attack_type_list
attack_type_list =[{"label":str(i) , "value":str(i)} for i in sorted(df["attacktype1_txt"].unique().tolist())]
global year_list
year_list = sorted(df['iyear'].unique().tolist())
global year_dict
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: indexing
conda: base (Python 3.8.3)
Line 19, Col 1
ASCII
CRLF
RW
Mem 78%
Type here to search
ENG
17:15
09-10-2020

Fig.2

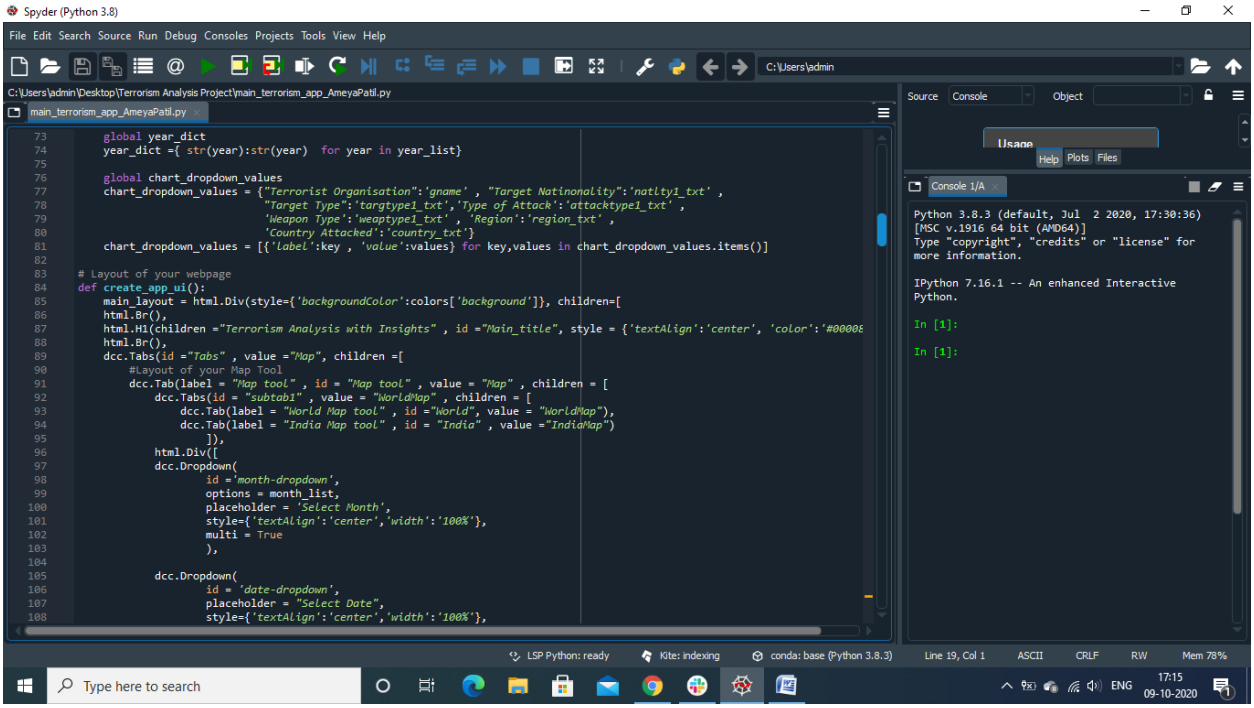
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py
global year_dict
year_dict ={ str(year):str(year) for year in year_list}
global chart_dropdown_values
chart_dropdown_values = {"Terrorist Organisation":'gname' , "Target Nationality":'natlty1_txt' ,
"Target Type":'targtype1_txt','Type of Attack':'attacktype1_txt' ,
'Weapon Type':'weaptype1_txt' , 'Region':'region_txt' ,
'Country Attacked':'country_txt'}
chart_dropdown_values = [{'label':key , 'value':values} for key,values in chart_dropdown_values.items()]
# Layout of your webpage
def create_app_ui():
main_layout = html.Div(style={'backgroundColor':colors['background']}, children=[
html.Br(),
html.H1(children ="Terrorism Analysis with Insights" , id ="Main_title", style = {'textAlign':'center', 'color':'#00008
html.Br(),
dcc.Tabs(id ="Tabs" , value ="Map", children =[
#Layout of your Map Tool
dcc.Tab(label = "Map tool" , id = "Map tool" , value = "Map" , children = [
dcc.Tabs(id = "subtab1" , value = "WorldMap" , children = [
dcc.Tab(label = "World Map tool" , id ="World", value = "WorldMap"),
dcc.Tab(label = "India Map tool" , id = "India" , value ="IndiaMap")
]),
html.Div([
dcc.Dropdown(
id ='month-dropdown',
options = month_list,
placeholder = 'Select Month',
style={'textAlign':'center','width':'100%'},
multi = True
),
dcc.Dropdown(
id = 'date-dropdown',
placeholder = "Select Date",
style={'textAlign':'center','width':'100%'},
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for
more information.
IPython 7.16.1 -- An enhanced Interactive
Python.
In [1]:
In [1]:
LSP Python: ready
Kite: indexing
conda: base (Python 3.8.3)
Line 19, Col 1
ASCII
CRLF
RW
Mem 78%
Type here to search
ENG
17:15
09-10-2020

Fig.3

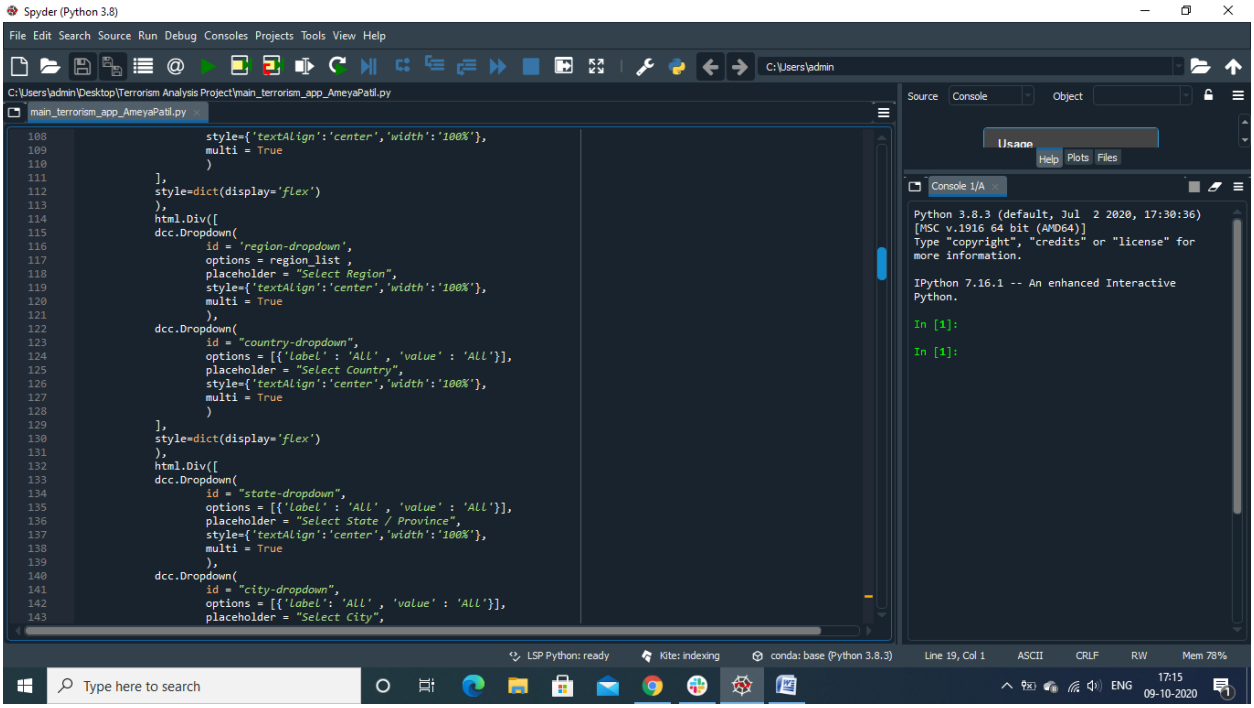
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py
style={'textAlign':'center','width':'100%'},
multi = True
)
],
style=dict(display='flex')
),
html.Div([
dcc.Dropdown(
id = 'region-dropdown',
options = region_list ,
placeholder = "Select Region",
style={'textAlign':'center','width':'100%'},
multi = True
),
dcc.Dropdown(
id = "country-dropdown",
options = [{'label' : 'All' , 'value' : 'All'}],
placeholder = "Select Country",
style={'textAlign':'center','width':'100%'},
multi = True
)
],
style=dict(display='flex')
),
html.Div([
dcc.Dropdown(
id = "state-dropdown",
options = [{'label' : 'All' , 'value' : 'All'}],
placeholder = "Select State / Province",
style={'textAlign':'center','width':'100%'},
multi = True
),
dcc.Dropdown(
id = "city-dropdown",
options = [{'label': 'All' , 'value' : 'All'}],
placeholder = "Select City",
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for
more information.
IPython 7.16.1 -- An enhanced Interactive
Python.
In [1]:
In [1]:
LSP Python: ready
Kite: indexing
conda: base (Python 3.8.3)
Line 19, Col 1
ASCII
CRLF
RW
Mem 78%
Type here to search
ENG
17:15
09-10-2020

Fig.4

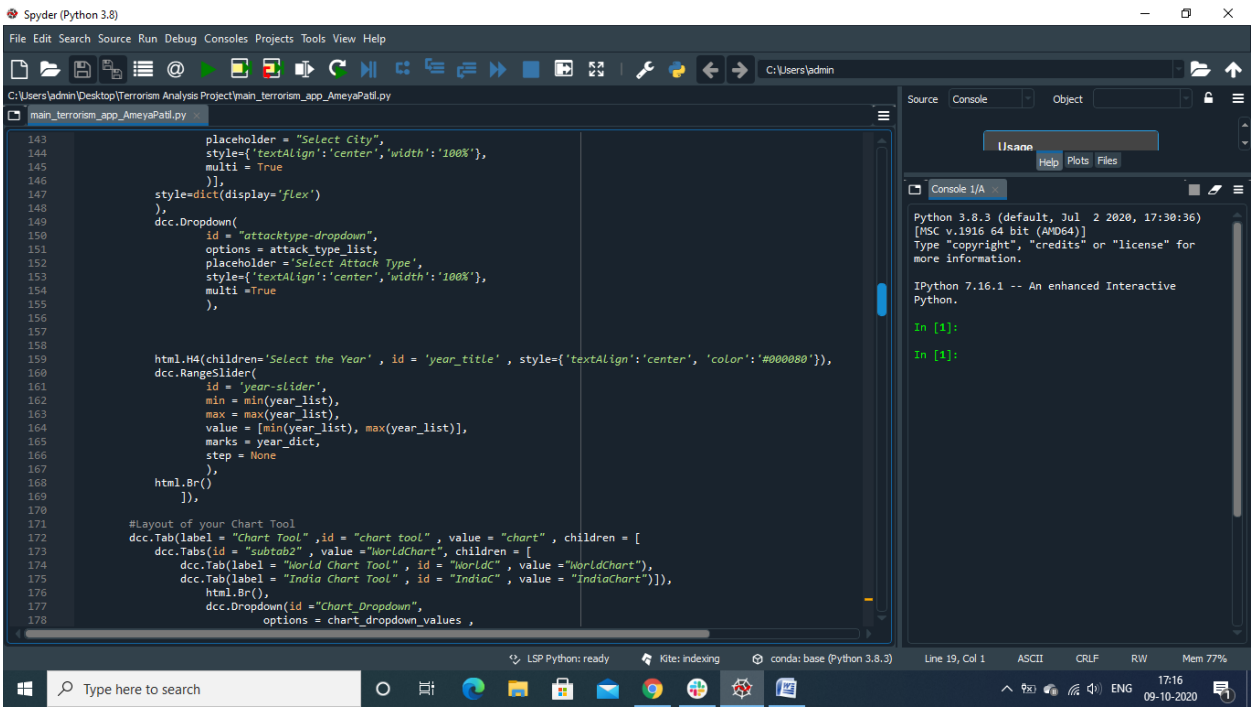
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py
placeholder = "Select City",
style={'textAlign':'center','width':'100%'},
multi = True
)],
style=dict(display='flex')
),
dcc.Dropdown(
id = "attacktype-dropdown",
options = attack_type_list,
placeholder ='Select Attack Type',
style={'textAlign':'center','width':'100%'},
multi =True
),
html.H4(children='Select the Year' , id = 'year_title' , style={'textAlign':'center', 'color':'#000080'}),
dcc.RangeSlider(
id = 'year-slider',
min = min(year_list),
max = max(year_list),
value = [min(year_list), max(year_list)],
marks = year_dict,
step = None
),
html.Br()
]),
#Layout of your Chart Tool
dcc.Tab(label = "Chart Tool" ,id = "chart tool" , value = "chart" , children = [
dcc.Tabs(id = "subtab2" , value ="WorldChart", children = [
dcc.Tab(label = "World Chart Tool" , id = "WorldC" , value ="WorldChart"),
dcc.Tab(label = "India Chart Tool" , id = "IndiaC" , value = "IndiaChart")]),
html.Br(),
dcc.Dropdown(id ="Chart_Dropdown",
options = chart_dropdown_values ,
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: indexing
conda: base (Python 3.8.3)
Line 19, Col 1
ASCII
CRLF
RW
Mem 77%
Type here to search
ENG
17:16
09-10-2020

Fig.5

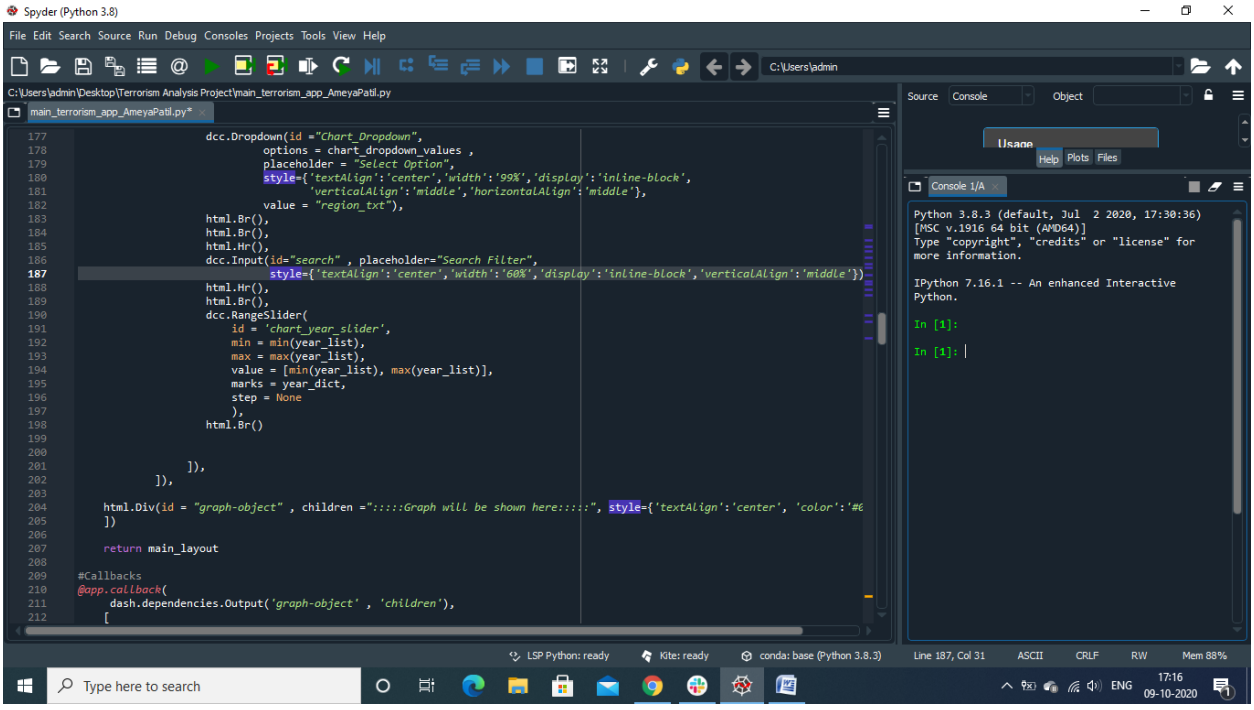
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
dcc.Dropdown(id ="Chart_Dropdown",
options = chart_dropdown_values ,
placeholder = "Select Option",
style={'textAlign':'center','width':'99%','display':'inline-block',
'verticalAlign':'middle','horizontalAlign':'middle'},
value = "region_txt"),
html.Br(),
html.Br(),
html.Hr(),
dcc.Input(id="search" , placeholder="Search Filter",
style={'textAlign':'center','width':'60%','display':'inline-block','verticalAlign':'middle'}),
html.Hr(),
html.Br(),
dcc.RangeSlider(
id = 'chart_year_slider',
min = min(year_list),
max = max(year_list),
value = [min(year_list), max(year_list)],
marks = year_dict,
step = None
),
html.Br()
]),
]),
html.Div(id = "graph-object" , children =":::::Graph will be shown here:::::", style={'textAlign':'center', 'color':'#0
])
return main_layout
#Callbacks
@app.callback(
dash.dependencies.Output('graph-object' , 'children'),
[
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 88%
Type here to search
ENG
17:16
09-10-2020

Fig.6

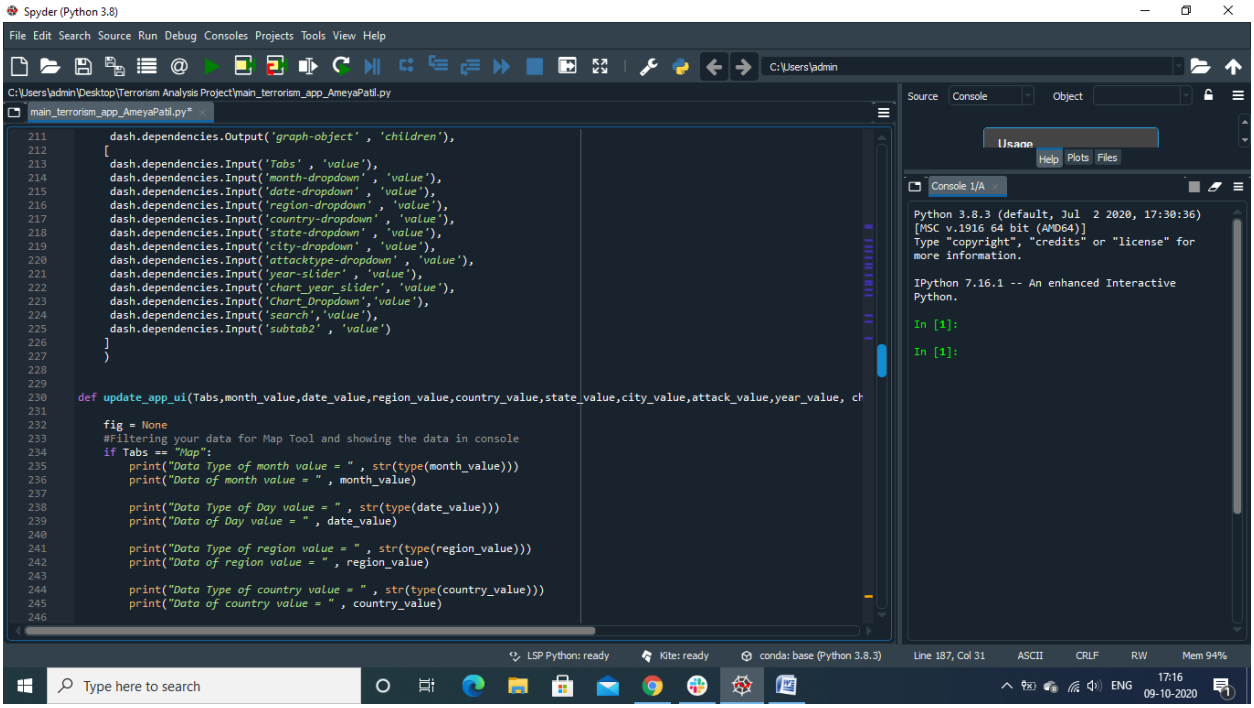
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
dash.dependencies.Output('graph-object' , 'children'),
[
dash.dependencies.Input('Tabs' , 'value'),
dash.dependencies.Input('month-dropdown' , 'value'),
dash.dependencies.Input('date-dropdown' , 'value'),
dash.dependencies.Input('region-dropdown' , 'value'),
dash.dependencies.Input('country-dropdown' , 'value'),
dash.dependencies.Input('state-dropdown' , 'value'),
dash.dependencies.Input('city-dropdown' , 'value'),
dash.dependencies.Input('attacktype-dropdown' , 'value'),
dash.dependencies.Input('year-slider' , 'value'),
dash.dependencies.Input('chart_year_slider', 'value'),
dash.dependencies.Input('Chart_Dropdown','value'),
dash.dependencies.Input('search','value'),
dash.dependencies.Input('subtab2' , 'value')
]
)
def update_app_ui(Tabs,month_value,date_value,region_value,country_value,state_value,city_value,attack_value,year_value, ch
fig = None
#Filtering your data for Map Tool and showing the data in console
if Tabs == "Map":
print("Data Type of month value = " , str(type(month_value)))
print("Data of month value = " , month_value)
print("Data Type of Day value = " , str(type(date_value)))
print("Data of Day value = " , date_value)
print("Data Type of region value = " , str(type(region_value)))
print("Data of region value = " , region_value)
print("Data Type of country value = " , str(type(country_value)))
print("Data of country value = " , country_value)
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 94%
Type here to search
ENG
17:16
09-10-2020

Fig.7

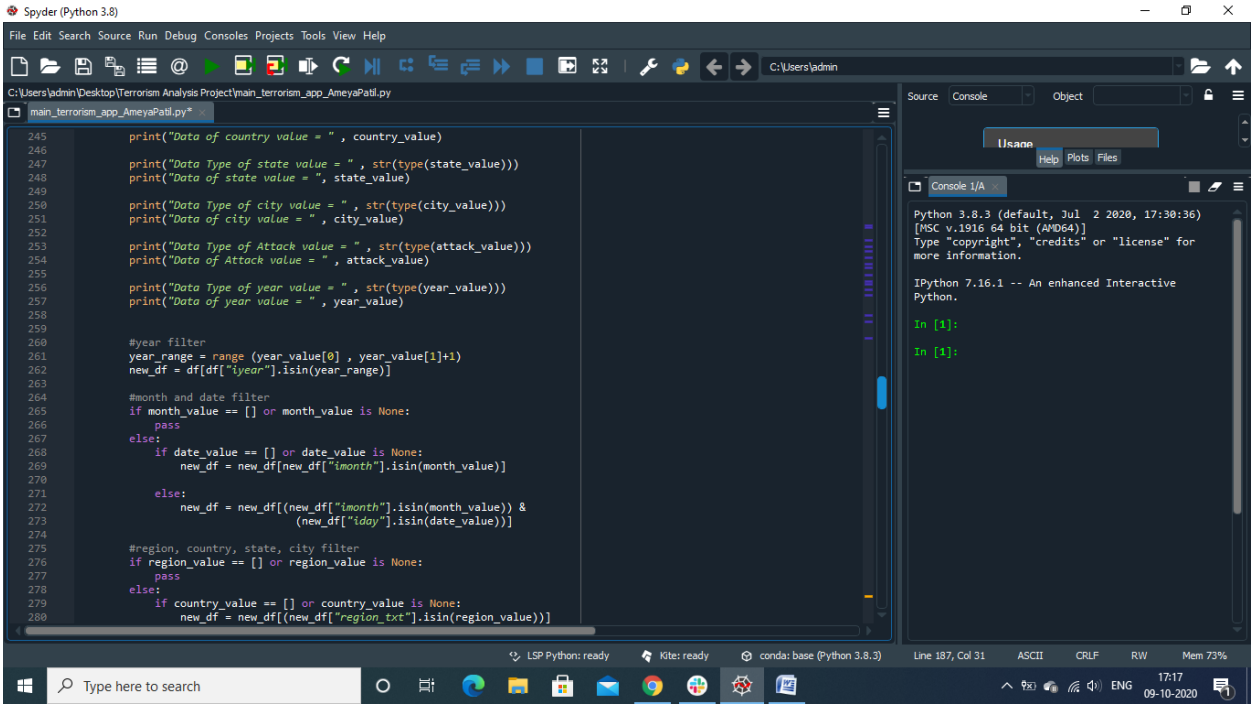
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
print("Data of country value = " , country_value)
print("Data Type of state value = " , str(type(state_value)))
print("Data of state value = ", state_value)
print("Data Type of city value = " , str(type(city_value)))
print("Data of city value = " , city_value)
print("Data Type of Attack value = " , str(type(attack_value)))
print("Data of Attack value = " , attack_value)
print("Data Type of year value = " , str(type(year_value)))
print("Data of year value = " , year_value)
#year filter
year_range = range (year_value[0] , year_value[1]+1)
new_df = df[df["iyear"].isin(year_range)]
#month and date filter
if month_value == [] or month_value is None:
pass
else:
if date_value == [] or date_value is None:
new_df = new_df[new_df["imonth"].isin(month_value)]
else:
new_df = new_df[(new_df["imonth"].isin(month_value)) &
(new_df["iday"].isin(date_value))]
#region, country, state, city filter
if region_value == [] or region_value is None:
pass
else:
if country_value == [] or country_value is None:
new_df = new_df[(new_df["region_txt"].isin(region_value))]
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 73%
Type here to search
ENG
17:17
09-10-2020

Fig.8

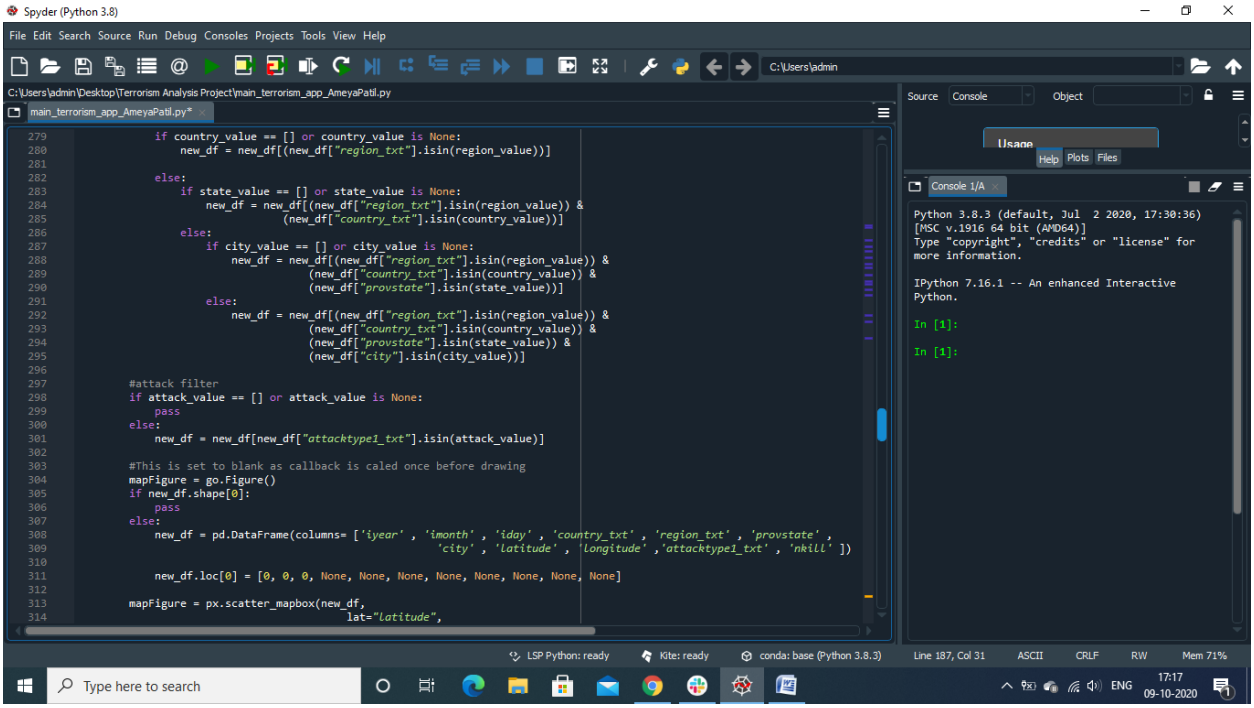
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
if country_value == [] or country_value is None:
new_df = new_df[(new_df["region_txt"].isin(region_value))]
else:
if state_value == [] or state_value is None:
new_df = new_df[(new_df["region_txt"].isin(region_value)) &
(new_df["country_txt"].isin(country_value))]
else:
if city_value == [] or city_value is None:
new_df = new_df[(new_df["region_txt"].isin(region_value)) &
(new_df["country_txt"].isin(country_value)) &
(new_df["provstate"].isin(state_value))]
else:
new_df = new_df[(new_df["region_txt"].isin(region_value)) &
(new_df["country_txt"].isin(country_value)) &
(new_df["provstate"].isin(state_value)) &
(new_df["city"].isin(city_value))]
#attack filter
if attack_value == [] or attack_value is None:
pass
else:
new_df = new_df[new_df["attacktype1_txt"].isin(attack_value)]
#This is set to blank as callback is caled once before drawing
mapFigure = go.Figure()
if new_df.shape[0]:
pass
else:
new_df = pd.DataFrame(columns= ['iyear' , 'imonth' , 'iday' , 'country_txt' , 'region_txt' , 'provstate' ,
'city' , 'latitude' , 'longitude' ,'attacktype1_txt' , 'nkill'])
new_df.loc[0] = [0, 0, 0, None, None, None, None, None, None, None, None]
mapFigure = px.scatter_mapbox(new_df,
lat="latitude",
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 71%
Type here to search
ENG
17:17
09-10-2020

Fig.9

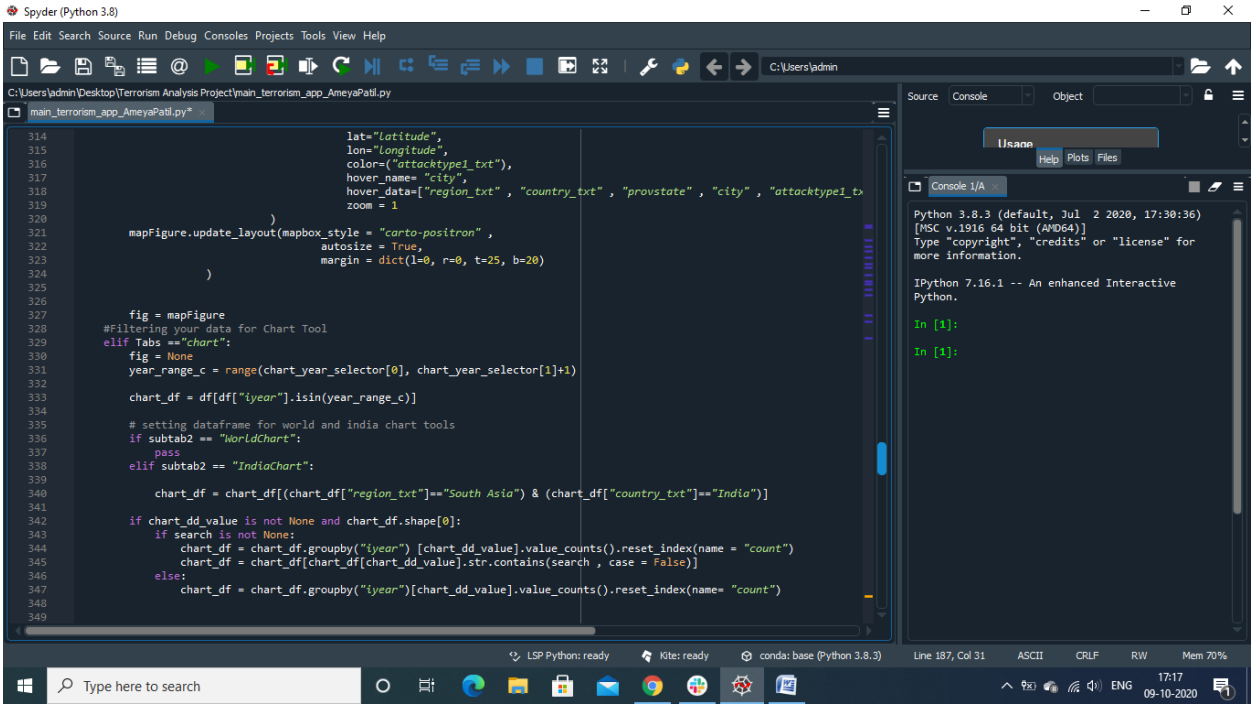
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
lat="latitude",
lon="longitude",
color=("attacktype1_txt"),
hover_name= "city",
hover_data=["region_txt" , "country_txt" , "provstate" , "city" , "attacktype1_tx
zoom = 1
)
mapFigure.update_layout(mapbox_style = "carto-positron" ,
autosize = True,
margin = dict(l=0, r=0, t=25, b=20)
)
fig = mapFigure
#Filtering your data for Chart Tool
elif Tabs =="chart":
fig = None
year_range_c = range(chart_year_selector[0], chart_year_selector[1]+1)
chart_df = df[df["iyear"].isin(year_range_c)]
# setting dataframe for world and india chart tools
if subtab2 == "WorldChart":
pass
elif subtab2 == "IndiaChart":
chart_df = chart_df[(chart_df["region_txt"]=="South Asia") & (chart_df["country_txt"]=="India")]
if chart_dd_value is not None and chart_df.shape[0]:
if search is not None:
chart_df = chart_df.groupby("iyear") [chart_dd_value].value_counts().reset_index(name = "count")
chart_df = chart_df[chart_df[chart_dd_value].str.contains(search , case = False)]
else:
chart_df = chart_df.groupby("iyear")[chart_dd_value].value_counts().reset_index(name= "count")
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 70%
Type here to search
ENG
17:17
09-10-2020

Fig.10

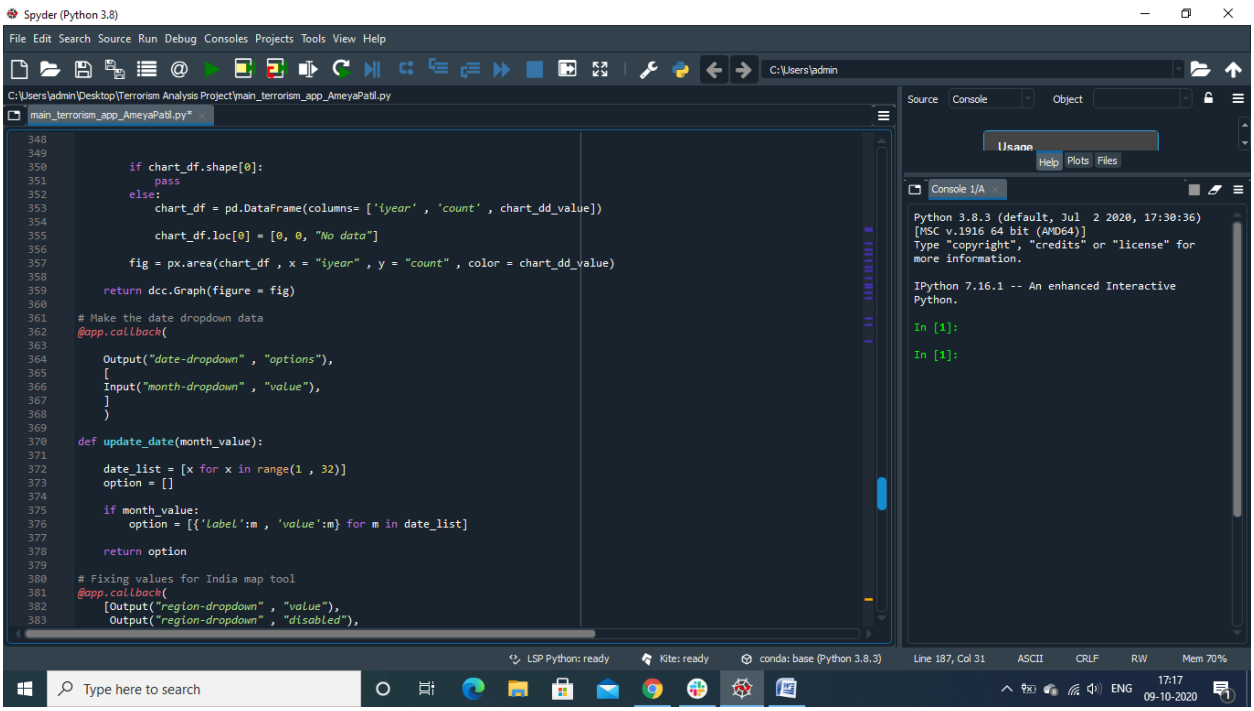
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
if chart_df.shape[0]:
pass
else:
chart_df = pd.DataFrame(columns= ['iyear' , 'count' , chart_dd_value])
chart_df.loc[0] = [0, 0, "No data"]
fig = px.area(chart_df , x = "iyear" , y = "count" , color = chart_dd_value)
return dcc.Graph(figure = fig)
Make the date dropdown data
@app.callback(
Output("date-dropdown" , "options"),
[
Input("month-dropdown" , "value"),
]
)
def update_date(month_value):
date_list = [x for x in range(1 , 32)]
option = []
if month_value:
option = [{'label':m , 'value':m} for m in date_list]
return option
Fixing values for India map tool
@app.callback(
[Output("region-dropdown" , "value"),
Output("region-dropdown" , "disabled"),
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 70%
Type here to search
ENG
17:17
09-10-2020

Fig.11

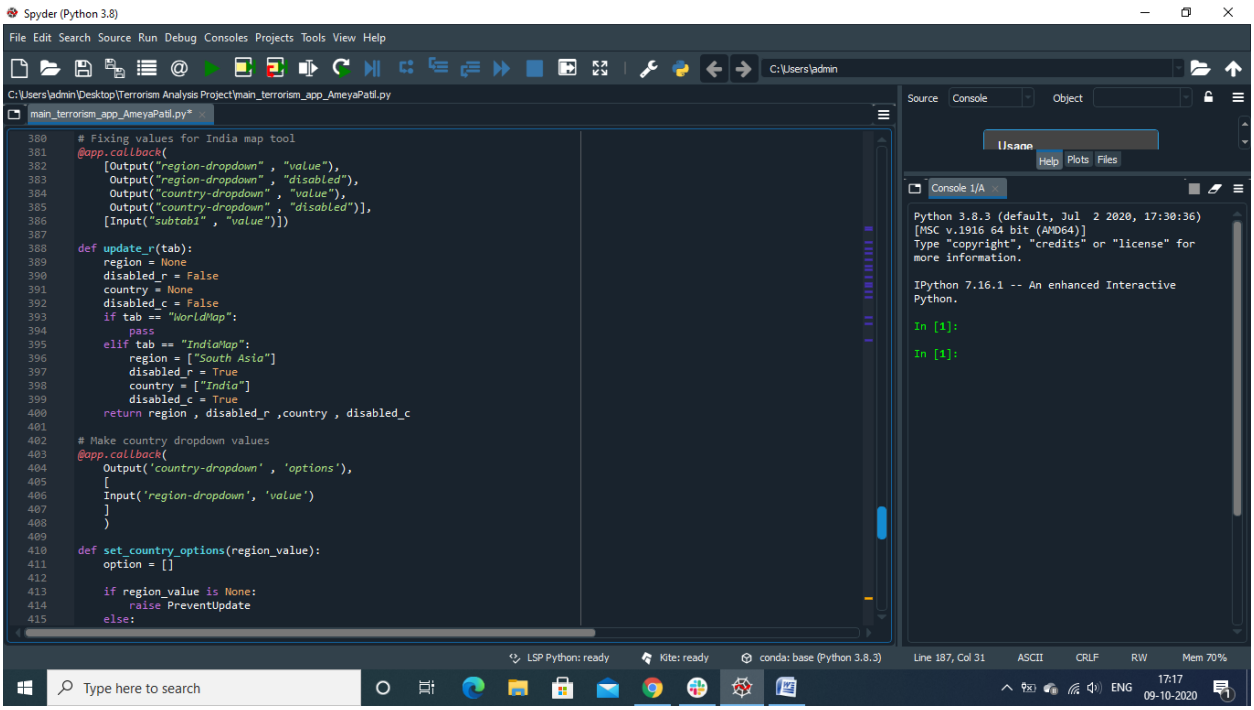
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
Fixing values for India map tool
@app.callback(
[Output("region-dropdown" , "value"),
Output("region-dropdown" , "disabled"),
Output("country-dropdown" , "value"),
Output("country-dropdown" , "disabled")],
[Input("subtab1" , "value")])
def update_r(tab):
region = None
disabled_r = False
country = None
disabled_c = False
if tab == "WorldMap":
pass
elif tab == "IndiaMap":
region = ["South Asia"]
disabled_r = True
country = ["India"]
disabled_c = True
return region , disabled_r ,country , disabled_c
Make country dropdown values
@app.callback(
Output('country-dropdown' , 'options'),
[
Input('region-dropdown', 'value')
]
)
def set_country_options(region_value):
option = []
if region_value is None:
raise PreventUpdate
else:
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 70%
Type here to search
ENG
17:17
09-10-2020

Fig.12

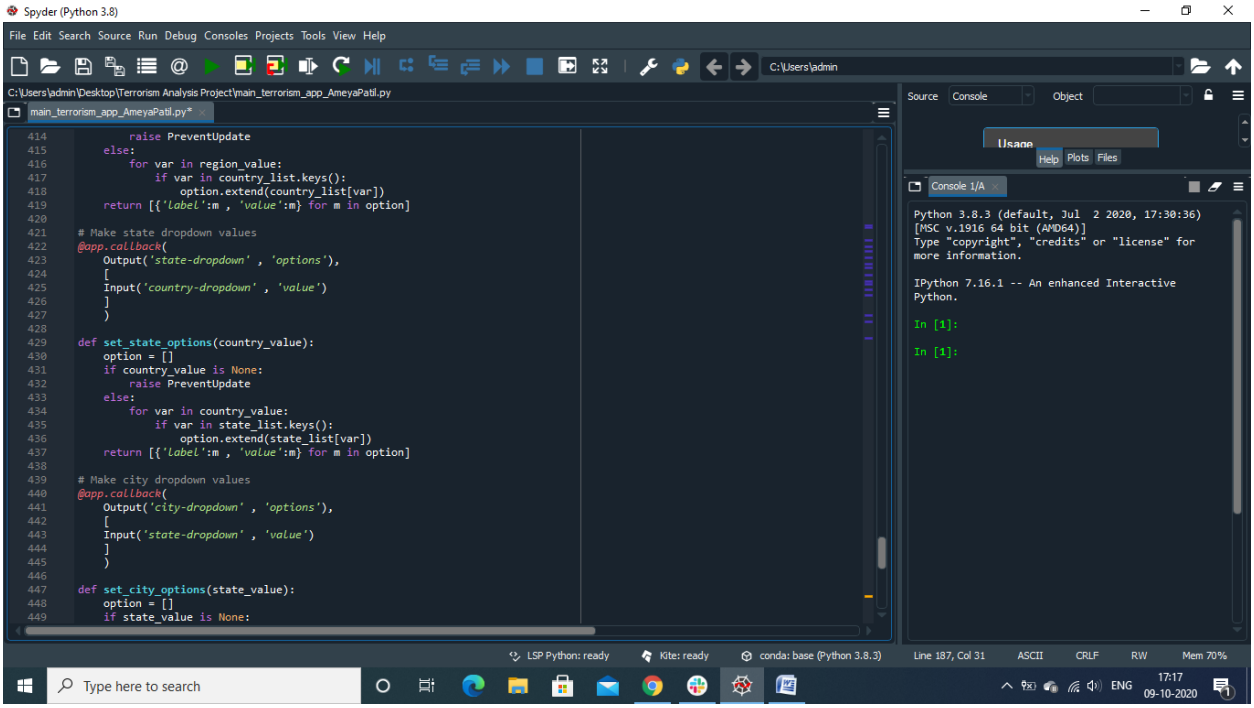
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py*
raise PreventUpdate
else:
for var in region_value:
if var in country_list.keys():
option.extend(country_list[var])
return [{'label':m , 'value':m} for m in option]
Make state dropdown values
@app.callback(
Output('state-dropdown' , 'options'),
[
Input('country-dropdown' , 'value')
]
)
def set_state_options(country_value):
option = []
if country_value is None:
raise PreventUpdate
else:
for var in country_value:
if var in state_list.keys():
option.extend(state_list[var])
return [{'label':m , 'value':m} for m in option]
Make city dropdown values
@app.callback(
Output('city-dropdown' , 'options'),
[
Input('state-dropdown' , 'value')
]
)
def set_city_options(state_value):
option = []
if state_value is None:
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 70%
Type here to search
ENG
17:17
09-10-2020

Fig.13

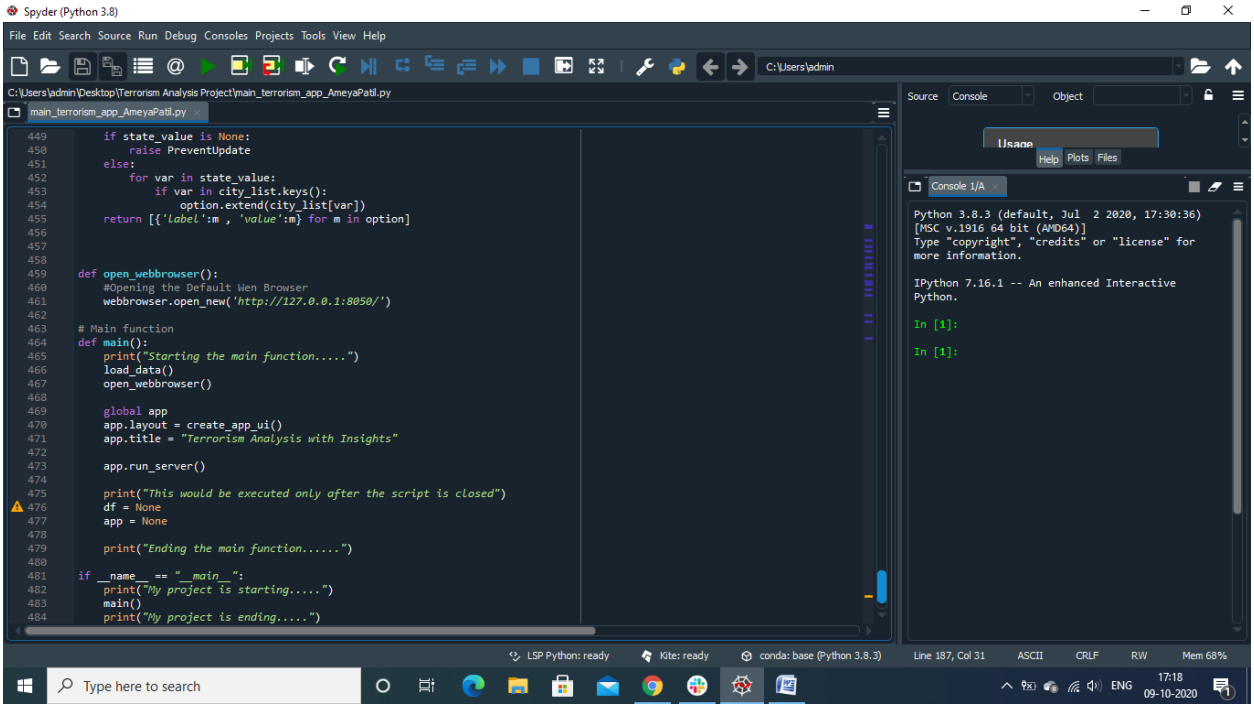
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\admin
C:\Users\admin\Desktop\Terrorism Analysis Project\main_terrorism_app_AmeyaPatil.py
main_terrorism_app_AmeyaPatil.py
if state_value is None:
raise PreventUpdate
else:
for var in state_value:
if var in city_list.keys():
option.extend(city_list[var])
return [{'label':m , 'value':m} for m in option]
def open_webbrowser():
#Opening the Default Wen Browser
webbrowser.open_new('http://127.0.0.1:8050/')
# Main function
def main():
print("Starting the main function.....")
load_data()
open_webbrowser()
global app
app.layout = create_app_ui()
app.title = "Terrorism Analysis with Insights"
app.run_server()
print("This would be executed only after the script is closed")
df = None
app = None
print("Ending the main function.....")
if __name__ == "__main__":
print("My project is starting.....")
main()
print("My project is ending.....")
Source Console Object
Usage
Help Plots Files
Console 1/A
Python 3.8.3 (default, Jul 2 2020, 17:30:36)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
In [1]:
LSP Python: ready
Kite: ready
conda: base (Python 3.8.3)
Line 187, Col 31
ASCII
CRLF
RW
Mem 68%
Type here to search
ENG
17:18
09-10-2020

Fig.14

5. **Results**:

After implementing the above code the output screenshot is displayed below:

THESE 4 ARE DEFAULT WITHOUT ANY INPUTS

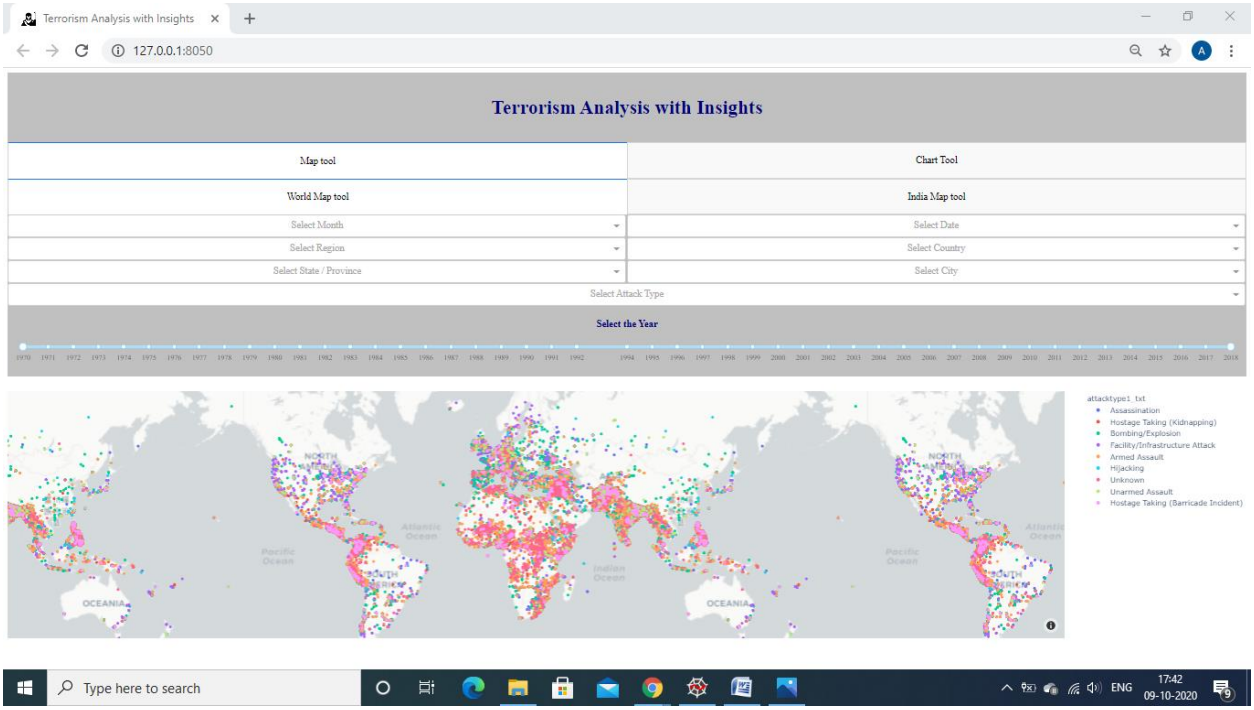

Fig.15

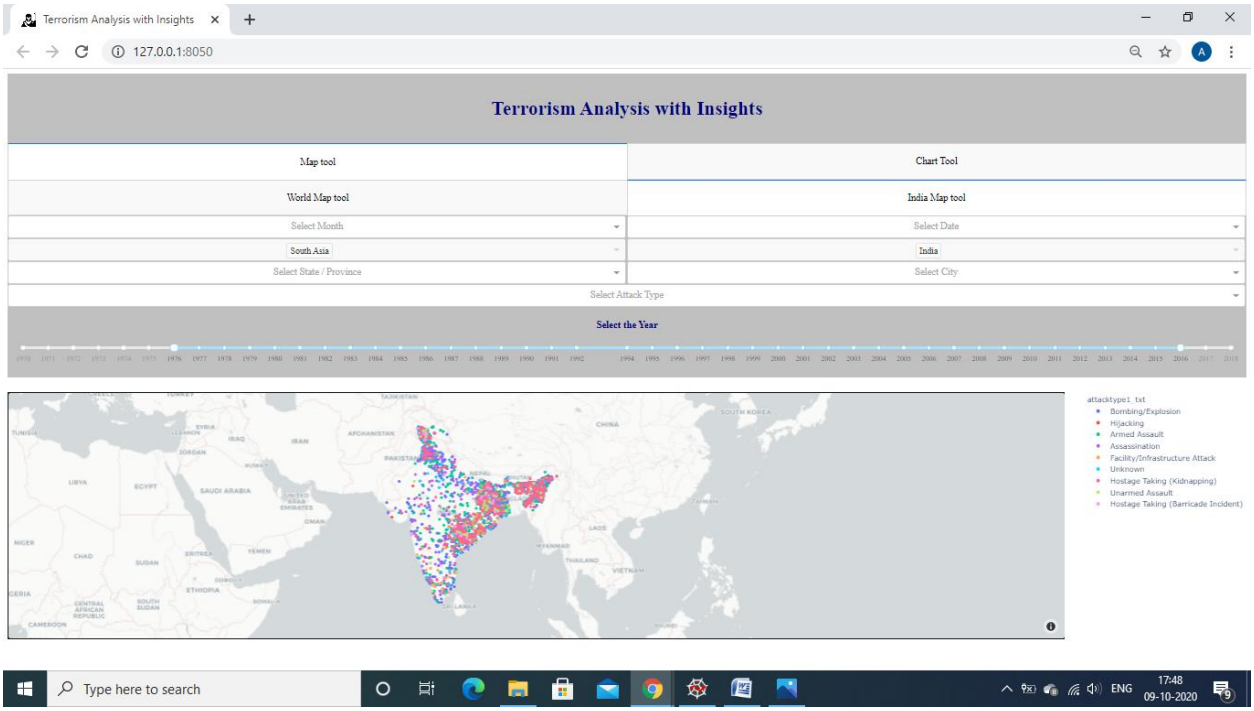
Terrorism Analysis with Insights
127.0.0.1:8050
Terrorism Analysis with Insights
Map tool
Chart Tool
World Map tool
India Map tool
Select Month
Select Date
South Asia
India
Select State / Province
Select City
Select Attack Type
Select the Year
attacktype1_txt
Bombing/Explosion
Hijacking
Armed Assault
Assassination
Facility/Infrastructure Attack
Unknown
Hostage Taking (Kidnapping)
Unarmed Assault
Hostage Taking (Barricade Incident)
Type here to search
ENG
17:48
09-10-2020

Fig.16

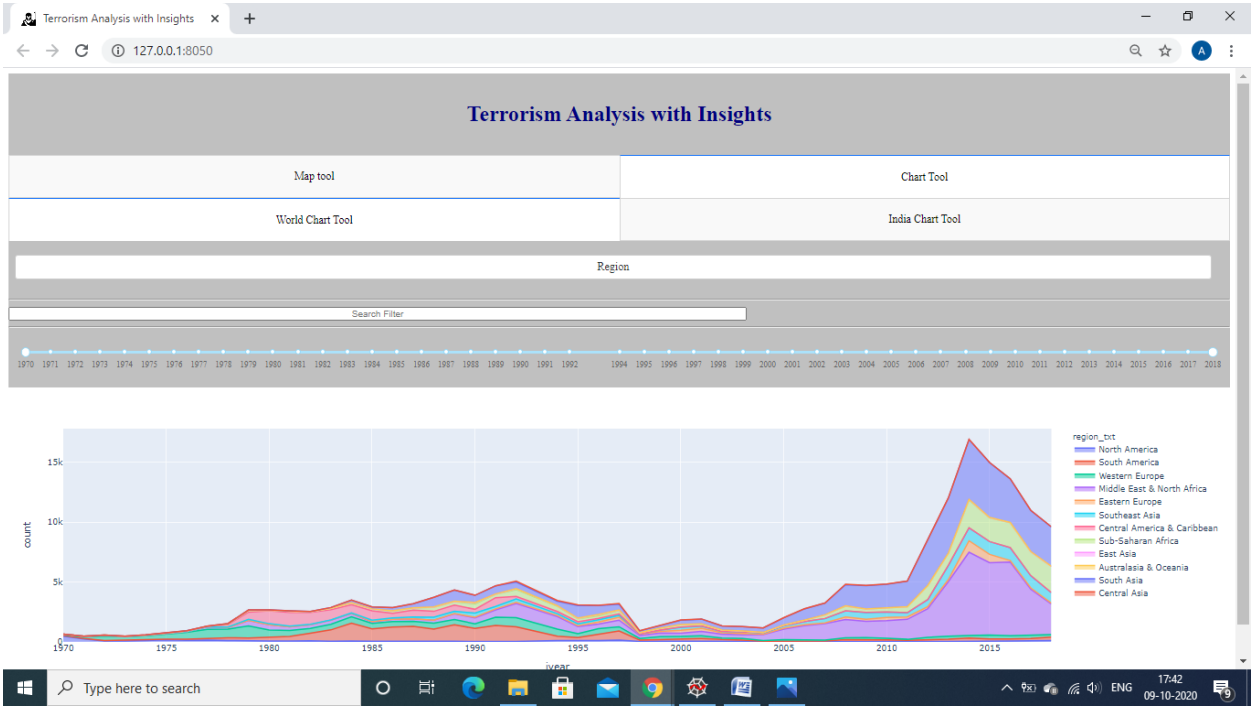
Terrorism Analysis with Insights
127.0.0.1:8050
Terrorism Analysis with Insights
Map tool
Chart Tool
World Chart Tool
India Chart Tool
Region
Search Filter
count
year
region_txt
North America
South America
Western Europe
Middle East & North Africa
Eastern Europe
Southeast Asia
Central America & Caribbean
Sub-Saharan Africa
East Asia
Australasia & Oceania
South Asia
Central Asia
Type here to search
ENG
17:42
09-10-2020

Fig.17

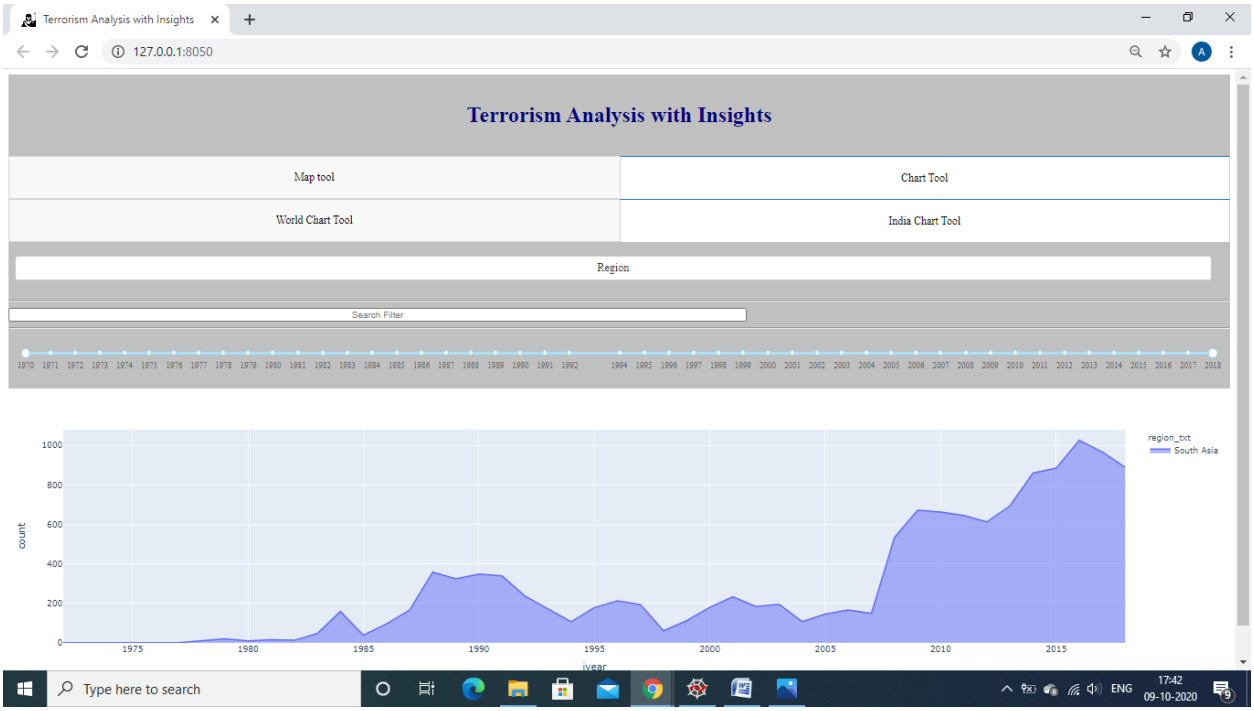

Fig.18

THESE NEXT ARE WITH SOME INPUTS IN THEM

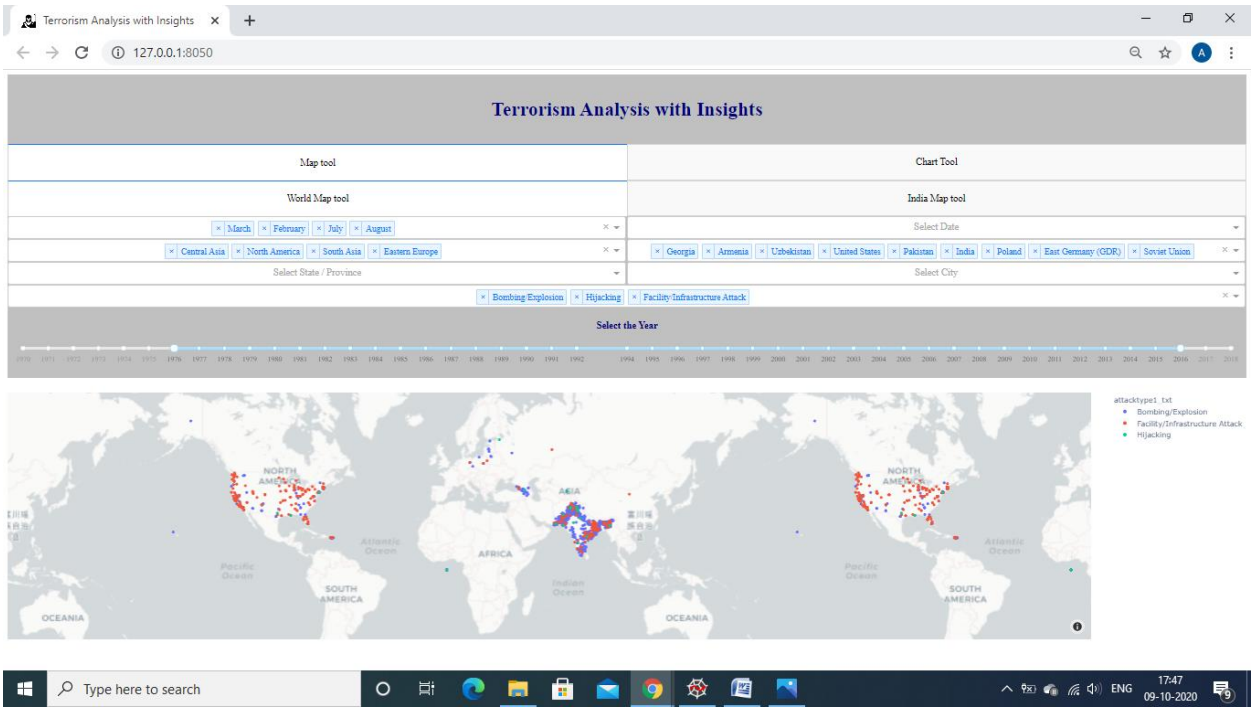

Fig.19

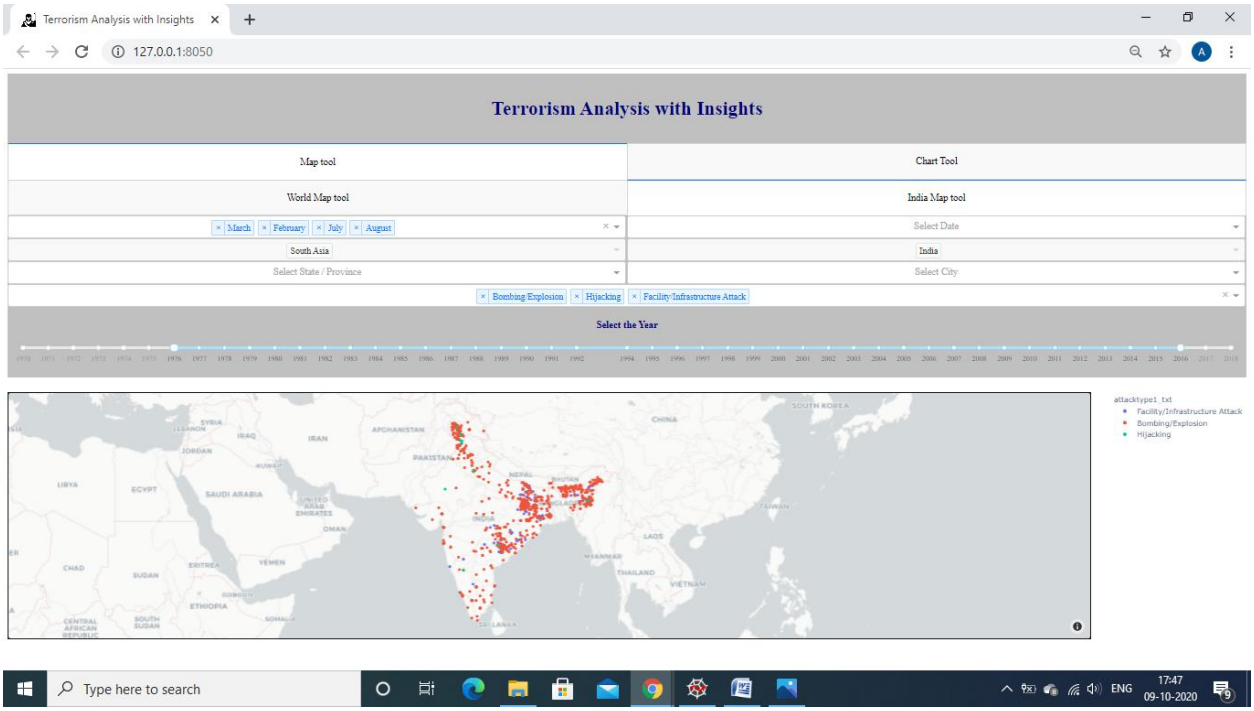
Terrorism Analysis with Insights
127.0.0.1:8050
Terrorism Analysis with Insights
Map tool
Chart Tool
World Map tool
India Map tool
March
February
July
August
Select Date
South Asia
India
Select State / Province
Select City
Bombing/Explosion
Hijacking
Facility/Infrastructure Attack
Select the Year
attacktype1_txt
Facility/Infrastructure Attack
Bombing/Explosion
Hijacking
Type here to search
ENG
17:47
09-10-2020

Fig.20

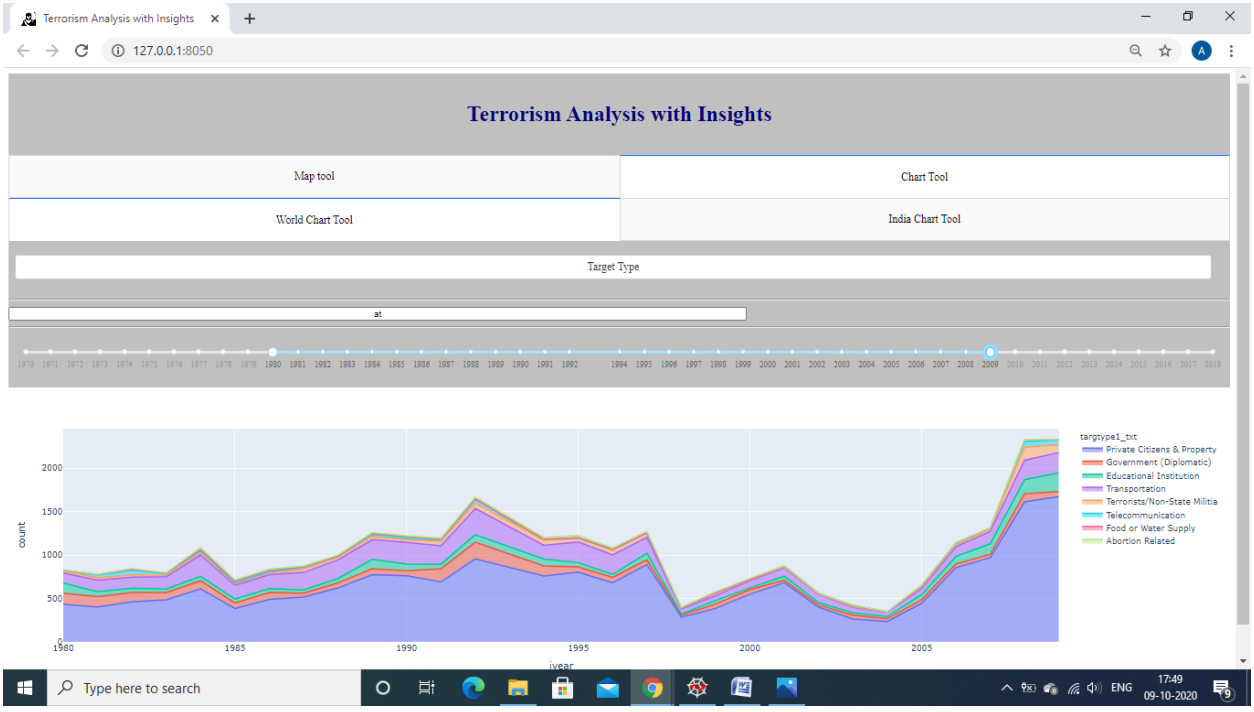
Terrorism Analysis with Insights
127.0.0.1:8050
Terrorism Analysis with Insights
Map tool
Chart Tool
World Chart Tool
India Chart Tool
Target Type
at
count
iyear
targtype1_txt
Private Citizens & Property
Government (Diplomatic)
Educational Institution
Transportation
Terrorists/Non-State Militia
Telecommunication
Food or Water Supply
Abortion Related
Type here to search
ENG
17:49
09-10-2020

Fig.21

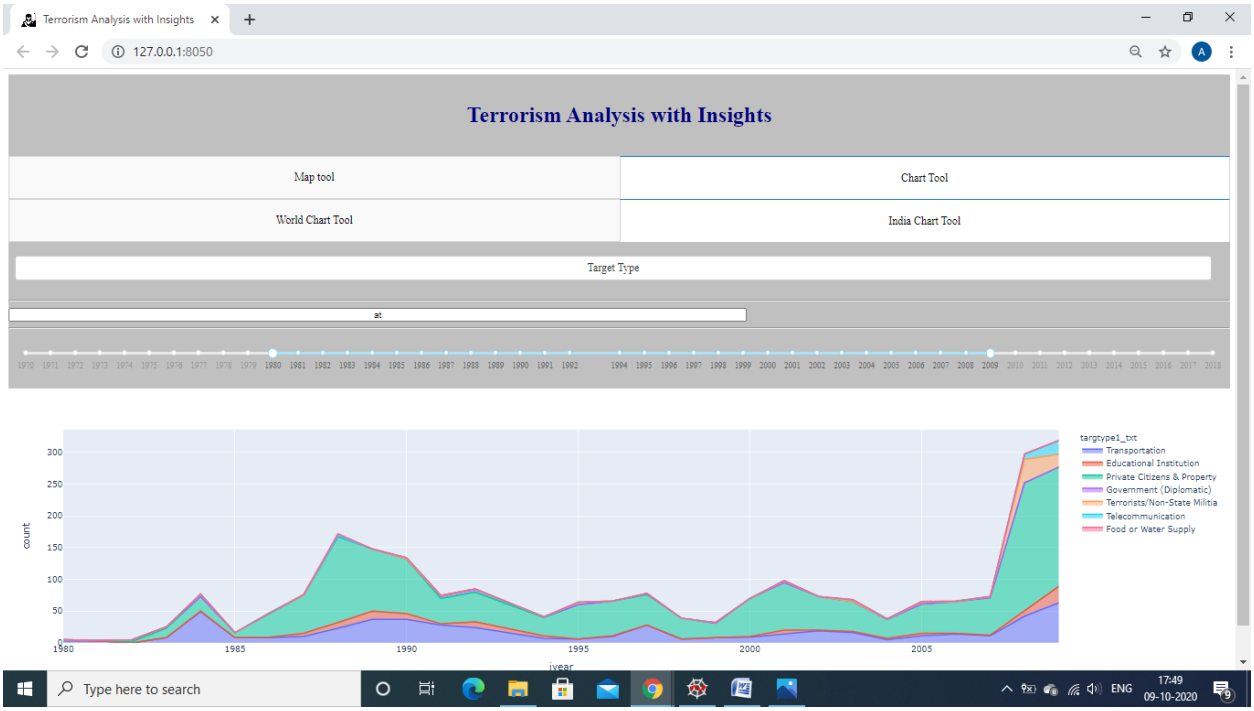

Fig.22

6. **Software Improvements**:

There are lots of improvements in this project. As the we can select any option from the dropdowns like Region, Country, State, and City. All these 4 dropdowns are interlinked to each other (i.e. Region filters the Country and so on), so if anyone is not selected the remaining will not show up any values. Similarly the Month and Date dropdowns are interlinked to each other, so if the month is not selected then the dates will not show up.

7. **Conclusion and Future Works**:

The conclusion of the project “Terrorism Analysis with Insights” is that it meets all our requirements and is ready to be used.

Future work includes using JAVASCRIPT and CSS for improving the user interface for making the web page look better.

9. **References**

- https://www.youtube.com/watch?v=Ma8tS4p27JI&list=PLH6mU1kedUy8fCzkTTJlwsf2EnV_UvOV-&index=1
- https://dash.plotly.com
- https://dash.plotly.com/layout
- Stack overflow
- https://community.plotly.com/

10. **Biography**:

Myself Barkha Verma pursuing BTECH in stream of Computer Science and engineering (Data Science) third year at Poornima University, Jaipur. My skills C, Python, MYSQL, R programming, Machine learning, deep learning.

My Kaggle profile: https://www.kaggle.com/barkhaverma

My Github profile: https://github.com/barkha000