LOVELY PROFESSIONAL UNIVERSITY

JALANDHAR-DELHI G.T.ROAD PHAGWARA PUNJAB (144411).

NAME: NITISH KUSHWAHA

Reg.no.12004972

Project: Analysis & Create Dashboards Using R Programming.

Section: KM045

Submitted by

NITISH KUSHWAHA

Reg.No: 12004972

Programme and Section: B.Tech (CSE Hons.),KM082

Course Code: INT232

Under the Guidance of

DR. MIR SALIM UL ISLAM (28897)

Discipline of CSE/IT

&Lovely School of Computer Science & Engineering

Lovely Professional University, Phagwara

CERTIFICATE

This is to certify that “NITISH KUSHWAHA” bearing Registration no.12004972 has completed “INT 232” project titled, “Covid -19 Case Analysis & Creating Dashboard Using R programming” under my guidance and supervision. To the best of my knowledge, the present work is the result of his/her original development, effort and study.

DR. MIR SALIM UL ISLAM (28897)

Designation of the Supervisor

School of Computer Science & Engineering

Lovely Professional University

Phagwara, Punjab.

DECLARATION

I, Nitish Kushwaha, student of B-Tech CSE under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

Date: 20/04/2023

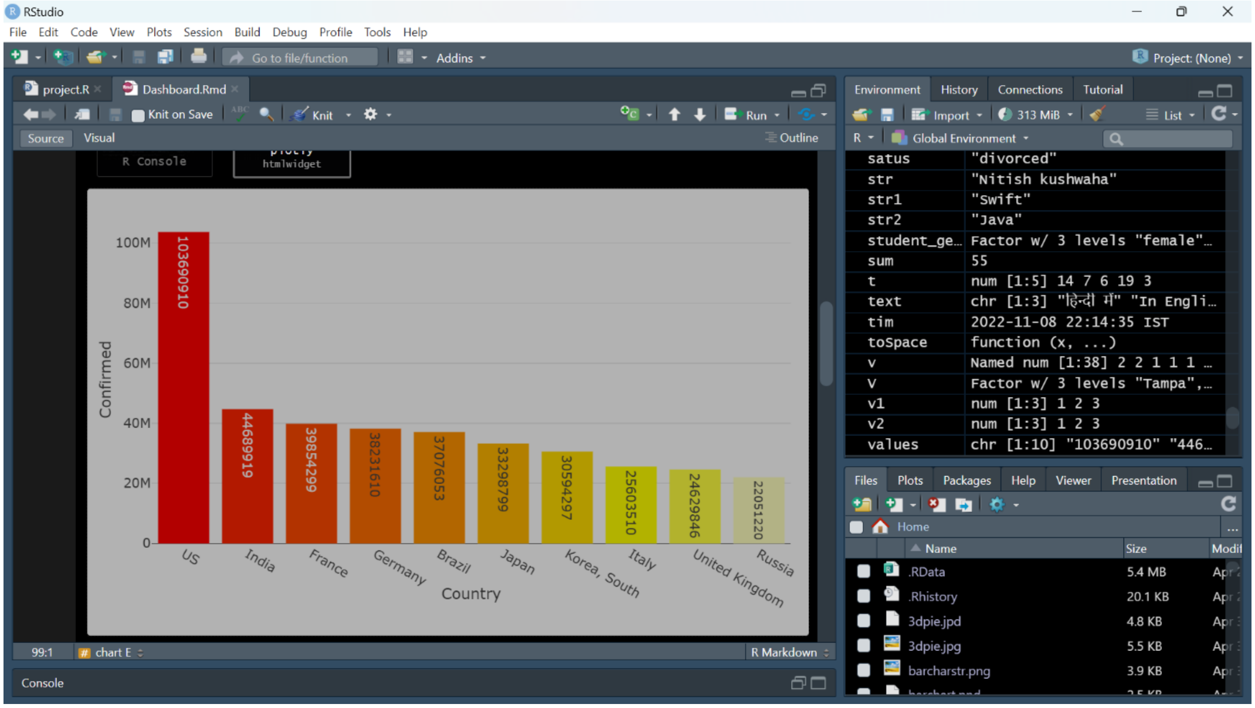
Nitish Kushwaha

Reg.No.:1200497

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my teacher “**Dr. Mir Salim Ul Islam**” Sir who gave me the golden opportunity to do this wonderful project of analysis of the data to “Create Dashboards Using R programming” which also helped me in doing a lot of research and I came to know about so many new things. I am thankful to them.

Secondly, I would also like to thank friends who helped me a lot in finalizing this project within the limited time frame.



```
satus          "divorced"
str            "Nitish kushwaha"
str1           "Swift"
str2           "Java"
student_ge…    Factor w/ 3 levels "female"…
sum            55
t              num [1:5] 14 7 6 19 3
text           chr [1:3] "हिन्दी में" "In Engli…
tim            2022-11-08 22:14:35 IST
toSpace        function (x, ...)
v              Named num [1:38] 2 2 1 1 1 …
V              Factor w/ 3 levels "Tampa",…
v1             num [1:3] 1 2 3
v2             num [1:3] 1 2 3
values         chr [1:10] "103690910" "446…
```

Name	Size	Modif
.RData	5.4 MB	Apr
.Rhistory	20.1 KB	Apr
3dpie.jpd	4.8 KB	Apr
3dpie.jpg	5.5 KB	Apr
barcharstr.png	3.9 KB	Apr

The first line of the code creates a new data frame called "latest" by filtering the "covid" data frame to only include rows where the "Date" column is equal to the maximum date in the data set. The rows are then arranged in descending order by the "Confirmed" column.

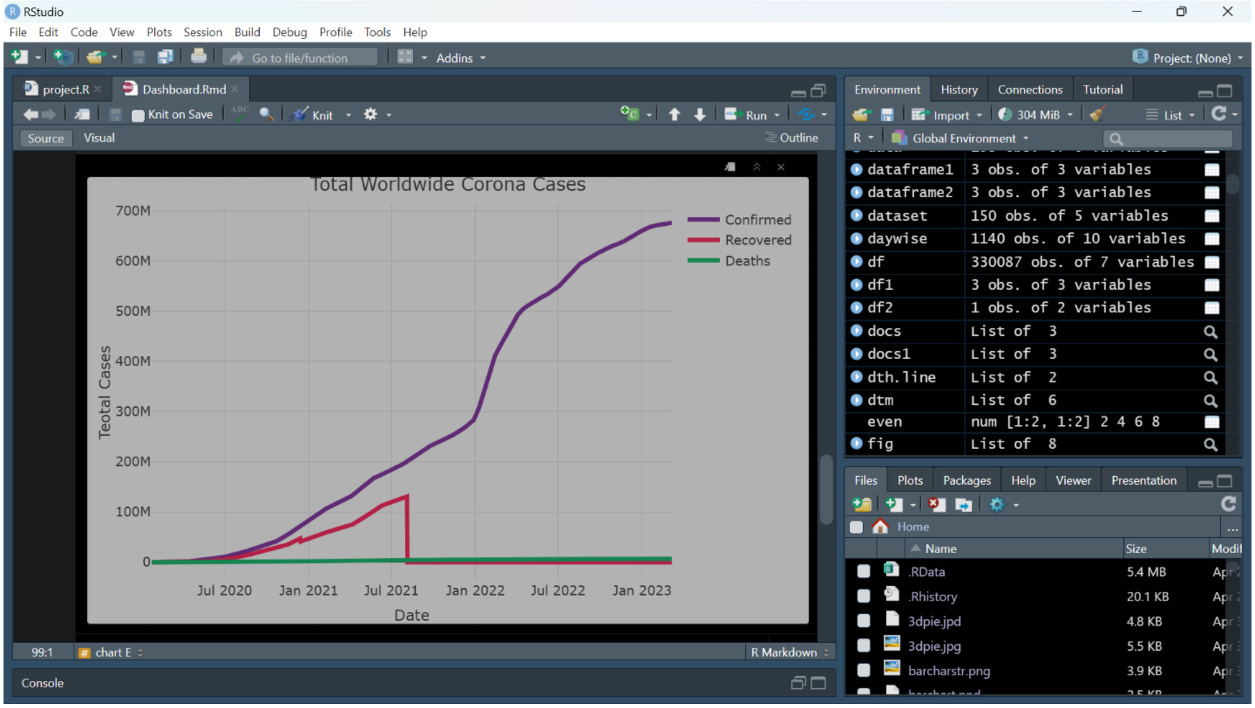
The second line of the code selects the top 10 rows from the "latest" data frame and assigns them to a new data frame called "top10". The "Country" column in "top10" is then converted to a factor with levels ordered by the same order as the countries appear in "top10".

The third line of the code creates a vector of the "Confirmed" column in "top10" and converts it to a character vector.

The fourth line of the code uses the plotly package to create an interactive bar plot. The "top10" data frame is used as the data source for the plot, and the "Country" and "Confirmed" columns are mapped to the x and y axes, respectively. The "type" argument specifies that a bar plot should be created, and the "names" argument specifies the title for the y-axis.

The "text" argument specifies that the values of the "Confirmed" column should be displayed as labels on the bars, and the "textposition" argument specifies that the label positions should be determined automatically by the plotly package.

Finally, the "marker" argument is used to customize the appearance of the bars. The "color" argument specifies a gradient of colors to be used for the bars, and the "line" argument specifies that the outline of the bars should be colored magenta and have a width of 0.



```
dataframe1   3 obs. of 3 variables
dataframe2   3 obs. of 3 variables
dataset      150 obs. of 5 variables
daywise      1140 obs. of 10 variables
df           330087 obs. of 7 variables
df1          3 obs. of 3 variables
df2          1 obs. of 2 variables
docs         List of  3
docs1        List of  3
dth.line     List of  2
dtm          List of  6
even         num [1:2, 1:2] 2 4 6 8
fig          List of  8
```

```
.RData          5.4 MB    Apr
.Rhistory       20.1 KB   Apr
3dpie.jpd       4.8 KB    Apr
3dpie.jpg       5.5 KB    Apr
barcharstr.png  3.9 KB    Apr
```

This code uses the R programming language to create a line plot of the total worldwide COVID-19 cases over time.

The first line of the code creates a plotly object called "fig" with the "daywise" data frame as the data source and the "Date" column mapped to the x-axis.

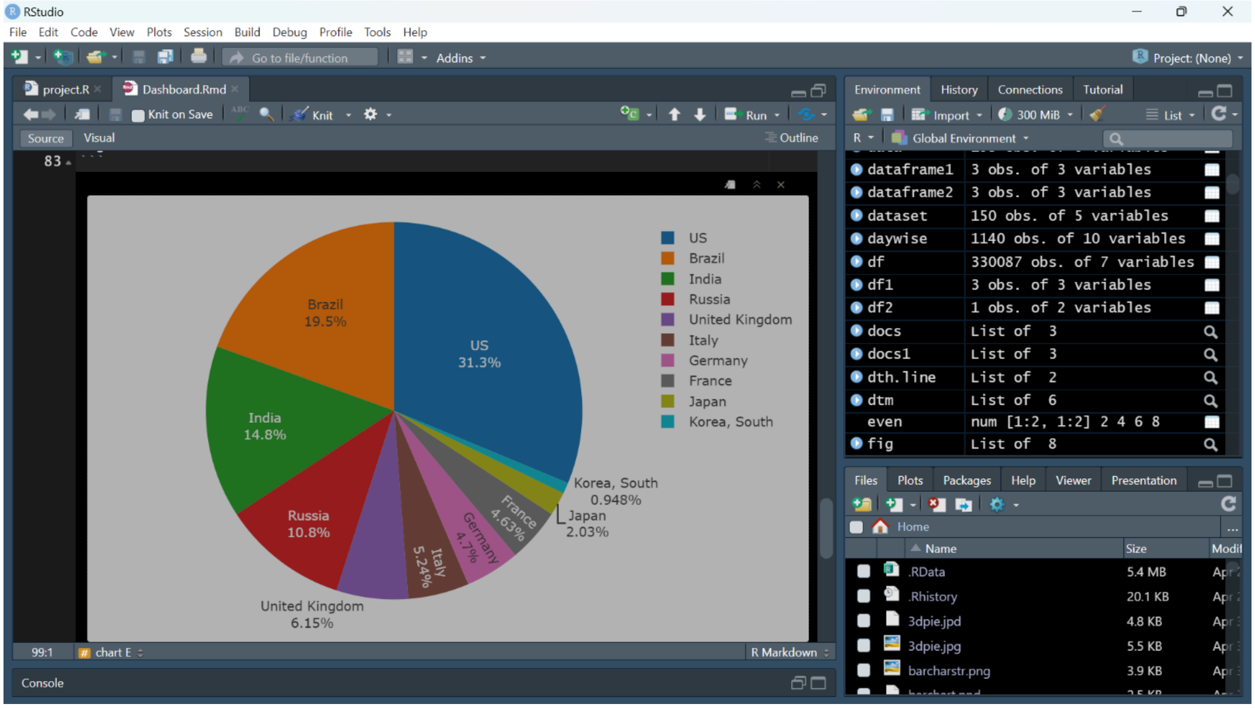
The second, third, and fourth lines of the code add three traces to the plotly object "fig". Each trace corresponds to a different variable ("Confirmed", "Recovered", "Deaths") and contains the data for that variable over time. The "name" argument specifies the label for the trace, and the "mode" argument specifies that a line plot should be used.

The fifth through eighth lines of the code create variables to specify the line colors and widths for each trace. The "cnf", "rec", and "dth" variables correspond to the colors for the "Confirmed", "Recovered", and "Deaths" traces, respectively. The "act" variable is not used in this code.

The ninth through twelfth lines of the code add the three traces to the plotly object "fig" again, but this time with the specified line colors and widths.

The thirteenth through fifteenth lines of the code use the "layout" function to add a title and axis labels to the plot. The "title" argument specifies the title for the plot, and the "xaxis" and "yaxis" arguments specify the labels for the x-axis and y-axis, respectively.

Finally, the "fig" object is displayed using the "fig" command.



```
dataframe1  3 obs. of 3 variables
dataframe2  3 obs. of 3 variables
dataset     150 obs. of 5 variables
daywise     1140 obs. of 10 variables
df          330087 obs. of 7 variables
df1         3 obs. of 3 variables
df2         1 obs. of 2 variables
docs        List of  3
docs1       List of  3
dth.line    List of  2
dtm         List of  6
even        num [1:2, 1:2] 2 4 6 8
fig         List of  8
```

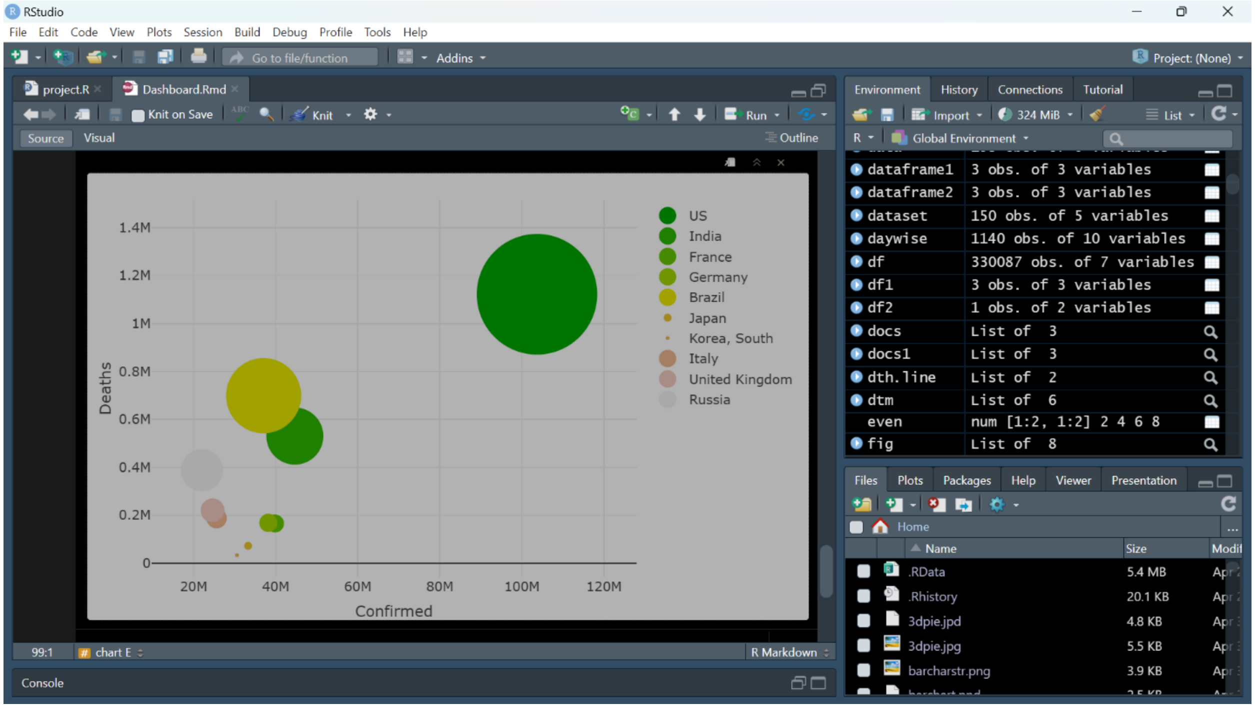
```
.RData          5.4 MB
.Rhistory       20.1 KB
3dpie.jpd       4.8 KB
3dpie.jpg       5.5 KB
barcharstr.png  3.9 KB
```

This code uses the R programming language to create a pie chart of the distribution of COVID-19 deaths among the top 10 countries with the highest number of confirmed cases.

The first line of the code creates a plotly object called "fig" with the "top10" data frame as the data source. The "labels" argument maps the "Country" column to the labels on the pie chart, and the "values" argument maps the "Deaths" column to the values on the pie chart.

The "type" argument specifies that a pie chart should be created, and the "textinfo" argument specifies that the labels on the pie chart should include both the label text and the percentage of the total that each label represents.

Overall, this code produces a pie chart that shows the distribution of COVID-19 deaths among the top 10 countries with the highest number of confirmed cases.



dataframe1	3 obs. of 3 variables
dataframe2	3 obs. of 3 variables
dataset	150 obs. of 5 variables
daywise	1140 obs. of 10 variables
df	330087 obs. of 7 variables
df1	3 obs. of 3 variables
df2	1 obs. of 2 variables
docs	List of 3
docs1	List of 3
dth.line	List of 2
dtm	List of 6
even	num [1:2, 1:2] 2 4 6 8
fig	List of 8

Name	Size	Modif
.RData	5.4 MB	Apr
.Rhistory	20.1 KB	Apr
3dpie.jpd	4.8 KB	Apr
3dpie.jpg	5.5 KB	Apr
barcharstr.png	3.9 KB	Apr

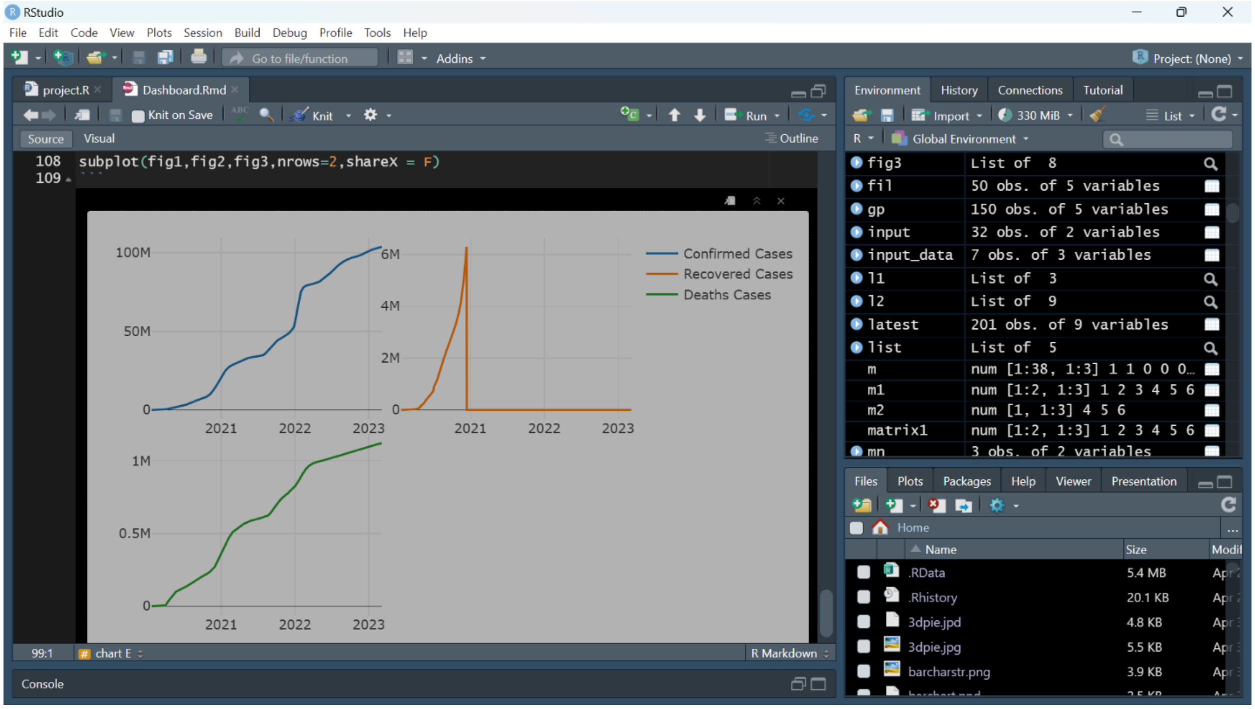
R programming language to create a scatter plot that compares the number of COVID-19 confirmed cases and deaths for each of the top 10 countries with the highest number of confirmed cases.

The first line of the code creates a plotly object called "fig" with the "top10" data frame as the data source. The "x" argument maps the "Confirmed" column to the x-axis, and the "y" argument maps the "Deaths" column to the y-axis.

The "type" argument specifies that a scatter plot should be created, and the "mode" argument specifies that markers should be used. The "color" argument maps the "Country" column to the marker colors on the plot, and the "colors" argument specifies the color palette to be used.

The "size" argument maps the "Confirmed" column to the size of the markers on the plot, and the "marker" argument specifies additional marker properties. In this case, the size of the markers is set to be proportional to the number of confirmed cases, and the size of the marker border is set to be proportional to the number of deaths.

Overall, this code produces a scatter plot that shows the relationship between the number of COVID-19 confirmed cases and deaths for each of the top 10 countries with the highest number of confirmed cases. The plot also provides information about the relative sizes of each country's outbreak, as well as the number of deaths associated with each outbreak.



```
subplot(fig1,fig2,fig3,nrows=2,shareX = F)
```

fig3	List of 8
fil	50 obs. of 5 variables
gp	150 obs. of 5 variables
input	32 obs. of 2 variables
input_data	7 obs. of 3 variables
l1	List of 3
l2	List of 9
latest	201 obs. of 9 variables
list	List of 5
m	num [1:38, 1:3] 1 1 0 0 0…
m1	num [1:2, 1:3] 1 2 3 4 5 6
m2	num [1, 1:3] 4 5 6
matrix1	num [1:2, 1:3] 1 2 3 4 5 6
mn	3 obs. of 2 variables

Name	Size	Modif
.RData	5.4 MB	Apr
.Rhistory	20.1 KB	Apr
3dpie.jpd	4.8 KB	Apr
3dpie.jpg	5.5 KB	Apr
barcharstr.png	3.9 KB	Apr

This code uses the R programming language to create a subplot with three line plots that show the progression of COVID-19 cases in the United States over time.

The first line of the code filters the "covid" data frame to only include data for the United States, and sorts the data by date. This creates a new data frame called "US".

The next three lines of the code create plotly objects called "fig1", "fig2", and "fig3" with the "US" data frame as the data source. Each plot shows the progression of a different type of COVID-19 case (confirmed, recovered, or deaths) over time.

The "x" argument maps the "Date" column to the x-axis, and the "y" argument maps the corresponding column for each plot to the y-axis.

The "type" argument specifies that a line plot should be created, and the "mode" argument specifies that lines should be used.

The "name" argument specifies the name of each plot. This will be used later in the "subplot" function to create a legend for the plot.

The final line of the code uses the "subplot" function to create a subplot with the three line plots created earlier. The "nrows" argument specifies that the plots should be arranged in two rows, and the "shareX" argument specifies that the x-axis should not be shared between the two rows. This creates a subplot that shows the progression of COVID-19 cases in the United States over time, with each row representing a different type of case. The legend at the top of the plot shows the names of each plot.

References:

Kaggle.com

Google.com

Youtube.com