

## **Assignment 3-Storytelling with Open Data - Tokyo Olympics 2021 (2020)**

**Student Name :** Karthik Kolume

**Student number :** S3857825

**URL :** <https://rpubs.com/s3857825/827814>

**Raw Code :**

```
---
title: "Assignment 3-Storytelling with Open Data - Tokyo Olympics 2021 (2020)"
output:
  flexdashboard::flex_dashboard:
    orientation: columns
    vertical_layout: fill
---

``{r setup, include=FALSE}
library(flexdashboard)
library(tidyverse)
library(readxl)
library(RColorBrewer)
library(plotly)
library(highcharter)
library(ggwordcloud)
theme_set(theme_bw())

setwd("~/Documents/RMIT/RMIT Work/Sem2/Data Viz/Assignment3")

athletes <- read_excel("Athletes.xlsx")
medals <- read_excel("Medals.xlsx")
teams <- read_excel("Teams.xlsx")
gender <- read_excel("EntriesGender.xlsx")

athletes <- athletes %>%
  rename(country = NOC) %>%
  mutate(country = replace(country, country == "Côte d'Ivoire", "Cote d'Ivoire"),
         country = replace(country, country == "ROC", "Russia")) %>%
  distinct()

teams <- teams %>%
  rename(country = Name, gender = Event) %>%
  distinct() %>%
  select(country, Discipline, gender)
```

```

medals <- medals %>%
  mutate(Rank = as.numeric(Rank),
         Gold = as.numeric(Gold),
         Silver = as.numeric(Silver),
         Bronze = as.numeric(Bronze),
         Total = as.numeric(Total),
         'Rank by Total' = as.numeric(`Rank by Total`))

```

```

```

```

```

Column{data-width=600}

```

---

### Number of Medals won by 25 Countries in Tokyo Olympics 2021 and the Medal type(Gold, Silver, Bronze). Select medal type in the dropdown available on the left side of the chart.

```

```{r}
names(medals)[2] <- 'country'

medals_longer <- medals %>%
  pivot_longer(cols = 3:6,
               names_to = "Medal_Type",
               values_to = "Number_of_Medals") %>%
  arrange('Rank by Total')%>%head(100)
plot_ly(medals_longer,
        x = ~Number_of_Medals,
        y = ~reorder(country, -Rank),
        type = 'bar',orientation='h', color= "gray33",
        hoverinfo = 'text',
        text = ~paste('</br> Country: ', country,
                      '</br> # Medals', Number_of_Medals),
        transforms = list(
          list(
            type = 'filter',
            target = ~Medal_Type,
            operation = 'in',
            value = unique(medals_longer$Medal_Type)[1]
          )
        )) %>%
  layout(
    updatemenus = list(
      list(
        type = 'dropdown',
        active = 0,
        buttons = list(
          list(method = "restyle",

```

```

      args = list("transforms[0].value", unique(medals_longer$Medal_Type)[1]),
      label = unique(medals_longer$Medal_Type)[1]),
    list(method = "restyle",
      args = list("transforms[0].value", unique(medals_longer$Medal_Type)[2]),
      label = unique(medals_longer$Medal_Type)[2]),
    list(method = "restyle",
      args = list("transforms[0].value", unique(medals_longer$Medal_Type)[3]),
      label = unique(medals_longer$Medal_Type)[3]),
    list(method = "restyle",
      args = list("transforms[0].value", unique(medals_longer$Medal_Type)[4]),
      label = unique(medals_longer$Medal_Type)[4])
  )
)
)
) %>%
layout(title = 'Countries by Rank <br><sup>Select type of Medals</sup>',
  xaxis = list(title = 'Number of Medals'), yaxis = list(title = 'Top 25 Countries'))

```

```

...

```

```

row

```

```

-----
### Top 20 countries with maximum number of athlete participation and its Count.

```

```

```{r fig.height=3}
ccolor <- "#009999"
athletes %>%
  group_by(country) %>%
  summarise(Count = n()) %>%
  arrange((desc(Count))) %>%
  head(20) %>%
  hchart('column', hcaes(x = country, y = Count, color = ccolor)) %>%
  hc_tooltip(pointFormat = '<b>Count: </b> {point.y} <br>') %>%
  hc_title(text = 'Number of Athletes by Country',
    style = list(fontSize = '25px', fontWeight = 'bold')) %>%
  hc_xAxis(title = list(text = 'Top 20 countries')) %>%
  hc_yAxis(title = list(text = 'Number of Athletes'))
...

```

```

-----
### Gender distribution: Number of Female participation to Number of Male Participation.

```

```

```{r}
gender$Female<-as.numeric(gender$Female)
gender$Male<-as.numeric(gender$Male)
gender$Total<-as.numeric(gender$Total)

plot_ly(gender,

```

```

x = ~Female,
y = ~reorder(Discipline,-Total),
type = 'bar', orientation = 'h',
name = 'Female',
marker = list(color = 'rgba(246, 78, 139, 0.6)',
              line = list(color = 'rgba(246, 78, 139, 1.0)',
                          width = 1))) %>%
add_trace(x = ~Male,
          name = 'Male',
          marker = list(color = 'rgba(58, 71, 80, 0.6)',
                        line = list(color = 'rgba(58, 71, 80, 1.0)',
                                    width = 1)))%>%
layout(barmode = 'stack',
       xaxis = list(title = "Number of Athletes"),
       yaxis = list(title = "Sport", size = 0.2))

```