````
---
title: "maps"
output: html_document
date: "2024-04-04"
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

```{r, echo=FALSE, warning=FALSE}
library(quantmod)
library(tidyquant)
library(tidyverse)
library(polite)
library(rvest)
library(highcharter)
library(magrittr)
library(gt)
library(quantmod)
library(gtExtras)
```

```{r, echo=FALSE, warning=FALSE}
rm(list=ls())
url <- "https://editorial.rottentomatoes.com/guide/best-netflix-shows-and-movies-to-binge-
watch-now/"

titles <- url%>%
  bow()%>%
  scrape()%>%
  html_nodes(".article_movie_title a")%>%
  html_text()

imgs<- url%>%
  bow()%>%
  scrape()%>%
  html_nodes(".article_poster")%>%
  html_attr("src")

ratings <- url%>%
  bow()%>%
  scrape()%>%
  html_nodes(".tMeterScore")%>%
  html_text()%>%
  gsub(pattern="%", replacement="")%>%
  as.numeric()/100

year <- url%>%
  bow()%>%
  scrape()%>%
  html_nodes(".start-year")%>%
  html_text()%>%
  gsub(pattern="\\(|\\)", replacement="")%>%
  as.numeric()

data.frame(titles,imgs, year, ratings)%>%
  arrange(., desc(year))%>%
  gt()%>%
  cols_label(titles="Titles",
             imgs="Posters",
             ratings="Ratings")%>%
  fmt_percent(columns = "ratings", decimals=0)%>%
  data_color(columns=c("ratings"),method = "numeric", palette = c("red", "green"))%>%
  gtExtras::gt_theme_guardian()%>%
  gtExtras::gt_img_rows(imgs, height=150)%>%
  tab_header(title="100 best Netflix series")%>%
  tab_footnote(footnote = "Source: Rotten Tomatoes")%>%
  opt_interactive(use_search = TRUE)
````
````

```
```

```{r, warning=FALSE, echo=FALSE}

ticker <-c("PEP", "KO")
df<- getSymbols(Symbols=ticker, auto.assign = TRUE)%>%
  map(~Ad(get(.)))%>%
  reduce(merge)%>%
  set_colnames(c("PEPSI", "COLA"))%>%
  to.monthly(indexAt = "lastof", OHLC=FALSE)

rm(list=ticker)

highchart(type="stock")%>%
  hc_add_series(data=df$PEPSI%>%round(digits=2),name="PEPSI")%>%
  hc_add_series(data=df$COLA%>%round(digits=2),name="COLA")%>%
  hc_colors(
    colors = c("#0E91A4",
               "#0869D4")
  )%>%
  hc_tooltip(pointFormat="<b>{series.name}: </b> ${point.y}<br>")%>%
  hc_add_theme(hc_theme_hcrt())%>%

  hc_plotOptions(
    line = list(
      lineWidth = 3.5
    )
  )



  highchart(type="stock")%>%
  hc_add_series(data=df$PEPSI%>%round(digits=2),name="PEPSI")%>%
  hc_add_series(data=df$COLA%>%round(digits=2),name="COLA")%>%
  hc_colors(
    colors = c("white",
               "yellow")
  )%>%
  hc_tooltip(pointFormat="<b>{series.name}: </b> ${point.y}<br>")%>%
  hc_add_theme(hc_theme_google())%>%

  hc_plotOptions(
    line = list(
      lineWidth = 1.5
    )
  )%>%
    hc_plotOptions(series = list(animation = list(duration = 7000, easing = "easeOutBounce")),
                   line=list(lineWidth=3.5))%>%
    hc_title(text=c("Stock price movements"),
             style=list(fontFamily='times new roman',
                        fontSize = "20px", fontWeight = "bold"))%>%
    hc_xAxis(
      title = list(text = "Dates",
                   style = list(fontFamily = "times new roman", fontSize = "15px")),
      labels = list(style = list(fontFamily = "times new roman", fontSize = "12px"))
      )%>%
    hc_yAxis(
      title = list(text = "Price ($)",
                   style = list(fontFamily = "times new roman", fontSize = "15px")),
      labels = list(style = list(fontFamily = "times new roman", fontSize = "12px"))
      )%>%
  hc_chart(
borderColor = "navy",
borderRadius = 10,
borderWidth = 2,
backgroundColor = list(
linearGradient = c(0, 0, 500, 500),
```

```
stops = list(
list(0, "purple"),
list(1, "navy")
)
)
)
```

```{r}
honeyproduction <- read.csv("honeyproduction.csv")%>%
  group_by(state, year)%>%
  mutate(production_growth=log(totalprod))%>%
  transform(production_growth=round(production_growth, digits=2))
df<- honeyproduction%>%
  group_by(state)%>%
  do(
    item= list(
      postalcode = first(.$state),
      sequence = .$production_growth,
      value = first(.$totalprod)
    )
  ) %>%
  .$item

map_data <- usgeojson

highchart(type = "map") %>%
  hc_add_series(
    data = df,
    mapData =map_data ,
    joinBy =c("postalcode"),
    borderWidth = 0.02
  ) %>%
  hc_colorAxis(stops = color_stops()) %>%
  hc_title(text = "Total Production of Honey") %>%
  hc_subtitle(text = "Honey Production over the years in each state") %>%
  hc_legend(
    layout = "horizontal",
    align = "right"
    ) %>%
  hc_motion(
    enabled = TRUE,
    autoPlay=TRUE,
    loop=TRUE,
    axisLabel = "year",
    labels = sort(unique(honeyproduction$year)),
    magnet = list(
      round = "floor",
      step = 0.1
    )
  ) %>%
  hc_chart(marginBottom  = 100)
```

```{r}
rm(list=ls())
library(jsonlite)
library(dplyr)
library(tidyr)
library(highcharter)
rm(list=ls())
URL <- "http://graphics8.nytimes.com/newsgraphics/2016/01/15/drug-
deaths/c23ba79c9c9599a103a8d60e2329be1a9b7d6994/data.json"

data("uscountygeojson")

data("unemployment")
```

```
data <-  fromJSON(URL) %>%
  as_tibble() %>%
  gather(year, value, -fips) %>%
  mutate(year = sub("^y", "", year),
         value = ifelse(is.na(value), 0, value))

ds <- data %>%
  group_by(fips) %>%
  do(item = list(
    fips = first(.$fips),
    sequence = .$value,
    value = first(.$value))) %>%
  .$item

r <-highchart(type = "map") %>%
  hc_add_series(
    data = ds,
    name = "drug deaths per 100,000",
    mapData = uscountygeojson,
    joinBy = "fips",
    borderWidth = 0.01
    ) %>%
  hc_motion(
    enabled = TRUE,
    axisLabel = "year",
    labels = sort(unique(data$year)),
    series = 0,
    updateIterval = 50,
    magnet = list(
      round = "floor",
      step = 0.1
    )
  ) %>%
  hc_chart(marginBottom  = 100) %>%
  hc_colorAxis(stops = color_stops()) %>%
  hc_title(text = "How the Epidemic of Drug Overdose Deaths Ripples") %>%
  hc_subtitle(text = "Overdose deaths per 100,000") %>%
  hc_legend(
    layout = "horizontal",
    reversed = TRUE,
    loop=TRUE,
    autoPlay=TRUE,
    floating = TRUE,
    align = "right"
    )
r
```