Assignment_Week_3&4_Venkidusamy_KesavAdithya

Kesav Adithya Venkidusamy

2022/06/24

```
knitr::opts_chunk$set(echo = TRUE)

library(readxl)
library(ggplot2)
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(treemapify)
```

```
## Warning: package 'treemapify' was built under R version 4.1.3
```

Data Loading

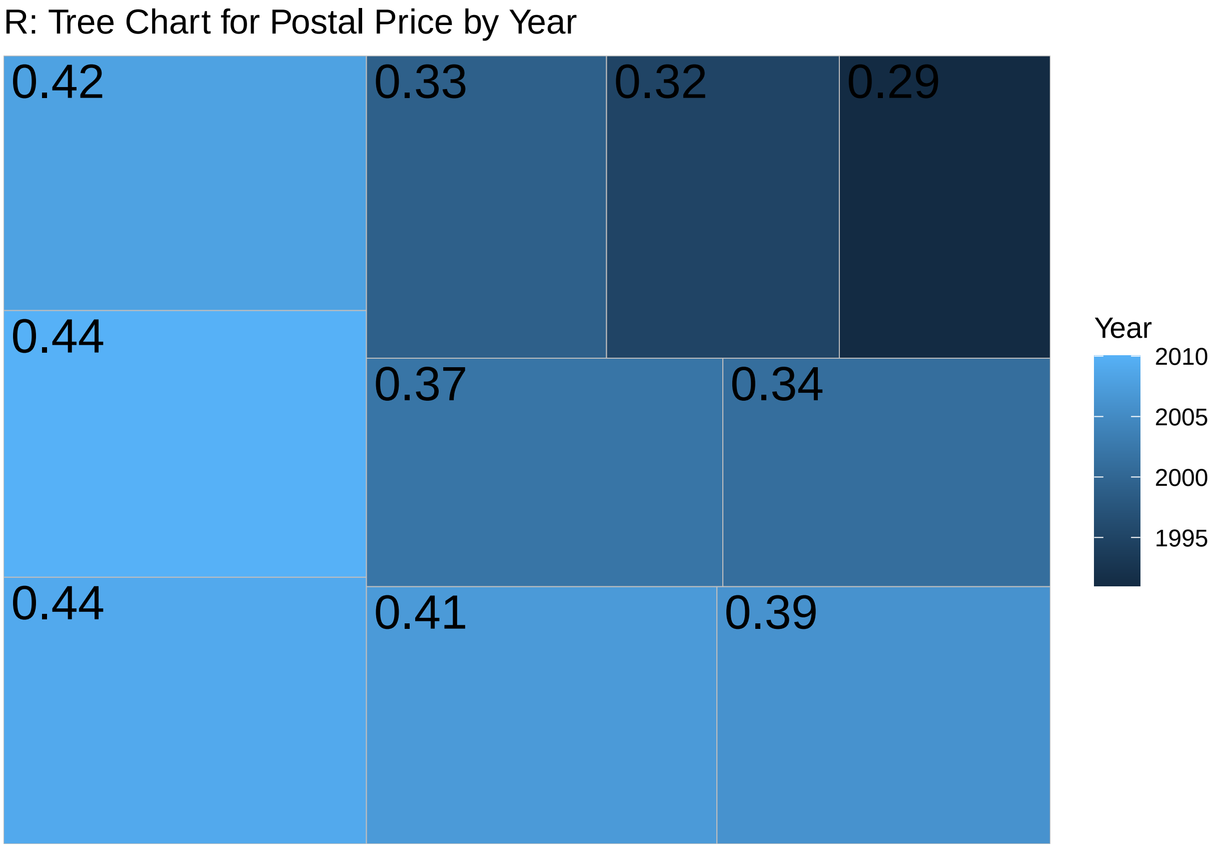
```
post_df <- read_excel("E:/Personal/Bellevue University/Course/github/dsc640/Week 3&4/us-postage.xlsm")
head(post_df)
```

```
## # A tibble: 6 x 2
##    Year Price
##   <dbl> <dbl>
## 1  1991  0.29
## 2  1995  0.32
## 3  1999  0.33
## 4  2001  0.34
## 5  2002  0.37
## 6  2006  0.39
```

```
# Total number of records present in the data set
nrow(post_df)
```

```
## [1] 10
```

```
## Create Tree Chart

ggplot(post_df, aes(area = Price, fill = Year, label = Price)) + geom_treemap() + geom_treemap_text() +
```



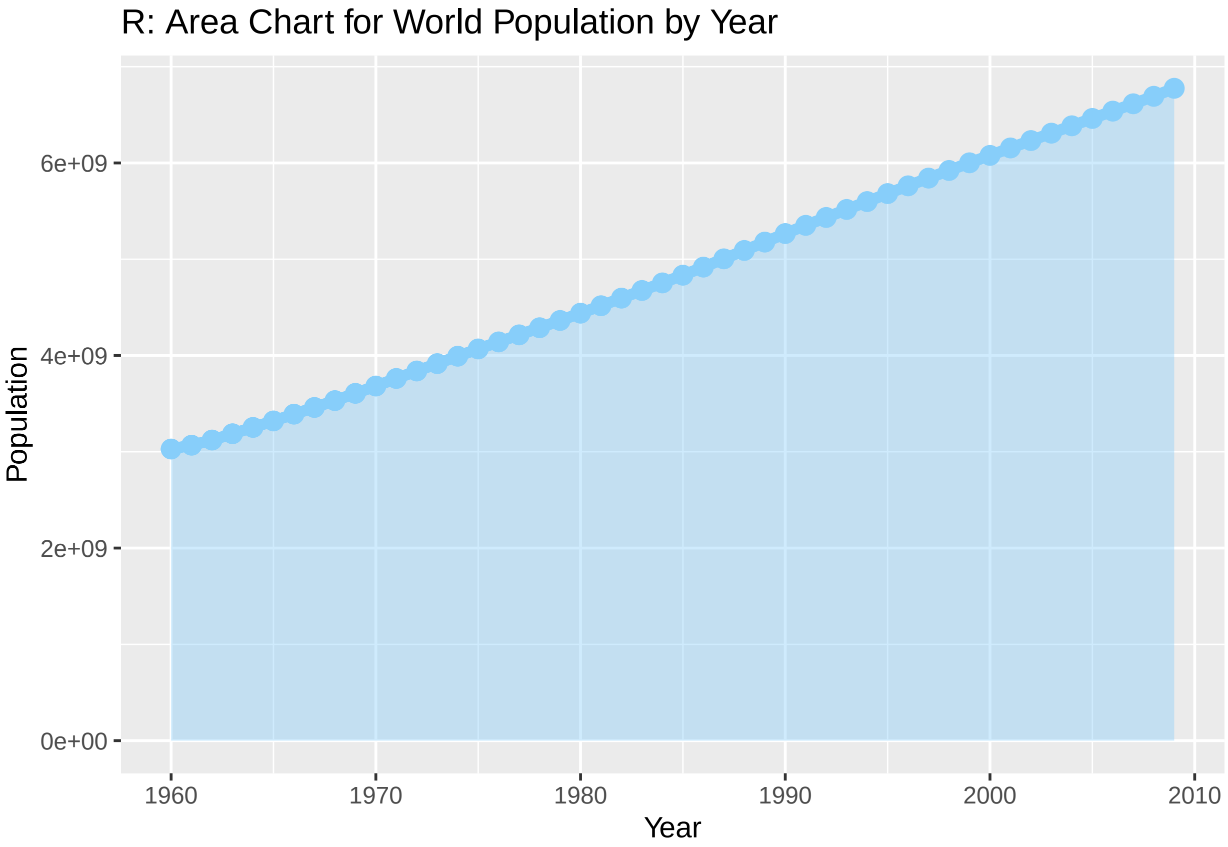
```
pop_df <- read_excel("E:/Personal/Bellevue University/Course/github/dsc640/Week 3&4/world-population.xl
head(pop_df)
```

```
## # A tibble: 6 x 2
##    Year Population
##   <dbl>      <dbl>
## 1  1960 3028654024
## 2  1961 3068356747
## 3  1962 3121963107
## 4  1963 3187471383
## 5  1964 3253112403
## 6  1965 3320396924
```

```
# Total number of records present in the data set
nrow(pop_df)
```

```
## [1] 50
```

```
## Create Area Chart

ggplot(pop_df, aes(x=Year, y=Population)) +
  geom_area( fill="#87CEFA", alpha=0.4) +
  geom_line(color="#87CEFA", size=2) +
  geom_point(size=3, color="#87CEFA") +
  ggtitle("R: Area Chart for World Population by Year")
```



```
unemp_df <- read_csv("E:/Personal/Bellevue University/Course/github/dsc640/Week 3&4/unemployement-rate-
```

```
## Rows: 746 Columns: 4

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Series id, Period
## dbl (2): Year, Value

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

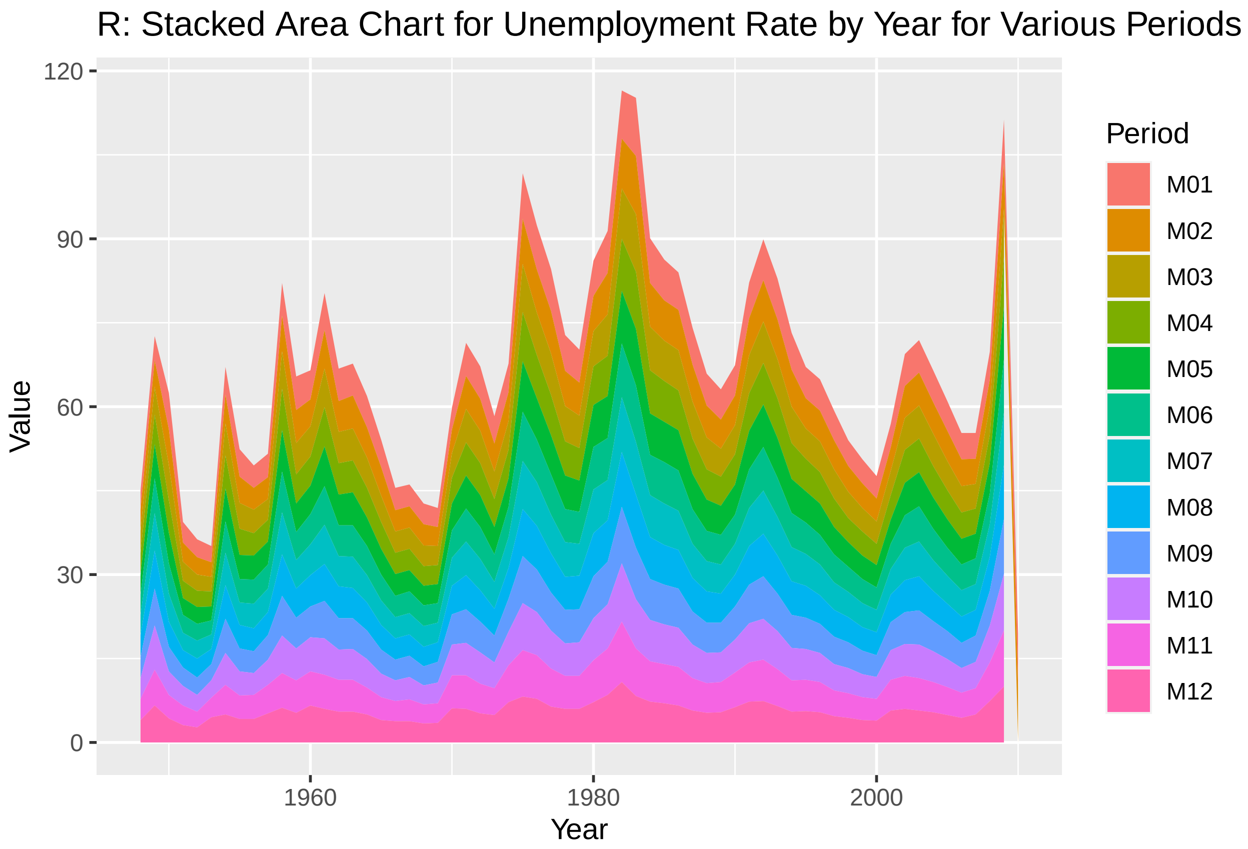
```
head(unemp_df)
```

```
## # A tibble: 6 x 4
##   `Series id`  Year Period Value
##   <chr>       <dbl> <chr>  <dbl>
## 1 LNS14000000  1948 M01      3.4
## 2 LNS14000000  1948 M02      3.8
## 3 LNS14000000  1948 M03      4
## 4 LNS14000000  1948 M04      3.9
## 5 LNS14000000  1948 M05      3.5
## 6 LNS14000000  1948 M06      3.6
```

```
# Total number of records present in the data set
nrow(unemp_df)
```

```
## [1] 746
```

```
## Create Stacked Area Chart

ggplot(unemp_df, aes(x=Year, y=Value, fill=Period)) +
    geom_area() + ggtitle("R: Stacked Area Chart for Unemployment Rate by Year for Various Periods")
```



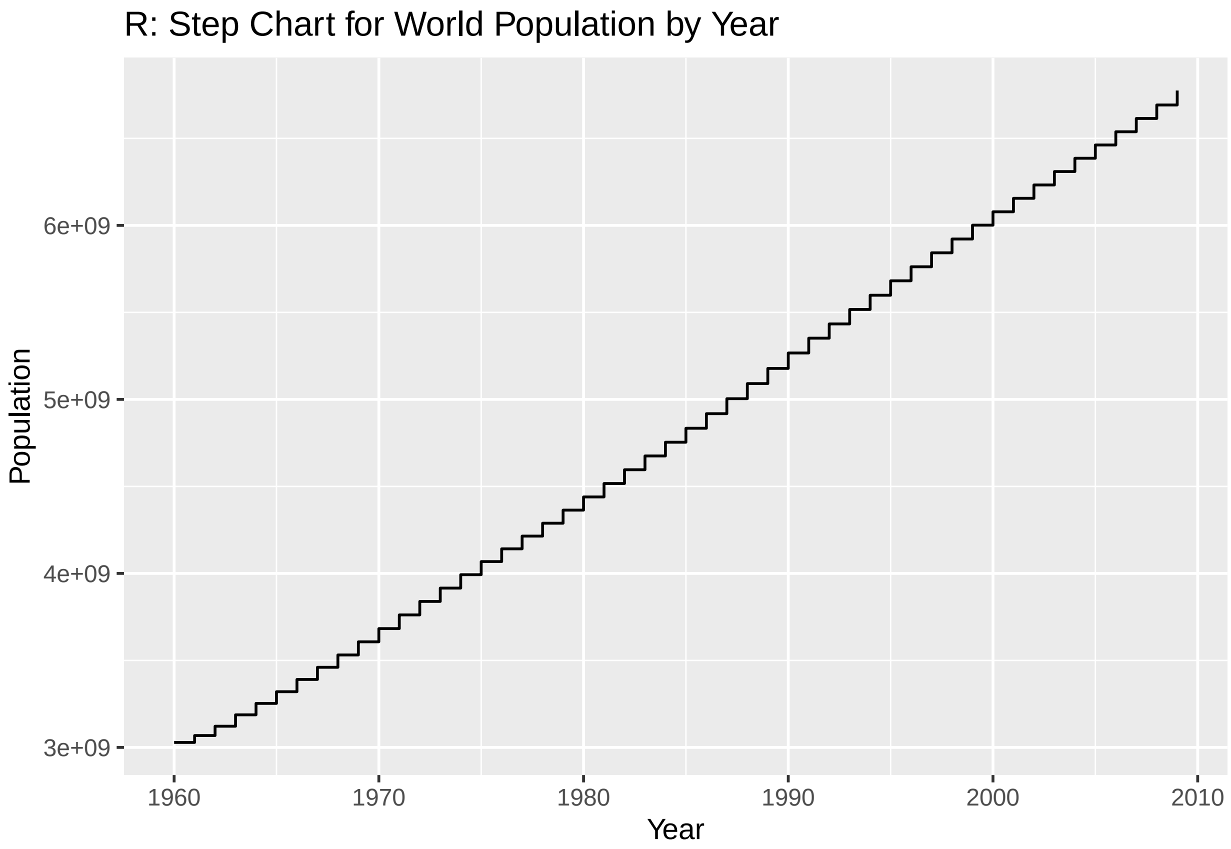
```
pop_df <- read_excel("E:/Personal/Bellevue University/Course/github/dsc640/Week 3&4/world-population.xls
head(pop_df)
```

```
## # A tibble: 6 x 2
##     Year Population
##    <dbl>      <dbl>
## 1  1960 3028654024
## 2  1961 3068356747
## 3  1962 3121963107
## 4  1963 3187471383
## 5  1964 3253112403
## 6  1965 3320396924
```

```
# Total number of records present in the data set
nrow(pop_df)
```

```
## [1] 50
```

```
## Create Step Chart

ggplot(pop_df, aes(x=Year, y=Population)) + geom_step() + ggtitle("R: Step Chart for World Population by
```



Week 3 & 4 Assignment - Python

Name: Kesav Adithya Venkidusamy
Course: DSC640 - Data Presentation and Visualization
Instructor: Catherine Williams

These two weeks we are going to be focused on tree maps, area charts, stacked area charts, and step charts and using various tools to create these visualizations. You must consolidate all the charts into ONE document with each chart labeled with the type of chart and technology - for example: Python - Bar Chart. Failure to label and consolidate the charts will resort in points being taken off or a 0 for the assignment. Sample Datasets (click on the Downloads tab.) You may also download them directly from this link: Exercise 2.2 Datasets (click the link to download a folder containing the datasets.)

You need to submit:

1 tree map, 1 area chart, 1 stacked area chart, and 1 step chart using Tableau or PowerBI
1 tree map, 1 area chart, 1 stacked area chart, and 1 step chart using Python
1 tree map, 1 area chart, 1 stacked area chart, and 1 step chart using R

You are welcome to use your own datasets or the data provided.

1 tree map, 1 area chart, 1 stacked area chart, and 1 step chart using Python

In [6]:

```
## Importing libraries required for this exercise
import pandas as pd
import numpy as np
import squarify
import matplotlib.pyplot as plt
%matplotlib inline
```

1. Python - Tree map Chart

Plotting Tree chart for postal price and year

In [3]:

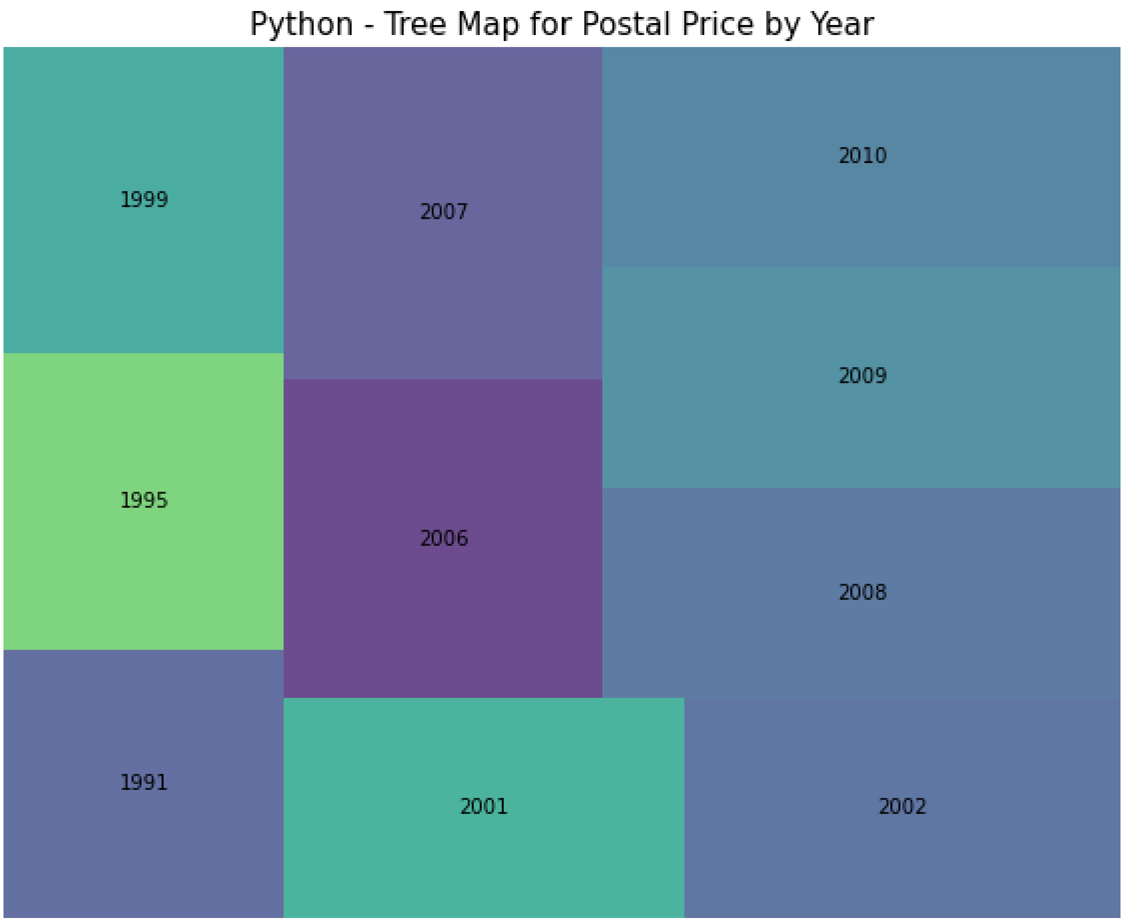
```
## Load the dataset into dataframe
post_df = pd.read_excel("us-postage.xlsm")
post_df.head()
```

Out[3]:

	Year	Price
0	1991	0.29
1	1995	0.32
2	1999	0.33
3	2001	0.34
4	2002	0.37

In [20]:

```
plt.figure(figsize=(10,8))
squarify.plot(sizes=post_df['Price'], label=post_df['Year'], alpha=.8 )
plt.title("Python - Tree Map for Postal Price by Year", fontsize = 15)
plt.axis('off')
plt.show()
```



2. Python - Area Chart

Plotting area chart for world population and year

In [15]:

```
## Reading us population dataset into dataframe
us_df = pd.read_excel("world-population.xlsm")
us_df.head()
```

Out[15]:

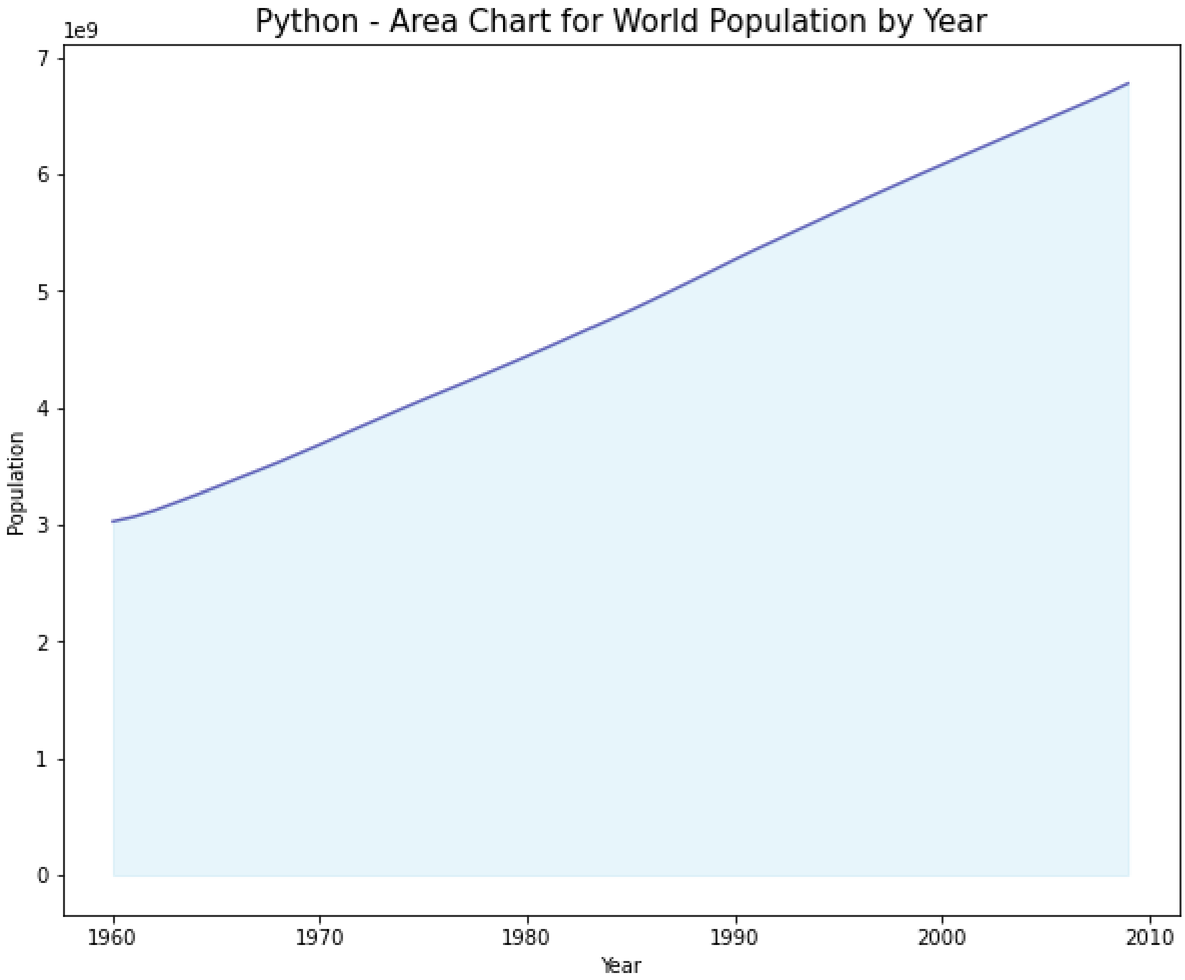
	Year	Population
0	1960	3028654024
1	1961	3068356747
2	1962	3121963107
3	1963	3187471383
4	1964	3253112403

In [21]:

```
# Create x and y values to plot
```

```
x = us_df['Year']
y = us_df['Population']

# Add a stronger line on top (edge)
plt.figure(figsize=(10,8))
plt.fill_between( x, y, color='skyblue', alpha=0.2)
plt.title('Python - Area Chart for World Population by Year', fontsize = 15)
plt.xlabel('Year')
plt.ylabel('Population')
plt.plot(x, y, color='darkblue', alpha=0.6)
```

```
Out[21]: [<matplotlib.lines.Line2D at 0x14e2607bfa0>]
```



3. Python - Stacked Area Chart

```
In [22]: # For this exercise I have considered the unemployment rate file
unemp_df = pd.read_csv("unemployement-rate-1948-2010.csv")
unemp_df.head()
```

Out[22]:

	Series id	Year	Period	Value
0	LNS14000000	1948	M01	3.4
1	LNS14000000	1948	M02	3.8
2	LNS14000000	1948	M03	4.0

	Series id	Year	Period	Value
3	LNS14000000	1948	M04	3.9
4	LNS14000000	1948	M05	3.5

In [27]:

```
# Creating X and Y axis for stacked area chart
unemp_sub_df =  unemp_df.pivot(index='Year', columns='Period', values='Value').reset_in
unemp_sub_df.head()
```

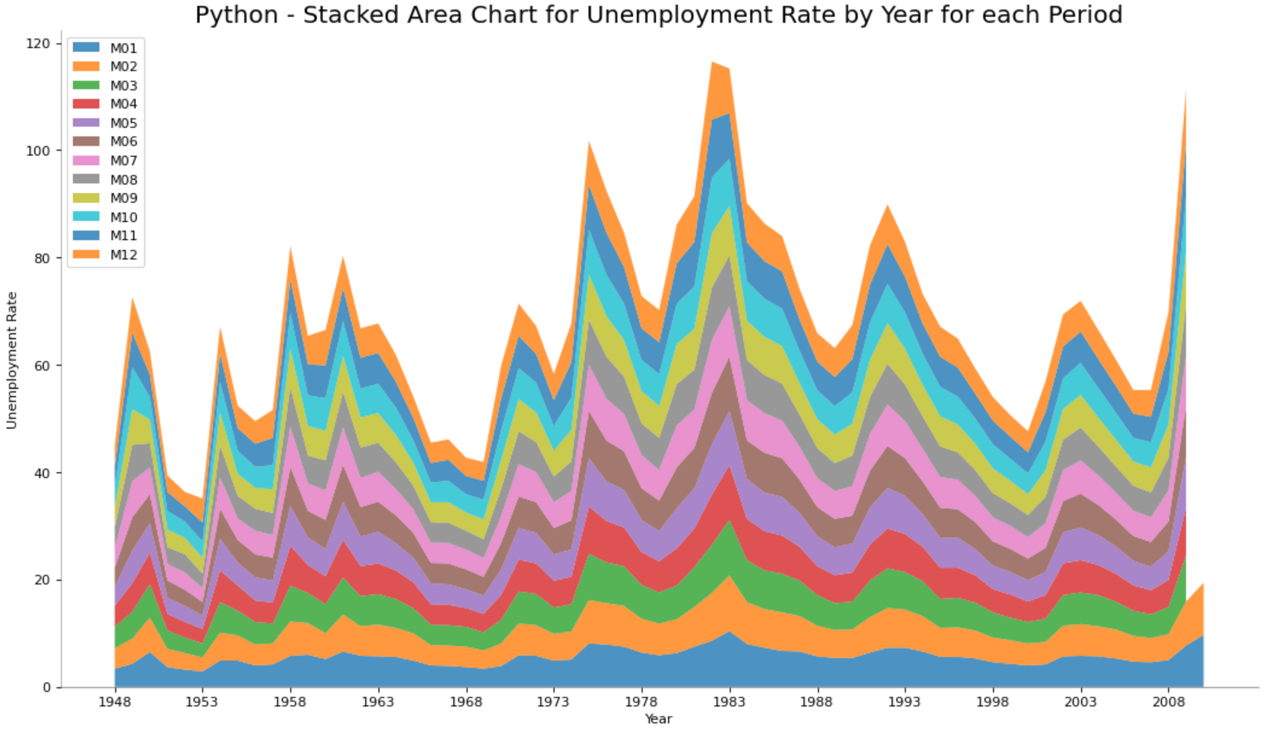
Out[27]:

Period	Year	M01	M02	M03	M04	M05	M06	M07	M08	M09	M10	M11	M12
0	1948	3.4	3.8	4.0	3.9	3.5	3.6	3.6	3.9	3.8	3.7	3.8	4.0
1	1949	4.3	4.7	5.0	5.3	6.1	6.2	6.7	6.8	6.6	7.9	6.4	6.6
2	1950	6.5	6.4	6.3	5.8	5.5	5.4	5.0	4.5	4.4	4.2	4.2	4.3
3	1951	3.7	3.4	3.4	3.1	3.0	3.2	3.1	3.1	3.3	3.5	3.5	3.1
4	1952	3.2	3.1	2.9	2.9	3.0	3.0	3.2	3.4	3.1	3.0	2.8	2.7

In [38]:

```
# Draw Plot and Annotate
fig, ax = plt.subplots(1,1,figsize=(16, 9), dpi= 80)
columns = unemp_sub_df.columns[1:]
labs = unemp_sub_df.values.tolist()

# Prepare data
x  = unemp_sub_df['Year'].values.tolist()
y0 = unemp_sub_df['M01'].values.tolist()
y1 = unemp_sub_df['M02'].values.tolist()
y2 = unemp_sub_df['M03'].values.tolist()
y3 = unemp_sub_df['M04'].values.tolist()
y4 = unemp_sub_df['M05'].values.tolist()
y5 = unemp_sub_df['M06'].values.tolist()
y6 = unemp_sub_df['M07'].values.tolist()
y7 = unemp_sub_df['M08'].values.tolist()
y8 = unemp_sub_df['M09'].values.tolist()
y9 = unemp_sub_df['M10'].values.tolist()
y10 = unemp_sub_df['M11'].values.tolist()
y11 = unemp_sub_df['M12'].values.tolist()
y = np.vstack([y0, y1, y2, y3, y4, y5, y6, y7, y8, y9, y10, y11])

# Plot for each column
labs = columns.values.tolist()
ax = plt.gca()
ax.stackplot(x, y, labels=labs, alpha=0.8)

# Create title
ax.set_title('Python - Stacked Area Chart for Unemployment Rate by Year for each Period
plt.xlabel('Year')
plt.ylabel('Unemployment Rate')

# Show Legend
ax.legend(fontsize=10, ncol=1, loc = 'upper left')
plt.xticks(x[::5], fontsize=10, horizontalalignment='center')

# Lighten borders
```

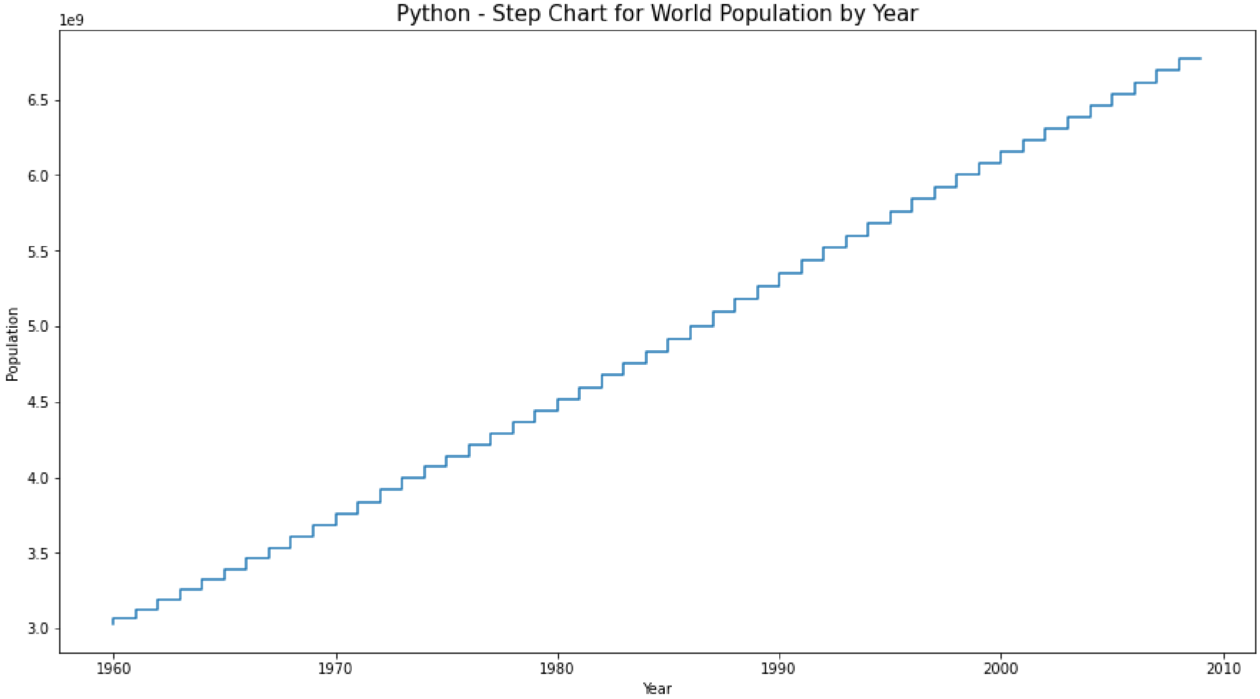
```
plt.gca().spines["top"].set_alpha(0)
plt.gca().spines["bottom"].set_alpha(.3)
plt.gca().spines["right"].set_alpha(0)
plt.gca().spines["left"].set_alpha(.3)

# Output graph
plt.show()
```



Python - Step Chart

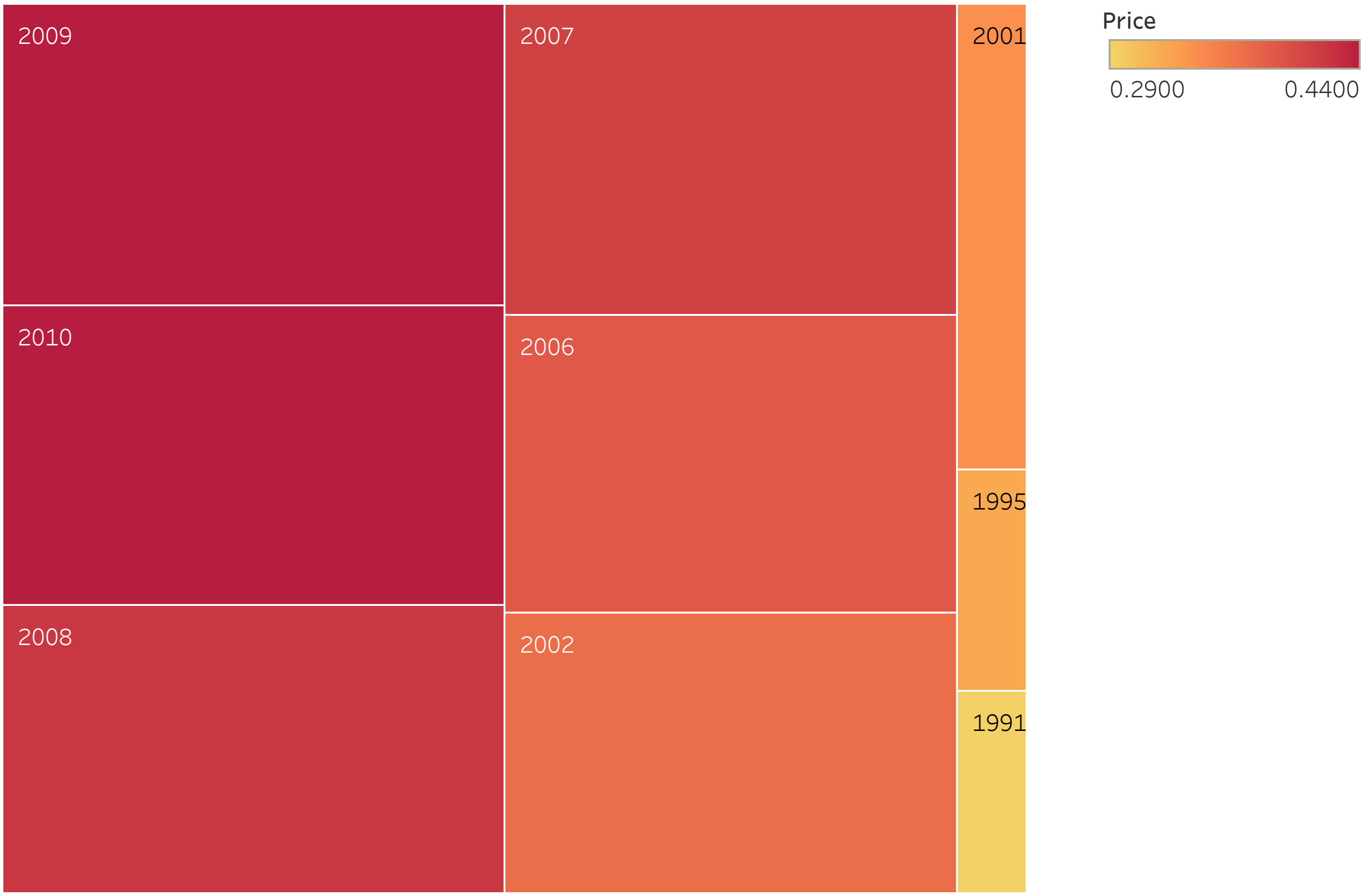
In [42]:

```
## For this chart, we will use world population dataframe created above
## Creating x and y for the plot
x = us_df['Year']
y = us_df['Population']

## Plotting the graph
plt.figure(figsize=(15,8))
plt.step(x, y)
plt.title("Python - Step Chart for World Population by Year",fontsize=15)
plt.xlabel('Year', fontsize=10)
plt.ylabel('Population',fontsize=10)
plt.show()
```


Python - Step Chart for World Population by Year
1e9
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
Population
1960
1970
1980
1990
2000
2010
Year

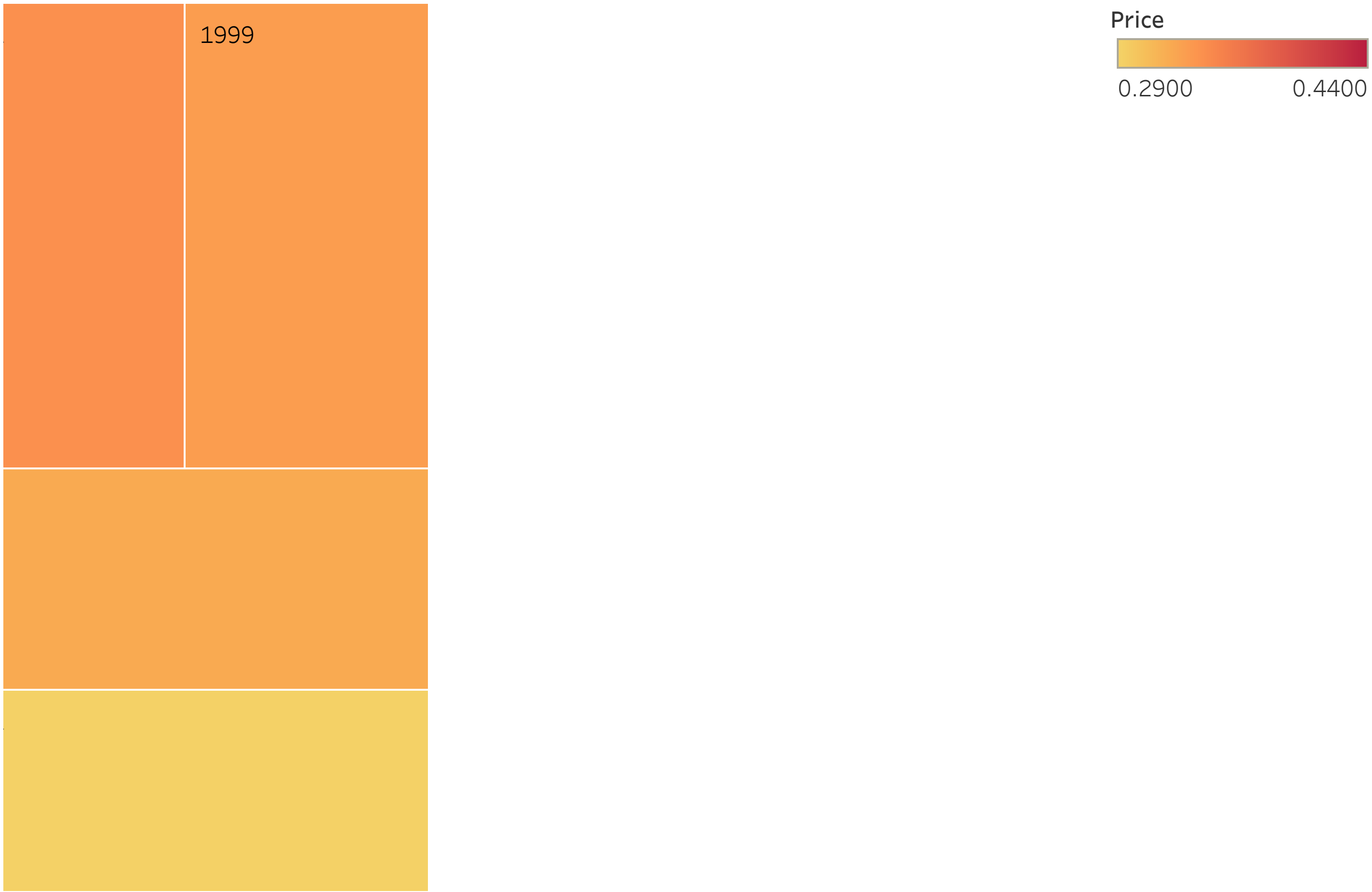
```
In [ ]:
```

Tableau: Tree Chart for USA Postal Rates



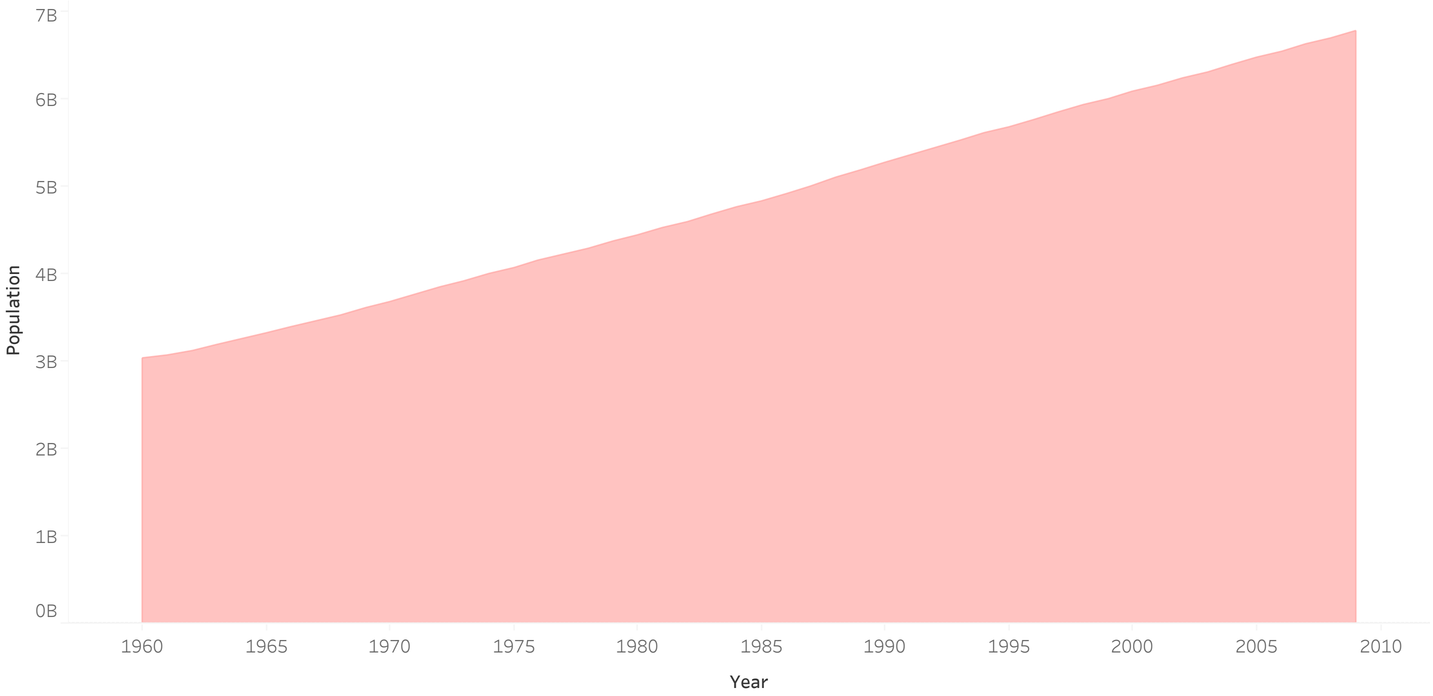
Year. Color shows sum of Price. Size shows sum of Price. The marks are labeled by Year.

Tableau: Tree Chart for USA Postal Rates



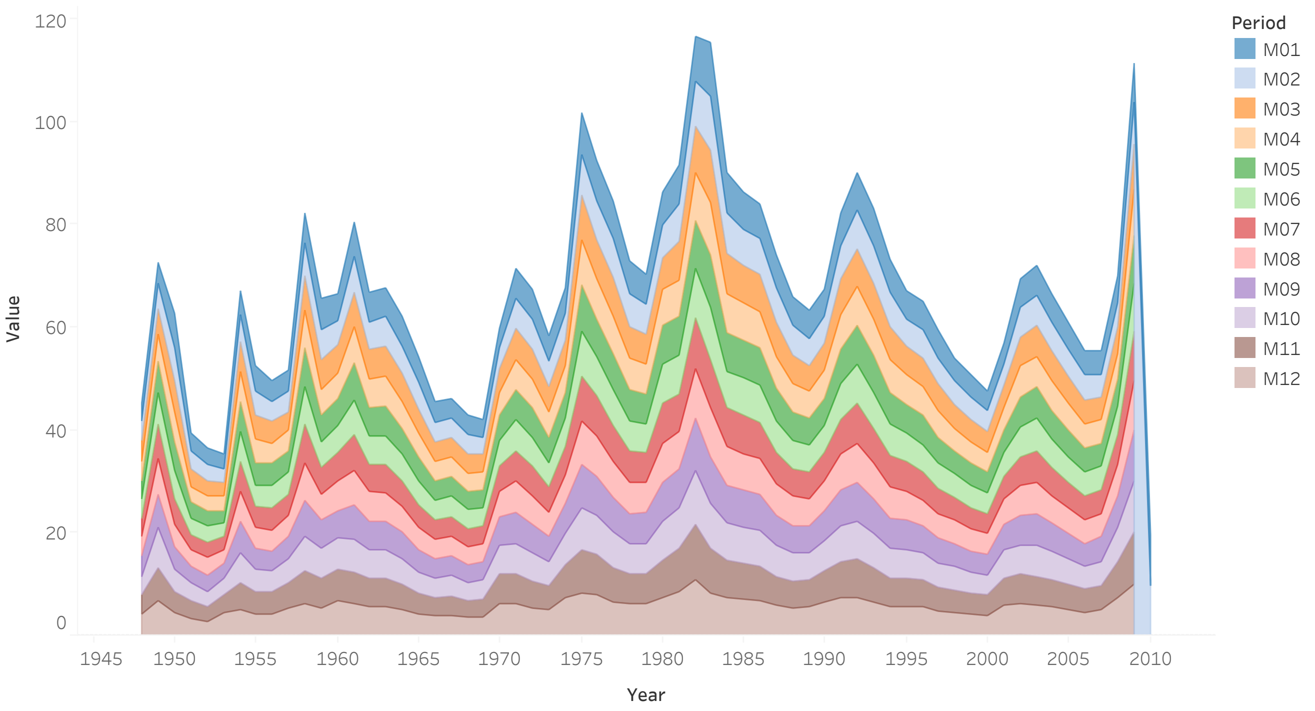
Year. Color shows sum of Price. Size shows sum of Price. The marks are labeled by Year.

Tableau: Area Chart for World Population by Year



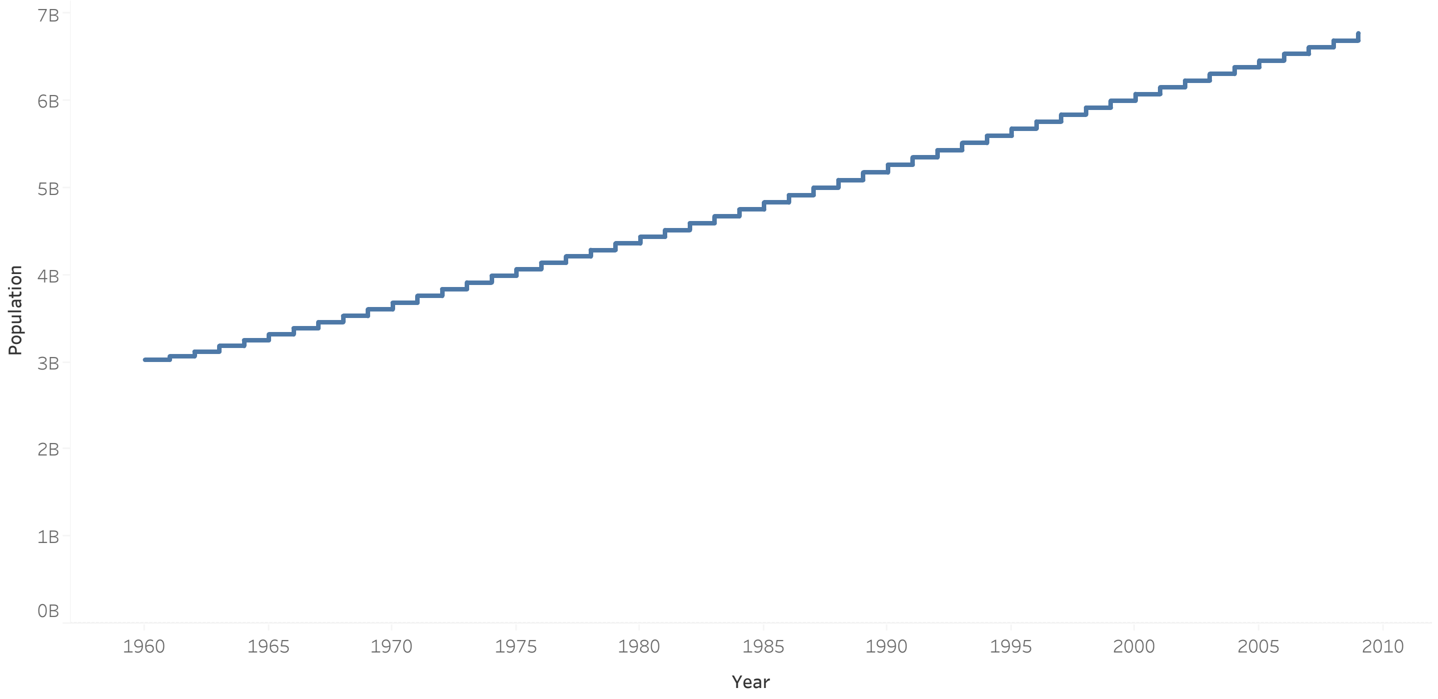
The plot of sum of Population for Year.

Tableau - Stacked Area Chart for Unemployment by Year for Each Period



The plot of sum of Value for Year. Color shows details about Period.

Tableau - Step Chart for World Population by Year



The trend of sum of Population for Year.