Python Plots

In [1]:
```
pip install squarify
```

Requirement already satisfied: squarify in c:\users\meena\anaconda3\lib\site-packages
(0.4.3)
Note: you may need to restart the kernel to use updated packages.

In [2]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import squarify
from datetime import datetime as dt
```

In [3]:
```python
df = pd.read_csv("unemployement-rate-1948-2010.csv")
```

In [4]:
```python
df
```

Out[4]:

	Series id	Year	Period	Value
0	LNS14000000	1948	M01	3.4
1	LNS14000000	1948	M02	3.8
2	LNS14000000	1948	M03	4.0
3	LNS14000000	1948	M04	3.9
4	LNS14000000	1948	M05	3.5
...	...	...	...	...
741	LNS14000000	2009	M10	10.1
742	LNS14000000	2009	M11	10.0
743	LNS14000000	2009	M12	10.0
744	LNS14000000	2010	M01	9.7
745	LNS14000000	2010	M02	9.7

746 rows × 4 columns

In [5]:
```python
df1 = pd.read_csv("expenditures.txt", sep = '\t', header=0)
```

In [6]:
```python
df1
```

Out[6]:

	year	category	expenditure	sex
0	2008	Food	6443	1

|	year	category	expenditure	sex
1	2008	Alcoholic Beverages	444	1
2	2008	Housing	17109	1
3	2008	Apparel	1801	1
4	2008	Transportation	8604	1
...	...	...	...	...
345	1984	Education	303	1
346	1984	Tobacco Products	228	1
347	1984	Miscellaneous	451	1
348	1984	Cash Contributions	706	1
349	1984	Personal Insurance	1897	1

350 rows × 4 columns

In [7]:
```python
# Calculate total expenditure for categories
expend_cat = df1.groupby(['category'])['expenditure'].sum().reset_index()

# Calculate total expenditure by year
expend_year = df1.groupby(['year'])['expenditure'].sum().reset_index()
```

Python- Tree Map

In [8]:
```python
squarify.plot(sizes=expend_cat['expenditure'], label=expend_cat['category'], alpha=0.6)
plt.axis('on')
plt.show()
```
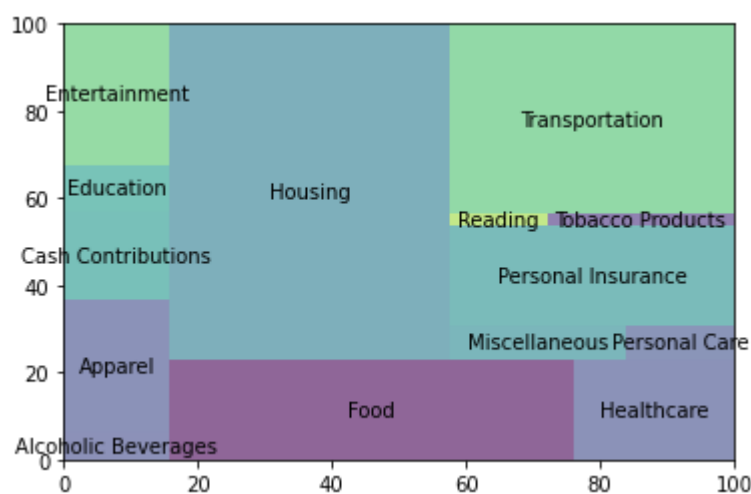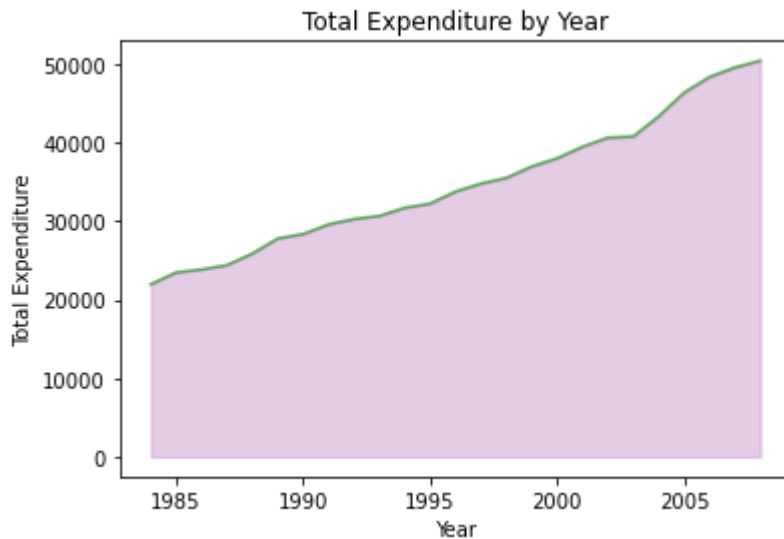

Python-Area chart

In [9]:
```python
a = expend_year['year']
b = expend_year['expenditure']
```

```python
plt.fill_between( a, b, color='purple', alpha=0.2)
plt.title('Total Expenditure by Year')
plt.xlabel('Year')
plt.ylabel('Total Expenditure')
plt.plot(a, b, color='green', alpha=0.6)
```

Out[9]:   [<matplotlib.lines.Line2D at 0x24380a53d00>]

Python- Stacked Area chart

In [10]:
```python
df2 = df1.loc[:, df1.columns != 'sex'].pivot(index='year', columns='category', values='
df2.reset_index(level=0, inplace=True)

# Draw Plot and Annotate
fig, ax = plt.subplots(1,1,figsize=(16, 9), dpi= 80)
columns = df2.columns[1:]
labs = df2.values.tolist()

# Prepare data
x  = df2['year'].values.tolist()
y0 = df2[columns[0]].values.tolist()
y1 = df2[columns[1]].values.tolist()
y2 = df2[columns[2]].values.tolist()
y3 = df2[columns[3]].values.tolist()
y4 = df2[columns[4]].values.tolist()
y5 = df2[columns[5]].values.tolist()
y6 = df2[columns[6]].values.tolist()
y7 = df2[columns[7]].values.tolist()
y8 = df2[columns[8]].values.tolist()
y9 = df2[columns[9]].values.tolist()
y10 = df2[columns[10]].values.tolist()
y11 = df2[columns[11]].values.tolist()
y12 = df2[columns[12]].values.tolist()

y = np.vstack([y0, y1, y2, y3, y4, y5, y6, y7, y8, y9, y10, y11, y12])

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

```python
# Create title
ax.set_title('Total Expenditure by year - category wise', fontsize=18)
plt.xlabel('Year')
plt.ylabel('Total Expenditure')

# 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()
```


Total Expenditure by year - category wise