1. Importing Libraries & Loading the Data

To start off with the analysis project the first step is indeed to load the dataset. We do this by firstly importing the necessary libraries below, among which we will use the pandas library to load and read the dataset.

In [47]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [34]:
```
import os
for dirname, _, filenames in os.walk('C:/Users/admin/Desktop/DATA Projrct'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
C:/Users/admin/Desktop/DATA Projrct\Nutrical Dataset.csv
```

In [14]:
```
data = pd.read_csv ("C:/Users/admin/Desktop/DATA Projrct/Nutrical Dataset.csv")

data.head(2)
```

Out[14]:

	Category	Item	Serving Size	Calories	Calories from Fat	Total Fat	Total Fat (% Daily Value)	Saturated Fat	Saturated Fat (% Daily Value)
0	Breakfast	Egg McMuffin	4.8 oz (136 g)	300	120	13.0	20	5.0	25
1	Breakfast	Egg White Delight	4.8 oz (135 g)	250	70	8.0	12	3.0	15

2 rows × 24 columns

In [42]:
```
data.shape
```

Out[42]:
```
(260, 25)
```

After loading the dataset, we used the .shape() function to determine the structure of the dataframe, which contains 260 menu items distributed across various categories. The dataset includes 22 columns representing different nutritional variables, which will assist us in conducting a detailed analysis.

2. Data Preprocessing

In this step, we will examine the dataset structure and assess whether the data types are appropriate for the given data. Additionally, we will check for any missing values in the dataset.

In [44]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 260 entries, 0 to 259
Data columns (total 25 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Category                        260 non-null    object
 1   Item                            260 non-null    object
 2   Serving Size                    260 non-null    object
 3   Calories                        260 non-null    int64
 4   Calories from Fat               260 non-null    int64
 5   Total Fat                       260 non-null    float64
 6   Total Fat (% Daily Value)       260 non-null    int64
 7   Saturated Fat                   260 non-null    float64
 8   Saturated Fat (% Daily Value)   260 non-null    int64
 9   Trans Fat                       260 non-null    float64
 10  Cholesterol                     260 non-null    int64
 11  Cholesterol (% Daily Value)     260 non-null    int64
 12  Sodium                          260 non-null    int64
 13  Sodium (% Daily Value)          260 non-null    int64
 14  Carbohydrates                   260 non-null    int64
 15  Carbohydrates (% Daily Value)   260 non-null    int64
 16  Dietary Fiber                   260 non-null    int64
 17  Dietary Fiber (% Daily Value)   260 non-null    int64
 18  Sugars                          260 non-null    int64
 19  Protein                         260 non-null    int64
 20  Vitamin A (% Daily Value)       260 non-null    int64
 21  Vitamin C (% Daily Value)       260 non-null    int64
 22  Calcium (% Daily Value)         260 non-null    int64
 23  Iron (% Daily Value)            260 non-null    int64
 24  saturated_cholesterol           222 non-null    float64
dtypes: float64(4), int64(18), object(3)
memory usage: 50.9+ KB
```

In [45]:
```
data.isnull()
```

Out[45]:

	Category	Item	Serving Size	Calories	Calories from Fat	Total Fat	Total Fat (% Daily Value)	Saturated Fat	Saturated Fat (% Daily Value)
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
255	False	False	False	False	False	False	False	False	False
256	False	False	False	False	False	False	False	False	False
257	False	False	False	False	False	False	False	False	False
258	False	False	False	False	False	False	False	False	False
259	False	False	False	False	False	False	False	False	False

260 rows × 25 columns

In [46]:
```
data.isnull().sum()
```

Out[46]:
```
Category                         0
Item                             0
Serving Size                     0
Calories                         0
Calories from Fat                0
Total Fat                        0
Total Fat (% Daily Value)        0
Saturated Fat                    0
Saturated Fat (% Daily Value)    0
Trans Fat                        0
Cholesterol                      0
Cholesterol (% Daily Value)      0
Sodium                           0
Sodium (% Daily Value)           0
Carbohydrates                    0
Carbohydrates (% Daily Value)    0
Dietary Fiber                    0
Dietary Fiber (% Daily Value)    0
Sugars                           0
Protein                          0
Vitamin A (% Daily Value)        0
Vitamin C (% Daily Value)        0
Calcium (% Daily Value)          0
Iron (% Daily Value)             0
saturated_cholesterol           38
dtype: int64
```

From the above code, we can gain a clear understanding of the dataset's basic structure. By identifying the number of rows and columns, as well as reviewing the data types assigned to each header, we can get an overview of the dataset's structure and orientation. Additionally, I checked for any missing values and verified whether the data types for each column were appropriate. As seen from the results of the .info() and .isnull() functions, there were no null values present, indicating that the dataset is clean and contains no missing data.

3. Exploratory Data Analysis

Based on the detailed Nutrical dataset for McDonald's menu items, the exploratory data analysis (EDA) will help uncover insights into the nutritional values across the entire menu.

A. Analyze the distribution of calorie counts across menu items.

In [52]:
```
data.head()
```

Out[52]:

	Category	Item	Serving Size	Calories	Calories from Fat	Total Fat	Total Fat (% Daily Value)	Saturated Fat	Saturated Fat (% Daily Value)
0	Breakfast	Egg McMuffin	4.8 oz (136 g)	300	120	13.0	20	5.0	25
1	Breakfast	Egg White Delight	4.8 oz (135 g)	250	70	8.0	12	3.0	15
2	Breakfast	Sausage McMuffin	3.9 oz (111 g)	370	200	23.0	35	8.0	42
3	Breakfast	Sausage McMuffin with Egg	5.7 oz (161 g)	450	250	28.0	43	10.0	52
4	Breakfast	Sausage McMuffin with Egg Whites	5.7 oz (161 g)	400	210	23.0	35	8.0	42

5 rows × 25 columns

a. Calories

In [54]:
```
Calories = data['Calories'].describe()
Calories
```

Out[54]:
```
count     260.000000
mean      368.269231
std       240.269886
min         0.000000
25%       210.000000
50%       340.000000
75%       500.000000
max      1880.000000
Name: Calories, dtype: float64
```

import matplotlib.pyplot as plt

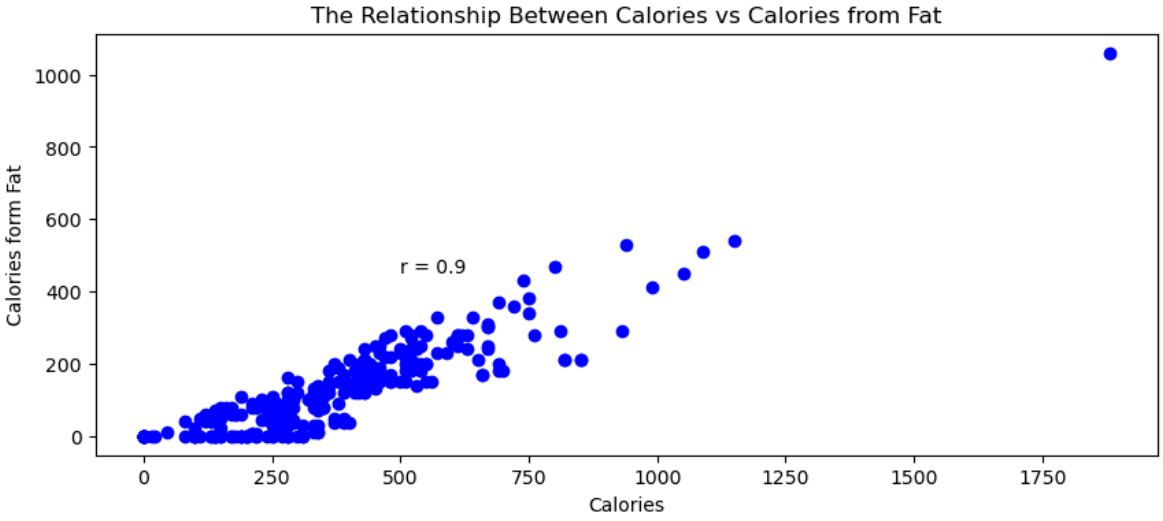
menu_category = data['Category'].value_counts()

plt.figure(figsize=(10, 4), dpi=100) menu_category.plot.bar(color=['green', 'blue', 'blue', 'blue', 'blue'])

plt.title("Number of Menu Items for Each Food Category") plt.ylabel("Number of Items") plt.xlabel("Category") plt.xticks(rotation=45)

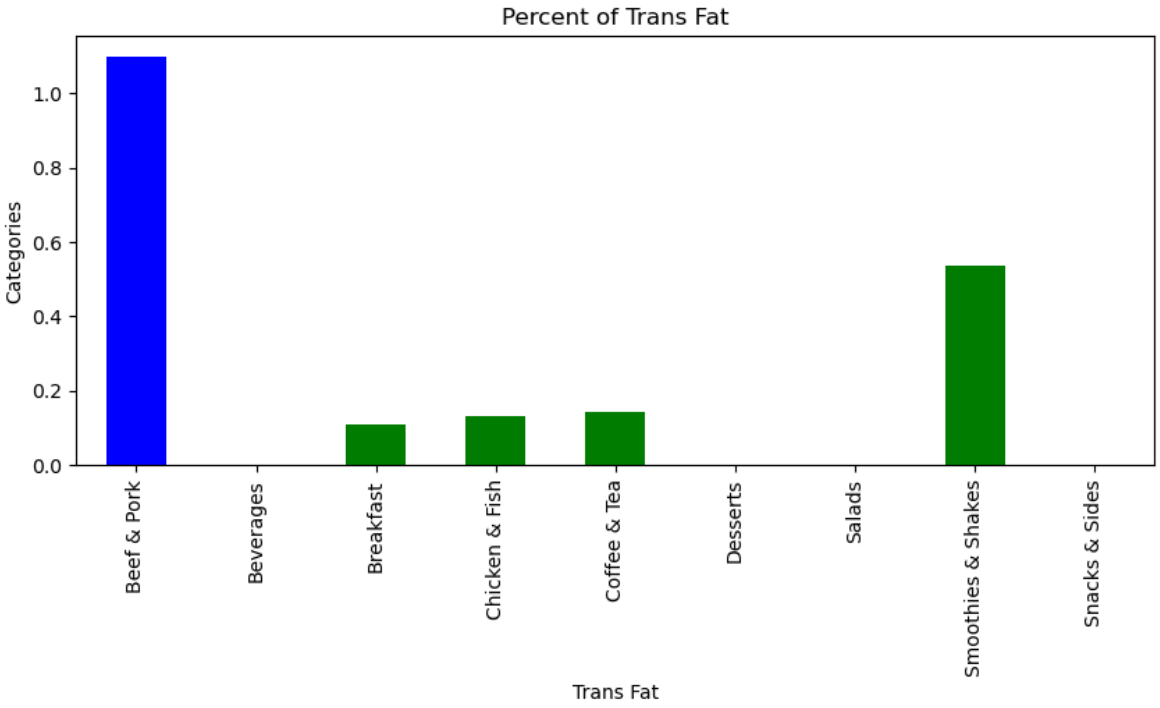
plt.show()

In [18]:
```
plt.figure(figsize=(10, 4), dpi=100)

correlation = data['Calories'].corr(data['Calories from Fat'])

plt.scatter(data.Calories, data['Calories from Fat'], color='blue')

plt.text(500,450, 'r = {}'.format(round(correlation,2)))
plt.xlabel("Calories")
plt.ylabel("Calories form Fat")
plt.title("The Relationship Between Calories vs Calories from Fat")

plt.show()
```



In [19]:
```
plt.figure (figsize=(10, 4), dpi=100)

menu_category = data.groupby('Category')['Trans Fat'].mean()

menu_category.plot.bar(color = ['blue','green','green','green','green','green','

plt.title("Percent of Trans Fat")
```

```
plt.xlabel("Trans Fat")
plt.ylabel("Categories")
plt.xticks(rotation=90)

plt.show()
```



In [20]:
```
data.groupby('Category')['Trans Fat'].mean()
```

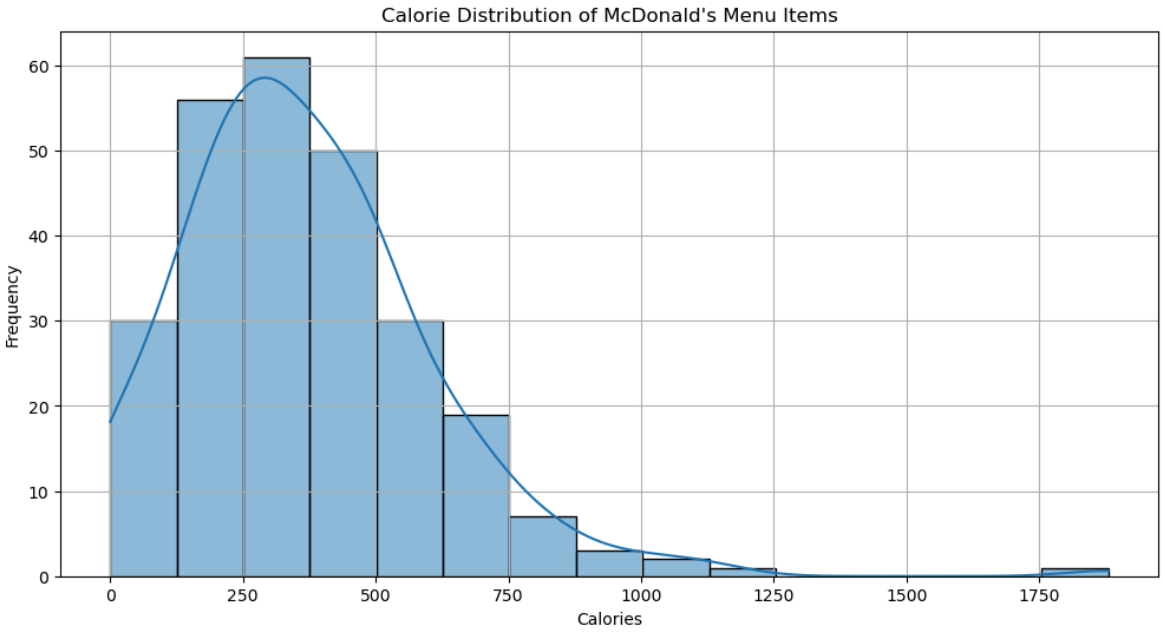
Out[20]:
```
Category
Beef & Pork           1.100000
Beverages             0.000000
Breakfast             0.107143
Chicken & Fish        0.129630
Coffee & Tea          0.142105
Desserts              0.000000
Salads                0.000000
Smoothies & Shakes    0.535714
Snacks & Sides        0.000000
Name: Trans Fat, dtype: float64
```

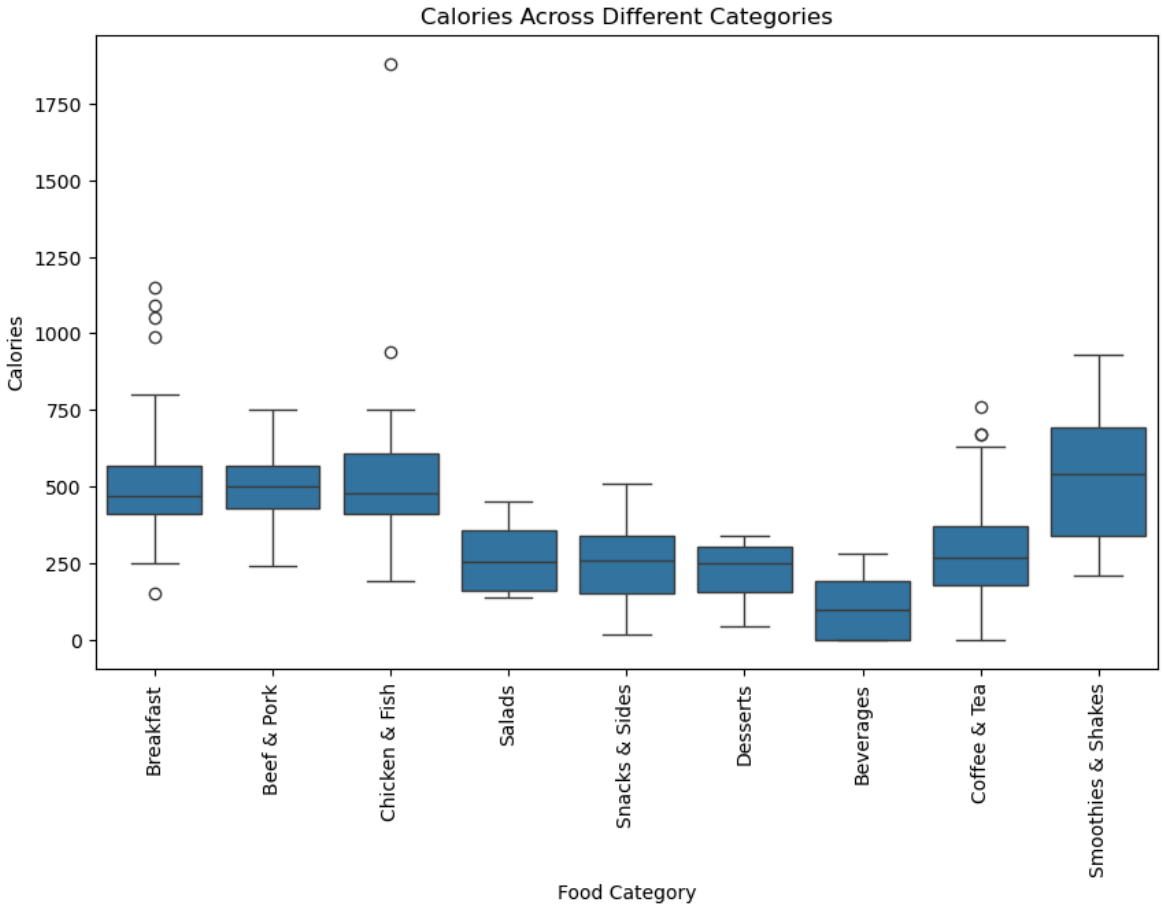
In [21]:
```
data['saturated_cholesterol'] = data['Saturated Fat']/data['Cholesterol']*100
```

In [22]:
```
saturated_cholesterol = data.groupby('Category')['saturated_cholesterol'].mean()
```

In [55]:
```
plt.figure(figsize=(12, 6))
sns.histplot(data=data, x= 'Calories', bins=15, kde=True)
plt.title('Calorie Distribution of McDonald\'s Menu Items')
plt.xlabel('Calories')
plt.ylabel('Frequency')
plt.grid()
plt.show()

plt.figure(figsize=(10, 6))
sns.boxplot(x='Category', y='Calories', data=data)
plt.title('Calories Across Different Categories')
plt.xlabel('Food Category')
plt.ylabel('Calories')
```

```
plt.xticks(rotation=90)
plt.show()
```





Descriptive Statistics -

The mean calorie count is 368.27 calories. The standard deviation is 240 calories, indicating the widespread in the calorie counts. 25% of the items have 210 or less calories, 50% of items have 340 or less calories & 75% of items have 500 or less calories. Visually analyzing the distribution, we can observe that most items fall within the 125-

500 calorie range, causing the distribution to be heavily skewed to the right, with a few outliers at both the lower and higher ends.

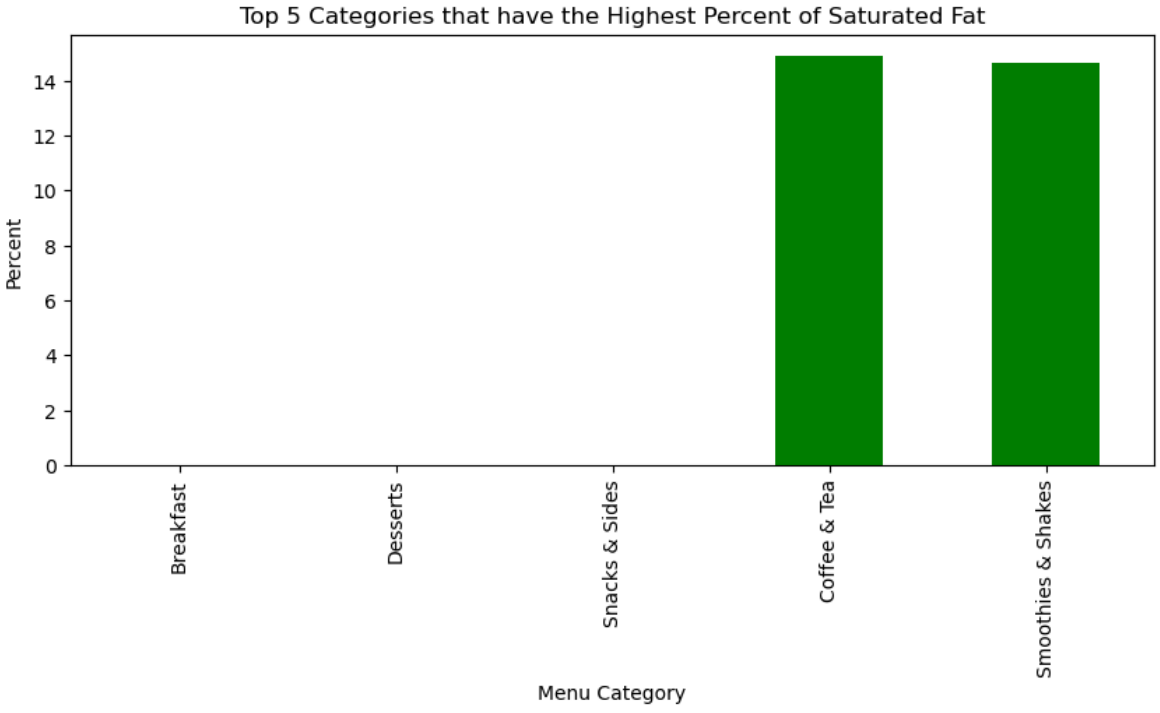
B. Explore the nutritional content (e.g., fat, protein, carbohydrates) of different items.

1. Total Fat(grams)

In [56]:
```
Total_fat=data['Total Fat'].describe()
Total_fat
```

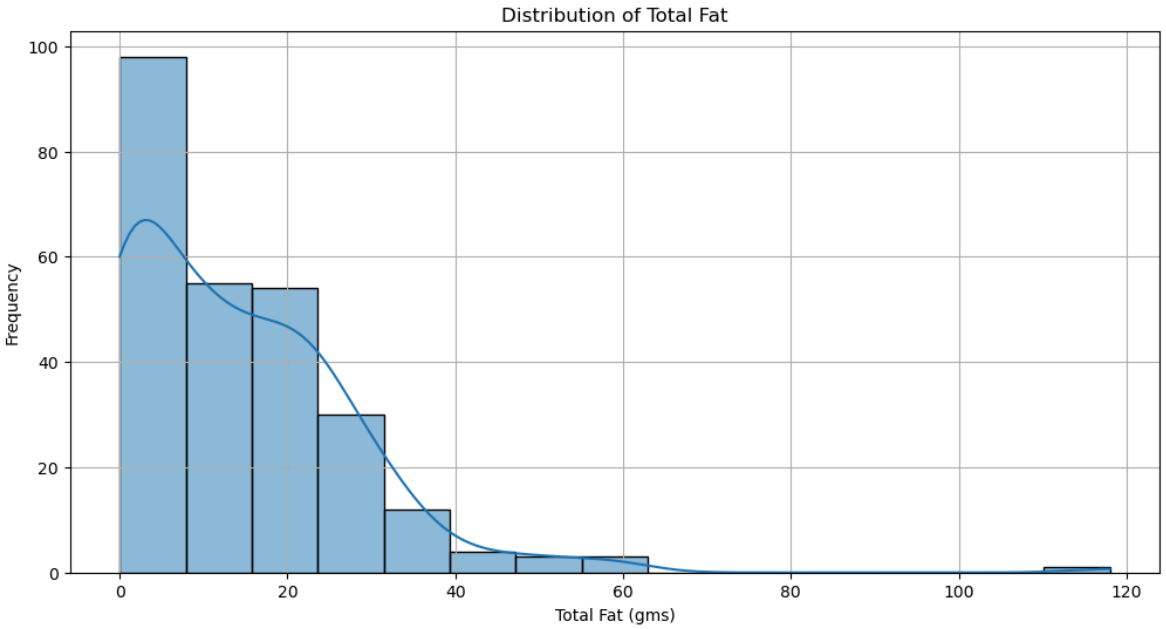
Out[56]:
```
count    260.000000
mean      14.165385
std       14.205998
min        0.000000
25%        2.375000
50%       11.000000
75%       22.250000
max      118.000000
Name: Total Fat, dtype: float64
```

In [23]:
```
plt.figure(figsize=(10, 4), dpi=100)

saturated_cholesterol.sort_values(ascending=False).plot.bar(color = 'green')

plt.title("Top 5 Categories that have the Highest Percent of Saturated Fat")
plt.ylabel("Percent")
plt.xlabel("Menu Category")
plt.xticks(rotation=90)

plt.show();
```



In [58]:
```
plt.figure(figsize=(12,6))
sns.histplot(data=data, x='Total Fat',bins=15, kde = True)
plt.title('Distribution of Total Fat')
```

```
plt.xlabel('Total Fat (gms)')
plt.ylabel('Frequency')
plt.grid()
plt.show()
```



Descriptive Statistics -

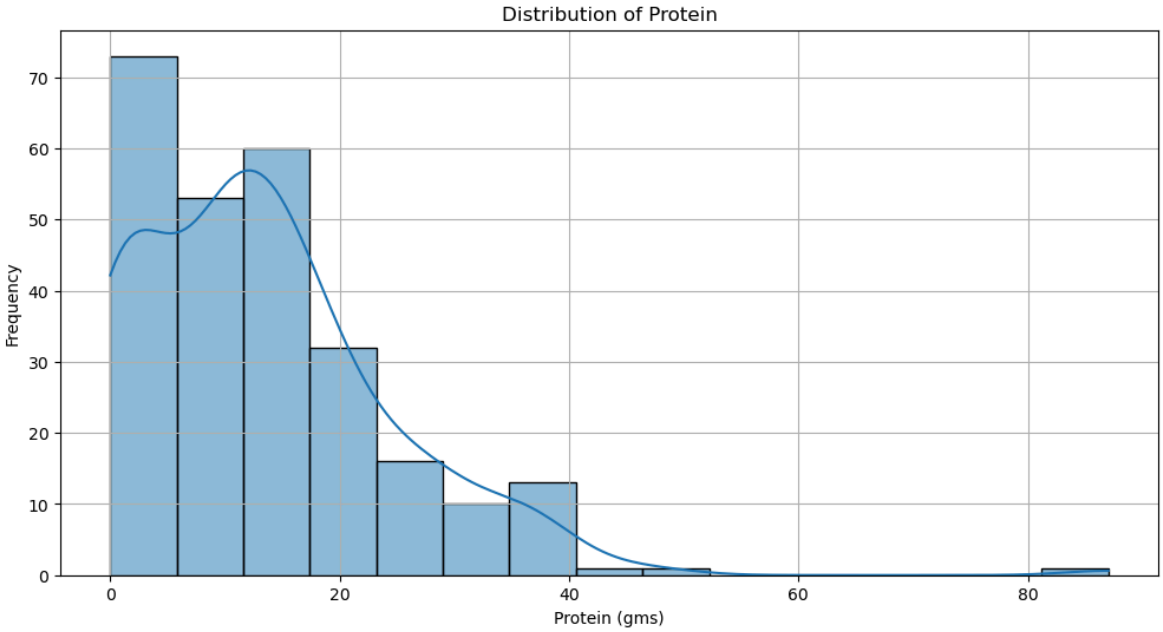
The mean of the Total fat content is 14.16 grams. The standard deviation is 14.2 grams, indicating the moderate spread in the total fat values. 25% of the items have 2.37 grams or less total fat, 50% of items have 11 grams or less total fat & 75% of items have 22.25 grams or less total fat.

2 Proteins (grams)

In [59]:
```
Proteins =data['Protein'].describe()
Proteins
```

Out[59]:
```
count    260.000000
mean      13.338462
std       11.426146
min        0.000000
25%        4.000000
50%       12.000000
75%       19.000000
max       87.000000
Name: Protein, dtype: float64
```

In [60]:
```
plt.figure(figsize=(12,6))
sns.histplot(data=data, x='Protein',bins=15, kde = True )
plt.title('Distribution of Protein')
plt.xlabel('Protein (gms)')
plt.ylabel('Frequency')
plt.grid()
plt.show()
```



Descriptive Statistics -

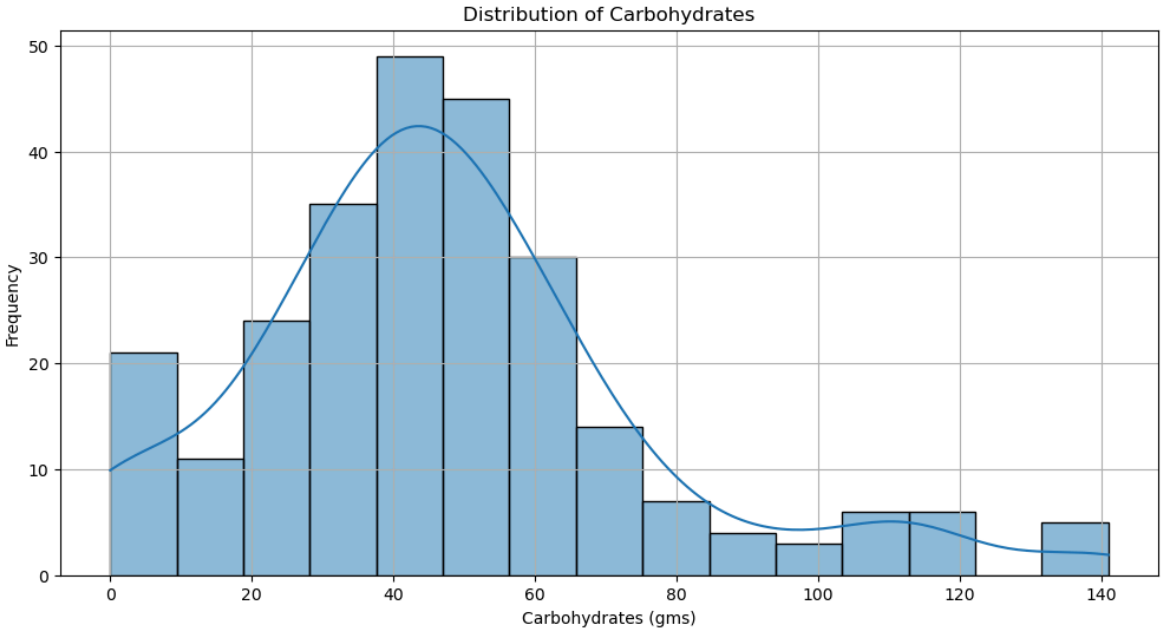
The mean of the Protein content is 13.33 grams. The standard deviation is 11.43 grams, indicating the moderate spread in the protein values. 25% of the items have 12 grams or less proteins, 50% of items have 19 grams or less proteins & 75% of items have 87 grams or less proteins.

3 Carbohydrates (grams)

In [61]:
```
Carbohydrates= data['Carbohydrates'].describe()
Carbohydrates
```

Out[61]:
```
count    260.000000
mean      47.346154
std       28.252232
min        0.000000
25%       30.000000
50%       44.000000
75%       60.000000
max      141.000000
Name: Carbohydrates, dtype: float64
```

In [74]:
```
plt.figure(figsize=(12,6))
sns.histplot(data=data, x='Carbohydrates',bins=15, kde = True )
plt.title('Distribution of Carbohydrates')
plt.xlabel('Carbohydrates (gms)')
plt.ylabel('Frequency')
plt.grid()
plt.show()
```



Descriptive Statistics -

The mean of the Carbohydrates content is 47.34 grams. The standard deviation is 28.25 grams, indicating the widespread in the carbohydrates values. 25% of the items have 30 grams or less carbohydrates, 50% of items have 44 grams or less carbohydrates & 75% of items have 60 grams or less carbohydrates.

C. Identify trends and patterns in the dataset.

In this step, we will examine how different nutritional variables across the items correlate with each other by visualizing the correlations using pair plots and heatmaps.

In [65]:
```
nutritional_vars = ['Calories','Total Fat', 'Cholesterol','Carbohydrates','Prote
nutritional_vars
```

Out[65]:
```
['Calories', 'Total Fat', 'Cholesterol', 'Carbohydrates', 'Proteins']
```

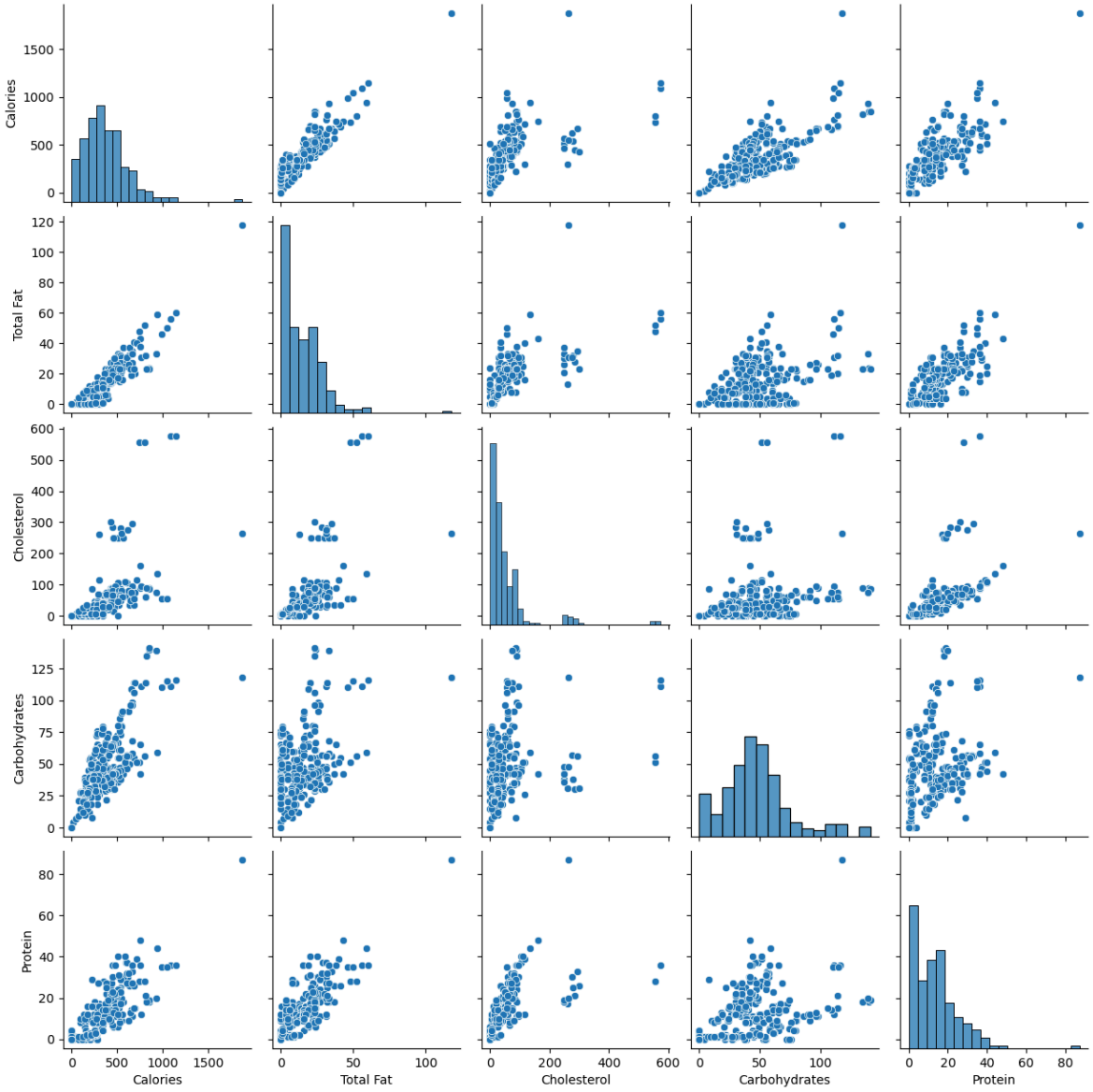
In [67]:
```
calories_vs_nutrients = data[['Calories','Total Fat', 'Cholesterol','Carbohydrat
print(calories_vs_nutrients)
```

```
             Calories  Total Fat  Cholesterol  Carbohydrates   Protein
Calories     1.000000   0.904409     0.596399       0.781539  0.787847
Total Fat    0.904409   1.000000     0.680547       0.461213  0.807773
Cholesterol  0.596399   0.680547     1.000000       0.270977  0.561561
Carbohydrates 0.781539  0.461213     0.270977       1.000000  0.352122
Protein      0.787847   0.807773     0.561561       0.352122  1.000000
```

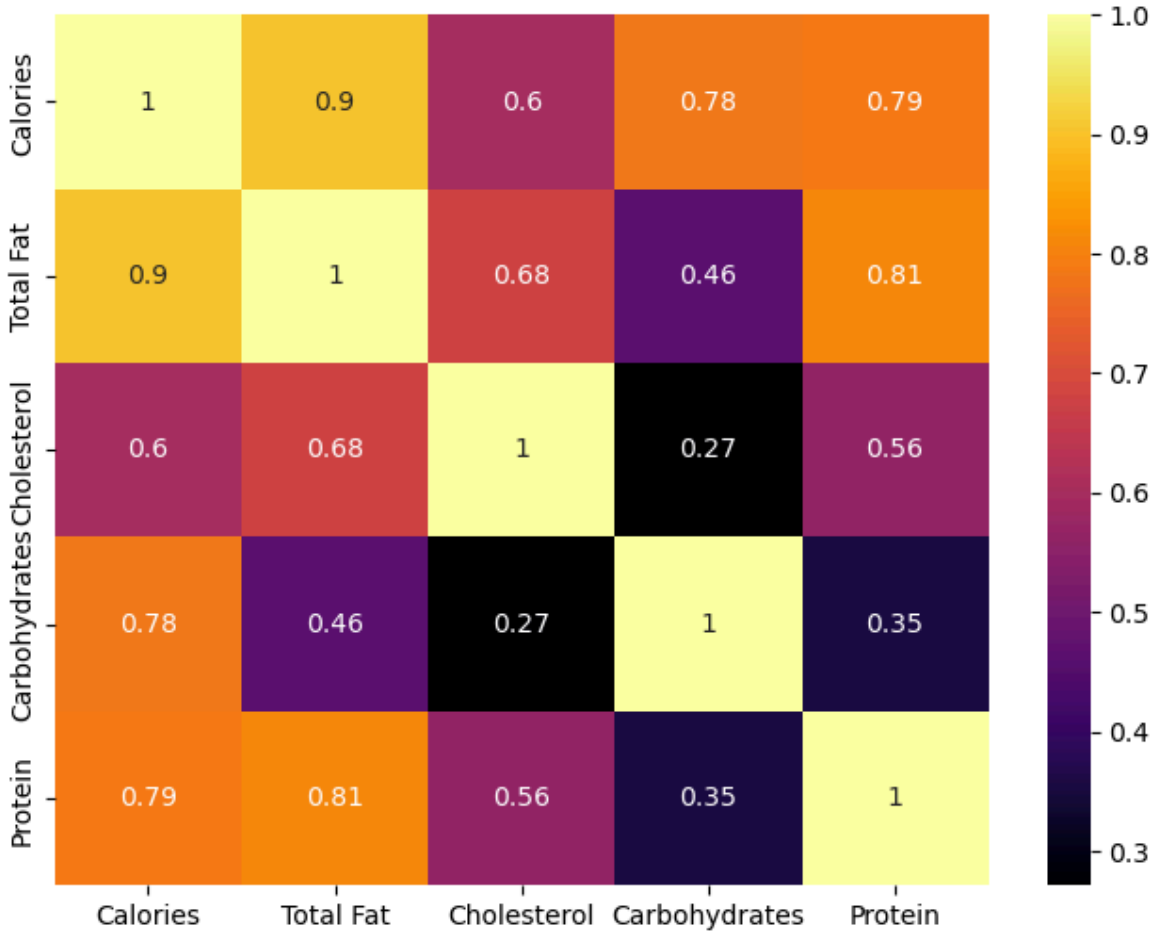
The above code provides a numerical matrix showing how the other nutrients correlate with calories, but it doesn't give us a clear visual understanding. Therefore, we will rely on visualizing the correlation matrix and plot the correlations for better insight.

In [71]:
```
sns.pairplot(data[['Calories','Total Fat', 'Cholesterol','Carbohydrates','Protei
```

Out[71]:
```
<seaborn.axisgrid.PairGrid at 0x1a26b07d730>
```



In [72]:
```
plt.figure(figsize=(8,6))
sns.heatmap(calories_vs_nutrients, annot=True, cmap='inferno')
plt.show()
```



The analysis reveals the following trends and patterns:

Calories and Total Fat: There is a strong positive correlation (0.904) between calorie count and total fat content, suggesting that menu items with higher calorie counts tend to have higher total fat.

Total Fat and Protein: The correlation between total fat and protein is also high (0.807), indicating that items rich in protein are likely to contain more total fat.

Calories and Protein: The positive correlation (0.787) between calories and protein suggests that menu items with more calories generally contain more protein.

Calories and Carbohydrates: A positive correlation (0.781) between calories and carbohydrates indicates that items with higher calorie counts tend to have higher carbohydrate values.

To summarize the correlation matrix, I have included a heatmap that visually represents the relationships between these nutrients in a simpler manner.

4. Data Visualization

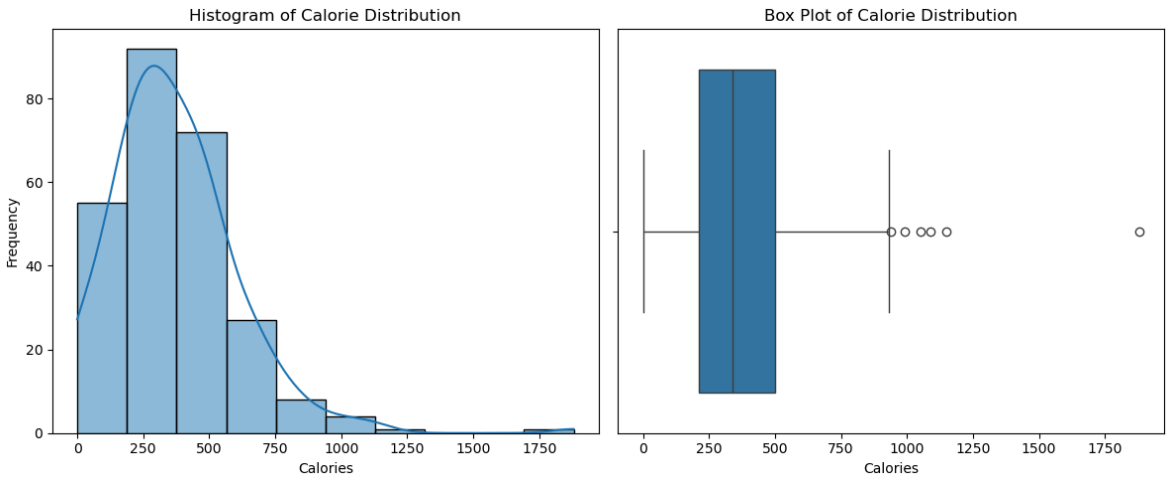
To gain insights into the calorie distribution and nutritional content of these items, I will create several data visualizations. First, I will generate a histogram and a box plot to illustrate the distribution of calorie counts across the menu items. This will help us understand the range of calorie values, identify any outliers or skewness in the data, and

provide an overview of the calorie distribution. Next, I will create a series of bar charts to compare the nutritional characteristics (total fat, saturated fat, carbohydrates, and protein) across different food categories, such as burgers, salads, and desserts. This will highlight any significant differences in the nutrient profiles of these food groups. Through these visualizations, we will gain a comprehensive understanding of the calorie and nutrient composition of the menu items, which can aid in consumer decision-making and inform future menu development.

A. Distribution of Calories

The histogram and boxplot for the distribution can be plotted together using a subplot, allowing us to compare both visuals side by side. This will provide more comprehensive insights into the distribution and help identify any outliers."

In [75]:
```
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
sns.histplot(data['Calories'], bins=10, kde=True)
plt.title('Histogram of Calorie Distribution')
plt.xlabel('Calories')
plt.ylabel('Frequency')

plt.subplot(1, 2, 2)
sns.boxplot(x=data['Calories'])
plt.title('Box Plot of Calorie Distribution')
plt.xlabel('Calories')

plt.tight_layout()
plt.show()
```



The histogram shows a right-skewed distribution, with a peak around 125-500 calories and a long tail extending towards higher calorie values. Most menu items are concentrated at the lower end of the calorie range, with only a few high-calorie outliers. These observations are confirmed by the box plot, where the median is approximately 340 calories, and the 25th and 75th percentiles are 210 and 500 calories, respectively. The box plot also highlights several outliers, including high-calorie items, starting with the 'McFlurry with Reese's Peanut Butter Cups (Medium)' at 810 calories.

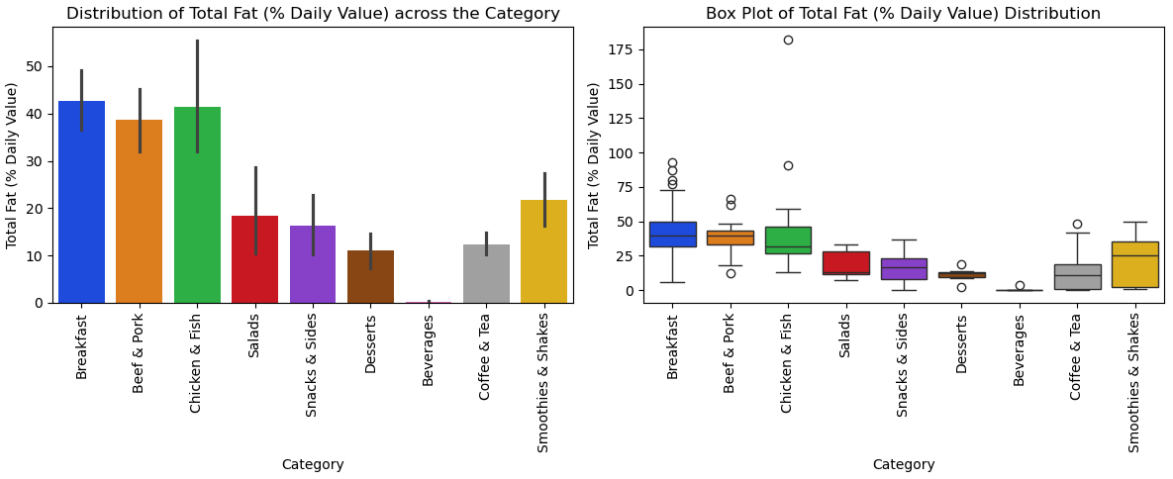
B. Nutritional Content Comparison

With the help of several bar-charts and boxplots we compare the nutritional characteristics of different food categories. As the count of nutrients is 22(which will not be as feasible), we create the visualizations for 4 selected list of nutrients namely,

Total Fat (% Daily Value) Cholesterol (% Daily Value) Sugars Protein

1 Total Fat (% Daily Value)

In [76]:
```
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
sns.barplot(data= data, x= 'Category', y= 'Total Fat (% Daily Value)', hue='Cate
plt.title('Distribution of Total Fat (% Daily Value) across the Category')
plt.xlabel('Category')
plt.ylabel('Total Fat (% Daily Value)')
plt.xticks(rotation= 90)

plt.subplot(1, 2, 2)
sns.boxplot(data= data, x= 'Category', y='Total Fat (% Daily Value)', hue='Categ
plt.title('Box Plot of Total Fat (% Daily Value) Distribution')
plt.xlabel('Category')
plt.ylabel('Total Fat (% Daily Value)')
plt.xticks(rotation= 90)
plt.tight_layout()
plt.show()
```



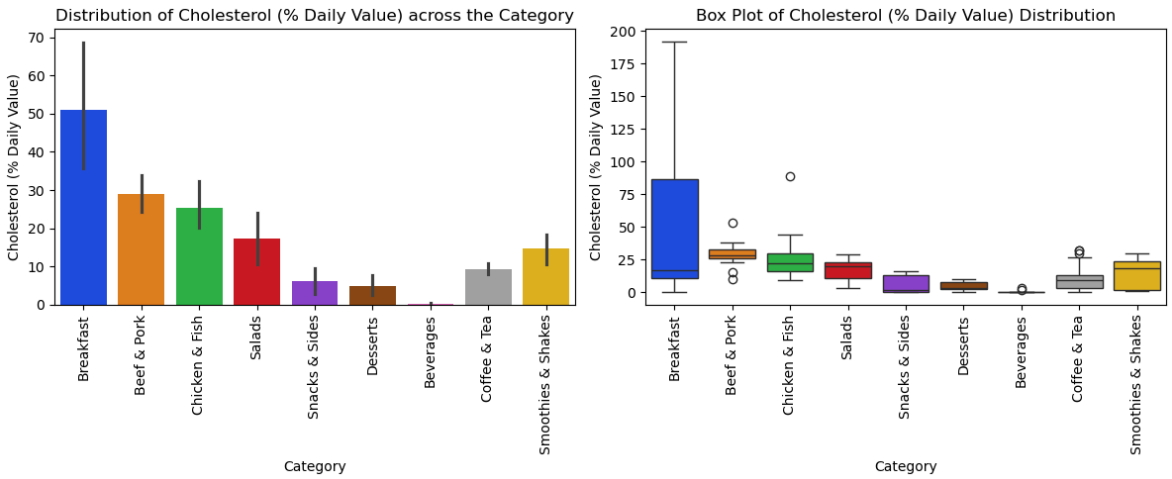
1. From the bar plot: The bar plot reveals that the Breakfast category has the highest average Total Fat (% Daily Value), at around 43%, followed closely by the Chicken & Fish category at approximately 40%. In contrast, the Beverages category has the lowest average Total Fat (% Daily Value), showing negligible values.

2. From the box plot (for the top two categories with the highest average Total Fat (% Daily Value)):

1. Breakfast: The box plot shows a relatively even distribution of Total Fat (% Daily Value) across breakfast items, with a median of 43%. The 25th percentile is around 30%, and the 75th percentile is about 50%. Several outliers are observed with higher Total Fat (% Daily Value) values.
2. Chicken & Fish: The box plot reveals a skewed distribution of Total Fat (% Daily Value) across chicken and fish items, with a median of about 40%. The 25th percentile is roughly 28%, and the 75th percentile is around 48%. A few outliers are present, reflecting higher Total Fat (% Daily Value) values.

2 Cholestrol (% Daily Value)

In [77]:
```
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
sns.barplot(data= data, x= 'Category', y= 'Cholesterol (% Daily Value)', hue='Ca
plt.title('Distribution of Cholesterol (% Daily Value) across the Category')
plt.xlabel('Category')
plt.ylabel('Cholesterol (% Daily Value)')
plt.xticks(rotation= 90)

plt.subplot(1, 2, 2)
sns.boxplot(data= data, x= 'Category', y='Cholesterol (% Daily Value)', hue='Cat
plt.title('Box Plot of Cholesterol (% Daily Value) Distribution')
plt.xlabel('Category')
plt.ylabel('Cholesterol (% Daily Value)')
plt.xticks(rotation= 90)
plt.tight_layout()

plt.show()
```



1. From the bar plot: The Breakfast category has the highest average Cholesterol (% Daily Value) at around 50%, followed by the Beef & Pork category at about 30%, and the Chicken & Fish category at around 25%. The Beverages category once again stands out as having the lowest cholesterol content, with very minimal cholesterol values.

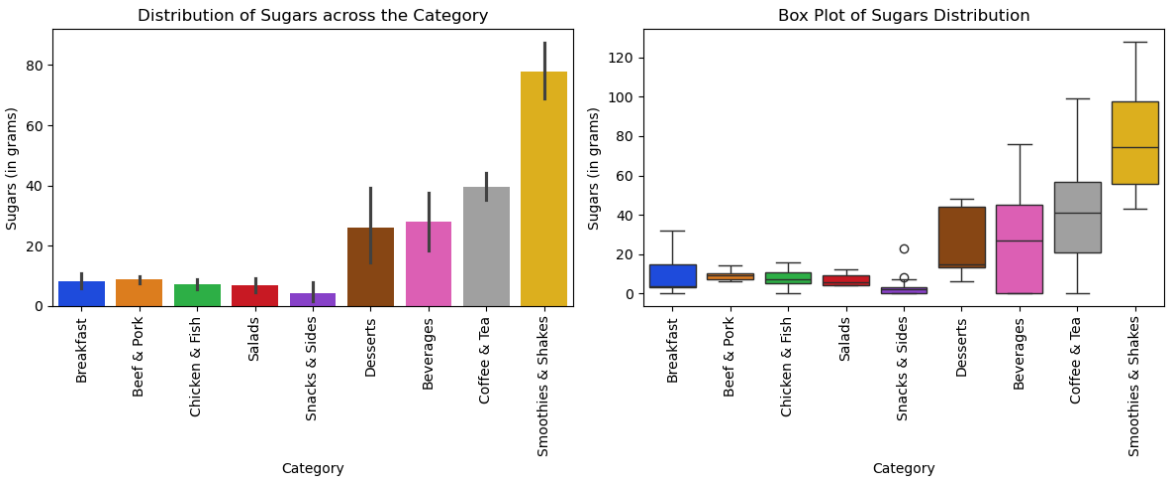
2. From the box plot (for the top two categories with the highest average Cholesterol (% Daily Value)):

1. Breakfast: The Breakfast category shows a broader range of cholesterol values, with some items reaching nearly 90% of the daily value. The median cholesterol level is also higher compared to other categories.
2. Beef & Pork: The Beef & Pork category exhibits a more compact distribution, with most items falling between 25-35% of the daily cholesterol value.

Additionally, there are a few outliers in both the Chicken & Fish and Beef & Pork categories, indicating that certain items in these groups have exceptionally high cholesterol content

3 Sugar (in grams)

In [78]:
```
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
sns.barplot(data= data, x= 'Category', y= 'Sugars', hue='Category', palette= 'br
plt.title('Distribution of Sugars across the Category')
plt.xlabel('Category')
plt.ylabel('Sugars (in grams)')
plt.xticks(rotation= 90)

plt.subplot(1, 2, 2)
sns.boxplot(data= data, x= 'Category', y='Sugars',hue='Category', palette= 'brig
plt.title('Box Plot of Sugars Distribution')
plt.xlabel('Category')
plt.ylabel('Sugars (in grams)')
plt.xticks(rotation= 90)
plt.tight_layout()

plt.show()
```



1. From the bar plot:

1. The Smoothies & Shakes category has the highest average sugar content, with around 80 grams of sugar per serving.
2. The Coffee & Tea category comes next, with an average of about 40 grams of sugar per serving.

3. The Snacks & Sides category has the lowest average sugar content, with around 8 grams of sugar per serving.

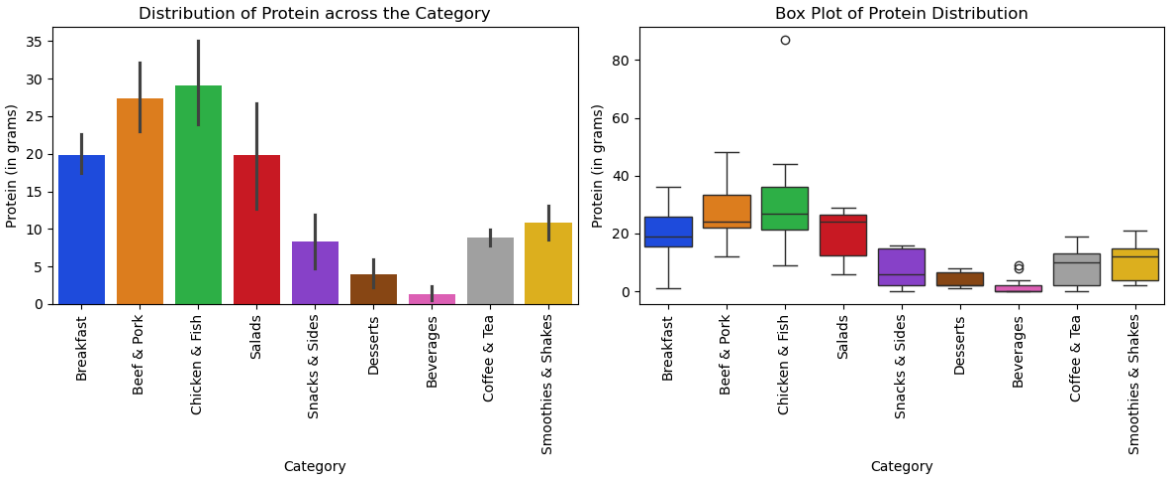
2. From the box plot (for the top two categories with the highest average Sugars content in grams):

1. The Smoothies & Shakes category displays the widest range of sugar content, with some items reaching up to 100 grams of sugar per serving.
2. The Coffee & Tea category also shows a broad range, with some items reaching up to 60 grams of sugar per serving.

Aside from the Snacks & Sides category, which ironically has the lowest sugar content, there are no significant outliers for sugar content across the other categories.

4 Protein (in grams)

In [84]:
```
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
sns.barplot(data= data, x= 'Category', y= 'Protein', hue='Category', palette= 'b
plt.title('Distribution of Protein across the Category')
plt.xlabel('Category')
plt.ylabel('Protein (in grams)')
plt.xticks(rotation= 90)

plt.subplot(1, 2, 2)
sns.boxplot(data= data, x= 'Category', y='Protein', hue='Category', palette= 'br
plt.title('Box Plot of Protein Distribution')
plt.xlabel('Category')
plt.ylabel('Protein (in grams)')
plt.xticks(rotation= 90)
plt.tight_layout()

plt.show()
```



1. From the bar plot:

1. The Chicken & Fish category has the highest average protein content, with around 29 grams of protein per serving.
2. The Beef & Pork category follows with the second-highest average protein content, at around 27 grams per serving.

3. The Beverages category has the lowest average protein content, with only 2-3 grams of protein per serving.

2. From the box plot (for the top two categories with the highest average protein content in grams):

1. The Chicken & Fish category shows the widest range of protein content, with some items reaching up to 40 grams of protein per serving.
2. The Beef & Pork category has a more compact distribution compared to Chicken & Fish, with most items falling between 22-37 grams of protein per serving.

There are no significant outliers indicating exceptionally high protein levels across the categories. However, a small portion of outliers is observed in both the Chicken & Fish and Beverages categories, with a limited number of items showing higher protein values.

5. Nutrition-Based Insights

The main goal is to analyze the data to identify the menu items with the highest and lowest calorie counts, as well as determine the average nutritional content of popular menu categories. To achieve this, we will use the .idxmax() and .idxmin() functions to identify the highest and lowest values, and the .describe() function to calculate the average nutritional content across the categories.

A. Identify menu items with the highest and lowest calorie counts.

In [86]:
```
high_calorie_item = data[data['Calories'] == data['Calories'].max()]

print("Highest Calorie Item:")
print(high_calorie_item[['Item', 'Calories']])
```

```
Highest Calorie Item:
                           Item  Calories
82  Chicken McNuggets (40 piece)      1880
```

In [87]:
```
low_calorie_item = data[data['Calories'] == data['Calories'].min()]

print("\nLowest Calorie Item:")
print(low_calorie_item[['Item', 'Calories']])
```

```
Lowest Calorie Item:
                        Item  Calories
114        Diet Coke (Small)         0
115       Diet Coke (Medium)         0
116        Diet Coke (Large)         0
117        Diet Coke (Child)         0
122   Diet Dr Pepper (Small)         0
123  Diet Dr Pepper (Medium)         0
124   Diet Dr Pepper (Large)         0
125   Diet Dr Pepper (Child)         0
136      Dasani Water Bottle         0
137         Iced Tea (Small)         0
138        Iced Tea (Medium)         0
139         Iced Tea (Large)         0
140         Iced Tea (Child)         0
145           Coffee (Small)         0
146          Coffee (Medium)         0
147           Coffee (Large)         0
```

B. Determine the average nutritional content of popular menu categories.

We'll start by making a new dataframe to achieve this task. In this dataframe we will exclude the 'Item' & 'Serving Size' columns as they both contain categorical values and won't be suitable in an aggregation step.

In [89]:
```
columns_to_drop = ['Item', 'Serving Size']
data_new = data.drop(columns_to_drop, axis=1)
data_new.head()
```

Out[89]:

	Category	Calories	Calories from Fat	Total Fat	Total Fat (% Daily Value)	Saturated Fat	Saturated Fat (% Daily Value)	Trans Fat	Cholesterol
0	Breakfast	300	120	13.0	20	5.0	25	0.0	260
1	Breakfast	250	70	8.0	12	3.0	15	0.0	25
2	Breakfast	370	200	23.0	35	8.0	42	0.0	45
3	Breakfast	450	250	28.0	43	10.0	52	0.0	285
4	Breakfast	400	210	23.0	35	8.0	42	0.0	50

5 rows × 23 columns

In [90]:
```
average_nutritional_content = data_new.groupby('Category').mean()
average_nutritional_content
```

Out[90]:

Category	Calories	Calories from Fat	Total Fat	Total Fat (% Daily Value)	Saturated Fat	Saturated Fat (% Daily Value)	Tran Fa
Beef & Pork	494.000000	224.666667	24.866667	38.600000	10.466667	52.000000	1.10000
Beverages	113.703704	0.740741	0.092593	0.148148	0.055556	0.296296	0.00000
Breakfast	526.666667	248.928571	27.690476	42.666667	10.654762	53.428571	0.10714
Chicken & Fish	552.962963	242.222222	26.962963	41.333333	6.166667	31.111111	0.12963
Coffee & Tea	283.894737	71.105263	8.021053	12.357895	4.921053	24.368421	0.14210
Desserts	222.142857	64.285714	7.357143	11.142857	4.285714	21.285714	0.00000
Salads	270.000000	108.333333	11.750000	18.333333	3.750000	18.500000	0.00000
Smoothies & Shakes	531.428571	127.678571	14.125000	21.714286	8.375000	41.785714	0.53571
Snacks & Sides	245.769231	94.615385	10.538462	16.230769	2.692308	13.384615	0.00000

9 rows × 22 columns

Summarizing the Analysis

Findings and Insights:

1. Menu Items Analysis:

1. The dataset includes a variety of menu items categorized into Breakfast, Beef & Pork, Chicken & Fish, Coffee & Tea, and other categories. Items like the "Chicken McNuggets (40 pieces)" from the Chicken & Fish category have the highest calorie counts, while the "Diet Coke (small)" from the Beverages category has the lowest.
2. The Beef & Pork category generally contains higher protein content, while the Breakfast category tends to have higher fat and carbohydrate levels.
3. Other categories like Beverages, Smoothies & Shakes, and Coffee & Tea have comparatively lower calorie, total fat, and carbohydrate values, but show a significant spike in sugar content, with a notably low average protein content.

2. Average Nutritional Content:

1. Average Calories: The average calorie count for Chicken & Fish items is 552 calories, making it one of the most calorie-dense categories.

2. Average Total Fat: The Breakfast and Chicken & Fish categories have higher total fat values, with averages of 27.69 and 26.9 grams, respectively.

3. Average Protein: Chicken & Fish items have the highest average protein content, with an average of 29.11 grams per serving.

4. Average Cholesterol: The Breakfast category has the highest average cholesterol level, calculated at 153 milligrams, compared to other categories.

5. Average Carbohydrates: While protein-rich categories such as Chicken & Fish, Beef & Pork, and Breakfast items typically contain moderate carbohydrate values, the Smoothies & Shakes category has much higher carbohydrate levels, averaging 90.5 grams per serving.

Conclusions

Healthier Options:

Access to detailed nutritional content allows customers to make healthier choices, such as selecting the "Egg White Delight," "Premium Grilled Chicken Classic Sandwich," or "Fruit & Maple Oatmeal without Brown Sugar." These items are lower in calories, fat, and sodium, helping to promote better eating habits and supporting customers in maintaining a balanced diet.

Unhealthy Side of the Table:

Based on the nutritional analysis, some of the less healthy food categories include:

1. Breakfast Category: Items in this category are high in calories, total fat, saturated fat, cholesterol, and sodium. For example, the "Big Breakfast with Hotcakes (Large Biscuit)" contains 1,150 calories, 60 grams of total fat, 20 grams of saturated fat, and 2,260 mg of sodium. Many other breakfast sandwiches and biscuits also contain high levels of these unhealthy nutrients.

2. Beef & Pork Category: This category includes high-calorie, high-fat, high-sodium items, such as the "Bacon Clubhouse Burger," which contains 720 calories, 40 grams of fat, 15 grams of saturated fat, and 1,470 mg of sodium. Other items like the "Double Quarter Pounder with Cheese" also contain high amounts of these unhealthy nutrients.

3. While the Chicken & Fish category typically contains lower levels of these unhealthy nutrients compared to Breakfast and Beef & Pork categories, some items in this group can still be considered less healthy due to their calorie, fat, and sodium content.

4. Overall, the Breakfast and Beef & Pork categories are the most unhealthy, based on the nutritional information provided in the dataset.

Advice for Customers:

1. Be cautious with Breakfast items: Many breakfast items are high in calories, fat, and sodium. Lighter options like "Fruit & Maple Oatmeal" or "Hash Brown" offer significantly lower calorie and fat content.
2. Opt for leaner protein in Beef & Pork items: Many burgers and sandwiches in this category are high in calories, fat, and sodium. Choosing leaner options like the "Hamburger" or "Cheeseburger" can provide a healthier alternative.
3. Prioritize the Chicken & Fish category: This category generally offers lower calorie, fat, and sodium options compared to Breakfast and Beef & Pork categories. Options like the "Premium Crispy Chicken Classic Sandwich" or "Premium Grilled Chicken Ranch BLT Sandwich" can be good choices for a healthier meal.
4. Be mindful of portion sizes: Many items come in different sizes. Opt for smaller portions to limit unhealthy nutrient intake
5. Supplement with healthier sides: Consider adding healthier side options like "Hash Browns" or "Fruit & Maple Oatmeal" to balance out the nutritional profile of your meal.

By following these guidelines, customers can make more informed decisions and select menu items that align with their health goals and dietary needs.

Recommendations to Improve McDonald's Menu Nutritional Profile:

1. Increase Healthy Options:

Introduce more low-calorie, low-fat, and low-sodium items to cater to health-conscious customers. Expand the selection of salads, grilled chicken, and fruit-based sides to provide healthier alternatives. 2. Enhance Nutritional Information Transparency: Clearly display nutritional information on menus and packaging to help customers make informed choices. Include allergen information to support individuals with dietary restrictions. 3. Reduce Added Sugars: Decrease the amount of added sugars in beverages, desserts, and breakfast items to align with dietary guidelines.

Benefit of Nutritional Analysis

Benefit for Customers:

1. Informed Food Choices: With access to detailed nutritional information, customers can make informed decisions that align with their dietary preferences and health goals.

2. Health-Conscious Decisions: Nutritional analysis allows customers to identify healthier options, such as those with lower calorie, fat, and sodium content, promoting better eating habits and overall well-being.

3. Tailored Dietary Needs: Customers with specific dietary restrictions, such as low-fat, low-sodium, or high-protein diets, can easily find menu items that meet their nutritional needs, helping them customize their meals according to individual preferences.

Benefit for McDonald's Organization:

1. Menu Development: The nutritional analysis can guide McDonald's in developing a balanced and diverse menu that appeals to a wide range of customer preferences. Understanding the nutritional profiles allows McDonald's to introduce healthier options and adjust existing recipes to meet the demand for healthier choices.

2. Customer Satisfaction: Providing transparent nutritional information demonstrates McDonald’s commitment to customer well-being. This can enhance the overall dining experience and improve customer satisfaction.

3. Health and Wellness Initiatives: McDonald's can leverage the nutritional analysis data to align with health and wellness trends. By promoting healthier menu options, McDonald's can position itself as a health-conscious brand, attracting customers who prioritize nutritious eating habits.

In conclusion, nutritional analysis benefits both customers and McDonald's by encouraging informed food choices, promoting health-conscious decisions, and guiding menu development that meets the diverse dietary preferences and needs of customers.

In [ ]: