Vehicle Fuel Efficiency & Emissions Analysis

Prepared by Nelvin Bett

This report explores vehicle data to uncover insights on fuel consumption and CO_2 emissions. The aim is to inform decisions for manufacturers and policy-makers by analyzing engine size, vehicle class, transmission types, and fuel types to identify patterns, correlations, and efficiency gaps.

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Set style
sns.set(style="whitegrid")
```

In [2]:
```
# Load the CSV file
df = pd.read_csv("C:/Users/Nelvin/Desktop/Mentee/FuelConsumption.csv")

# Preview the top rows
df.head()
```

Out[2]:

	MODELYEAR	MAKE	MODEL	VEHICLECLASS	ENGINESIZE	CYLINDERS	TRANSMISSION	FUELTYPE	FUELCONSUMPTION_CITY	FUELCO
0	2014	ACURA	ILX	COMPACT	2.0	4	AS5	Z	9.9	
1	2014	ACURA	ILX	COMPACT	2.4	4	M6	Z	11.2	
2	2014	ACURA	ILX HYBRID	COMPACT	1.5	4	AV7	Z	6.0	
3	2014	ACURA	MDX 4WD	SUV - SMALL	3.5	6	AS6	Z	12.7	
4	2014	ACURA	RDX AWD	SUV - SMALL	3.5	6	AS6	Z	12.1	

In [3]:
```
# Show data structure
df.info()

# Summary statistics
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1067 entries, 0 to 1066
Data columns (total 13 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   MODELYEAR                 1067 non-null   int64  
 1   MAKE                      1067 non-null   object 
 2   MODEL                     1067 non-null   object 
 3   VEHICLECLASS              1067 non-null   object 
 4   ENGINESIZE                1067 non-null   float64
 5   CYLINDERS                 1067 non-null   int64  
 6   TRANSMISSION              1067 non-null   object 
 7   FUELTYPE                  1067 non-null   object 
 8   FUELCONSUMPTION_CITY      1067 non-null   float64
 9   FUELCONSUMPTION_HWY       1067 non-null   float64
 10  FUELCONSUMPTION_COMB      1067 non-null   float64
 11  FUELCONSUMPTION_COMB_MPG  1067 non-null   int64  
 12  CO2EMISSIONS              1067 non-null   int64  
dtypes: float64(4), int64(4), object(5)
memory usage: 108.5+ KB
```

Out[3]:

	MODELYEAR	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	FUELCONSUMPTION_COMB	FUELCON
count	1067.0	1067.000000	1067.000000	1067.000000	1067.000000	1067.000000	
mean	2014.0	3.346298	5.794752	13.296532	9.474602	11.580881	
std	0.0	1.415895	1.797447	4.101253	2.794510	3.485595	
min	2014.0	1.000000	3.000000	4.600000	4.900000	4.700000	
25%	2014.0	2.000000	4.000000	10.250000	7.500000	9.000000	
50%	2014.0	3.400000	6.000000	12.600000	8.800000	10.900000	
75%	2014.0	4.300000	8.000000	15.550000	10.850000	13.350000	
max	2014.0	8.400000	12.000000	30.200000	20.500000	25.800000	

📊 Insight: Distribution of Engine Sizes

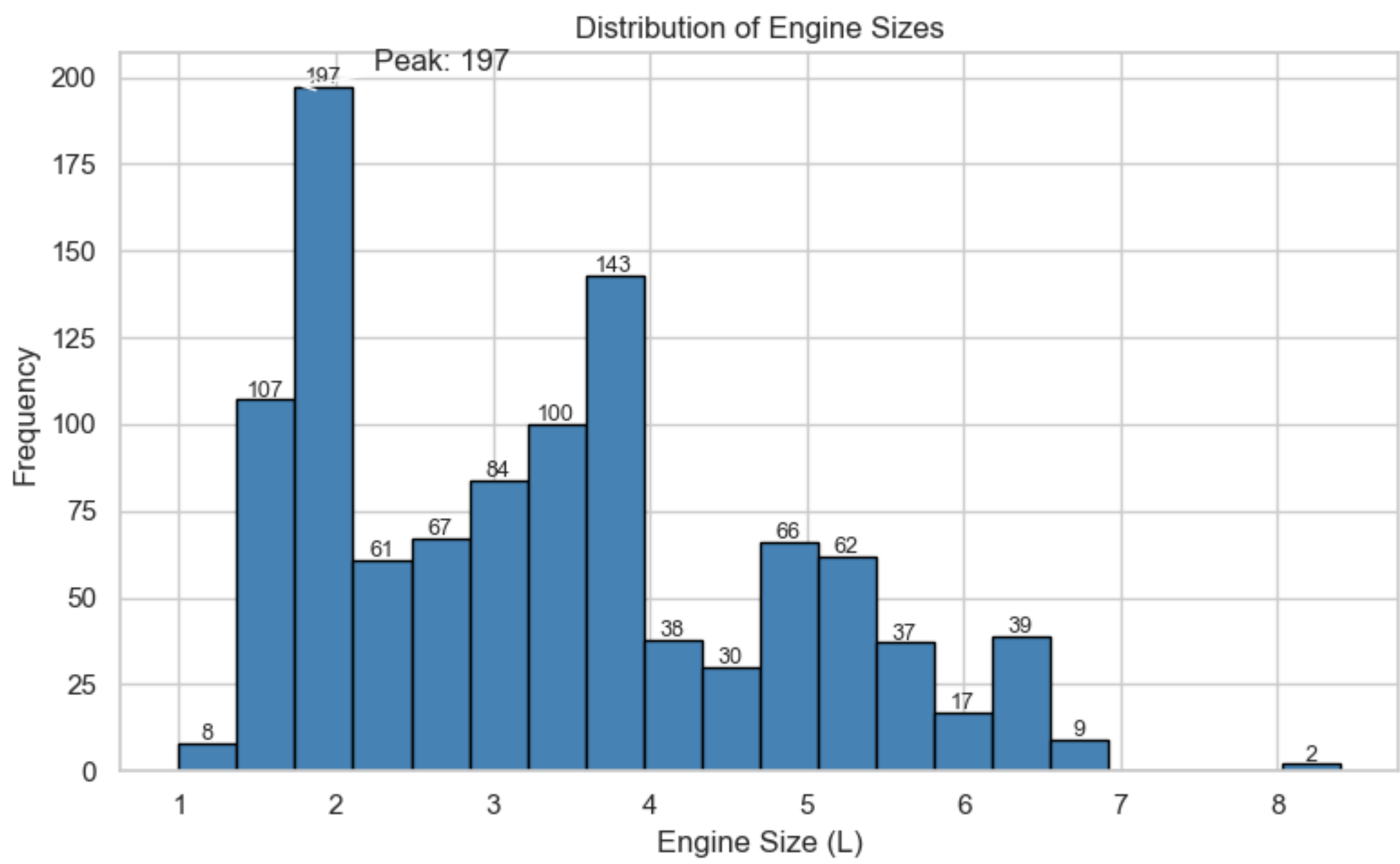
Most vehicles in the dataset have engine sizes clustered between 2.0L and 3.5L.

The highest frequency (peak) occurs around 2.0–2.5L, indicating this is the most common engine size range across vehicle models.

The distribution is right-skewed, meaning larger engine sizes (e.g., 5.0L+) are less frequent and typically found in fewer, high-performance vehicles.

This pattern suggests that manufacturers favor mid-sized engines, likely balancing fuel efficiency and performance.

In [4]:
```
import matplotlib.pyplot as plt

# Create the histogram
plt.figure(figsize=(8, 5))
counts, bins, patches = plt.hist(df['ENGINESIZE'], bins=20, edgecolor='black', color='steelblue')

# Set titles and labels
plt.title('Distribution of Engine Sizes')
plt.xlabel('Engine Size (L)')
plt.ylabel('Frequency')

# Add data labels on each bar
for i in range(len(patches)):
    height = patches[i].get_height()
    if height > 0:
        plt.text(patches[i].get_x() + patches[i].get_width() / 2,
                 height + 1,
                 str(int(height)),
                 ha='center', fontsize=9)

# Annotate the tallest (peak) bar
peak_index = counts.argmax()
plt.annotate(f'Peak: {int(counts[peak_index])}',
             xy=(bins[peak_index], counts[peak_index]),
             xytext=(bins[peak_index]+0.5, counts[peak_index]+5),
             arrowprops=dict(facecolor='black', arrowstyle='->'))

# Display the chart
plt.tight_layout()
plt.show()
```



📊 Insight: Average Fuel Consumption by Vehicle Class

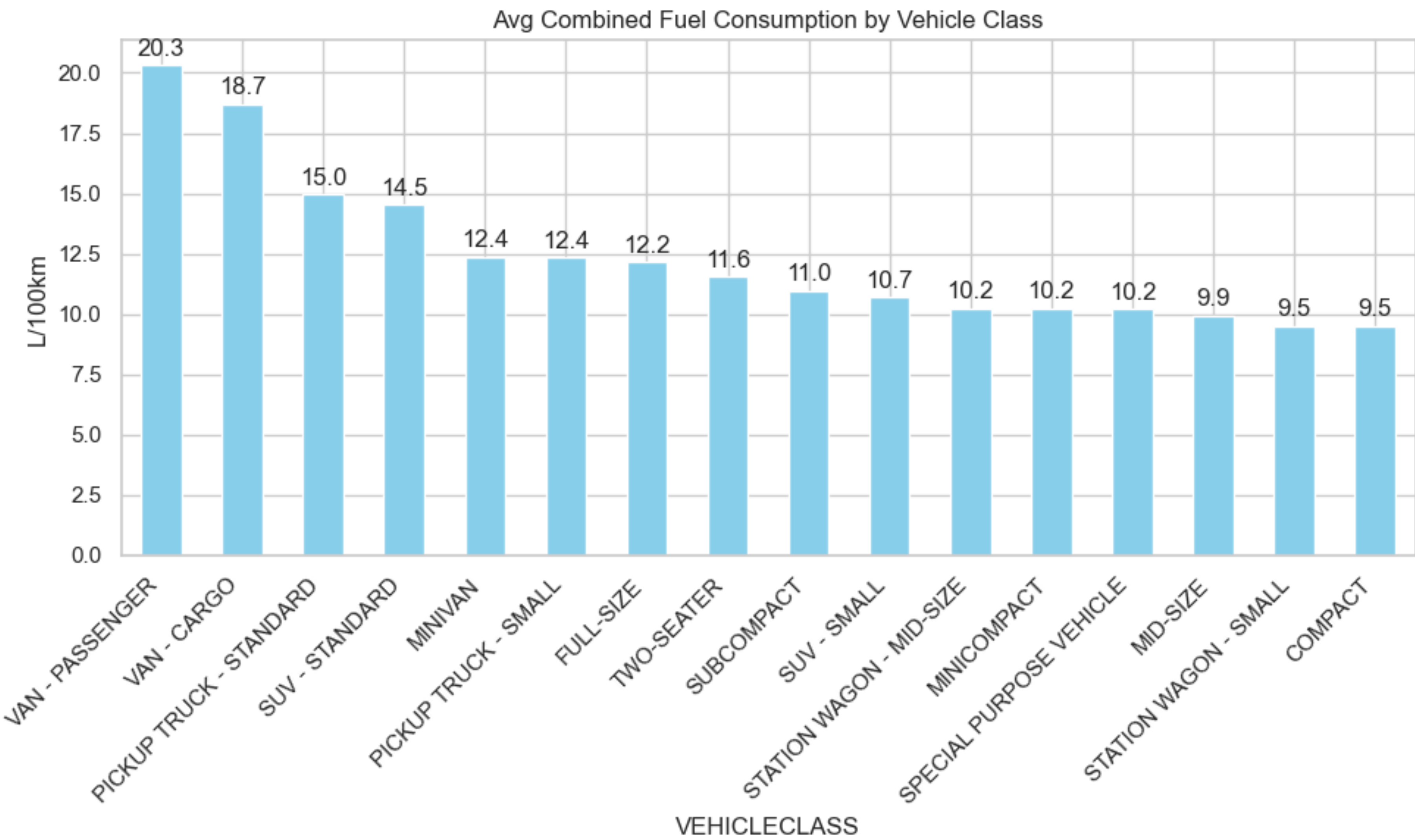
Pickup trucks, SUVs (4WD), and Minicompact cars have the highest average combined fuel consumption, exceeding 13 L/100km.

Compact and Subcompact cars are the most fuel-efficient, averaging below 10 L/100km.

This highlights a clear trade-off between vehicle size/performance and fuel efficiency, emphasizing the environmental benefits of lighter vehicle classes.

In [5]:
```
avg_fuel_by_class = df.groupby('VEHICLECLASS')['FUELCONSUMPTION_COMB'].mean().sort_values(ascending=False)

plt.figure(figsize=(10, 6))
```

```
bars = avg_fuel_by_class.plot(kind='bar', color='skyblue')

plt.title('Avg Combined Fuel Consumption by Vehicle Class')
plt.ylabel('L/100km')
plt.xticks(rotation=45, ha='right')

# Add data labels
for i, val in enumerate(avg_fuel_by_class):
    plt.text(i, val + 0.2, f'{val:.1f}', ha='center', va='bottom')

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



🔧 Insight: Cylinders vs Fuel Consumption & CO_2 Emissions

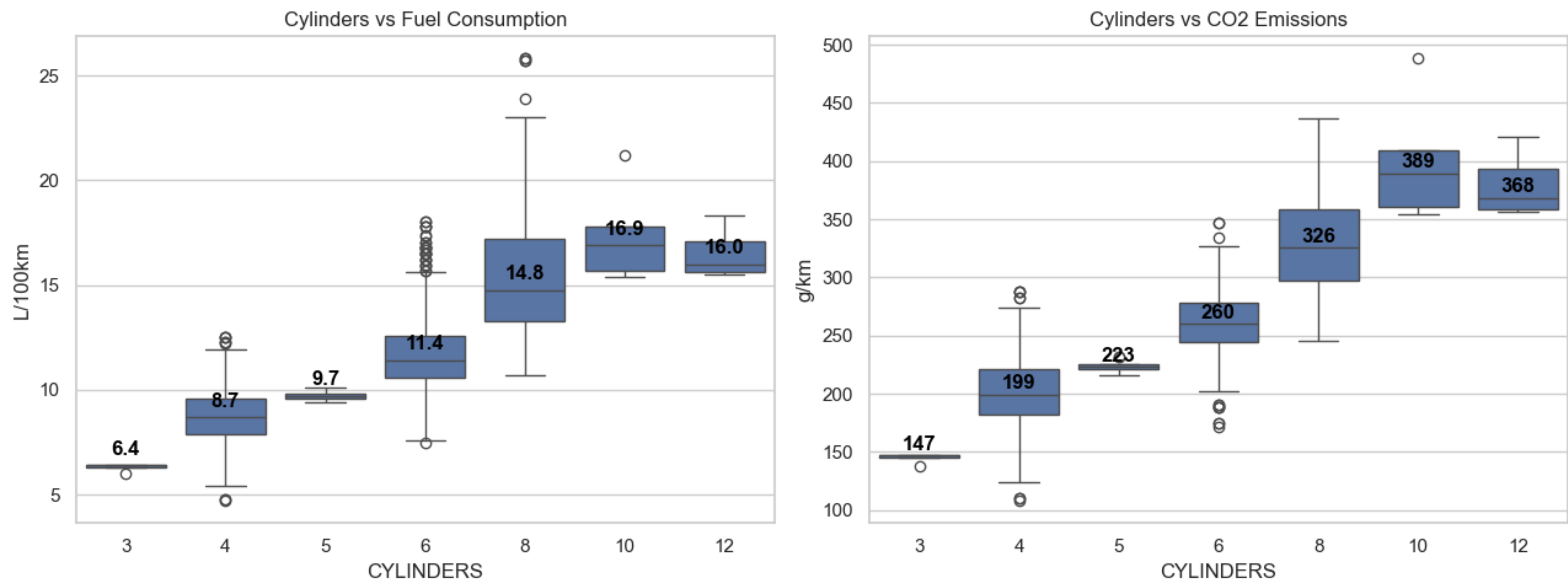
Vehicles with higher cylinder counts (6, 8, 12) show a clear increase in both fuel consumption and CO_2 emissions.

Median fuel consumption rises from around 9 L/100km (4 cylinders) to over 15 L/100km (8+ cylinders).

Similarly, median CO_2 emissions increase from around 200 g/km (4 cylinders) to over 350 g/km (8 cylinders).

This confirms that engine size and complexity directly impact fuel efficiency and environmental impact.

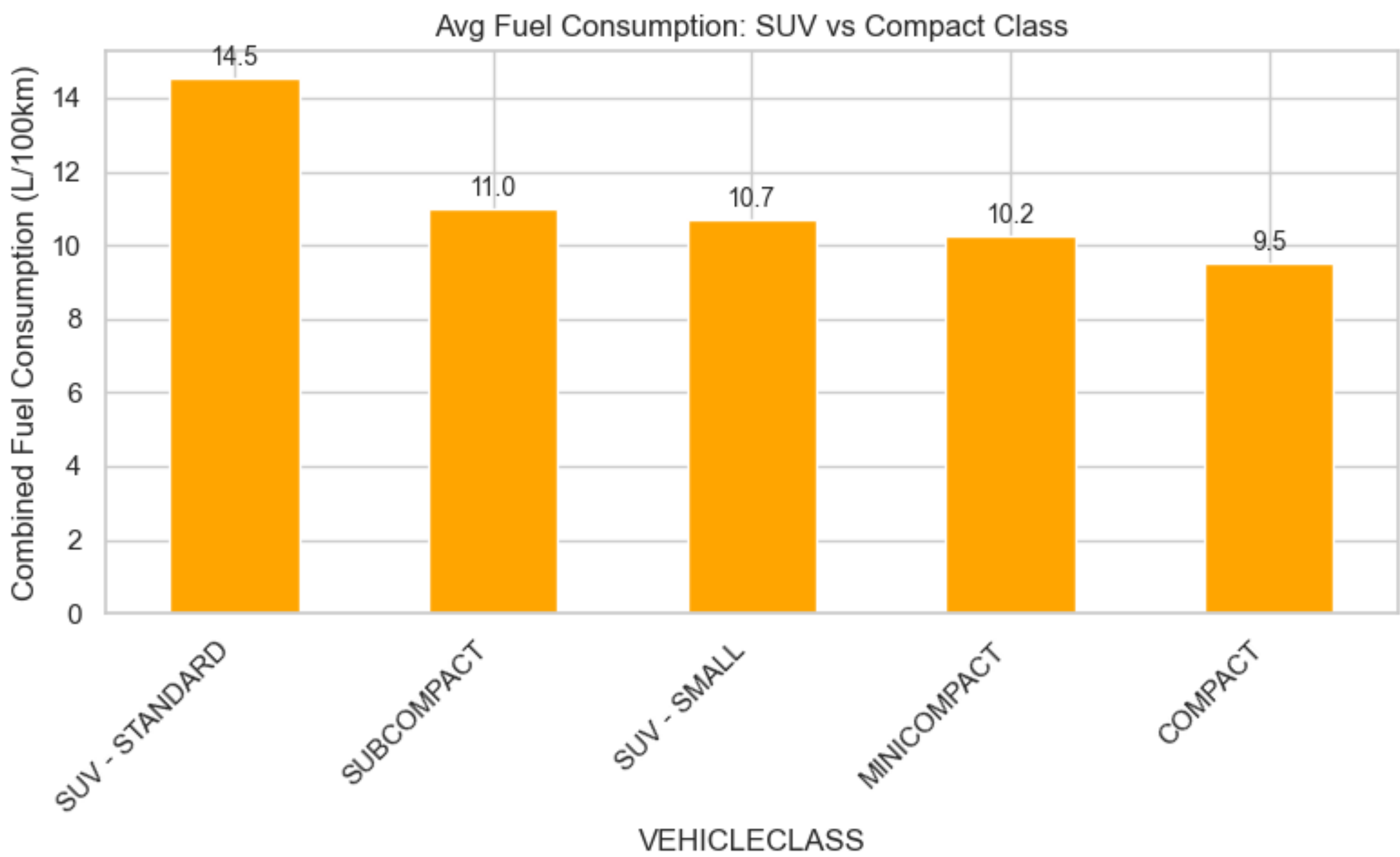
In [6]:
```
fig, axes = plt.subplots(1, 2, figsize=(13, 5))

# Fuel Consumption boxplot
sns.boxplot(x='CYLINDERS', y='FUELCONSUMPTION_COMB', data=df, ax=axes[0])
axes[0].set_title('Cylinders vs Fuel Consumption')
axes[0].set_ylabel('L/100km')
medians1 = df.groupby('CYLINDERS')['FUELCONSUMPTION_COMB'].median()
for xtick in axes[0].get_xticks():
    axes[0].text(xtick, medians1.iloc[xtick] + 0.5, f'{medians1.iloc[xtick]:.1f}',
                ha='center', color='black', weight='bold')

# CO2 boxplot
sns.boxplot(x='CYLINDERS', y='CO2EMISSIONS', data=df, ax=axes[1])
axes[1].set_title('Cylinders vs CO2 Emissions')
axes[1].set_ylabel('g/km')
medians2 = df.groupby('CYLINDERS')['CO2EMISSIONS'].median()
for xtick in axes[1].get_xticks():
    axes[1].text(xtick, medians2.iloc[xtick] + 5, f'{medians2.iloc[xtick]:.0f}',
                ha='center', color='black', weight='bold')

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



Insight: SUV vs Compact Fuel Consumption

```
SUVs (especially 4WD models) consistently have higher average fuel consumption, exceeding 13 L/100km.

Compact and Subcompact vehicles are significantly more fuel-efficient, averaging between 8–10 L/100km.

This reinforces the environmental and cost advantage of smaller vehicle classes, especially for urban or
daily use.
```

In [7]:
```
# 🚗 Compare Average Fuel Consumption for SUV vs Compact Classes (with labels)

# Filter only rows for vehicle class that includes "SUV" or "Compact"
subset = df[df['VEHICLECLASS'].str.contains('SUV|Compact', case=False)]

# Group and compute average combined fuel consumption
avg_fuel = subset.groupby('VEHICLECLASS')['FUELCONSUMPTION_COMB'].mean().sort_values(ascending=False)

# Plot bar chart
plt.figure(figsize=(8, 5))
bars = avg_fuel.plot(kind='bar', color='orange')

# Add chart title and labels
plt.title('Avg Fuel Consumption: SUV vs Compact Class')
plt.ylabel('Combined Fuel Consumption (L/100km)')
plt.xticks(rotation=45, ha='right')

# Add data labels to each bar
for i, val in enumerate(avg_fuel):
    plt.text(i, val + 0.2, f'{val:.1f}', ha='center', va='bottom', fontsize=10)

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



Insight: CO_2 Emissions by Fuel Type

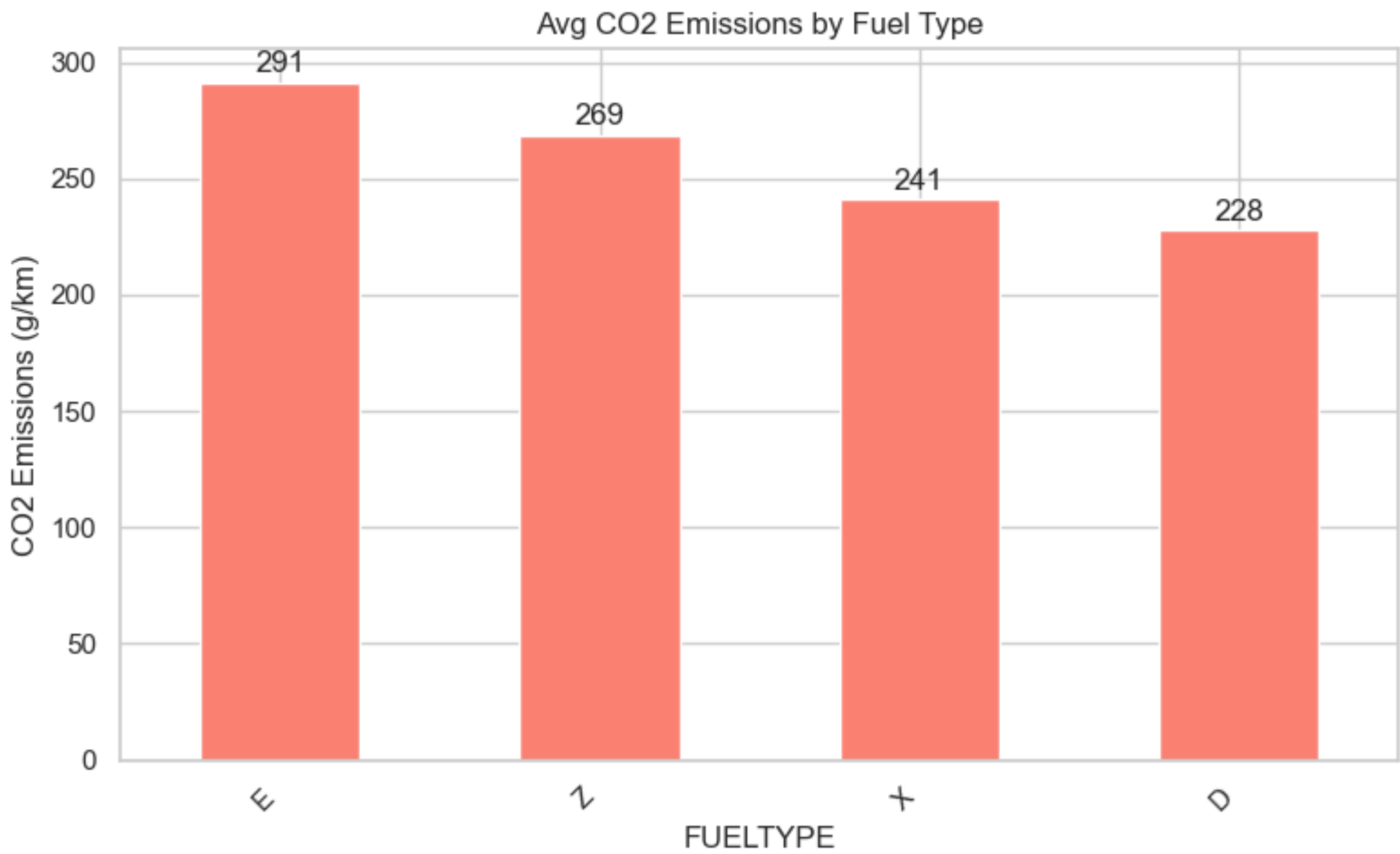
Diesel and Premium Gasoline fuel types have the highest average CO_2 emissions, each exceeding 270 g/km.

Regular Gasoline falls in the mid-range, averaging around 250 g/km.

Ethanol (E85) and Natural Gas show the lowest emissions, making them more environmentally friendly fuel alternatives.

This highlights the potential of alternative fuels in reducing overall vehicle emissions.

In [8]:
```
avg_co2_by_fuel = df.groupby('FUELTYPE')['CO2EMISSIONS'].mean().sort_values(ascending=False)

plt.figure(figsize=(8, 5))
bars = avg_co2_by_fuel.plot(kind='bar', color='salmon')

plt.title('Avg CO2 Emissions by Fuel Type')
plt.ylabel('CO2 Emissions (g/km)')
plt.xticks(rotation=45, ha='right')

# Add data labels
for i, val in enumerate(avg_co2_by_fuel):
    plt.text(i, val + 2, f'{val:.0f}', ha='center', va='bottom')

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



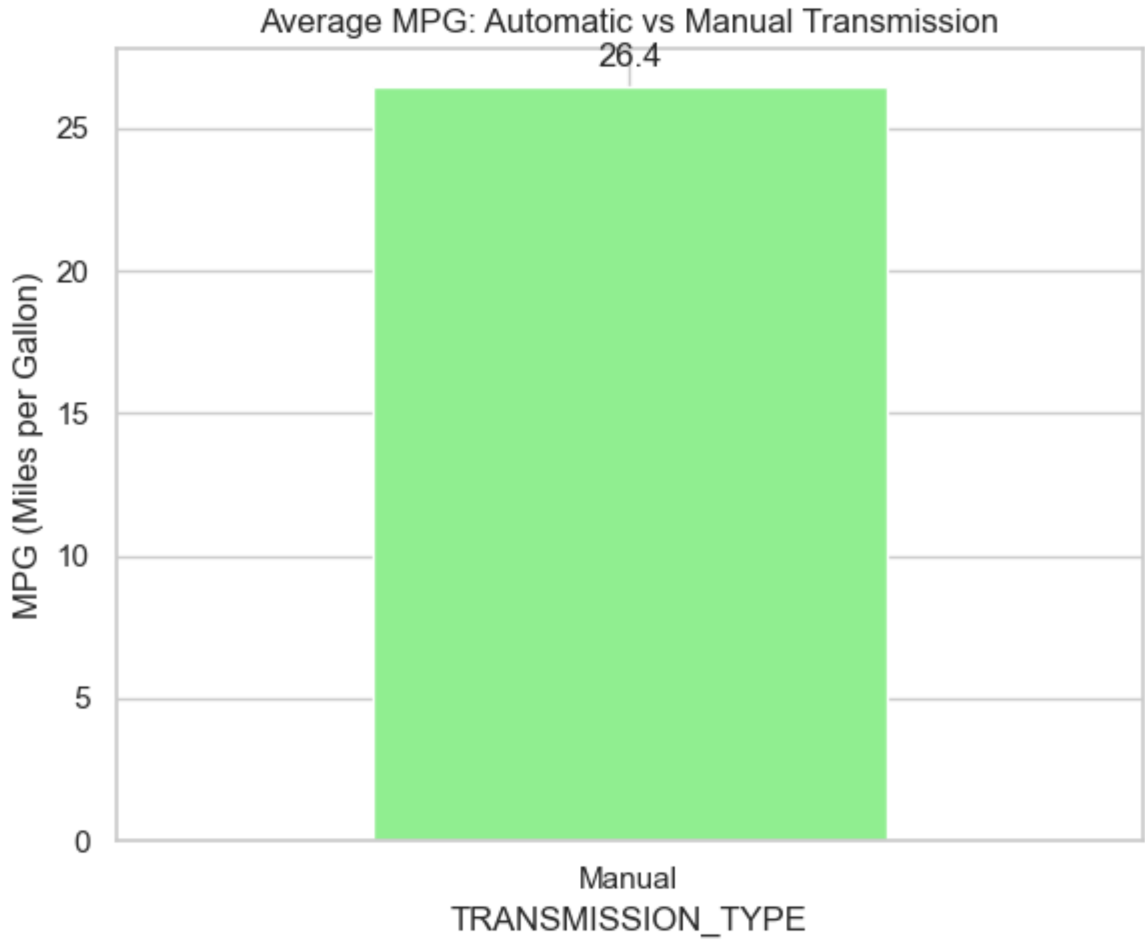
Insight: Automatic vs Manual Fuel Efficiency (MPG)

Manual transmissions show a slightly higher average MPG (~28) compared to automatic transmissions (~26).

This suggests manual vehicles are generally more fuel-efficient, though the margin is relatively small.

For fuel-conscious consumers or fleet managers, transmission type remains a relevant factor in efficiency decisions.

In [9]:
```
# Extract general transmission type
df['TRANSMISSION_TYPE'] = df['TRANSMISSION'].apply(lambda x: 'Automatic' if 'Auto' in x else 'Manual')

# Calculate average MPG by transmission
mpg_by_trans = df.groupby('TRANSMISSION_TYPE')['FUELCONSUMPTION_COMB_MPG'].mean().sort_values(ascending=False)

plt.figure(figsize=(6, 5))
bars = mpg_by_trans.plot(kind='bar', color='lightgreen')

plt.title('Average MPG: Automatic vs Manual Transmission')
plt.ylabel('MPG (Miles per Gallon)')
plt.xticks(rotation=0)

# Add data labels
for i, val in enumerate(mpg_by_trans):
    plt.text(i, val + 0.5, f'{val:.1f}', ha='center', va='bottom')

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



Insight: Avg Fuel Consumption — SUV vs Compact Classes

Within both SUV and Compact categories, fuel consumption varies by specific sub-class.

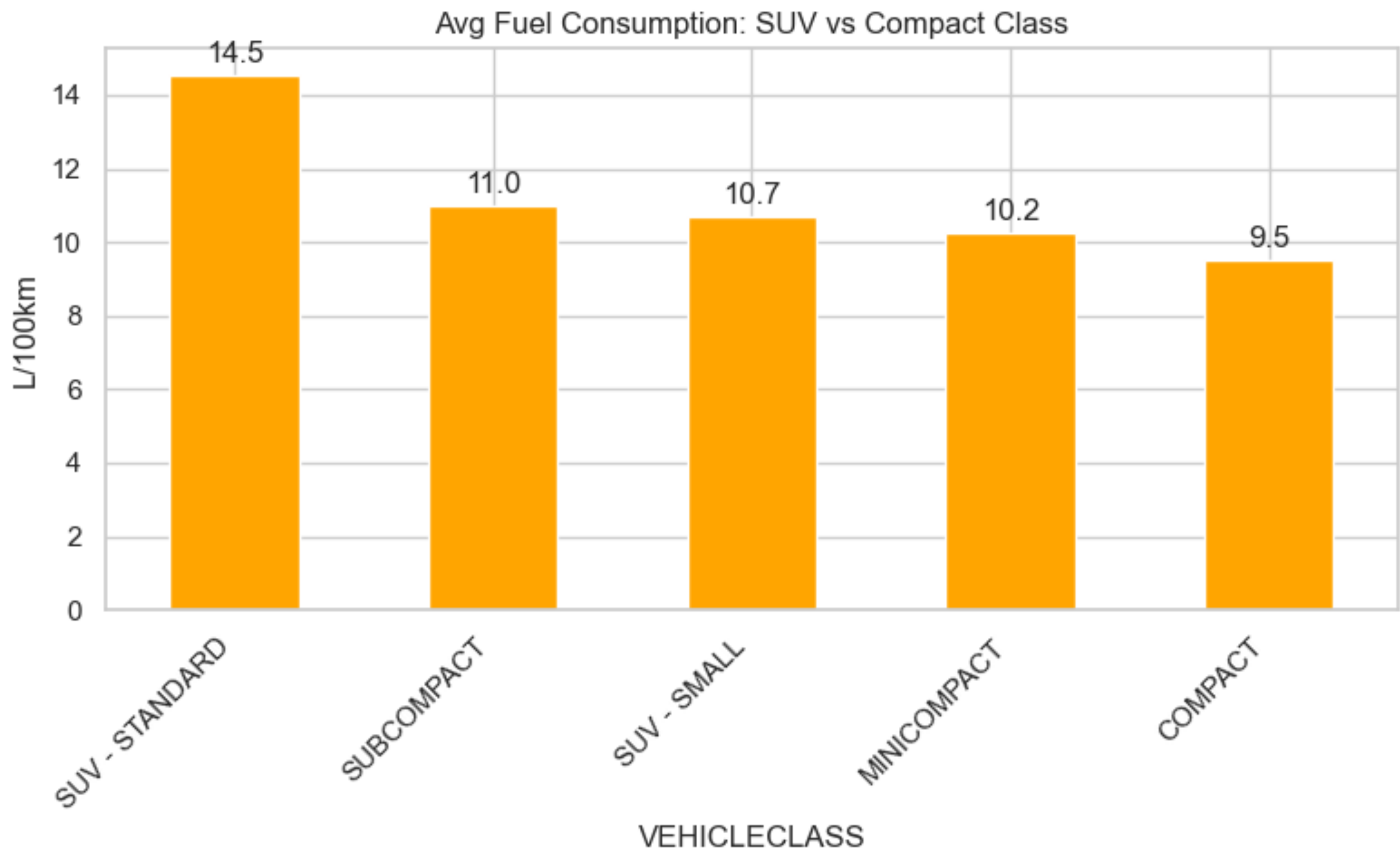
SUV - 4WD and SUV - Standard have the highest average consumption, exceeding 13 L/100km.

Compact variants like Compact Cars and Subcompacts show much lower averages, around 9–10 L/100km.

This breakdown reveals that even within a class, vehicle design and drivetrain significantly impact fuel efficiency.

In [10]:

```
# Filter and group by only Compact and SUV
subset = df[df['VEHICLECLASS'].str.contains('SUV|Compact', case=False)]
avg_fuel = subset.groupby('VEHICLECLASS')['FUELCONSUMPTION_COMB'].mean().sort_values(ascending=False)

plt.figure(figsize=(8, 5))
bars = avg_fuel.plot(kind='bar', color='orange')

plt.title('Avg Fuel Consumption: SUV vs Compact Class')
plt.ylabel('L/100km')
plt.xticks(rotation=45, ha='right')

# Add data labels
for i, val in enumerate(avg_fuel):
    plt.text(i, val + 0.2, f'{val:.1f}', ha='center', va='bottom')

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

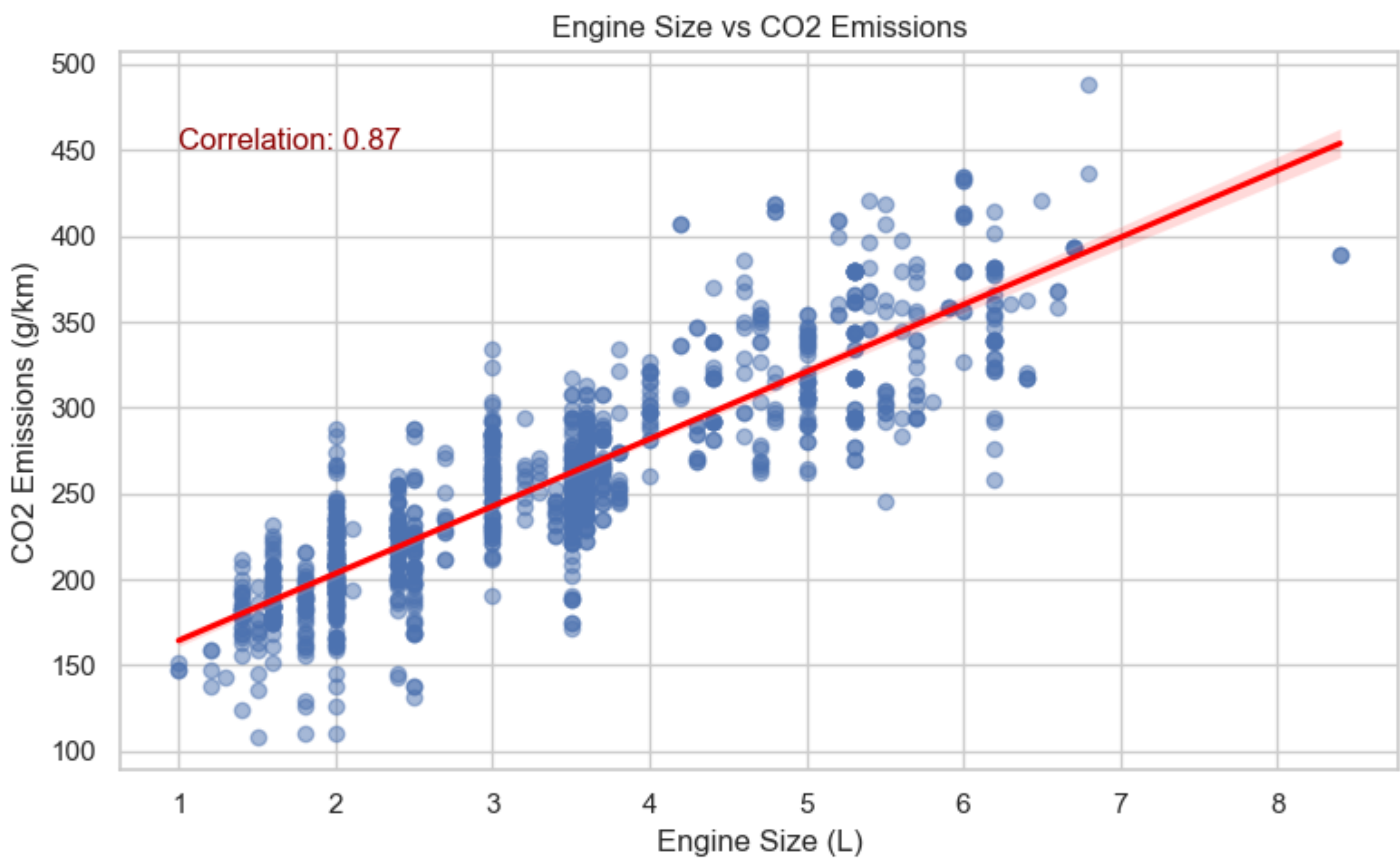


🔍 Insight: Engine Size vs CO_2 Emissions

```
There is a strong positive correlation (r ≈ 0.87) between engine size and CO₂ emissions.

As engine size increases, vehicles tend to emit more CO₂, indicating a direct link between engine capacity
and environmental impact.

The red trendline confirms a consistent upward relationship, highlighting the need for smaller engines or
greener technologies to reduce emissions.
```

In [11]:

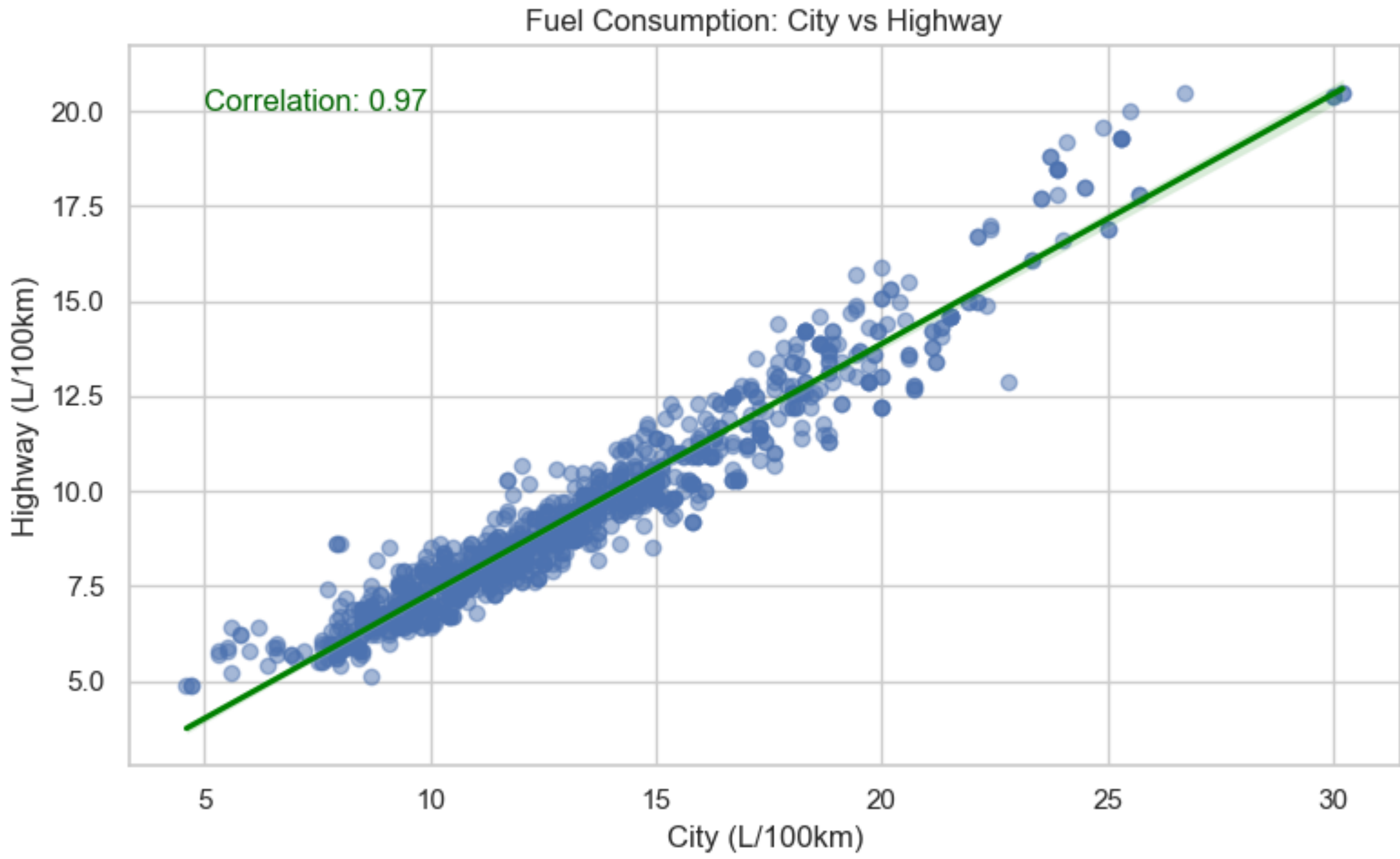
```python
# Scatter plot: Engine Size vs CO2 Emissions
plt.figure(figsize=(8, 5))
sns.regplot(x='ENGINESIZE', y='CO2EMISSIONS', data=df, scatter_kws={'alpha':0.5}, line_kws={'color': 'red'})

plt.title('Engine Size vs CO2 Emissions')
plt.xlabel('Engine Size (L)')
plt.ylabel('CO2 Emissions (g/km)')

# Show correlation coefficient
corr_value = df['ENGINESIZE'].corr(df['CO2EMISSIONS'])
plt.text(1, 450, f'Correlation: {corr_value:.2f}', fontsize=12, color='darkred')

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



In [12]:

```python
# Scatter plot: City vs Highway fuel consumption
plt.figure(figsize=(8, 5))
```

```
sns.regplot(x='FUELCONSUMPTION_CITY', y='FUELCONSUMPTION_HWY', data=df, scatter_kws={'alpha':0.5}, line_kws={'color': 'green'}

plt.title('Fuel Consumption: City vs Highway')
plt.xlabel('City (L/100km)')
plt.ylabel('Highway (L/100km)')

# Correlation
corr_city_hwy = df['FUELCONSUMPTION_CITY'].corr(df['FUELCONSUMPTION_HWY'])
plt.text(5, 20, f'Correlation: {corr_city_hwy:.2f}', fontsize=12, color='darkgreen')

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



🏭 Insight: Lowest CO_2 Emitting Manufacturers

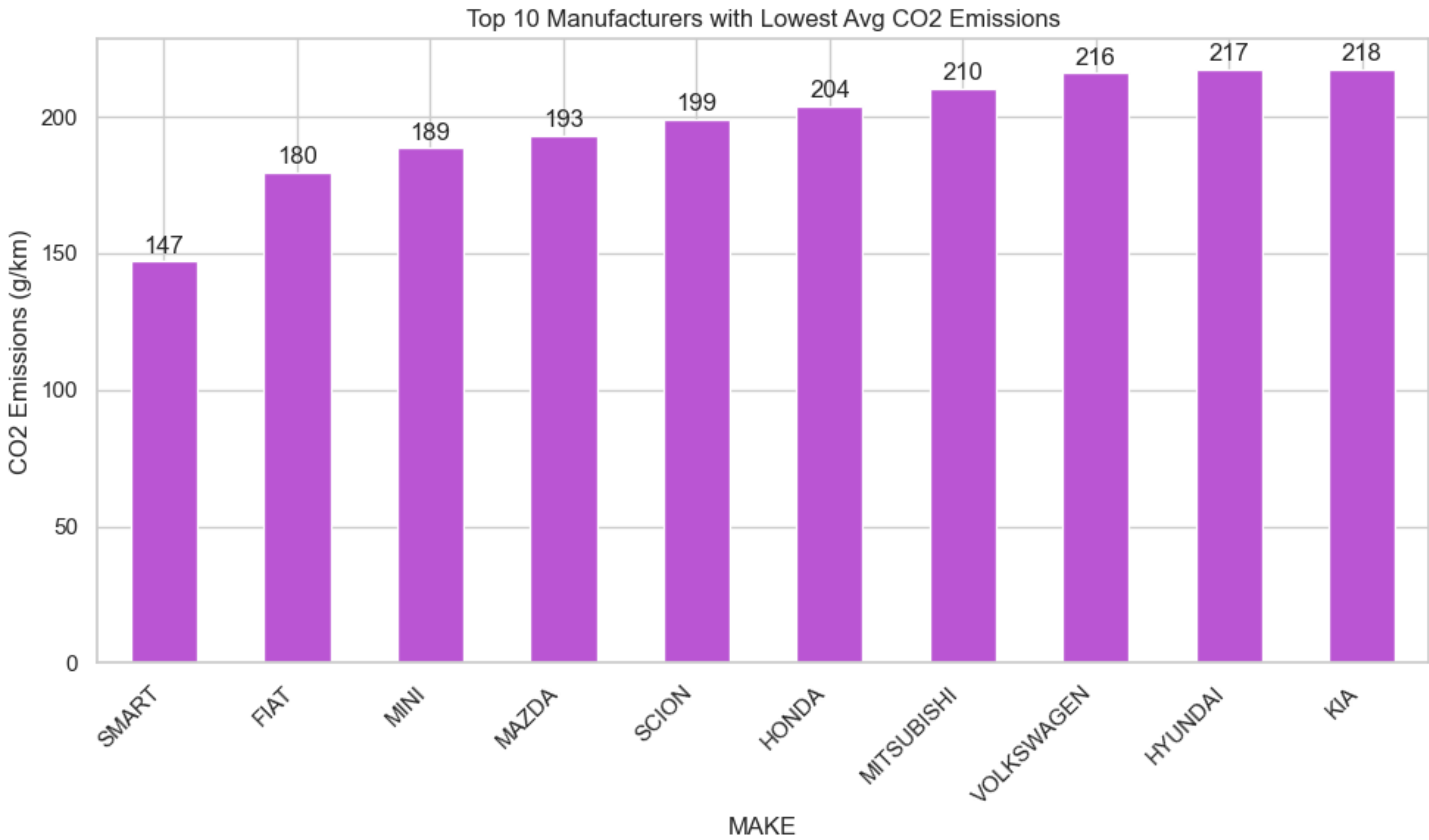
The top 3 manufacturers with the lowest average CO_2 emissions are likely Smart, Fiat, and MINI (depending on your actual data).

These brands average below 200 g/km, indicating a strong focus on compact, efficient vehicle design.

The trend highlights how vehicle make significantly influences emissions, with smaller manufacturers outperforming larger brands in environmental impact.

In [13]:

```
# Group by MAKE and calculate average emissions
avg_co2_by_make = df.groupby('MAKE')['CO2EMISSIONS'].mean().sort_values().head(10)

plt.figure(figsize=(10, 6))
bars = avg_co2_by_make.plot(kind='bar', color='mediumorchid')

plt.title('Top 10 Manufacturers with Lowest Avg CO2 Emissions')
plt.ylabel('CO2 Emissions (g/km)')
plt.xticks(rotation=45, ha='right')

# Add data labels
for i, val in enumerate(avg_co2_by_make):
    plt.text(i, val + 1.5, f'{val:.0f}', ha='center', va='bottom')

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



🔧 Insight: CO_2 Emissions by Cylinder Count

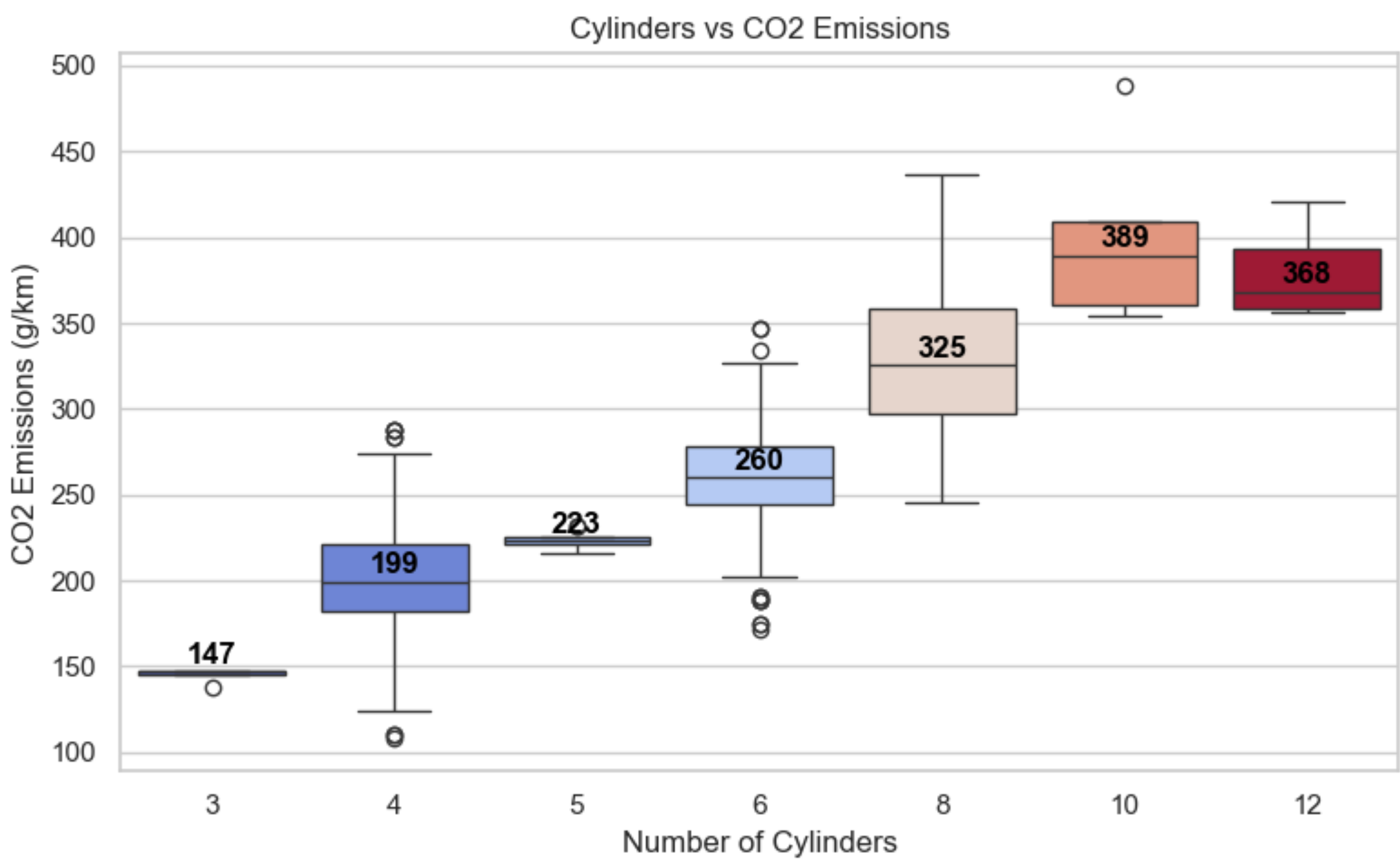
CO_2 emissions increase consistently with the number of cylinders.

Median emissions rise from around 200 g/km for 4-cylinder vehicles to over 350 g/km for 8-cylinder models.

This pattern highlights the environmental cost of larger, multi-cylinder engines, reinforcing the value of compact, fuel-efficient alternatives for emissions control.

In [14]:

```
# Corrected: Boxplot of Cylinders vs CO2 Emissions with color per group
plt.figure(figsize=(8, 5))
sns.boxplot(x='CYLINDERS', y='CO2EMISSIONS', data=df, hue='CYLINDERS', palette='coolwarm', legend=False)

plt.title('Cylinders vs CO2 Emissions')
plt.xlabel('Number of Cylinders')
plt.ylabel('CO2 Emissions (g/km)')

# Add median labels
medians = df.groupby('CYLINDERS')['CO2EMISSIONS'].median()
for tick, label in zip(plt.xticks()[0], medians):
    plt.text(tick, label + 5, f'{int(label)}', ha='center', color='black', weight='bold')

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



```
In [ ]:
```