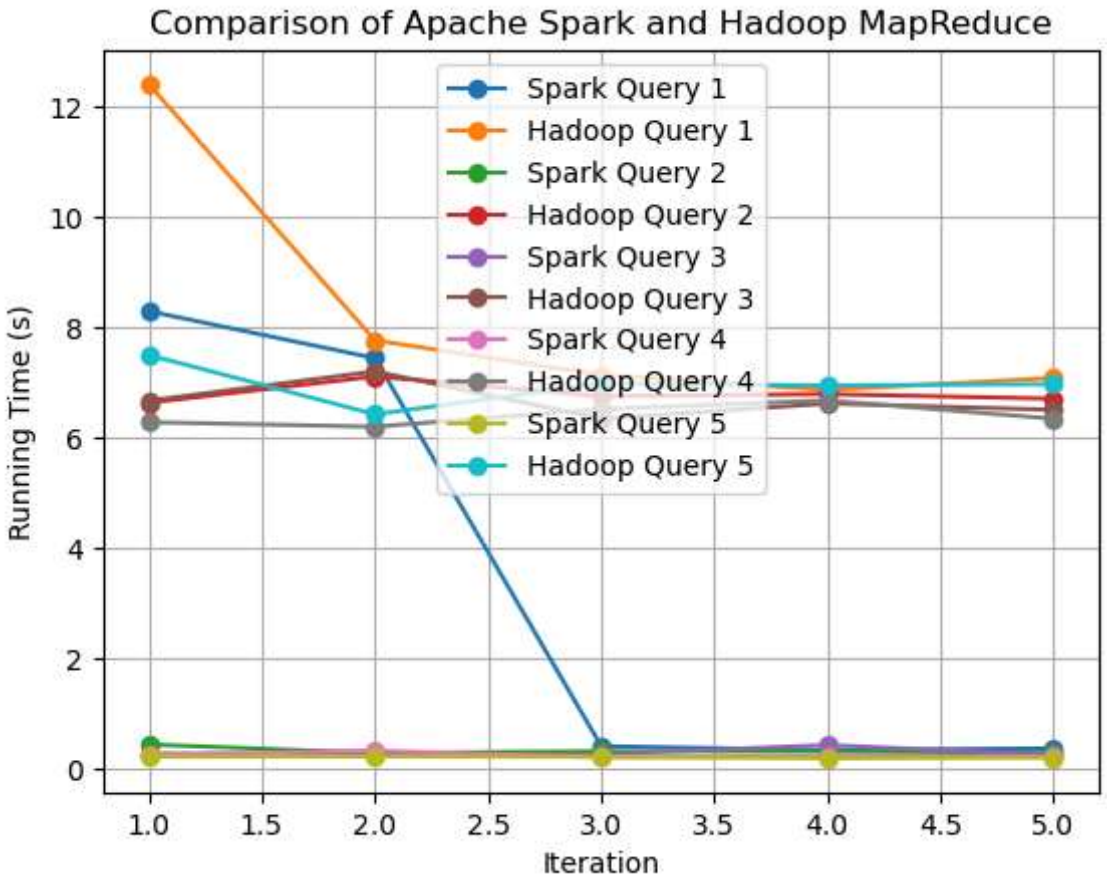
In [2]:
```python
import pandas as pd

# Read the CSV file into a DataFrame
df = pd.read_csv('Apache Spark and MapReduce.csv', header=0)

# Print the column names
print(df.columns)
```

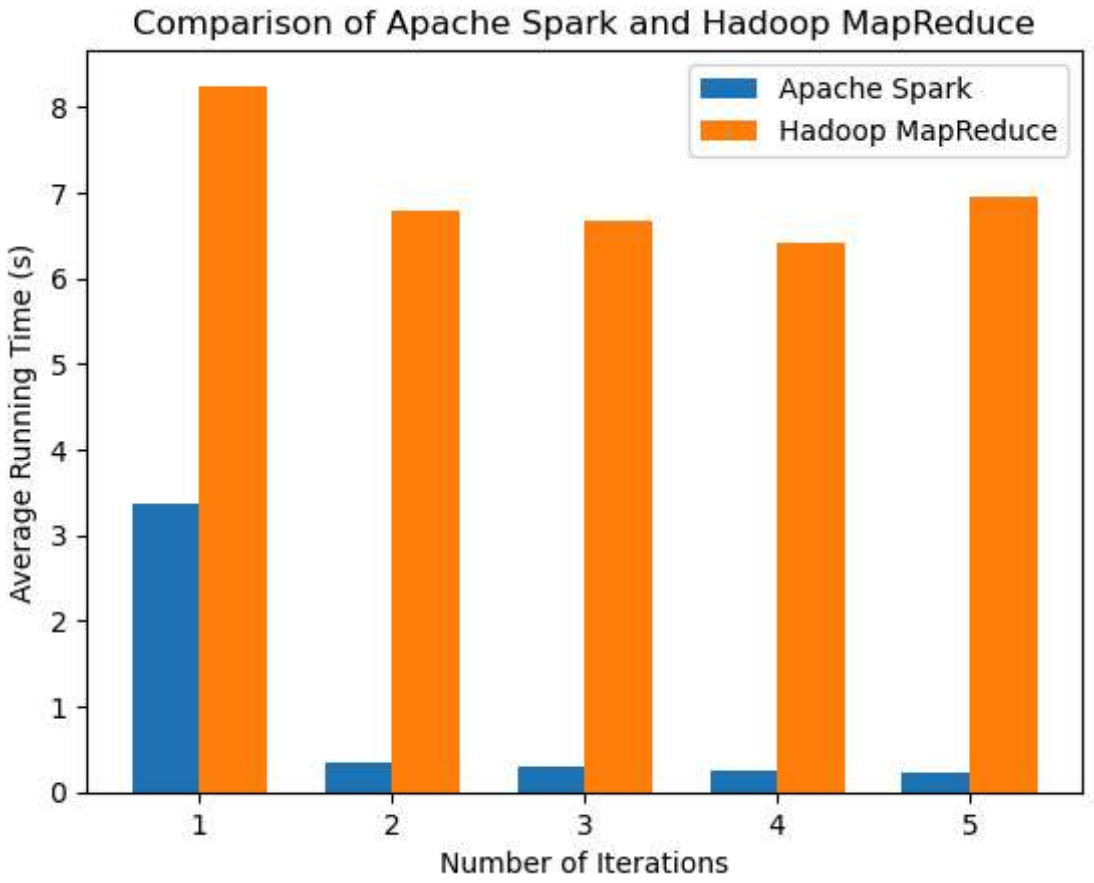
```
Index(['Query', 'Apache Spark Time (s)', 'Hadoop MapReduce Time (s)'], dtype='objec
t')
```

In [9]:
```python
import matplotlib.pyplot as plt

# Data for Apache Spark and Hadoop MapReduce running times for each query
spark_times = [
    [8.291, 7.441, 0.422, 0.35, 0.382],
    [0.452, 0.299, 0.33, 0.325, 0.299],
    [0.288, 0.275, 0.276, 0.443, 0.26],
    [0.262, 0.341, 0.221, 0.241, 0.232],
    [0.231, 0.236, 0.23, 0.201, 0.215]
]

hadoop_times = [
    [12.398, 7.764, 7.138, 6.864, 7.084],
    [6.635, 7.111, 6.754, 6.792, 6.707],
    [6.672, 7.202, 6.345, 6.621, 6.507],
    [6.283, 6.201, 6.519, 6.677, 6.345],
    [7.498, 6.425, 6.982, 6.948, 6.972]
]

# Iterations for each query
iterations = list(range(1, 6))

# Plotting the graph
for query, (spark, hadoop) in enumerate(zip(spark_times, hadoop_times), start=1):
    plt.plot(iterations, spark, marker='o', label=f'Spark Query {query}')
    plt.plot(iterations, hadoop, marker='o', label=f'Hadoop Query {query}')

plt.xlabel('Iteration')
plt.ylabel('Running Time (s)')
plt.title('Comparison of Apache Spark and Hadoop MapReduce')
plt.legend()
plt.grid(True)
plt.show()
```



In [12]:

```
import numpy as np

# Iterations for each query
queries = list(range(1, 6))

# Calculate the average times for each query
spark_avg_times = [np.mean(times) for times in spark_times]
hadoop_avg_times = [np.mean(times) for times in hadoop_times]

# Width of the bars
bar_width = 0.35

# Plotting the bar chart
fig, ax = plt.subplots()
bar_spark = ax.bar(np.arange(len(queries)) - bar_width/2, spark_avg_times, bar_width,
bar_hadoop = ax.bar(np.arange(len(queries)) + bar_width/2, hadoop_avg_times, bar_width

# Add labels, title, and legend
ax.set_xlabel('Number of Iterations')
ax.set_ylabel('Average Running Time (s)')
ax.set_title('Comparison of Apache Spark and Hadoop MapReduce')
ax.set_xticks(np.arange(len(queries)))
ax.set_xticklabels(queries)
ax.legend()

# Show plot
plt.show()
```



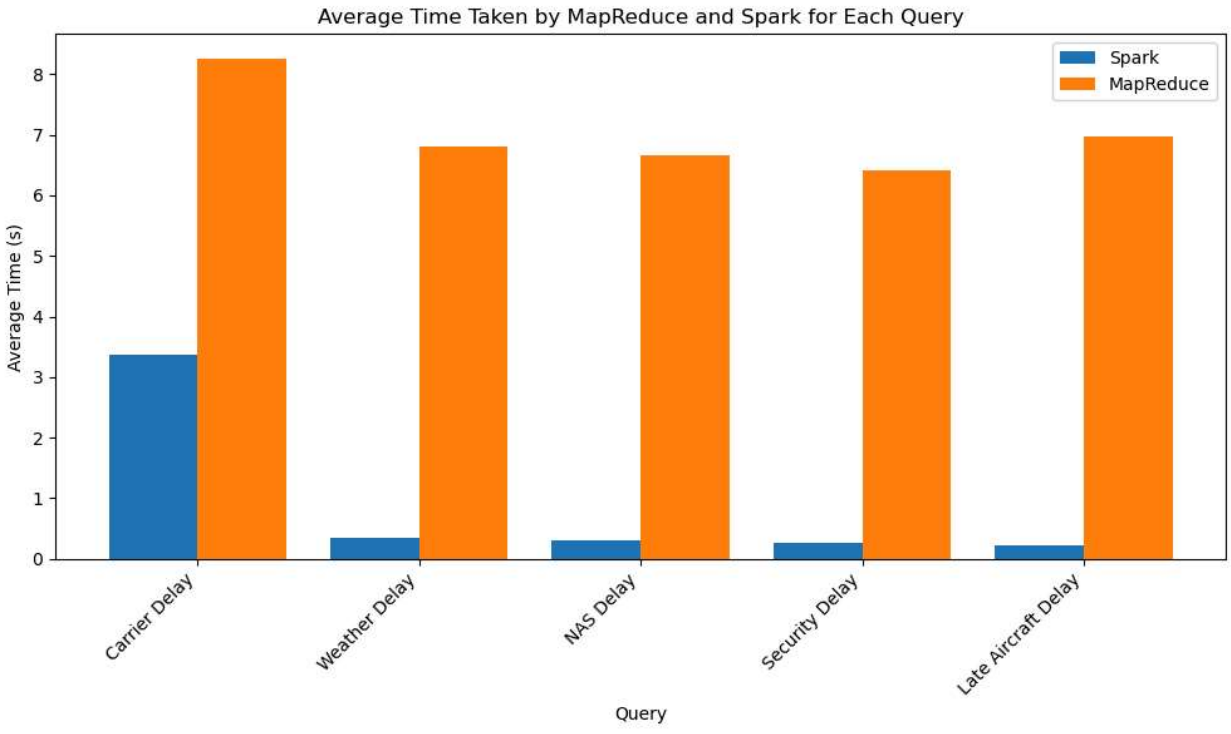
In [17]:

```
from tabulate import tabulate

# Sample data for Apache Spark and Hadoop MapReduce running times for each query
all_spark_times = [
    [8.291, 7.441, 0.422, 0.35, 0.382],  # Carrier Delay query
    [0.452, 0.299, 0.33, 0.325, 0.299],  # Weather Delay query
    [0.288, 0.275, 0.276, 0.443, 0.26],  # NAS Delay query
    [0.262, 0.341, 0.221, 0.241, 0.232],  # Security Delay query
    [0.231, 0.236, 0.23, 0.201, 0.215]   # Late Aircraft Delay query
]

all_hadoop_times = [
    [12.398, 7.764, 7.138, 6.864, 7.084],  # Carrier Delay query
    [6.635, 7.111, 6.754, 6.792, 6.707],   # Weather Delay query
    [6.672, 7.202, 6.345, 6.621, 6.507],   # NAS Delay query
    [6.283, 6.201, 6.519, 6.677, 6.345],   # Security Delay query
    [7.498, 6.425, 6.982, 6.948, 6.972]    # Late Aircraft Delay query
]

# Queries
queries = ['Carrier Delay', 'Weather Delay', 'NAS Delay', 'Security Delay', 'Late Airc

# Calculate the average times for each query
spark_avg_times = [sum(times) / len(times) for times in all_spark_times]
hadoop_avg_times = [sum(times) / len(times) for times in all_hadoop_times]

# Create table data
table_data = []
for query, spark_time, hadoop_time in zip(queries, spark_avg_times, hadoop_avg_times):
    table_data.append([query, spark_time, hadoop_time])
```

```
# Display table
print(tabulate(table_data, headers=['Query', 'Spark Avg Time (s)', 'Hadoop Avg Time (s
```

```
+--------------------+----------------------+-----------------------+
| Query              |   Spark Avg Time (s) |   Hadoop Avg Time (s) |
+====================+======================+=======================+
| Carrier Delay      |               3.3772 |                8.2496 |
+--------------------+----------------------+-----------------------+
| Weather Delay      |               0.341  |                6.7998 |
+--------------------+----------------------+-----------------------+
| NAS Delay          |               0.3084 |                6.6694 |
+--------------------+----------------------+-----------------------+
| Security Delay     |               0.2594 |                6.405  |
+--------------------+----------------------+-----------------------+
| Late Aircraft Delay |              0.2226 |                6.965  |
+--------------------+----------------------+-----------------------+
```

In [18]:
```
# Calculate the average times for each query
spark_avg_times = [np.mean(times) for times in all_spark_times]
hadoop_avg_times = [np.mean(times) for times in all_hadoop_times]

# Plotting the bar graph
plt.figure(figsize=(10, 6))
plt.bar(np.arange(len(queries)) - 0.2, spark_avg_times, width=0.4, label='Spark', alig
plt.bar(np.arange(len(queries)) + 0.2, hadoop_avg_times, width=0.4, label='MapReduce',

# Adding labels and title
plt.xlabel('Query')
plt.ylabel('Average Time (s)')
plt.title('Average Time Taken by MapReduce and Spark for Each Query')
plt.xticks(np.arange(len(queries)), queries, rotation=45, ha='right')
plt.legend()

# Show plot
plt.tight_layout()
plt.show()
```



In [ ]: