REPORT-2016

In [9]:

```
import pandas as pd
import matplotlib.pyplot as plt
```

In [10]:

```
# Load the dataset into a pandas DataFrame
data = {
    "Year": ["2007/08", "2008/09", "2009/10", "2010/11", "2011/12", "2012/13", "20
    "Scope1": [6424, 6831, 6720, 6659, 6099, 6338, 5315, 8672, 9417],
    "Scope2": [9939, 10444, 10107, 9791, 9348, 7957, 8350, 3477, 2288],
    "Scope3": [289, 355, 340, 704, 572, 448, 403, 397, 285]
}

df_emissions = pd.DataFrame(data)
```

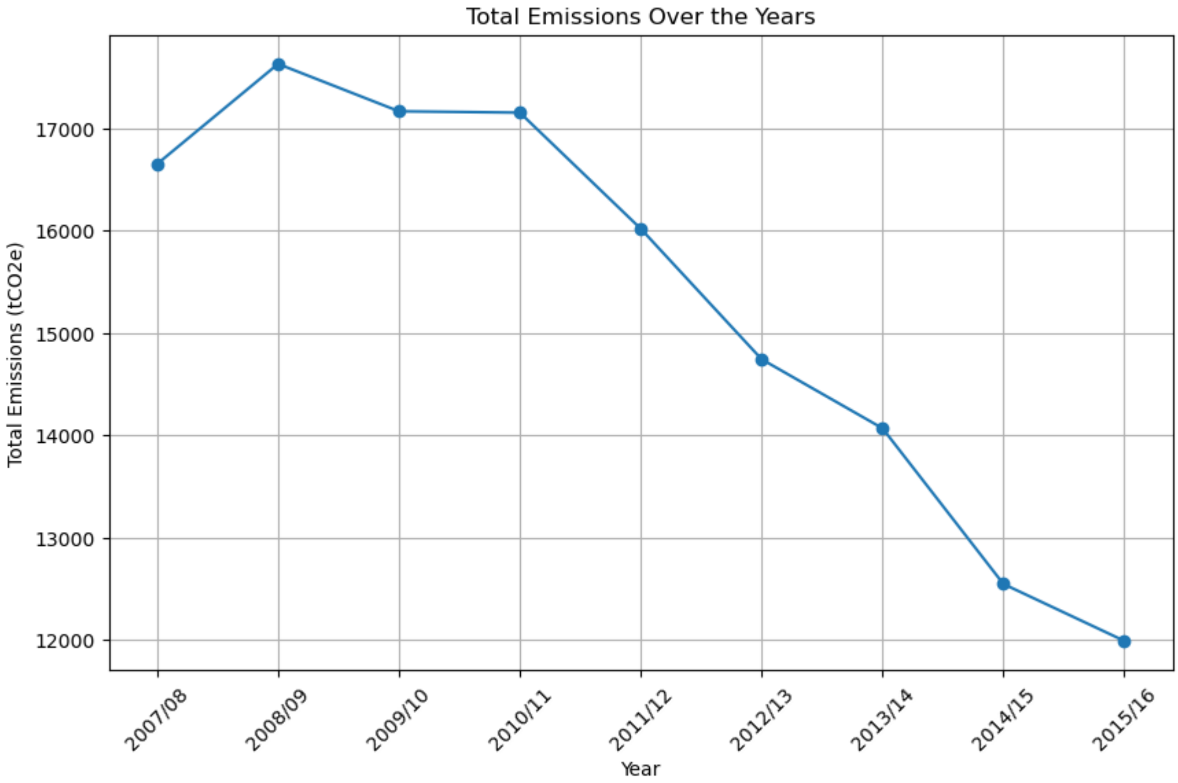
In [11]:

```
# Calculate the total emissions for each year
df_emissions["Total"] = df_emissions["Scope1"] + df_emissions["Scope2"] + df_emiss

# Print the DataFrame with emissions data
print(df_emissions)
```

```
      Year  Scope1  Scope2  Scope3  Total
0  2007/08    6424    9939     289  16652
1  2008/09    6831   10444     355  17630
2  2009/10    6720   10107     340  17167
3  2010/11    6659    9791     704  17154
4  2011/12    6099    9348     572  16019
5  2012/13    6338    7957     448  14743
6  2013/14    5315    8350     403  14068
7  2014/15    8672    3477     397  12546
8  2015/16    9417    2288     285  11990
```

In [12]:

```
# Plot emissions over the years
plt.figure(figsize=(10, 6))
plt.plot(df_emissions["Year"], df_emissions["Total"], marker='o')
plt.xlabel("Year")
plt.ylabel("Total Emissions (tCO2e)")
plt.title("Total Emissions Over the Years")
plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```



In [13]:

```
# Breakdown of emission sources
data_breakdown = {
    "Emission source": ["Grid Electricity (generation)", "Grid Electricity (transmi
                        "Natural Gas", "Gas Oil", "Burning Oil (Kerosene)", "Petrol
                        "Diesel (average biofuel blend)", "Water - Supply", "Water
                        "Mixed recycling", "Refuse Municipal /Commercial /Industria
                        "Organic Food & Drink AD", "Biomass (Wood Pellets)", "Outsi
    "Scope": ["Scope 2", "Scope 3", "Scope 1", "Scope 1", "Scope 1", "Scope 1", "Sc
              "Scope 3", "Scope 3", "Scope 3", "Scope 1", "Outside of Scopes"],
    "Consumption data": [5553075, 5553075, 49673195, 35803, 38320, 3491, 25511, 23
    "Units": ["kWh", "kWh", "kWh", "litres", "litres", "litres", "litres", "m3", "r
    "Emission factor": [0.41205, 0.03727, 0.183996818, 2.965717775, 2.532318775, 2
}
```

In [14]:

```
df_breakdown = pd.DataFrame(data_breakdown)
```

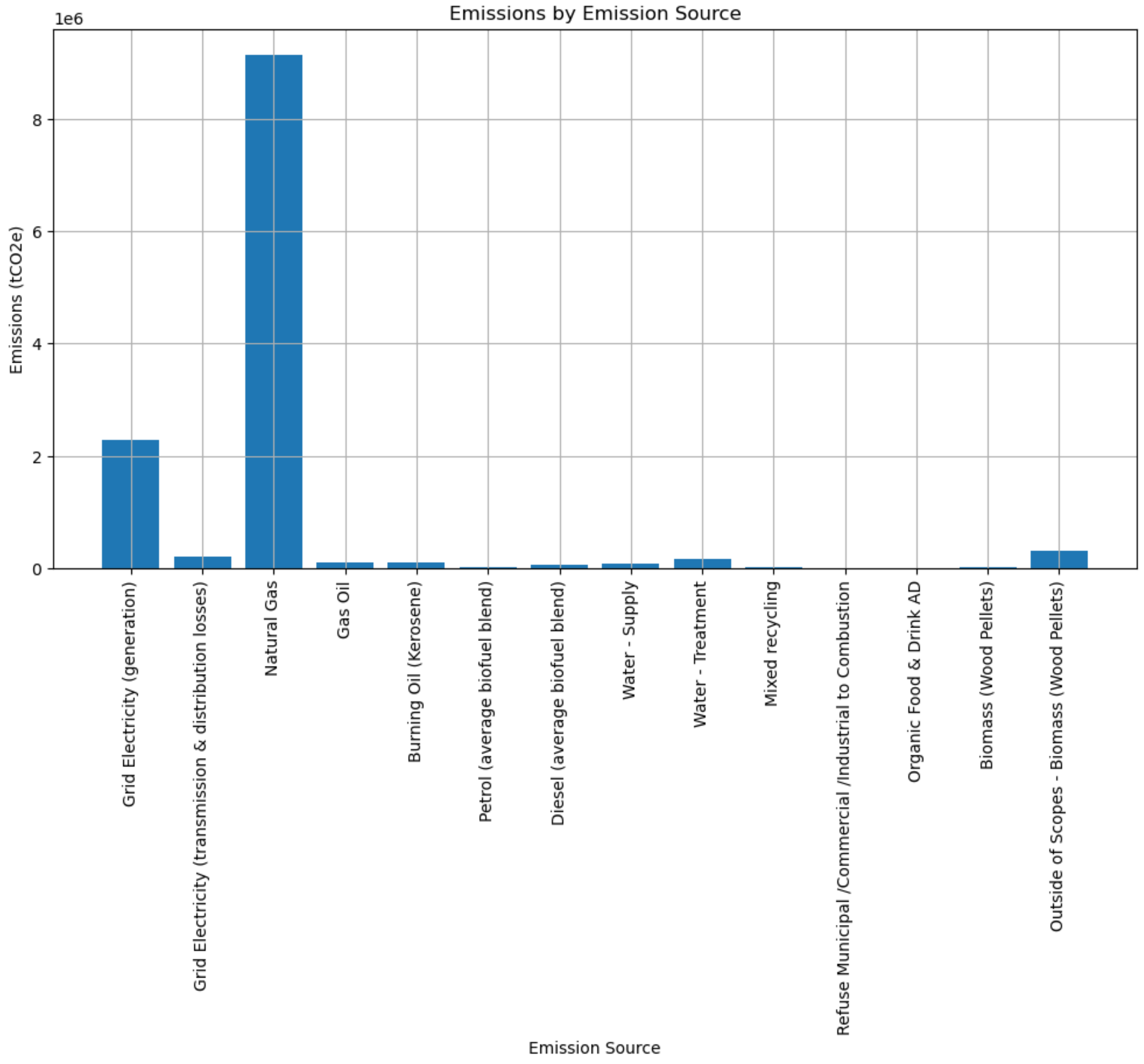
In [15]:

```
# Calculate emissions for each emission source
df_breakdown["Emissions (tCO2e)"] = df_breakdown["Consumption data"] * df_breakdowr

# Print the DataFrame with emission source breakdown
print(df_breakdown)
```

```
                                      Emission source              Scope  \
0                       Grid Electricity (generation)            Scope 2   
1   Grid Electricity (transmission & distribution ...            Scope 3   
2                                         Natural Gas            Scope 1   
3                                             Gas Oil            Scope 1   
4                              Burning Oil (Kerosene)            Scope 1   
5                      Petrol (average biofuel blend)            Scope 1   
6                      Diesel (average biofuel blend)            Scope 1   
7                                      Water - Supply            Scope 3   
8                                   Water - Treatment            Scope 3   
9                                     Mixed recycling            Scope 3   
10  Refuse Municipal /Commercial /Industrial to Co...            Scope 3   
11                            Organic Food & Drink AD            Scope 3   
12                             Biomass (Wood Pellets)            Scope 1   
13         Outside of Scopes - Biomass (Wood Pellets)  Outside of Scopes   

    Consumption data   Units  Emission factor  Emissions (tCO2e)  
0            5553075     kWh         0.412050       2.288145e+06  
1            5553075     kWh         0.037270       2.069631e+05  
2           49673195     kWh         0.183997       9.139710e+06  
3              35803  litres         2.965718       1.061816e+05  
4              38320  litres         2.532319       9.703846e+04  
5               3491  litres         2.196974       7.669636e+03  
6              25511  litres         2.611625       6.662517e+04  
7             238448      m3         0.344000       8.202611e+04  
8             230623      m3         0.708000       1.632811e+05  
9               1121  tonnes        21.000000       2.354100e+04  
10               165  tonnes        21.000000       3.465000e+03  
11                38  tonnes        21.000000       7.980000e+02  
12            882760     kWh         0.013070       1.153767e+04  
13            882760     kWh         0.349000       3.080832e+05  
```

In [16]:

```python
# Plot emissions by emission source
plt.figure(figsize=(12, 6))
plt.bar(df_breakdown["Emission source"], df_breakdown["Emissions (tCO2e)"])
plt.xlabel("Emission Source")
plt.ylabel("Emissions (tCO2e)")
plt.title("Emissions by Emission Source")
plt.xticks(rotation=90)
plt.grid(True)
plt.show()
```



In [17]:

```
# Summary of renewable energy generation and consumption
data_renewable = {
    "Technology": ["Biomass"],
    "Total consumed by the organisation (kWh)": [0],
    "Total exported (kWh)": [0],
    "Total consumed by the organisation (kWh) [Heat]": [882760],
    "Total exported (kWh) [Heat]": [0],
    "Comments": ["Lyons Crescent Biomass Boiler"]
}
```

In [19]:

```
df_renewable = pd.DataFrame(data_renewable)
```

In [21]:

```
# Print the DataFrame with renewable energy information
df_renewable.head()
```

Out[21]:

	Technology	Total consumed by the organisation (kWh)	Total exported (kWh)	Total consumed by the organisation (kWh) [Heat]	Total exported (kWh) [Heat]	Comments
0	Biomass	0	0	882760	0	Lyons Crescent Biomass Boiler

In [22]:

```
# Climate change targets
data_targets = {
    "Name of Target": ["Carbon Reduction", "Recycling of Waste"],
    "Type of Target": ["percentage", "percentage"],
    "Target": [38, 90],
    "Units": ["total % reduction", "percentage"],
    "Boundary/scope of Target": ["All emissions", "Waste"],
    "Progress against target": [28, 85],
    "Year used as baseline": ["2007/08", "2018/19"],
    "Baseline figure": [16651, None],
    "Units of baseline": ["tCO2e", None],
    "Target completion year": ["2020/21", "2018/19"],
    "Comments": ["", "Reach a 90% recycling level of all waste produced by the Univ
}
```

In [23]:

```
df_targets = pd.DataFrame(data_targets)

# Print the DataFrame with climate change targets
print(df_targets)
```

```
       Name of Target Type of Target  Target              Units  \
0    Carbon Reduction     percentage      38  total % reduction   
1  Recycling of Waste     percentage      90         percentage   

  Boundary/scope of Target  Progress against target Year used as baseline  \
0            All emissions                       28               2007/08   
1                    Waste                       85               2018/19   

   Baseline figure Units of baseline Target completion year  \
0          16651.0             tCO2e                2020/21   
1              NaN              None                2018/19   

                                            Comments  
0                                                     
1  Reach a 90% recycling level of all waste produ...  
```

Report 2017

In [33]:

```
import pandas as pd

# Creating the dataset for the "2017 Perform EDA" report
data = {
#     # Section 1 - General Information
#     "Name of reporting body": ["University of Stirling"],
#     "Type of body": ["Educational Institutions"],
#     "Highest number of full-time equivalent staff": [1618.5],
#     "Metrics used by the body": [
#         {"Metric": "Floor area", "Unit": "m2", "Value": 158688, "Comments": "Excl
#         {"Metric": "Number of full-time students", "Unit": "number FTS", "Value"
#         {"Metric": "Other (specify in comments)", "Unit": "other (specify in comm
#     ],
#     "Overall budget of the body (£/annum)": [117337000],

    # Section 3 - Emissions, Targets, and Projects
    # Emissions data
    "Reference Year": ["2007/08", "2008/09", "2009/10", "2010/11", "2011/12", "201
    "Year": [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017],
    "Scope1": [6424, 6831, 6720, 6659, 6099, 6338, 5315, 8672, 9417, 8863],
    "Scope2": [9939, 10444, 10107, 9791, 9348, 7957, 8350, 3477, 2288, 2174],
    "Scope3": [289, 355, 340, 704, 572, 448, 403, 397, 285, 284]}



# Create the DataFrame
df = pd.DataFrame(data)
```

```
# Print the DataFrame
print(df)
```

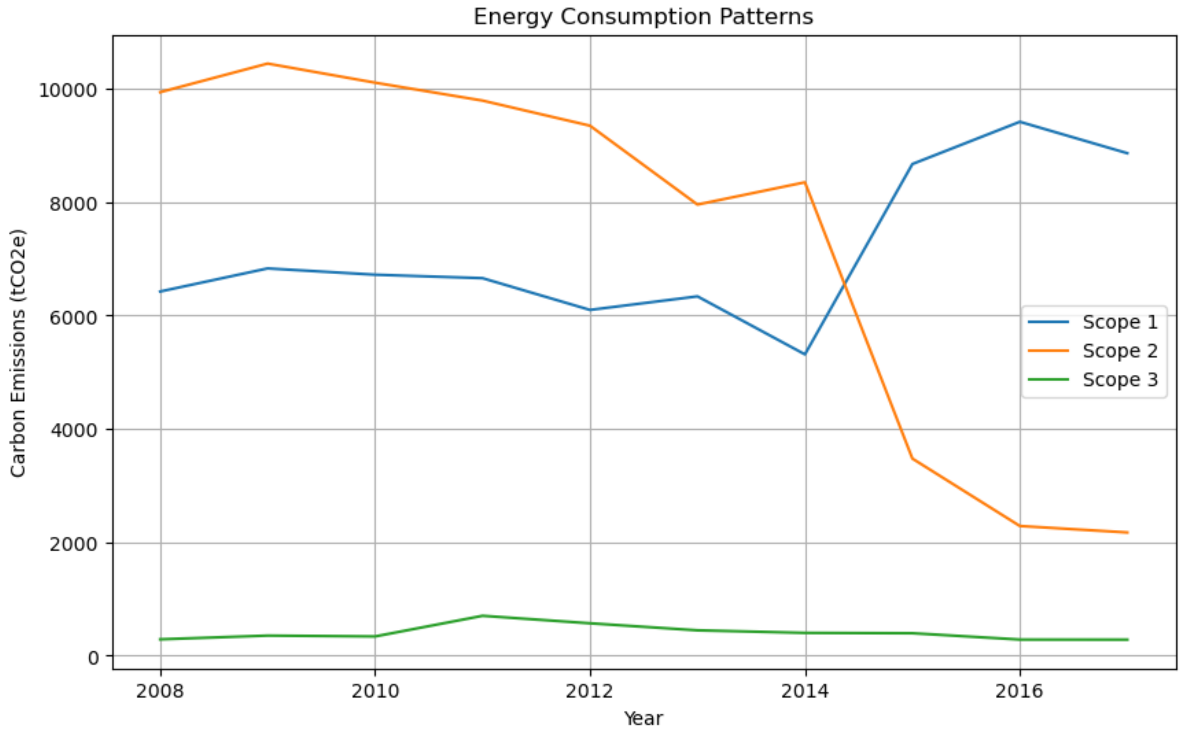
```
  Reference Year  Year  Scope1  Scope2  Scope3
0        2007/08  2008    6424    9939     289
1        2008/09  2009    6831   10444     355
2        2009/10  2010    6720   10107     340
3        2010/11  2011    6659    9791     704
4        2011/12  2012    6099    9348     572
5        2012/13  2013    6338    7957     448
6        2013/14  2014    5315    8350     403
7        2014/15  2015    8672    3477     397
8        2015/16  2016    9417    2288     285
9        2016/17  2017    8863    2174     284
```

In [34]:

```
# Calculate total carbon emissions (Scope1 + Scope2 + Scope3) for each year
df["Total Emissions"] = df["Scope1"] + df["Scope2"] + df["Scope3"]
```

In [35]:

```
# Plot energy consumption patterns
plt.figure(figsize=(10, 6))
plt.plot(df["Year"], df["Scope1"], label="Scope 1")
plt.plot(df["Year"], df["Scope2"], label="Scope 2")
plt.plot(df["Year"], df["Scope3"], label="Scope 3")
plt.xlabel("Year")
plt.ylabel("Carbon Emissions (tCO2e)")
plt.title("Energy Consumption Patterns")
plt.legend()
plt.grid(True)
plt.show()
```



In [36]:

```
# Assess carbon costs based on a given carbon price per tonne
carbon_price_per_tonne = 50  # Example: Assume carbon price is $50 per tonne
df["Carbon Cost ($)"] = df["Total Emissions"] * carbon_price_per_tonne
```

In [37]:

```
# Evaluate energy efficiency by dividing total carbon emissions by highest number
highest_staff = 1618.5  # Given highest number of full-time equivalent staff
df["Energy Efficiency (tCO2e per Staff)"] = df["Total Emissions"] / highest_staff
```

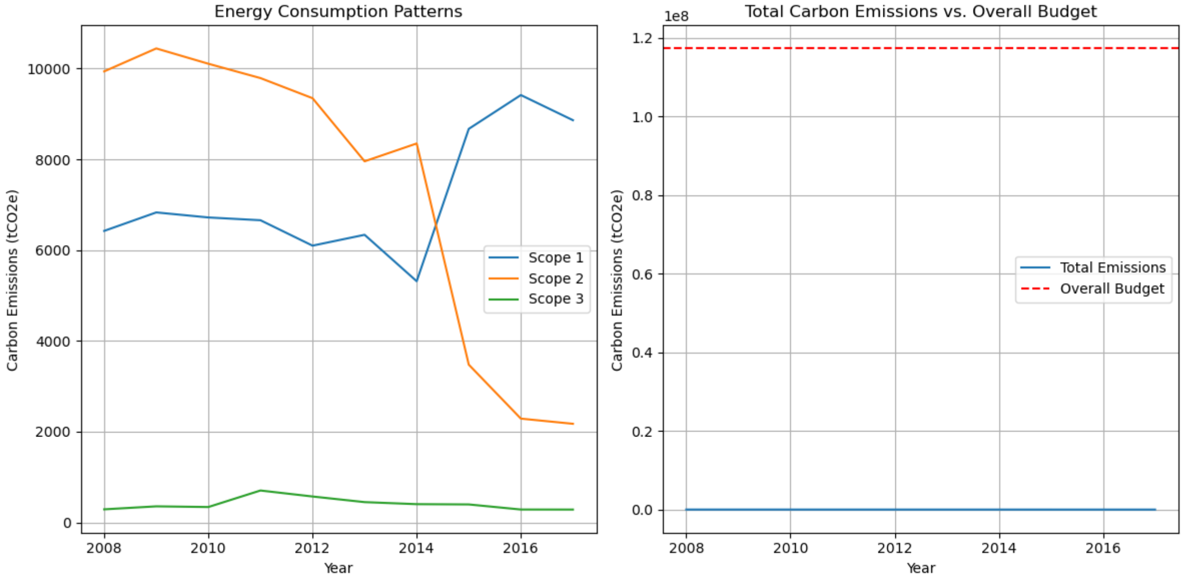
In [38]:

```python
# Provide Data-driven Insights
print("Data-driven Insights:")
print(df)
```

```
Data-driven Insights:
   Reference Year  Year  Scope1  Scope2  Scope3  Total Emissions  \
0         2007/08  2008    6424    9939     289            16652   
1         2008/09  2009    6831   10444     355            17630   
2         2009/10  2010    6720   10107     340            17167   
3         2010/11  2011    6659    9791     704            17154   
4         2011/12  2012    6099    9348     572            16019   
5         2012/13  2013    6338    7957     448            14743   
6         2013/14  2014    5315    8350     403            14068   
7         2014/15  2015    8672    3477     397            12546   
8         2015/16  2016    9417    2288     285            11990   
9         2016/17  2017    8863    2174     284            11321   

   Carbon Cost ($)  Energy Efficiency (tCO2e per Staff)  
0           832600                            10.288539  
1           881500                            10.892802  
2           858350                            10.606735  
3           857700                            10.598703  
4           800950                             9.897436  
5           737150                             9.109052  
6           703400                             8.691999  
7           627300                             7.751622  
8           599500                             7.408094  
9           566050                             6.994748  
```

In [40]:

```python
# Plot energy consumption patterns
plt.figure(figsize=(12, 6))
plt.subplot(1, 2, 1)
plt.plot(df["Year"], df["Scope1"], label="Scope 1")
plt.plot(df["Year"], df["Scope2"], label="Scope 2")
plt.plot(df["Year"], df["Scope3"], label="Scope 3")
plt.xlabel("Year")
plt.ylabel("Carbon Emissions (tCO2e)")
plt.title("Energy Consumption Patterns")
plt.legend()
plt.grid(True)

# Plot total carbon emissions vs. overall budget
plt.subplot(1, 2, 2)
plt.plot(df["Year"], df["Total Emissions"], label="Total Emissions")
plt.axhline(y=117337000, color="red", linestyle="--", label="Overall Budget")
plt.xlabel("Year")
plt.ylabel("Carbon Emissions (tCO2e)")
plt.title("Total Carbon Emissions vs. Overall Budget")
plt.legend()
plt.grid(True)

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



Report 2018

```
In [42]: # Part 1: PROFILE OF REPORTING BODY
         reporting_body_data = {
             "Name of reporting body": "University of Stirling",
             "Type of body": "Educational Institutions",
             "Highest number of full-time equivalent staff in the body during the report yea
             "Metrics used by the body": [
                 {"Metric": "Floor area", "Unit": "m2", "Value": 164811, "Comments": "Exclu
                 {"Metric": "Number of full-time equivalent students", "Unit": "number FTES
                 {"Metric": "Other (Please specify in the comments)", "Unit": "other (speci
             ],
             "Overall budget of the body": "https://www.stir.ac.uk/about/professional-servi
             "Report year": "Academic"
         }
```

```
In [43]: # Part 3a: EMISSIONS, TARGETS AND PROJECTS
         emissions_data = {
             "Year": ["2007/08", "2008/09", "2009/10", "2010/11", "2011/12", "2012/13", "20
             "Scope1": [6424, 6831, 6720, 6659, 6099, 6338, 5315, 8672, 9417, 8863, 9188, 8
             "Scope2": [9939, 10444, 10107, 9791, 9348, 7957, 8350, 3477, 2288, 2174, 1680,
             "Scope3": [289, 355, 340, 704, 572, 448, 403, 397, 285, 284, 303, 271]
         }
```

```
In [44]: # Part 3b: Breakdown of emission sources
         emission_sources_data = {
             "Emission source": ["Natural Gas", "Gas Oil", "Burning Oil (Kerosene)", "Petro
                                 "Water - Supply", "Water - Treatment", "Mixed recycling",
                                 "Organic Food & Drink AD", "Biomass (Wood Pellets)", "Grid
             "Scope": ["Scope 1", "Scope 1", "Scope 1", "Scope 1", "Scope 1", "Scope 3", "S
             "Consumption data": ["44894030 kWh", "356515 kWh", "462425 kWh", "6845 litres"
             "Emission factor": ["0.18385 kg CO2e/kWh", "0.25676 kg CO2e/kWh", "0.24675 kg 
                                 "0.344 kg CO2e/m3", "0.708 kg CO2e/m3", "21.354 kg CO2e/to
             "Emissions (tCO2e)": [8253.8, 91.5, 114.1, 15.1, 60.9, 65.4, 163.3, 8.1, 19.4,
             "Comments": ["Total gas incl CHP.Data from supplier's invoices.", "Based on re
                          "Data from fuel supplier.", "Data from fuel supplier.", "Data fro
                          "Data from waste contractor.", "Data from waste contractor.", "Da
                          "100% procured green grid energy.EV Sub Meter.Doesn't include Tra
         }
```

In [45]:

```
# Part 3c: Generation, consumption, and export of renewable energy
renewable_energy_data = {
    "Technology": ["Biomass"],
    "Total consumed by the organisation (kWh)": [944500],
    "Total exported (kWh)": [0],
    "Total consumed by the organisation (kWh)_Heat": [0],
    "Total exported (kWh)_Heat": [0],
    "Comments": ["Lyons Crescent"]
}
```

In [46]:

```
# Part 3d: Targets
targets_data = {
    "Name of Target": ["UOS Carbon Reduction", "UOS Waste Recycling"],
    "Type of Target": ["percentage", "percentage"],
    "Target": [38, 90],
    "Units": ["total % reduction", "%"],
    "Boundary/scope of Target": ["All emissions", "Waste"],
    "Progress against target": [38.1, 90.98],
    "Year used as baseline": ["2007/08", "2018/19"],
    "Baseline figure": [16652, 90],
    "Units of baseline": ["tCO2e", "%"],
    "Target completion year": ["2019/20", None],
    "Comments": ["The baseline year was set under the guidance of the Carbon Trust
                 "90% recycling of all waste target. Includes waste used to create
}
```

In [47]:

```
# Part 3e: Estimated total annual carbon savings from all projects implemented by t
carbon_savings_data = {
    "Emissions Source": ["Electricity", "Natural gas", "Other heating fuels", "Wast
    "Total estimated annual carbon savings (tCO2e)": [152, 182, 5, 0, 62, 0, 0, 0]
    "Comments": [None, None, None, None, None, None, None, None]
}
```

In [48]:

```
# Part 3f: Detail the top 10 carbon reduction projects to be carried out by the boo
carbon_reduction_projects_data = {
    "Project name": ["Replacement of Lighting in Cottrell with LEDs", "Building Man
                     "Building Management System (BMS) Improvements", "Energy, Wate
                     "Energy, Water & Waste Reduction Campaign", "Energy, Water & W
                     "Interhall Competition"],
    "Funding source": ["Salix", "University of Stirling", "University of Stirling",
                       "University of Stirling", "University of Stirling", "Univers
    "First full year of CO2e savings": ["2019/20", "2019/20", "2019/20", "2019/20",
    "Are these savings figures estimated or actual?": ["Estimated", "Estimated", "E
    "Capital cost (£)": [180000, 0, 0, 0, 0, 0, 0, 0, 1400000, 0],
    "Operational cost (£/annum)": [0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
    "Project lifetime (years)": [15, 15, 15, 15, 1, 1, 1, 1, 50, 1],
    "Primary fuel/emission source saved": ["Grid Electricity", "Natural Gas", "Gri
    "Estimated carbon savings per year (tCO2e/annum)": [91, 92, 19, 3, 90, 17, 3, 1
    "Estimated costs savings (£/annum)": [20000, 12800, 4800, 5300, 11000, 4300, 40
    "Behaviour Change": ["Demonstrates that the University supports carbon reductio
                         "Encouraging users to notify Estates when there are heatin
                         "Encouraging users to notify Estates when there are heatin
                         "Encouraging users to notify Estates when there are heatin
                         "Promoted reduction through various communication channels
                         "Promoted reduction through various communication channels
                         "Promoted reduction through various communication channels
                         "Promoted reduction through various communication channels
                         "This significant investment by the University demonstrate
                         "Engaged with students promoting energy efficiency. A £1k 
    "Comments": ["Project was postponed due to resources being diverted to deal wit
                 "Interrogating BMS to identify conflict between heating and coolin
                 "Interrogating BMS to identify conflict between heating and coolin
                 "Interrogating BMS to identify conflict between heating and coolin
                 "Funded from internal staff resource.",
                 "Funded from internal staff resource.",
                 "Funded from internal staff resource.",
                 "Funded from internal staff resource.",
                 "It is anticipated that savings will further increase to 72 tCO2 a
                 "This project was run by the Students Union in collaboration with 
}
```

In [49]:

```
# Create DataFrames
reporting_body_df = pd.DataFrame([reporting_body_data])
emissions_df = pd.DataFrame(emissions_data)
emission_sources_df = pd.DataFrame(emission_sources_data)
renewable_energy_df = pd.DataFrame(renewable_energy_data)
targets_df = pd.DataFrame(targets_data)
carbon_savings_df = pd.DataFrame(carbon_savings_data)
carbon_reduction_projects_df = pd.DataFrame(carbon_reduction_projects_data)
```

In [50]:

```
# Printing the DataFrames (You can use these DataFrames for further analysis and pr
print("Reporting Body Data:")
print(reporting_body_df)

print("\nEmissions Data:")
print(emissions_df)

print("\nEmission Sources Data:")
print(emission_sources_df)

print("\nRenewable Energy Data:")
print(renewable_energy_df)

print("\nTargets Data:")
print(targets_df)

print("\nCarbon Savings Data:")
print(carbon_savings_df)

print("\nCarbon Reduction Projects Data:")
print(carbon_reduction_projects_df)
```

```
Reporting Body Data:
   Name of reporting body              Type of body  \
0  University of Stirling  Educational Institutions

   Highest number of full-time equivalent staff in the body during the report year
\
0                                               1436

                            Metrics used by the body  \
0  [{'Metric': 'Floor area', 'Unit': 'm2', 'Value...

                          Overall budget of the body Report year
0  https://www.stir.ac.uk/about/professional-serv...    Academic

Emissions Data:
       Year  Scope1  Scope2  Scope3
0   2007/08    6424    9939     289
1   2008/09    6831   10444     355
2   2009/10    6720   10107     340
3   2010/11    6659    9791     704
4   2011/12    6099    9348     572
5   2012/13    6338    7957     448
6   2013/14    5315    8350     403
7   2014/15    8672    3477     397
8   2015/16    9417    2288     285
9   2016/17    8863    2174     284
10  2017/18    9188    1680     303
11  2018/19    8535    1503     271

Emission Sources Data:
                                      Emission source    Scope  \
0                                         Natural Gas  Scope 1
1                                             Gas Oil  Scope 1
2                               Burning Oil (Kerosene)  Scope 1
3                       Petrol (average biofuel blend)  Scope 1
4                       Diesel (average biofuel blend)  Scope 1
5                                       Water - Supply  Scope 3
6                                    Water - Treatment  Scope 3
7                                      Mixed recycling  Scope 3
8   Refuse Municipal /Commercial /Industrial to Co...  Scope 3
9                             Organic Food & Drink AD  Scope 3
10                             Biomass (Wood Pellets)  Scope 1
11                       Grid Electricity (generation)  Scope 2
12                       Grid Electricity (generation)  Scope 2

   Consumption data        Emission factor  Emissions (tCO2e)  \
0      44894030 kWh    0.18385 kg CO2e/kWh             8253.8
1        356515 kWh    0.25676 kg CO2e/kWh               91.5
2        462425 kWh    0.24675 kg CO2e/kWh              114.1
3       6845 litres  2.20904 kg CO2e/litre               15.1
4      23473 litres  2.59411 kg CO2e/litre               60.9
5         190068 m3       0.344 kg CO2e/m3               65.4
6         230623 m3       0.708 kg CO2e/m3              163.3
7        379 tonnes   21.354 kg CO2e/tonne                8.1
8        908 tonnes  21.3538 kg CO2e/tonne               19.4
9         56 tonnes  10.2039 kg CO2e/tonne                0.6
10       944500 kWh    0.01563 kg CO2e/kWh               14.8
11      5851159 kWh     0.2556 kg CO2e/kWh             1495.6
12        29420 kWh     0.2556 kg CO2e/kWh                7.5

                                              Comments
0   Total gas incl CHP.Data from supplier's invoices.
1     Based on readings from University owned meters.
2       Based on readings from University owned meter
```

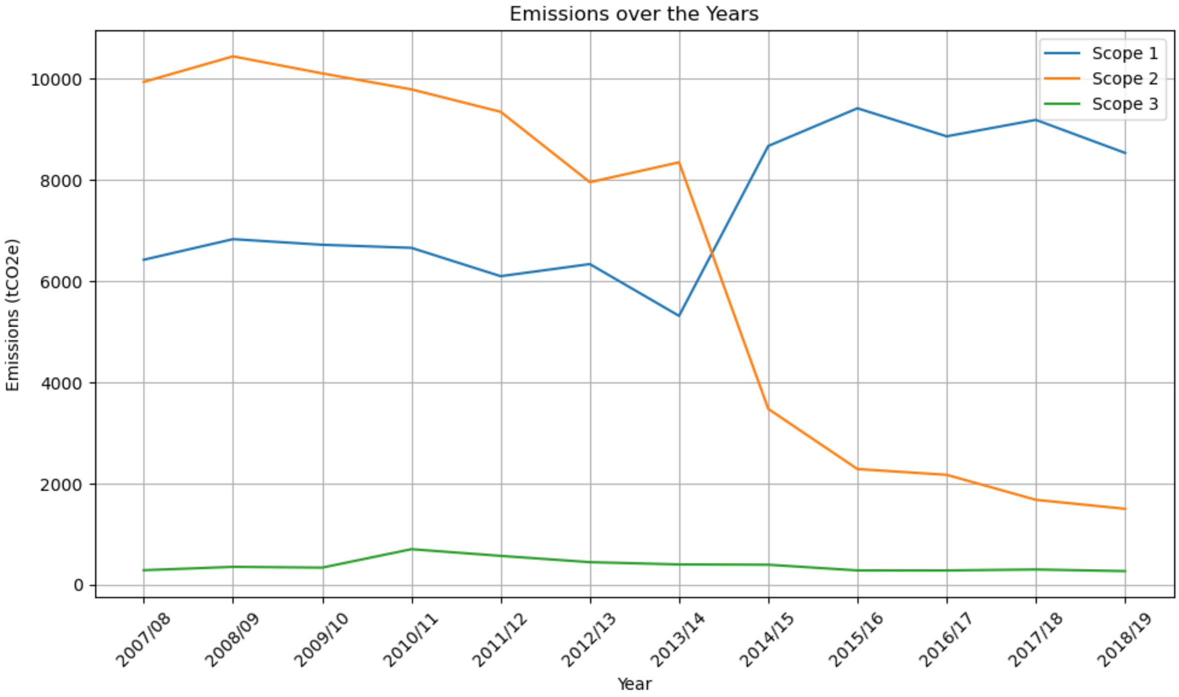
```
3                            Data from fuel supplier.
4                            Data from fuel supplier.
5                      Data from supplier's invoices.
6                      Data from supplier's invoices.
7                         Data from waste contractor.
8                         Data from waste contractor.
9                         Data from waste contractor.
10                   Data from Link Housing invoices.
11  100% procured green grid energy.Total Imported...
12  100% procured green grid energy.EV Sub Meter.D...

Renewable Energy Data:
  Technology  Total consumed by the organisation (kWh)  Total exported (kWh)  \
0    Biomass                                    944500                     0   

   Total consumed by the organisation (kWh)_Heat  Total exported (kWh)_Heat  \
0                                              0                          0   

         Comments  
0  Lyons Crescent  

Targets Data:
        Name of Target Type of Target  Target              Units  \
0  UOS Carbon Reduction     percentage      38  total % reduction   
1   UOS Waste Recycling     percentage      90                  %   

  Boundary/scope of Target  Progress against target Year used as baseline  \
0            All emissions                    38.10              2007/08   
1                    Waste                    90.98              2018/19   

   Baseline figure Units of baseline Target completion year  \
0            16652             tCO2e                2019/20   
1               90                 %                   None   

                                            Comments  
0  The baseline year was set under the guidance o...  
1  90% recycling of all waste target. Includes wa...  

Carbon Savings Data:
               Emissions Source  Total estimated annual carbon savings (tCO2e)  \
0                   Electricity                                            152   
1                   Natural gas                                            182   
2            Other heating fuels                                             5   
3                         Waste                                              0   
4             Water and sewerage                                            62   
5                Business Travel                                             0   
6                Fleet transport                                             0   
7  Other (specify in comments)                                               0   

  Comments  
0     None  
1     None  
2     None  
3     None  
4     None  
5     None  
6     None  
7     None  

Carbon Reduction Projects Data:
                                  Project name          Funding source  \
0  Replacement of Lighting in Cottrell with LEDs                   Salix   
1  Building Management System (BMS) Improvements  University of Stirling   
2  Building Management System (BMS) Improvements  University of Stirling   
```

```
3  Building Management System (BMS) Improvements  University of Stirling
4       Energy, Water & Waste Reduction Campaign  University of Stirling
5       Energy, Water & Waste Reduction Campaign  University of Stirling
6       Energy, Water & Waste Reduction Campaign  University of Stirling
7       Energy, Water & Waste Reduction Campaign  University of Stirling
8               Replacement of Campus Water Main  University of Stirling
9                          Interhall Competition  University of Stirling

  First full year of CO2e savings  \
0                         2019/20   
1                         2019/20   
2                         2019/20   
3                         2019/20   
4                         2018/19   
5                         2018/19   
6                         2018/19   
7                         2018/19   
8                         2019/20   
9                         2018/19   

  Are these savings figures estimated or actual?  Capital cost (£)  \
0                                      Estimated            180000   
1                                      Estimated                 0   
2                                      Estimated                 0   
3                                      Estimated                 0   
4                                      Estimated                 0   
5                                      Estimated                 0   
6                                      Estimated                 0   
7                                      Estimated                 0   
8                                      Estimated           1400000   
9                                      Estimated                 0   

   Operational cost (£/annum)  Project lifetime (years)  \
0                           0                        15   
1                           0                        15   
2                           0                        15   
3                           0                        15   
4                           0                         1   
5                           0                         1   
6                           0                         1   
7                           0                         1   
8                           0                        50   
9                           0                         1   

  Primary fuel/emission source saved  \
0                   Grid Electricity   
1                        Natural Gas   
2                   Grid Electricity   
3                            Gas Oil   
4                        Natural Gas   
5                   Grid Electricity   
6                     Water - Supply   
7                            Gas Oil   
8                     Water - Supply   
9                   Grid Electricity   

   Estimated carbon savings per year (tCO2e/annum)  \
0                                               91   
1                                               92   
2                                               19   
3                                                3   
4                                               90   
5                                               17   
6                                                3   
```

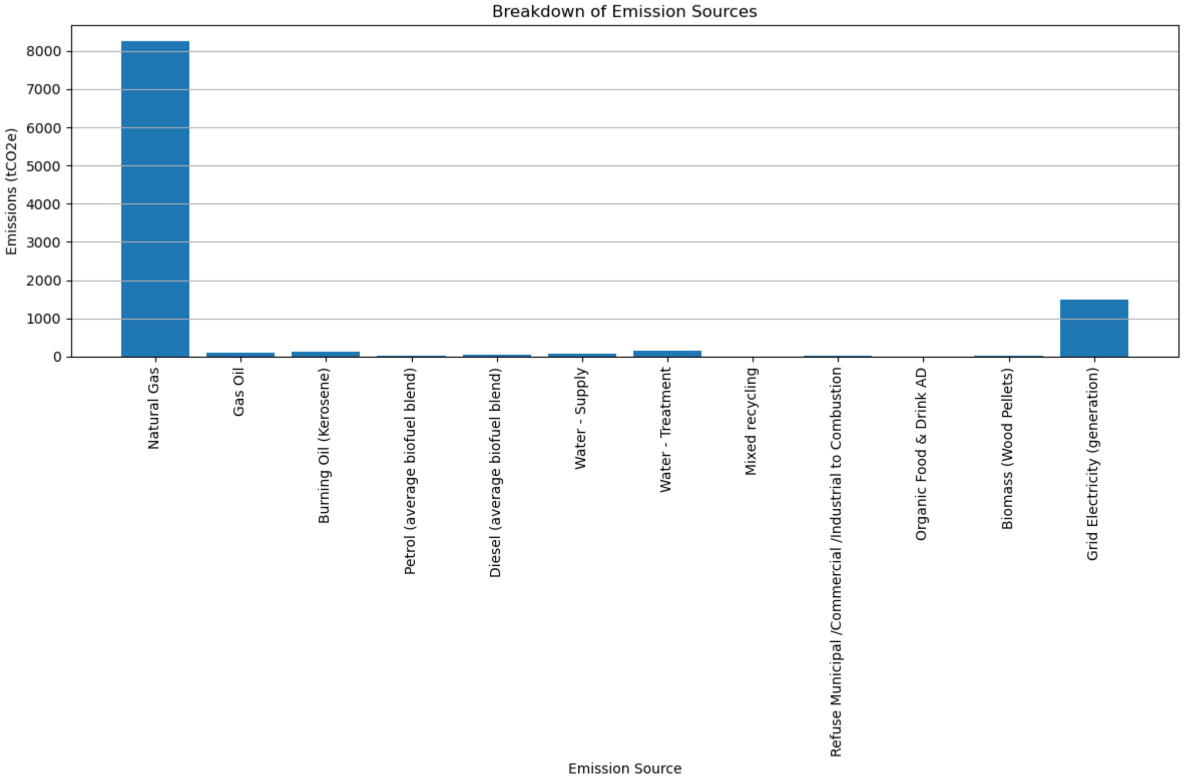
```
7                                                2
8                                               59
9                                               25

   Estimated costs savings (£/annum)  \
0                              20000   
1                              12800   
2                               4800   
3                               5300   
4                              11000   
5                               4300   
6                               4600   
7                                470   
8                              41000   
9                               5800   

                                    Behaviour Change  \
0  Demonstrates that the University supports carb...   
1  Encouraging users to notify Estates when there...   
2  Encouraging users to notify Estates when there...   
3  Encouraging users to notify Estates when there...   
4  Promoted reduction through various communicati...   
5  Promoted reduction through various communicati...   
6  Promoted reduction through various communicati...   
7  Promoted reduction through various communicati...   
8  This significant investment by the University ...   
9  Engaged with students promoting energy efficie...   

                                            Comments  
0  Project was postponed due to resources being d...  
1  Interrogating BMS to identify conflict between...  
2  Interrogating BMS to identify conflict between...  
3  Interrogating BMS to identify conflict between...  
4               Funded from internal staff resource.  
5               Funded from internal staff resource.  
6               Funded from internal staff resource.  
7               Funded from internal staff resource.  
8  It is anticipated that savings will further in...  
9  This project was run by the Students Union in ...  
```

In [51]:

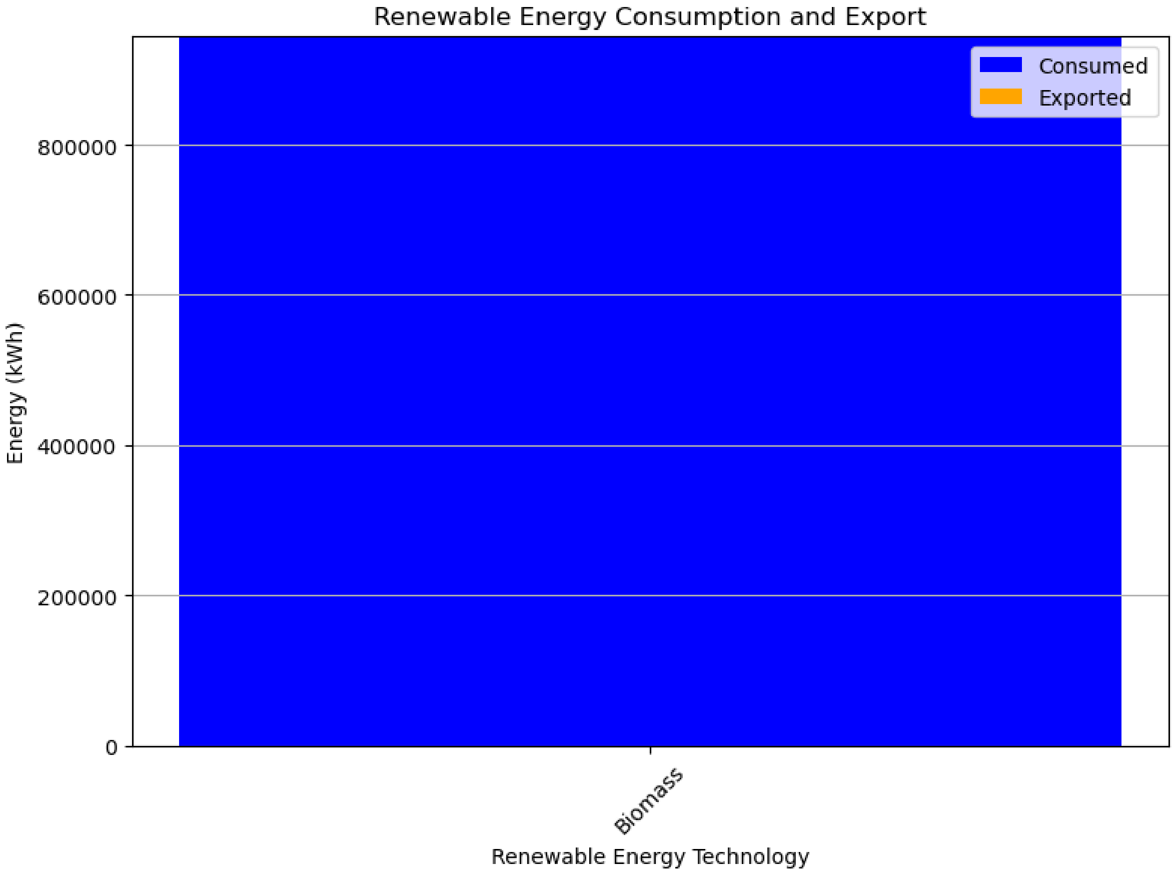
```python
plt.figure(figsize=(10, 6))
plt.plot(emissions_df["Year"], emissions_df["Scope1"], label="Scope 1")
plt.plot(emissions_df["Year"], emissions_df["Scope2"], label="Scope 2")
plt.plot(emissions_df["Year"], emissions_df["Scope3"], label="Scope 3")
plt.xlabel("Year")
plt.ylabel("Emissions (tCO2e)")
plt.title("Emissions over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True)
plt.tight_layout()
plt.show()
```



In [52]:
```
plt.figure(figsize=(12, 8))
plt.bar(emission_sources_df["Emission source"], emission_sources_df["Emissions (tC
plt.xlabel("Emission Source")
plt.ylabel("Emissions (tCO2e)")
plt.title("Breakdown of Emission Sources")
plt.xticks(rotation=90)
plt.grid(axis="y")
plt.tight_layout()
plt.show()
```

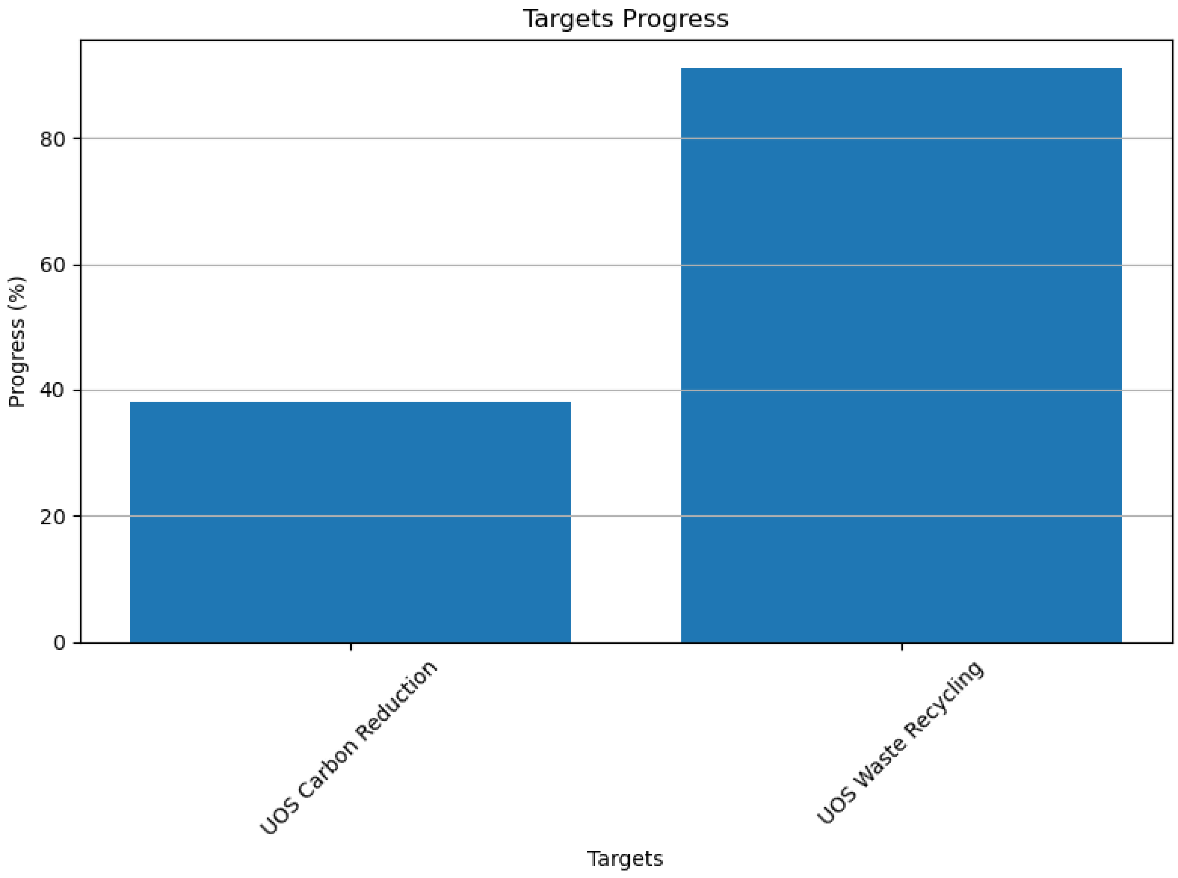


In [53]:
```
plt.figure(figsize=(8, 6))
plt.bar(renewable_energy_df["Technology"], renewable_energy_df["Total consumed by 
plt.bar(renewable_energy_df["Technology"], renewable_energy_df["Total exported (kWh
plt.xlabel("Renewable Energy Technology")
plt.ylabel("Energy (kWh)")
plt.title("Renewable Energy Consumption and Export")
plt.legend()
plt.xticks(rotation=45)
plt.grid(axis="y")
plt.tight_layout()
plt.show()
```



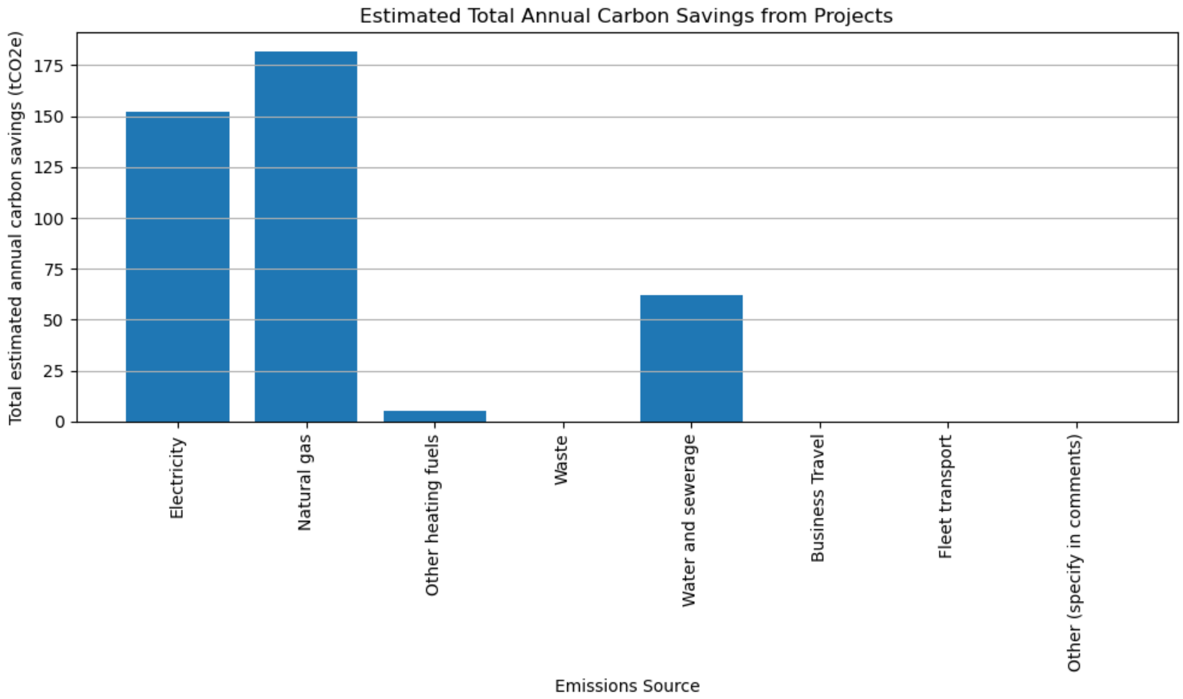
In [54]:

```
plt.figure(figsize=(8, 6))
plt.bar(targets_df["Name of Target"], targets_df["Progress against target"])
plt.xlabel("Targets")
plt.ylabel("Progress (%)")
plt.title("Targets Progress")
plt.xticks(rotation=45)
plt.grid(axis="y")
plt.tight_layout()
plt.show()
```



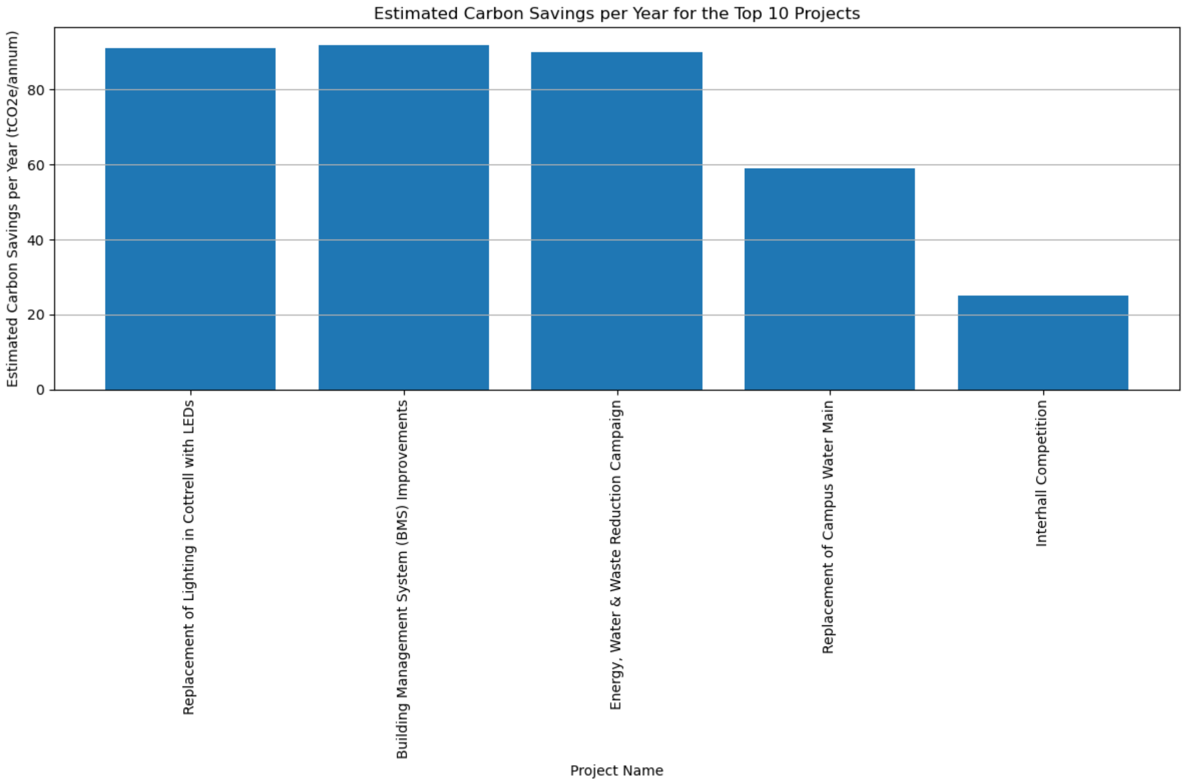
In [55]:

```
plt.figure(figsize=(10, 6))
plt.bar(carbon_savings_df["Emissions Source"], carbon_savings_df["Total estimated
plt.xlabel("Emissions Source")
plt.ylabel("Total estimated annual carbon savings (tCO2e)")
plt.title("Estimated Total Annual Carbon Savings from Projects")
plt.xticks(rotation=90)
plt.grid(axis="y")
plt.tight_layout()
plt.show()
```



In [56]:

```
plt.figure(figsize=(12, 8))
plt.bar(carbon_reduction_projects_df["Project name"], carbon_reduction_projects_df
plt.xlabel("Project Name")
plt.ylabel("Estimated Carbon Savings per Year (tCO2e/annum)")
plt.title("Estimated Carbon Savings per Year for the Top 10 Projects")
plt.xticks(rotation=90)
plt.grid(axis="y")
plt.tight_layout()
plt.show()
```



Report 2019

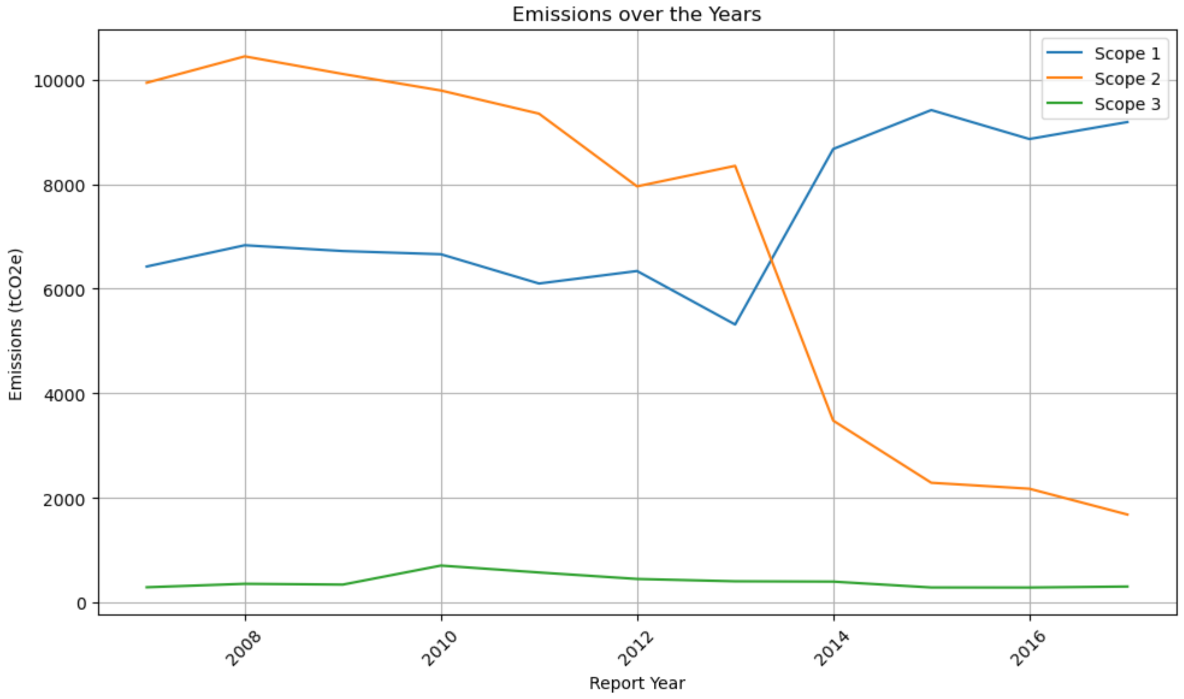
In [57]:

```
import pandas as pd

# Replace the data with the provided table data
data = {
    "Year": ["2007/08", "2008/09", "2009/10", "2010/11", "2011/12", "2012/13", "20
    "Scope1": [6424, 6831, 6720, 6659, 6099, 6338, 5315, 8672, 9417, 8863, 9188],
    "Scope2": [9939, 10444, 10107, 9791, 9348, 7957, 8350, 3477, 2288, 2174, 1680]
    "Scope3": [289, 355, 340, 704, 572, 448, 403, 397, 285, 284, 303]
}

emissions_df = pd.DataFrame(data)

# Renaming columns to be more descriptive
emissions_df.rename(columns={"Year": "Report Year"}, inplace=True)

# Converting the Report Year to numeric year
emissions_df["Report Year"] = emissions_df["Report Year"].apply(lambda x: int(x.sp

# Displaying the emissions data
print(emissions_df)
```

```
    Report Year  Scope1  Scope2  Scope3
0          2007    6424    9939     289
1          2008    6831   10444     355
2          2009    6720   10107     340
3          2010    6659    9791     704
4          2011    6099    9348     572
5          2012    6338    7957     448
6          2013    5315    8350     403
7          2014    8672    3477     397
8          2015    9417    2288     285
9          2016    8863    2174     284
10         2017    9188    1680     303
```

In [58]:

```python
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
plt.plot(emissions_df["Report Year"], emissions_df["Scope1"], label="Scope 1")
plt.plot(emissions_df["Report Year"], emissions_df["Scope2"], label="Scope 2")
plt.plot(emissions_df["Report Year"], emissions_df["Scope3"], label="Scope 3")
plt.xlabel("Report Year")
plt.ylabel("Emissions (tCO2e)")
plt.title("Emissions over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True)
plt.tight_layout()
plt.show()
```



In [59]:

```python
# Replace the data with the provided table data
emission_sources_data = {
    "Emission Source": ["Natural Gas", "Gas Oil", "Burning Oil (Kerosene)", "Petro
                        "Diesel (average biofuel blend)", "Water - Supply", "Water
                        "Refuse Municipal /Commercial /Industrial to Combustion", 
                        "Grid Electricity (generation)", "Grid Electricity (genera
    "Scope": ["Scope 1", "Scope 1", "Scope 1", "Scope 1", "Scope 1", "Scope 3", "S
    "Consumption data": [48314659, 342348, 539582, 2742, 25240, 272080, 230623, 38
    "Units": ["kWh", "kWh", "kWh", "litres", "litres", "m3", "m3", "tonnes", "tonn
    "Emission factor": [0.18396, 0.27652, 0.24665, 2.20307, 2.62694, 0.344, 0.708,
    "Units of baseline": ["kg CO2e/kWh", "kg CO2e/kWh", "kg CO2e/kWh", "kg CO2e/li
    "Emissions (tCO2e)": [8888.0, 94.7, 133.1, 6.0, 66.3, 93.6, 163.3, 8.3, 22.5, 
    "Comments": ["Total gas incl CHP. Data from supplier's invoices.", "Based on r
```

```
                 "Based on readings from University-owned meter.", "Data from fuel
                 "Data from supplier's invoices.", "Data from supplier's invoices.'
                 "Data from waste contractor.", "Data from waste contractor.", "Dat
                 "Total Imported Grid minus EV Sub Meter. Doesn't include Transmiss
                 "EV Sub Meter. Doesn't include Transmission and Distribution so co
}

emission_sources_df = pd.DataFrame(emission_sources_data)

# Displaying the breakdown of emission sources
print(emission_sources_df)
```

```
                                      Emission Source    Scope  \
0                                         Natural Gas  Scope 1   
1                                             Gas Oil  Scope 1   
2                              Burning Oil (Kerosene)  Scope 1   
3                      Petrol (average biofuel blend)  Scope 1   
4                      Diesel (average biofuel blend)  Scope 1   
5                                      Water - Supply  Scope 3   
6                                   Water - Treatment  Scope 3   
7                                     Mixed recycling  Scope 3   
8   Refuse Municipal /Commercial /Industrial to Co...  Scope 3   
9                             Organic Food & Drink AD  Scope 3   
10                             Biomass (Wood Pellets)  Scope 1   
11                      Grid Electricity (generation)  Scope 2   
12                      Grid Electricity (generation)  Scope 2   

    Consumption data   Units  Emission factor Units of baseline  \
0           48314659     kWh          0.18396       kg CO2e/kWh   
1             342348     kWh          0.27652       kg CO2e/kWh   
2             539582     kWh          0.24665       kg CO2e/kWh   
3               2742  litres          2.20307     kg CO2e/litre   
4              25240  litres          2.62694     kg CO2e/litre   
5             272080      m3          0.34400        kg CO2e/m3   
6             230623      m3          0.70800        kg CO2e/m3   
7                389  tonnes         21.38420     kg CO2e/tonne   
8               1052  tonnes         21.38420     kg CO2e/tonne   
9                 59  tonnes         21.38420     kg CO2e/tonne   
10            937640     kWh          0.01506       kg CO2e/kWh   
11           5912738     kWh          0.28307       kg CO2e/kWh   
12             21904     kWh          0.28307       kg CO2e/kWh   

    Emissions (tCO2e)                                           Comments  
0              8888.0  Total gas incl CHP. Data from supplier's invoi...  
1                94.7     Based on readings from University-owned meter.  
2               133.1     Based on readings from University-owned meter.  
3                 6.0                           Data from fuel supplier.  
4                66.3                           Data from fuel supplier.  
5                93.6                     Data from supplier's invoices.  
6               163.3                     Data from supplier's invoices.  
7                 8.3                        Data from waste contractor.  
8                22.5                        Data from waste contractor.  
9                 1.3                        Data from waste contractor.  
10               14.1                   Data from Link Housing invoices.  
11             1673.7  Total Imported Grid minus EV Sub Meter. Doesn'...  
12                6.2  EV Sub Meter. Doesn't include Transmission and...  
```

In [60]:

```
# Replace the data with the provided table data
renewable_energy_data = {
    "Technology": ["Biomass"],
    "Total consumed by the organisation (kWh)": [937640],
    "Total exported (kWh)": [0],
    "Comments": ["Lyons Cres Residences"]
}
```

```
renewable_energy_df = pd.DataFrame(renewable_energy_data)

# Displaying the renewable energy consumption and export
print(renewable_energy_df)
```

```
  Technology  Total consumed by the organisation (kWh)  Total exported (kWh)  \
0    Biomass                                    937640                     0

               Comments
0  Lyons Cres Residences
```

In [62]:

```
# Replace the data with the provided table data
carbon_savings_data = {
    "Emissions Source": ["Electricity", "Natural gas", "Other heating fuels", "Was
                         "Fleet transport", "Other (specify in comments)"],
    "Total estimated annual carbon savings (tCO2e)": [59.1, 202.3, 5.2, 0.5, 2.6, 
    "Comments": ["Students' Union Interhall competition, Energy, Water & Waste Redu
                 "Energy, Water & Waste Reduction Campaign, Building Management Sys
                 "Energy, Water & Waste Reduction Campaign, Building Management Sys
                 "Energy, Water & Waste Reduction Campaign, Building Management Sys
                 "Energy, Water & Waste Reduction Campaign, Building Management Sys
                 "",
                 "",
                 ""]
}

carbon_savings_df = pd.DataFrame(carbon_savings_data)

# Displaying the estimated total annual carbon savings from projects
print(carbon_savings_df)
```

```
              Emissions Source  Total estimated annual carbon savings (tCO2e)  \
0                  Electricity                                           59.1
1                  Natural gas                                          202.3
2          Other heating fuels                                            5.2
3                        Waste                                            0.5
4           Water and sewerage                                            2.6
5              Business Travel                                            0.0
6              Fleet transport                                            0.0
7  Other (specify in comments)                                            0.0

                                            Comments
0  Students' Union Interhall competition, Energy,...
1  Energy, Water & Waste Reduction Campaign, Buil...
2  Energy, Water & Waste Reduction Campaign, Buil...
3  Energy, Water & Waste Reduction Campaign, Buil...
4  Energy, Water & Waste Reduction Campaign, Buil...
5
6
7
```

In [63]:

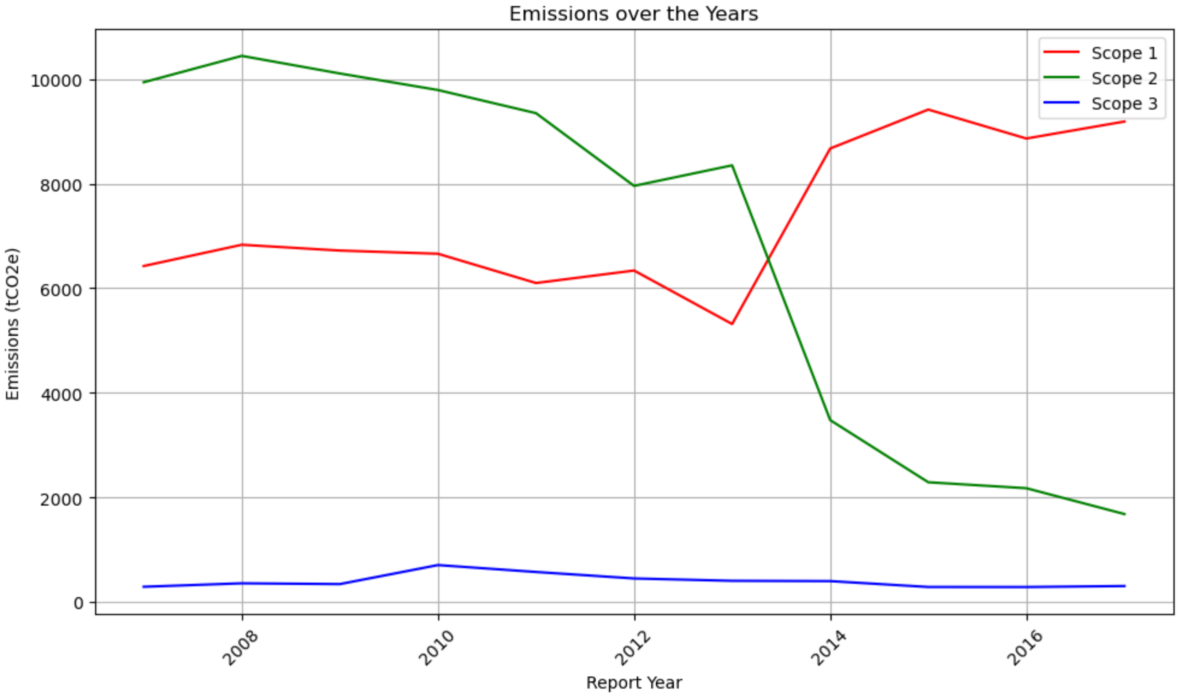
```
# Replace the data with the provided table data
carbon_reduction_projects_data = {
    "Project name": ["Building Management System (BMS) Improvements", "Building Ma
                     "Building Management System (BMS) Improvements", "Energy, Wat
                     "Energy, Water & Waste Reduction Campaign", "Energy, Water & 
                     "Energy, Water & Waste Reduction Campaign", "Interhall Compet
    "Funding source": ["University of Stirling", "University of Stirling", "Univer
                       "University of Stirling", "University of Stirling", "Univer
    "First full year of CO2e savings": ["2018/19", "2018/19", "2018/19", "2018/19"
    "Are these savings figures estimated or actual?": ["Estimated", "Estimated", "
    "Capital cost (£)": [0, 0, 0, 0, 0, 0, 0, 0],
    "Operational cost (£/annum)": [0, 0, 0, 0, 0, 0, 0, 0],
    "Project lifetime (years)": [0, 0, 0, 0, 0, 0, 0, 0],
    "Primary fuel/emission source saved": ["Grid Electricity", "Natural Gas", "Gas
    "Estimated carbon savings per year (tCO2e/annum)": [21.3, 112.5, 2.9, 17, 89.8
    "Estimated costs savings (£/annum)": [4910, 13784, 5265, 3918, 10998, 420, 459
    "Behaviour Change": ["Encouraging users to notify Estates when areas are out o
                         "Encouraging users to notify Estates when areas are out o
                         "Encouraging users to notify Estates when areas are out o
                         "Promoted reduction through various communication channel
                         "Promoted reduction through various communication channel
                         "Promoted reduction through various communication channel
                         "Changing culture in a fun way",
                         ""],
    "Comments": ["Interrogating BMS to identify conflict between heating and coolin
                 "Interrogating BMS to identify conflict between heating and coolin
                 "Interrogating BMS to identify conflict between heating and coolin
                 "Nominal costs", "Nominal costs", "Nominal costs", "Nominal costs
}

carbon_reduction_projects_df = pd.DataFrame(carbon_reduction_projects_data)

# Displaying the top 10 carbon reduction projects
print(carbon_reduction_projects_df)
```

```
                                    Project name          Funding source  \
0  Building Management System (BMS) Improvements  University of Stirling   
1  Building Management System (BMS) Improvements  University of Stirling   
2  Building Management System (BMS) Improvements  University of Stirling   
3       Energy, Water & Waste Reduction Campaign  University of Stirling   
4       Energy, Water & Waste Reduction Campaign  University of Stirling   
5       Energy, Water & Waste Reduction Campaign  University of Stirling   
6       Energy, Water & Waste Reduction Campaign  University of Stirling   
7                          Interhall Competition  University of Stirling   

  First full year of CO2e savings  \
0                         2018/19   
1                         2018/19   
2                         2018/19   
3                         2018/19   
4                         2018/19   
5                         2018/19   
6                         2018/19   
7                         2018/19   

  Are these savings figures estimated or actual?  Capital cost (£)  \
0                                      Estimated                 0   
1                                      Estimated                 0   
2                                      Estimated                 0   
3                                      Estimated                 0   
4                                      Estimated                 0   
5                                      Estimated                 0   
6                                      Estimated                 0   
7                                      Estimated                 0   

   Operational cost (£/annum)  Project lifetime (years)  \
0                           0                         0   
1                           0                         0   
2                           0                         0   
3                           0                         0   
4                           0                         0   
5                           0                         0   
6                           0                         0   
7                           0                         0   

  Primary fuel/emission source saved  \
0                   Grid Electricity   
1                        Natural Gas   
2                            Gas Oil   
3                   Grid Electricity   
4                        Natural Gas   
5                            Gas Oil   
6                     Water - Supply   
7                   Grid Electricity   

   Estimated carbon savings per year (tCO2e/annum)  \
0                                             21.3   
1                                            112.5   
2                                              2.9   
3                                             17.0   
4                                             89.8   
5                                              2.3   
6                                              2.6   
7                                             20.9   

   Estimated costs savings (£/annum)  \
0                               4910   
1                              13784   
2                               5265   
```

```
3                                3918
4                               10998
5                                 420
6                                4595
7                                2558

                                    Behaviour Change  \
0  Encouraging users to notify Estates when areas...
1  Encouraging users to notify Estates when areas...
2  Encouraging users to notify Estates when areas...
3  Promoted reduction through various communicati...
4  Promoted reduction through various communicati...
5  Promoted reduction through various communicati...
6                        Changing culture in a fun way
7

                                            Comments
0  Interrogating BMS to identify conflict between...
1  Interrogating BMS to identify conflict between...
2  Interrogating BMS to identify conflict between...
3                                      Nominal costs
4                                      Nominal costs
5                                      Nominal costs
6                                      Nominal costs
7
```

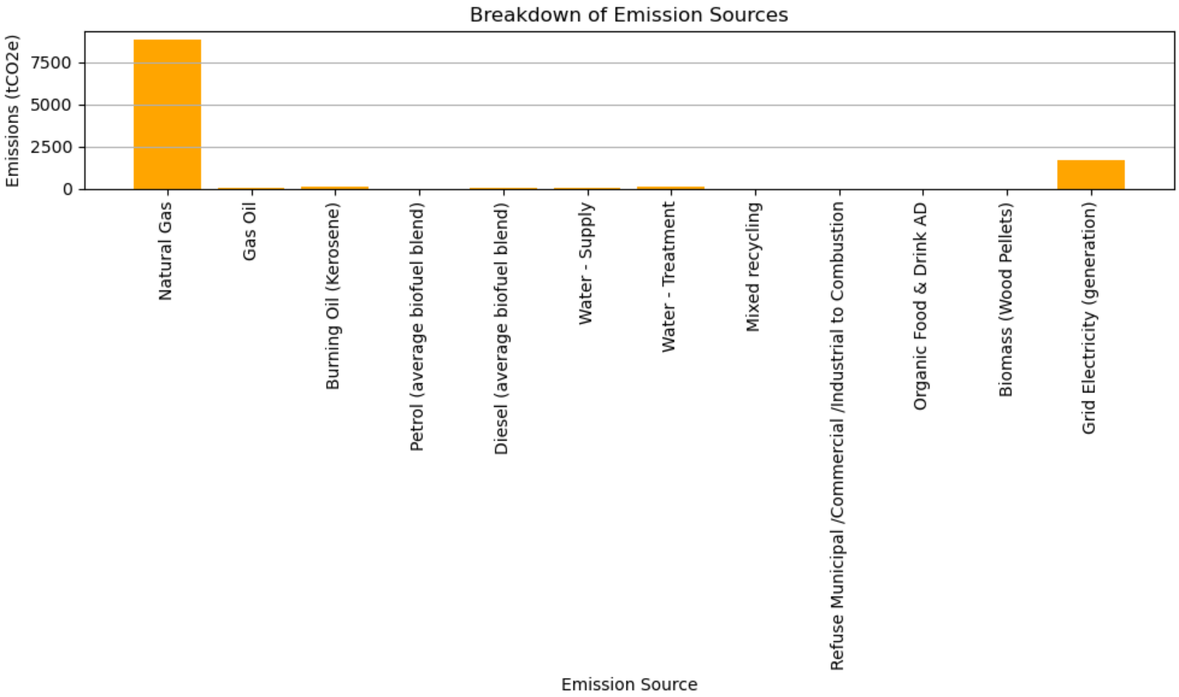
In [64]:

```python
plt.figure(figsize=(10, 6))
plt.plot(emissions_df["Report Year"], emissions_df["Scope1"], label="Scope 1", col
plt.plot(emissions_df["Report Year"], emissions_df["Scope2"], label="Scope 2", col
plt.plot(emissions_df["Report Year"], emissions_df["Scope3"], label="Scope 3", col
plt.xlabel("Report Year")
plt.ylabel("Emissions (tCO2e)")
plt.title("Emissions over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True)
plt.tight_layout()
plt.show()
```

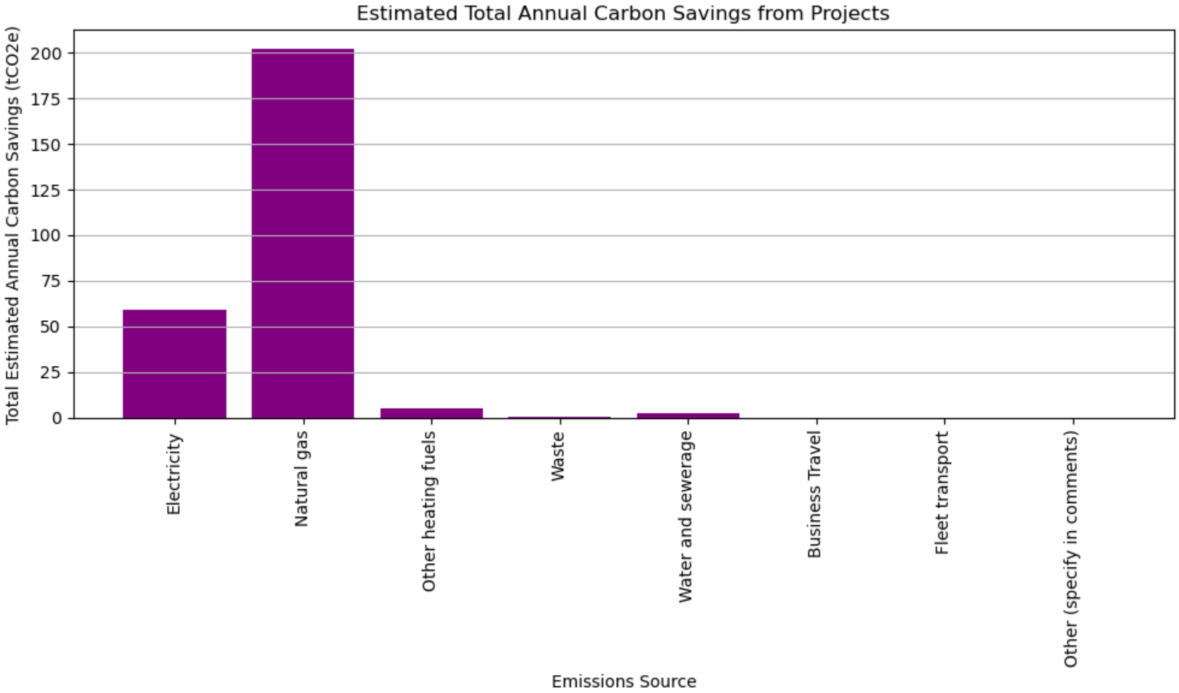


In [65]:

```python
plt.figure(figsize=(10, 6))
plt.bar(emission_sources_df["Emission Source"], emission_sources_df["Emissions (tC
plt.xlabel("Emission Source")
plt.ylabel("Emissions (tCO2e)")
plt.title("Breakdown of Emission Sources")
plt.xticks(rotation=90)
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```



```
In [66]: plt.figure(figsize=(10, 6))
         plt.bar(carbon_savings_df["Emissions Source"], carbon_savings_df["Total estimated 
         plt.xlabel("Emissions Source")
         plt.ylabel("Total Estimated Annual Carbon Savings (tCO2e)")
         plt.title("Estimated Total Annual Carbon Savings from Projects")
         plt.xticks(rotation=90)
         plt.grid(True, axis="y")
         plt.tight_layout()
         plt.show()
```



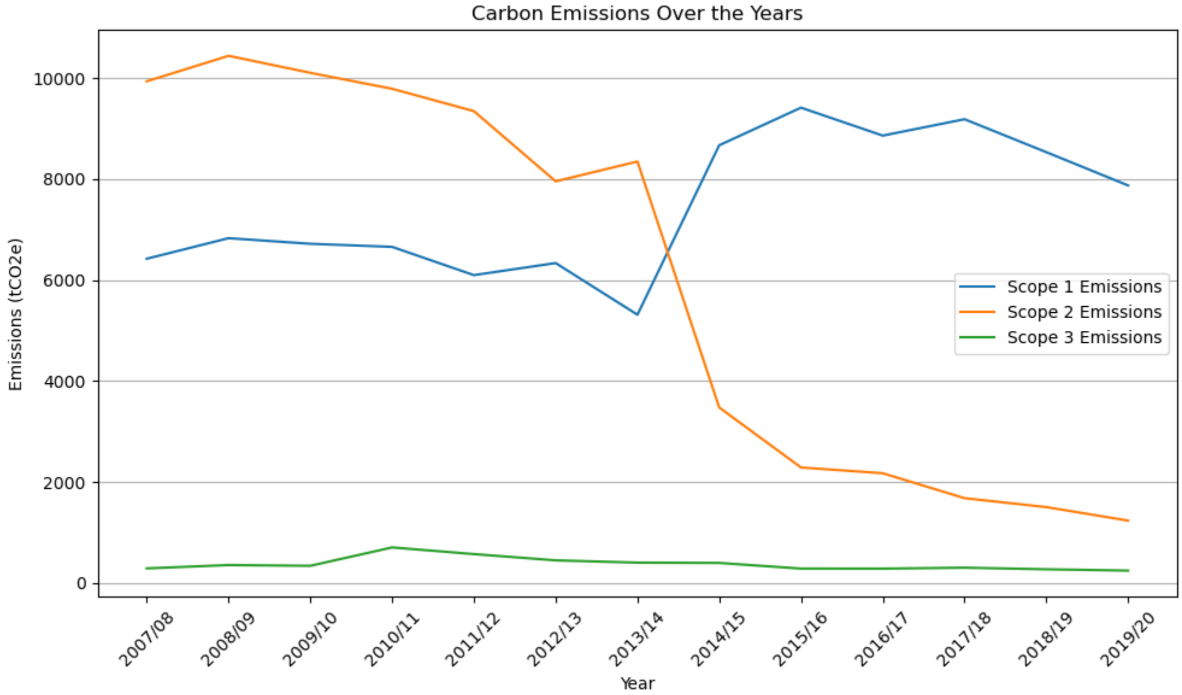
Report 2020

In [71]:

```
import matplotlib.pyplot as plt

# Data for carbon emissions from the dataset (taken from PART 3: EMISSIONS, TARGETS
years = [
    "2007/08",
    "2008/09",
    "2009/10",
    "2010/11",
    "2011/12",
    "2012/13",
    "2013/14",
    "2014/15",
    "2015/16",
    "2016/17",
    "2017/18",
    "2018/19",
    "2019/20",
]
scope1_emissions = [
    6424,
    6831,
    6720,
    6659,
    6099,
    6338,
    5315,
    8672,
    9417,
    8863,
    9188,
    8535,
    7876,
]
scope2_emissions = [
    9939,
    10444,
    10107,
    9791,
    9348,
    7957,
    8350,
    3477,
    2288,
    2174,
    1680,
    1503,
    1236,
]
scope3_emissions = [
    289,
    355,
    340,
    704,
    572,
    448,
    403,
    397,
    285,
    284,
    303,
    271,
    245,
]
```

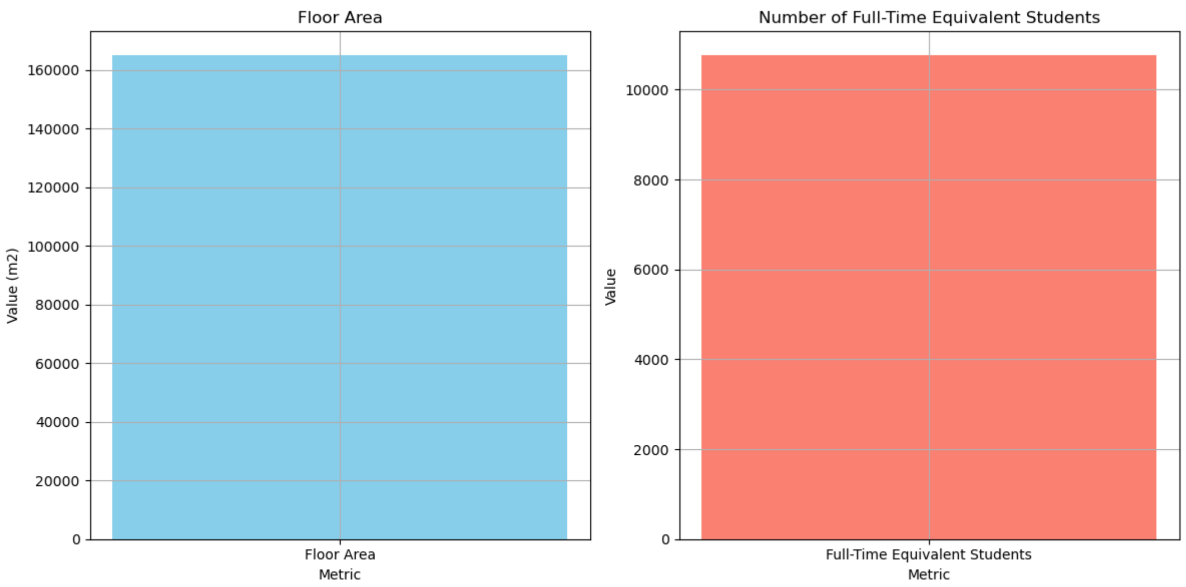
In [72]:

```
# Plot carbon emissions over the years
plt.figure(figsize=(10, 6))
plt.plot(years, scope1_emissions, label="Scope 1 Emissions")
plt.plot(years, scope2_emissions, label="Scope 2 Emissions")
plt.plot(years, scope3_emissions, label="Scope 3 Emissions")
plt.xlabel("Year")
plt.ylabel("Emissions (tCO2e)")
plt.title("Carbon Emissions Over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```

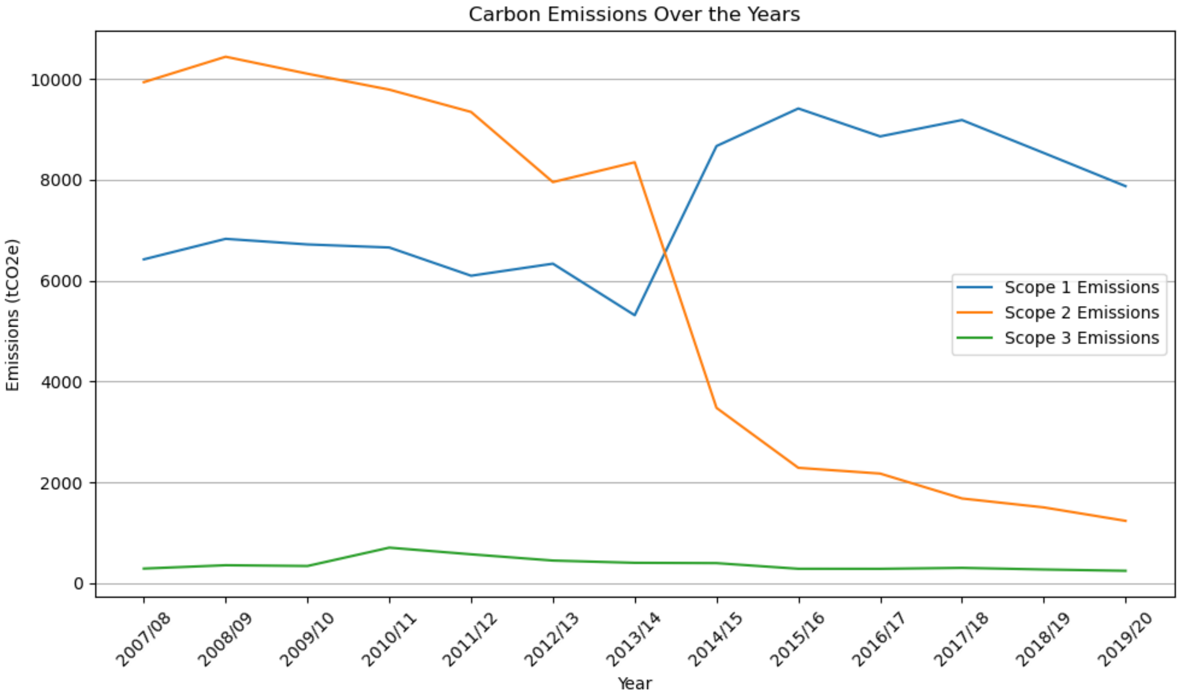


In [73]:

```
# Bar chart for floor area and number of full-time equivalent students
floor_area = 164811
fte_students = 10756

plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
plt.bar(["Floor Area"], [floor_area], color="skyblue")
plt.xlabel("Metric")
plt.ylabel("Value (m2)")
plt.title("Floor Area")
plt.grid(True)

plt.subplot(1, 2, 2)
plt.bar(["Full-Time Equivalent Students"], [fte_students], color="salmon")
plt.xlabel("Metric")
plt.ylabel("Value")
plt.title("Number of Full-Time Equivalent Students")
plt.grid(True)

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



In [74]:
```
# Line plot for carbon emissions over the years
plt.figure(figsize=(10, 6))
plt.plot(years, scope1_emissions, label="Scope 1 Emissions")
plt.plot(years, scope2_emissions, label="Scope 2 Emissions")
plt.plot(years, scope3_emissions, label="Scope 3 Emissions")
plt.xlabel("Year")
plt.ylabel("Emissions (tCO2e)")
plt.title("Carbon Emissions Over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```

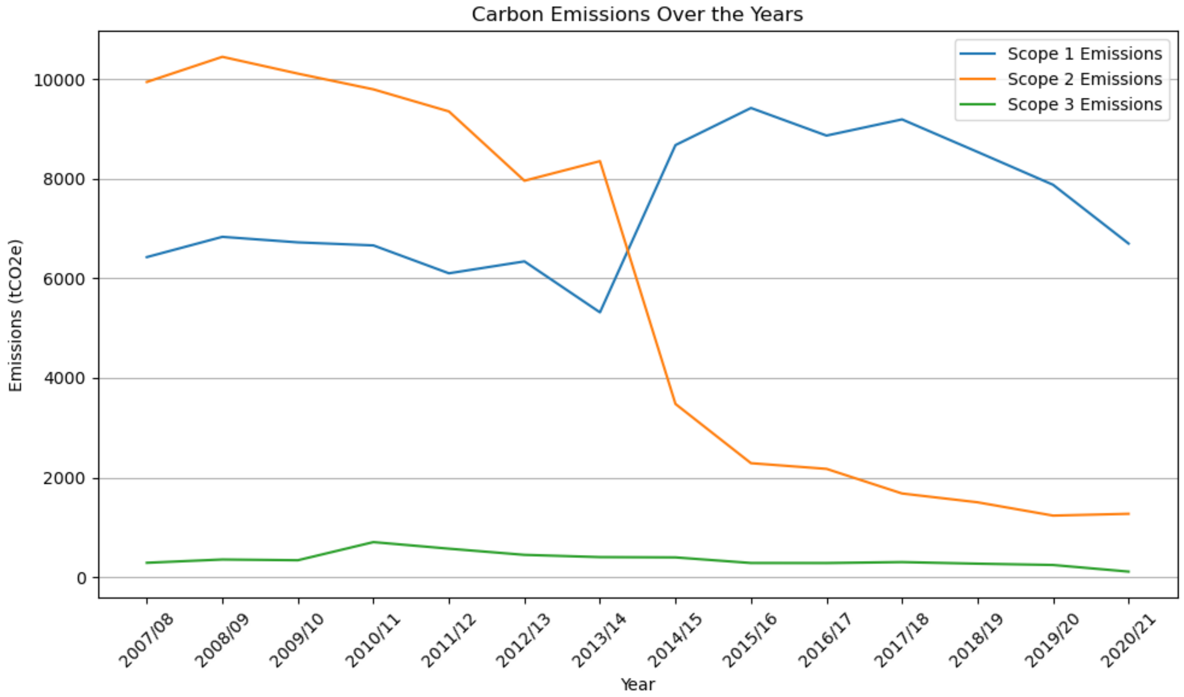


Report 2021

In [75]:
```
# Data for emissions from the dataset (taken from PART 3: EMISSIONS, TARGETS AND Pl
years = [
    "2007/08", "2008/09", "2009/10", "2010/11", "2011/12", "2012/13", "2013/14",
    "2014/15", "2015/16", "2016/17", "2017/18", "2018/19", "2019/20", "2020/21"
]

scope1_emissions = [
    6424, 6831, 6720, 6659, 6099, 6338, 5315, 8672, 9417, 8863, 9188, 8535, 7876,
]

scope2_emissions = [
    9939, 10444, 10107, 9791, 9348, 7957, 8350, 3477, 2288, 2174, 1680, 1503, 1236
]

scope3_emissions = [
    289, 355, 340, 704, 572, 448, 403, 397, 285, 284, 303, 271, 245, 112
]
```

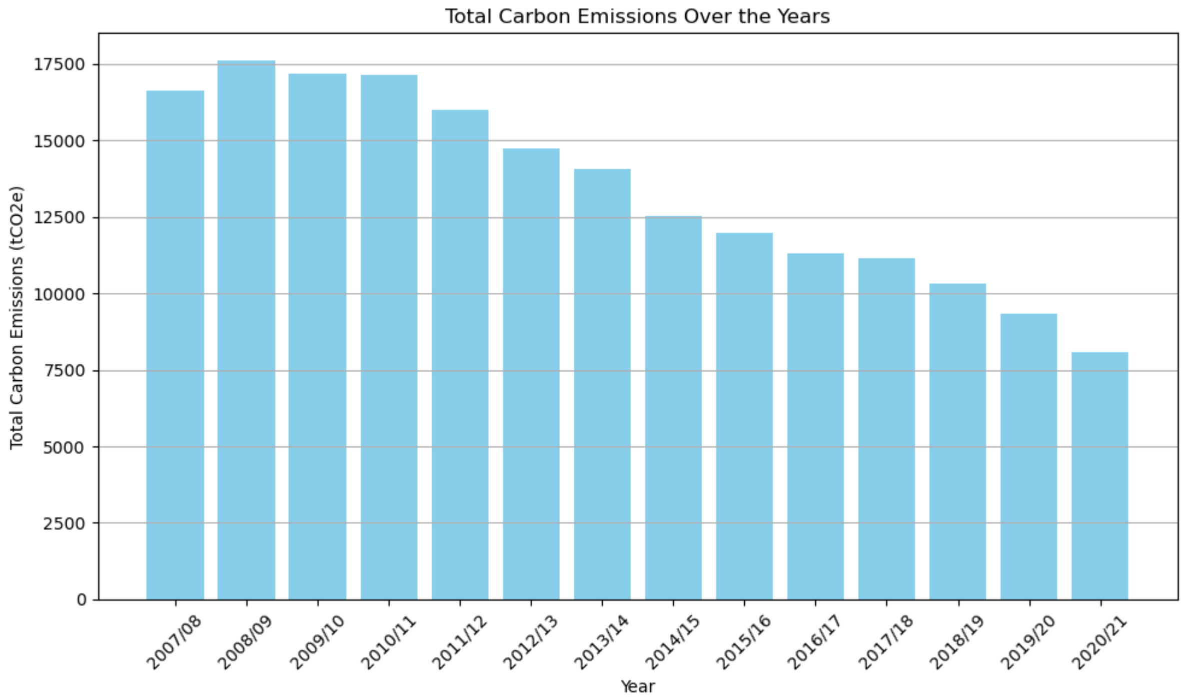
In [76]:

```
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
plt.plot(years, scope1_emissions, label="Scope 1 Emissions")
plt.plot(years, scope2_emissions, label="Scope 2 Emissions")
plt.plot(years, scope3_emissions, label="Scope 3 Emissions")
plt.xlabel("Year")
plt.ylabel("Emissions (tCO2e)")
plt.title("Carbon Emissions Over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```



In [77]:

```
total_emissions = [s1 + s2 + s3 for s1, s2, s3 in zip(scope1_emissions, scope2_emi

plt.figure(figsize=(10, 6))
plt.bar(years, total_emissions, color="skyblue")
plt.xlabel("Year")
plt.ylabel("Total Carbon Emissions (tCO2e)")
plt.title("Total Carbon Emissions Over the Years")
plt.xticks(rotation=45)
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```

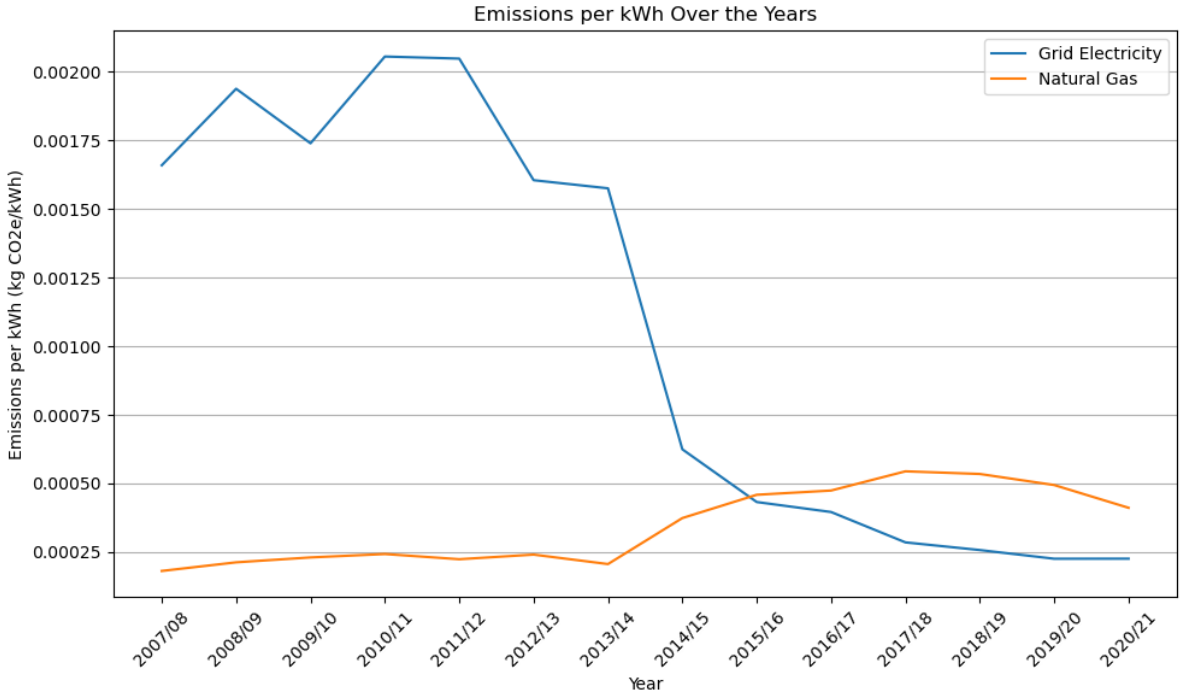


In [78]:

```
grid_electricity_consumption_kWh = [5990065, 5388985, 5810800, 4763300, 4564700, 4
natural_gas_consumption_kWh = [35418976, 32106281, 29140115, 27402022, 27216739, 2

# Calculate emissions per kWh for grid electricity and natural gas
grid_electricity_emissions_per_kWh = [s2 / c for s2, c in zip(scope2_emissions, gr
natural_gas_emissions_per_kWh = [s1 / c for s1, c in zip(scope1_emissions, natural_
```

In [79]:

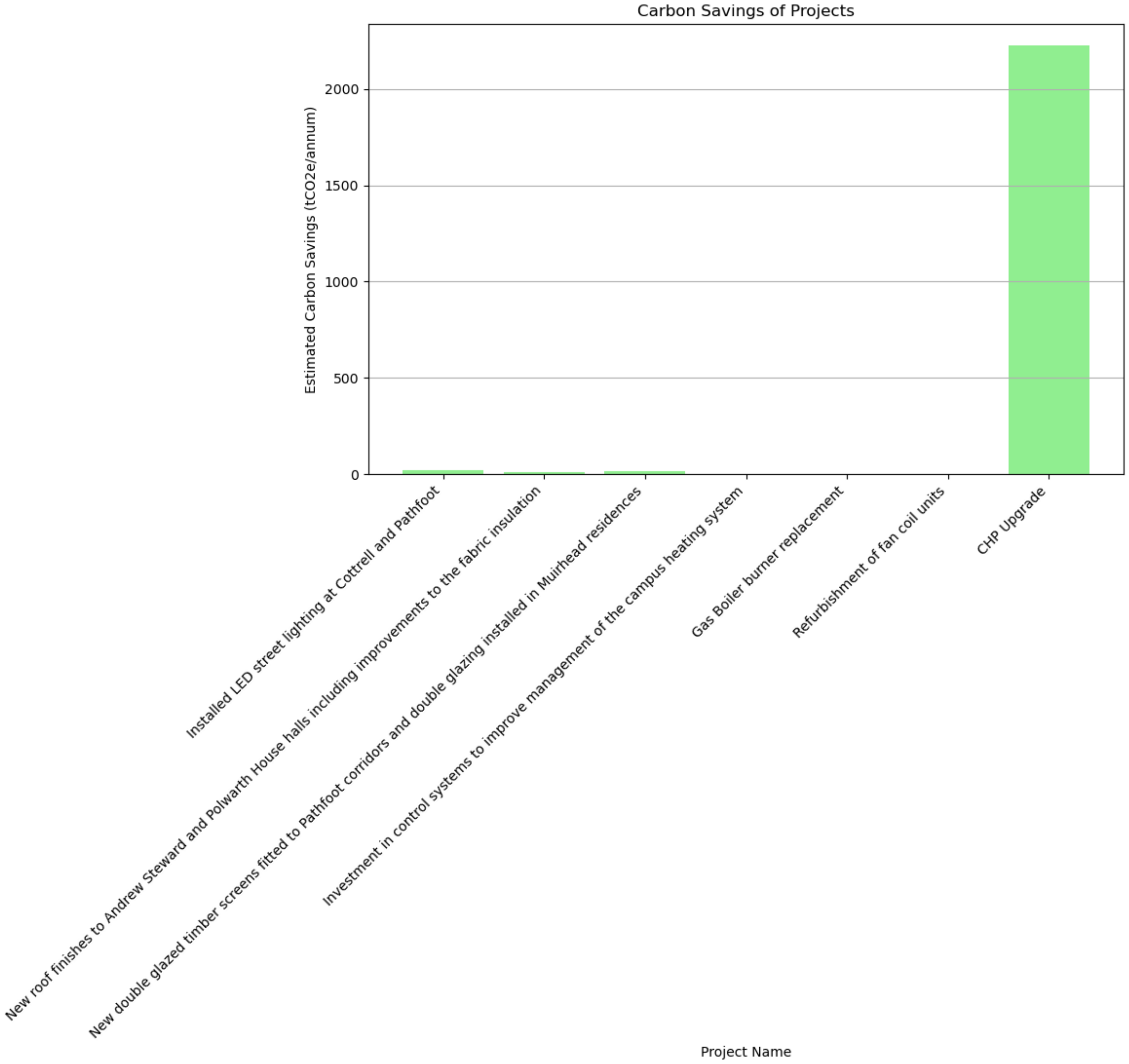
```
# Plot the emissions per kWh over the years
plt.figure(figsize=(10, 6))
plt.plot(years, grid_electricity_emissions_per_kWh, label="Grid Electricity")
plt.plot(years, natural_gas_emissions_per_kWh, label="Natural Gas")
plt.xlabel("Year")
plt.ylabel("Emissions per kWh (kg CO2e/kWh)")
plt.title("Emissions per kWh Over the Years")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```



In [80]:

```
# Data for projects from the dataset (taken from PART 3: EMISSIONS, TARGETS AND PRO
project_names = [
    "Installed LED street lighting at Cottrell and Pathfoot",
    "New roof finishes to Andrew Steward and Polwarth House halls including improve
    "New double glazed timber screens fitted to Pathfoot corridors and double glazi
    "Investment in control systems to improve management of the campus heating syst
    "Gas Boiler burner replacement",
    "Refurbishment of fan coil units",
    "CHP Upgrade",
]

carbon_savings_tCO2e = [21, 12, 15, 0.075, 0.2, 0.075, 2225]
cost_savings_annum = [13550, 2516, 2500, 50, 35, 50, 397000]

plt.figure(figsize=(10, 6))
plt.bar(project_names, carbon_savings_tCO2e, color="lightgreen")
plt.xlabel("Project Name")
plt.ylabel("Estimated Carbon Savings (tCO2e/annum)")
plt.title("Carbon Savings of Projects")
plt.xticks(rotation=45, ha='right')
plt.grid(True, axis="y")
plt.tight_layout()
plt.show()
```

```
/var/folders/dk/b7lfx_r11ds2wkc4dvr4zdbh0000gn/T/ipykernel_92152/605620396.py:22:
UserWarning: Tight layout not applied. The bottom and top margins cannot be made l
arge enough to accommodate all axes decorations.
  plt.tight_layout()
```



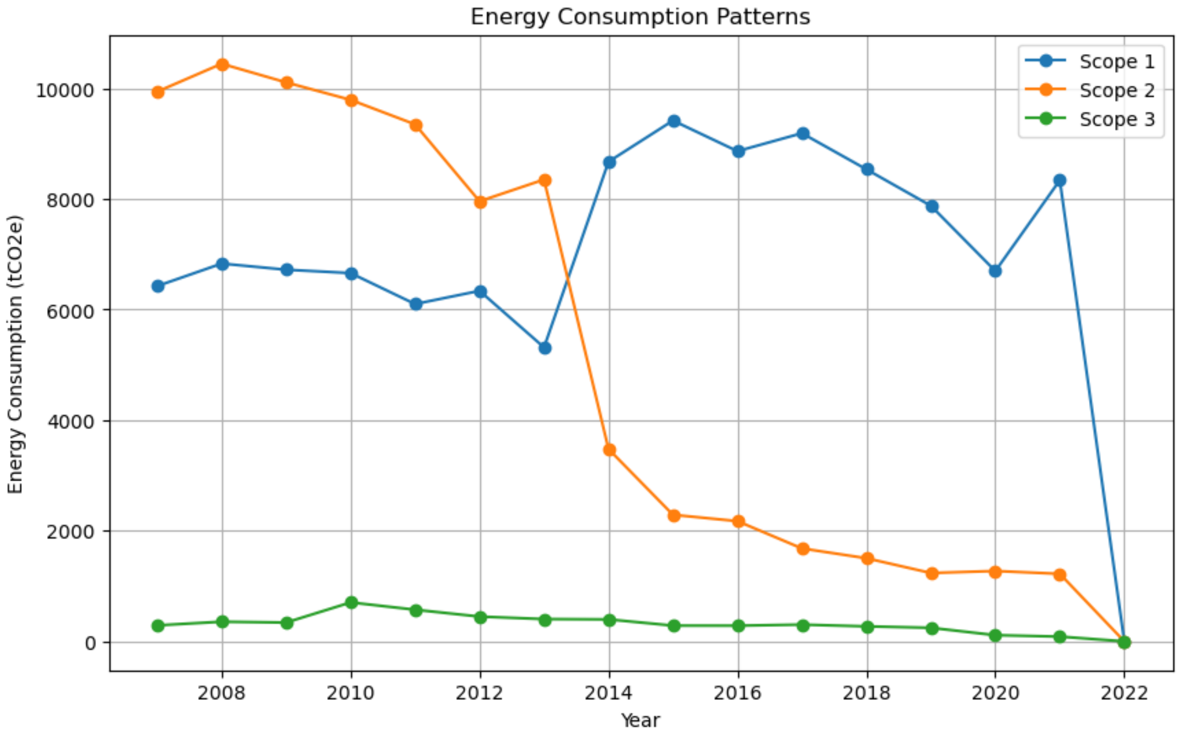
Year 2021

In [84]:

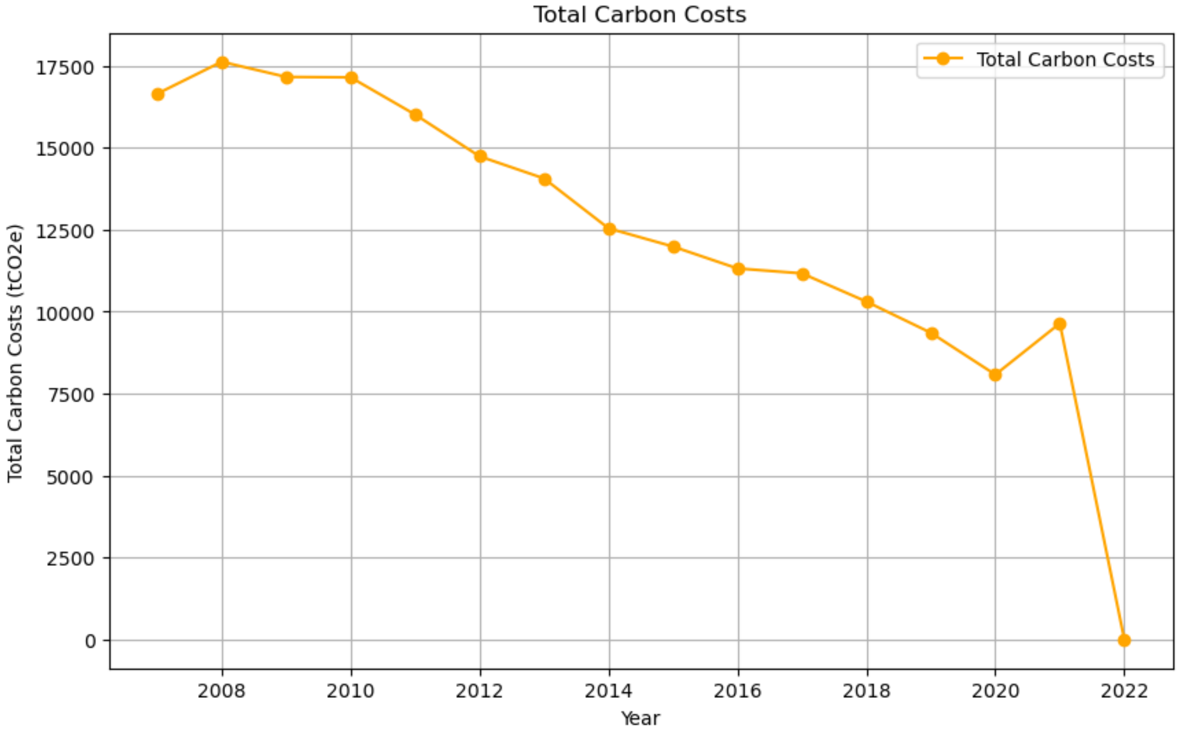
```
import matplotlib.pyplot as plt

# Data for energy consumption
years = [2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 20
scope_1 = [6424, 6831, 6720, 6659, 6099, 6338, 5315, 8672, 9417, 8863, 9188, 8535,
scope_2 = [9939, 10444, 10107, 9791, 9348, 7957, 8350, 3477, 2288, 2174, 1680, 150
scope_3 = [289, 355, 340, 704, 572, 448, 403, 397, 285, 284, 303, 271, 245, 112, 8
```

In [85]:

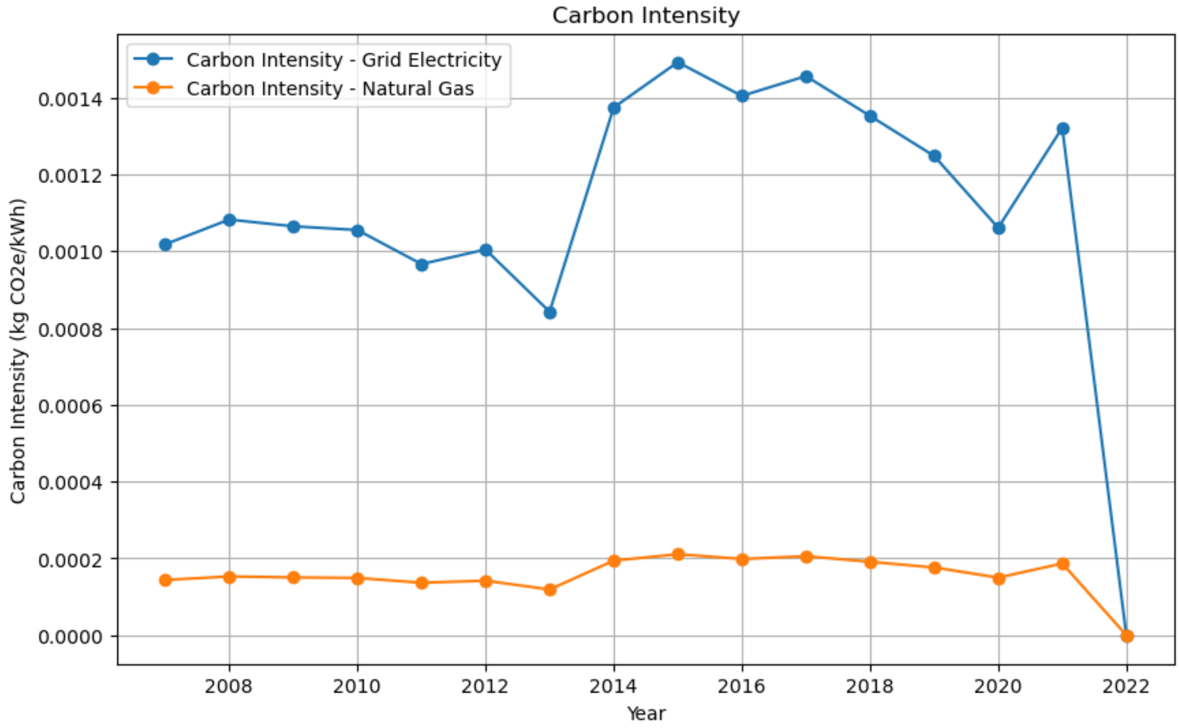
```
# Plotting energy consumption over the years
plt.figure(figsize=(10, 6))
plt.plot(years, scope_1, label='Scope 1', marker='o')
plt.plot(years, scope_2, label='Scope 2', marker='o')
plt.plot(years, scope_3, label='Scope 3', marker='o')
plt.xlabel('Year')
plt.ylabel('Energy Consumption (tCO2e)')
plt.title('Energy Consumption Patterns')
plt.legend()
plt.grid(True)
plt.show()
```



In [86]:
```
# Calculate total carbon costs
total_carbon_costs = [s1 + s2 + s3 for s1, s2, s3 in zip(scope_1, scope_2, scope_3

# Plotting total carbon costs over the years
plt.figure(figsize=(10, 6))
plt.plot(years, total_carbon_costs, label='Total Carbon Costs', marker='o', color=
plt.xlabel('Year')
plt.ylabel('Total Carbon Costs (tCO2e)')
plt.title('Total Carbon Costs')
plt.legend()
plt.grid(True)
plt.show()
```



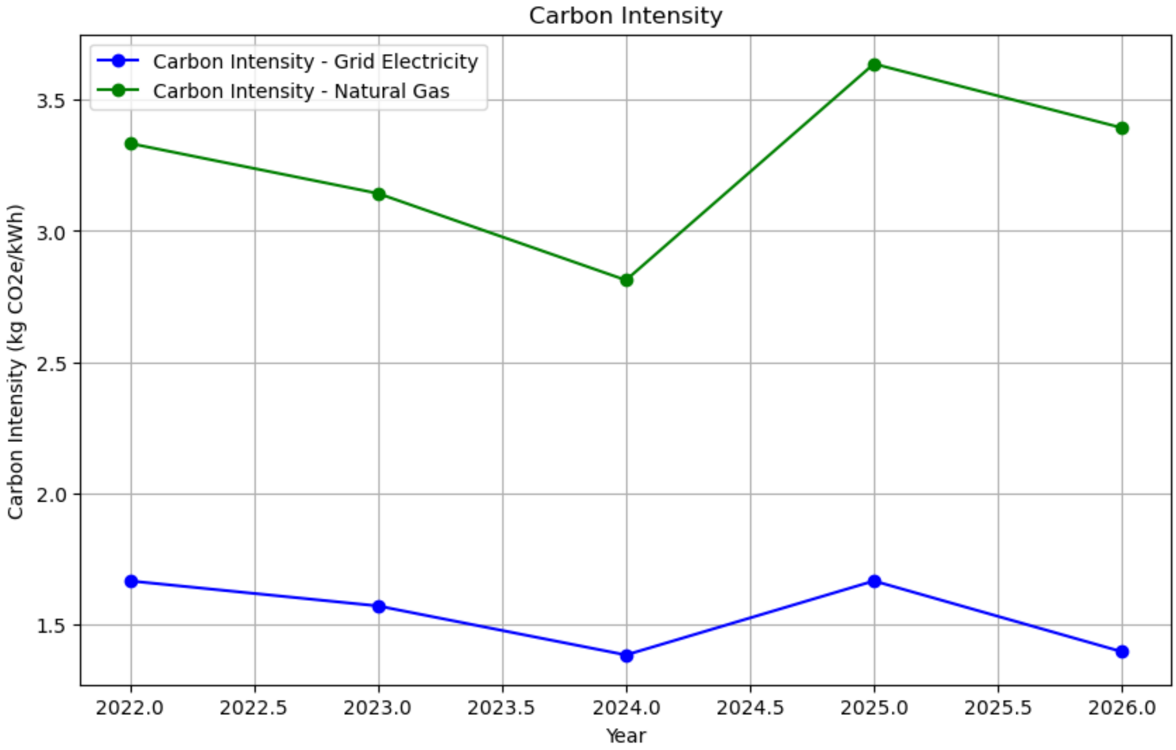
In [89]:
```
# Calculate carbon intensity for each year for grid electricity
carbon_intensity_grid = [s1 / consumption for s1, consumption in zip(scope_1, grid_
# Calculate carbon intensity for each year for natural gas (handling zero values)
carbon_intensity_gas = []
for s1, consumption in zip(scope_1, natural_gas):
    if consumption == 0:
        carbon_intensity_gas.append(0)  # Replace with 0 or any other appropriate 
    else:
        carbon_intensity_gas.append(s1 / consumption)

# Plotting carbon intensity over the years
plt.figure(figsize=(10, 6))
plt.plot(years, carbon_intensity_grid, label='Carbon Intensity - Grid Electricity'
plt.plot(years, carbon_intensity_gas, label='Carbon Intensity - Natural Gas', mark
plt.xlabel('Year')
plt.ylabel('Carbon Intensity (kg CO2e/kWh)')
plt.title('Carbon Intensity')
plt.legend()
plt.grid(True)
plt.show()
```



In [90]:

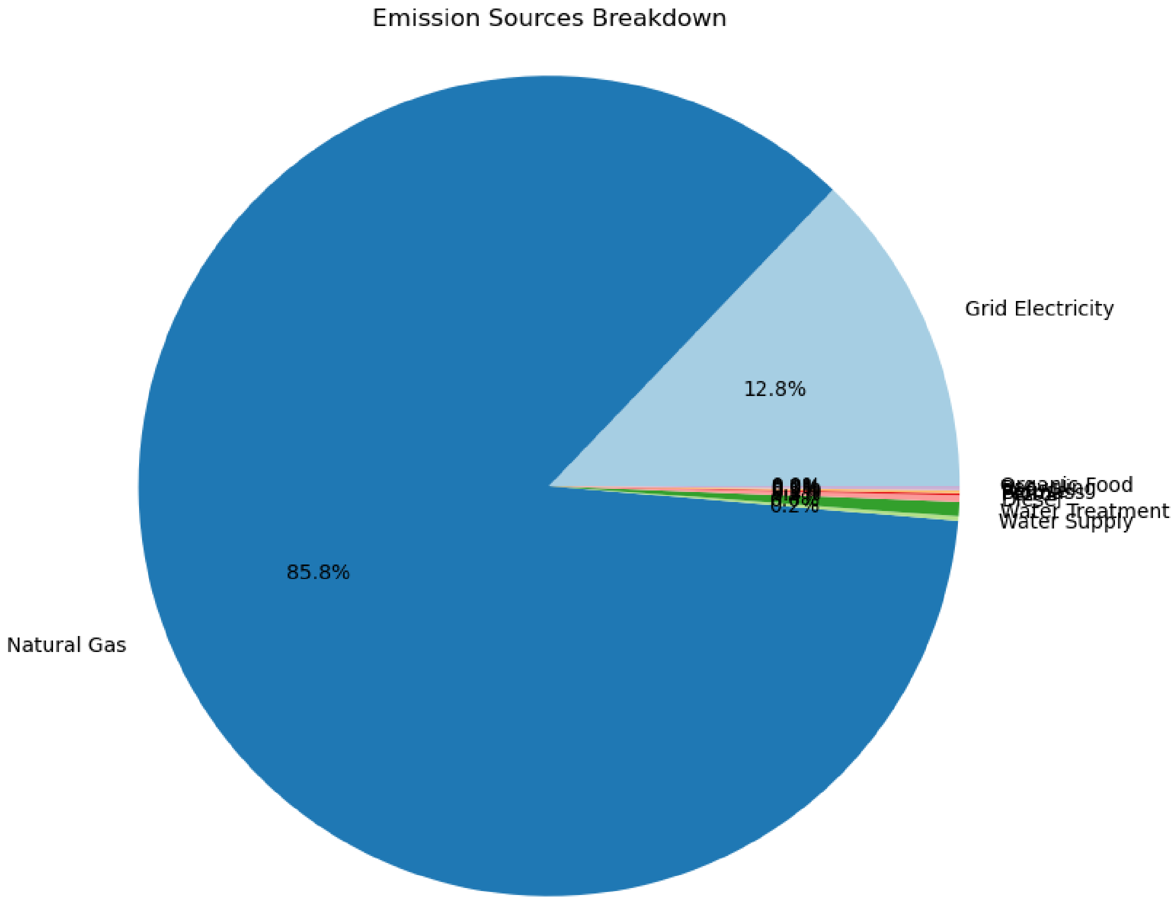
```
import matplotlib.pyplot as plt

# Sample data
years = [2022, 2023, 2024, 2025, 2026]
scope_1 = [10000, 11000, 9000, 12000, 9500]
grid_electricity_generation = [6000, 7000, 6500, 7200, 6800]
natural_gas = [3000, 3500, 3200, 3300, 2800]

# Calculate carbon intensity for each year for grid electricity
carbon_intensity_grid = [s1 / consumption for s1, consumption in zip(scope_1, grid_

# Calculate carbon intensity for each year for natural gas (handling zero values)
carbon_intensity_gas = []
for s1, consumption in zip(scope_1, natural_gas):
    if consumption == 0:
        carbon_intensity_gas.append(0)  # Replace with 0 or any other appropriate 
    else:
        carbon_intensity_gas.append(s1 / consumption)

# Plotting carbon intensity over the years
plt.figure(figsize=(10, 6))
```

```python
plt.plot(years, carbon_intensity_grid, label='Carbon Intensity - Grid Electricity'
plt.plot(years, carbon_intensity_gas, label='Carbon Intensity - Natural Gas', colo
plt.xlabel('Year')
plt.ylabel('Carbon Intensity (kg CO2e/kWh)')
plt.title('Carbon Intensity')
plt.legend()
plt.grid(True)
plt.show()
```



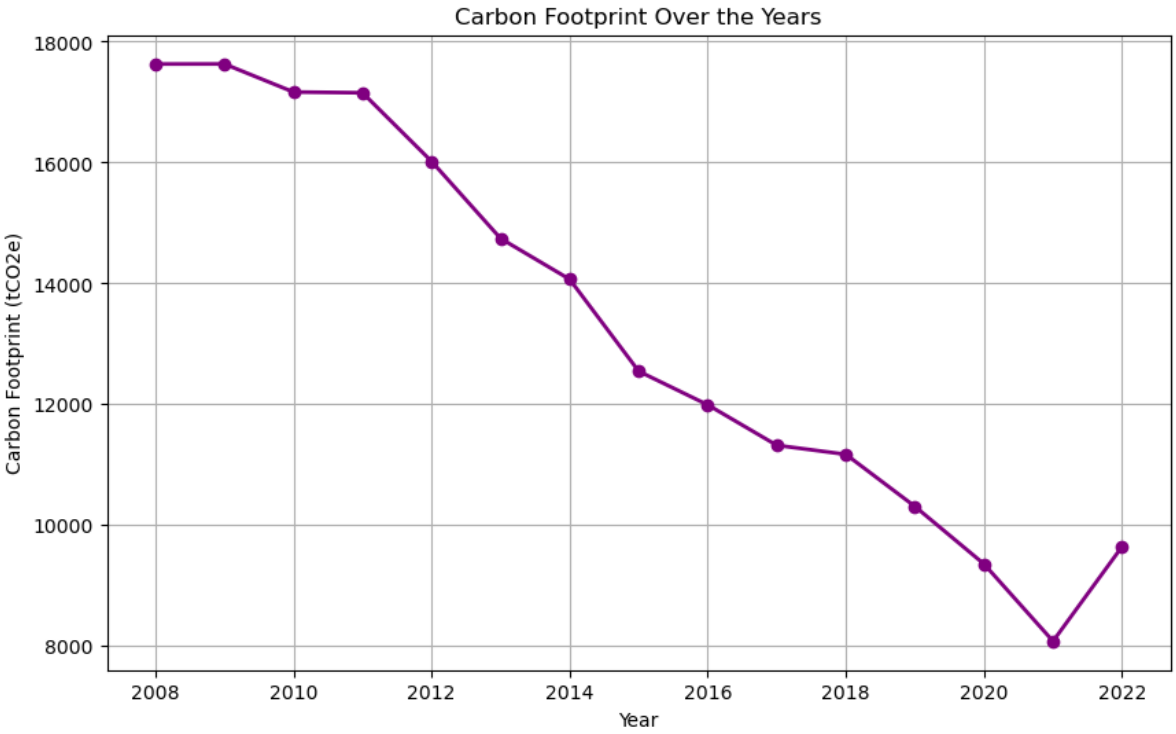
In [91]:

```python
import matplotlib.pyplot as plt

# Sample data
emission_sources = ['Grid Electricity', 'Natural Gas', 'Water Supply', 'Water Trea
emissions = [1220.7, 8155.8, 16.9, 53.0, 25.2, 7.9, 9.4, 1.0, 15.2, 0.2]

# Create a pie chart
plt.figure(figsize=(8, 8))
colors = plt.cm.Paired.colors  # Use a colormap for colors
plt.pie(emissions, labels=emission_sources, autopct='%1.1f%%', colors=colors)
plt.title('Emission Sources Breakdown')
plt.axis('equal')
plt.show()
```



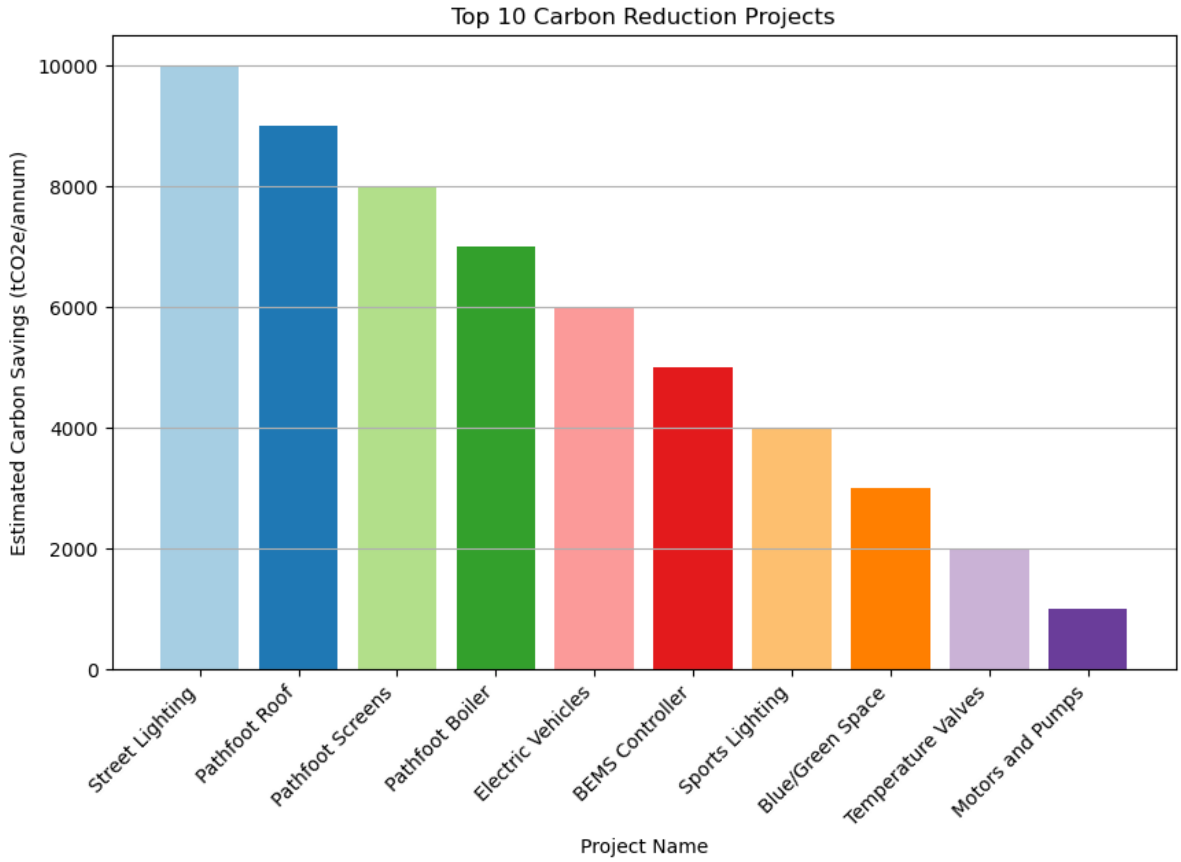
In [92]:

```
import matplotlib.pyplot as plt

# Sample data
years = [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2(
carbon_footprint = [17630, 17631, 17167, 17154, 16019, 14743, 14068, 12546, 11990,

# Create a line chart
plt.figure(figsize=(10, 6))
plt.plot(years, carbon_footprint, color='purple', marker='o', linewidth=2)
plt.xlabel('Year')
plt.ylabel('Carbon Footprint (tCO2e)')
plt.title('Carbon Footprint Over the Years')
plt.grid(True)
plt.show()
```



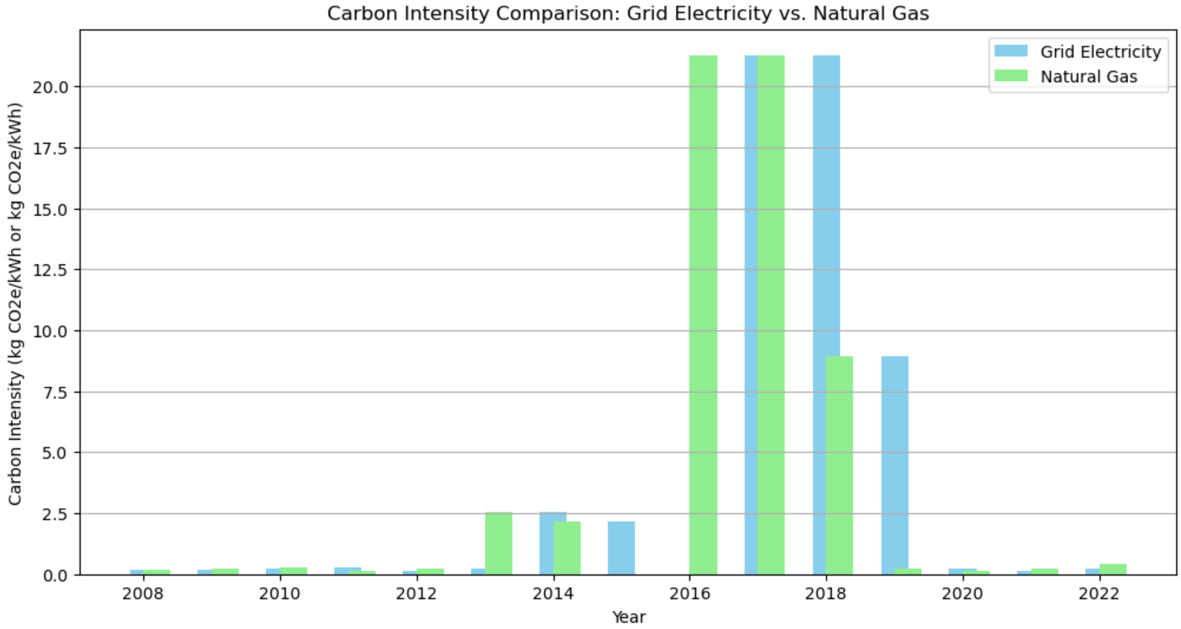
In [93]:

```
import matplotlib.pyplot as plt

# Sample data
projects = ['Street Lighting', 'Pathfoot Roof', 'Pathfoot Screens', 'Pathfoot Boil
carbon_savings = [10000, 9000, 8000, 7000, 6000, 5000, 4000, 3000, 2000, 1000]

# Create a bar chart
plt.figure(figsize=(10, 6))
colors = plt.cm.Paired.colors[:10]  # Use a colormap for colors
plt.bar(projects, carbon_savings, color=colors)
plt.xlabel('Project Name')
plt.ylabel('Estimated Carbon Savings (tCO2e/annum)')
plt.title('Top 10 Carbon Reduction Projects')
plt.xticks(rotation=45, ha='right')
plt.grid(axis='y')
plt.show()
```



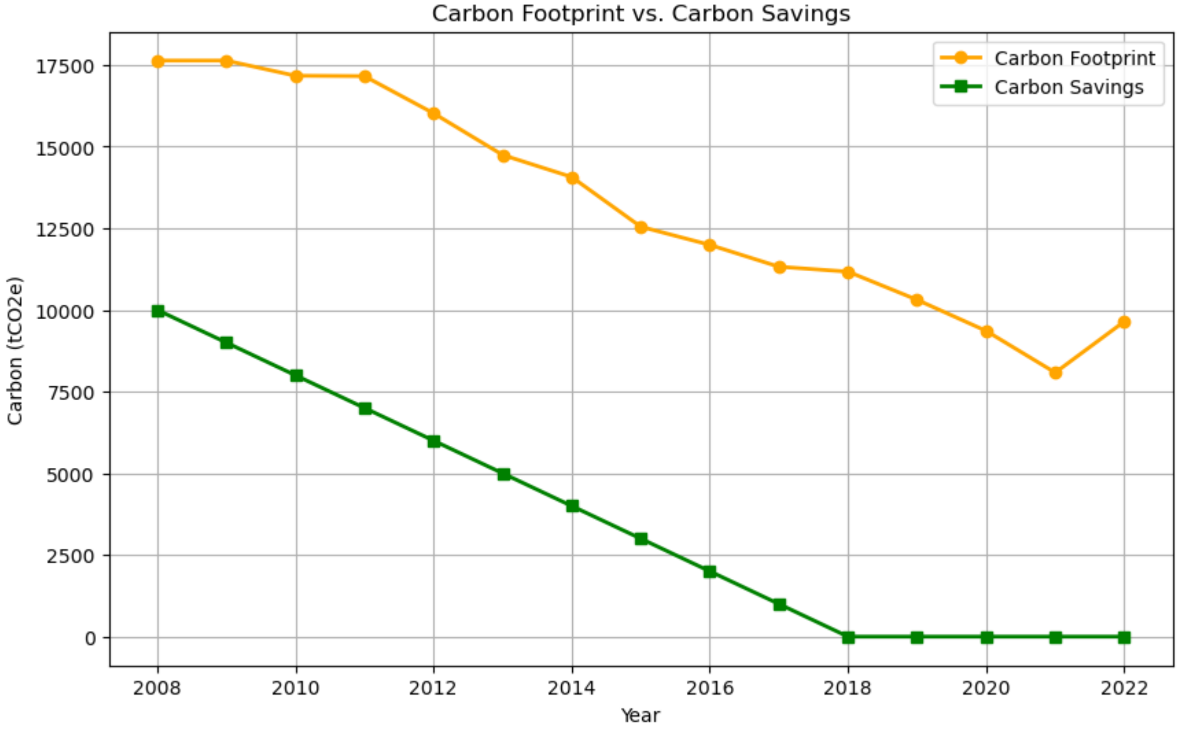
In [94]:

```
import matplotlib.pyplot as plt

# Sample data
years = [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 20
carbon_intensity_grid = [0.19338, 0.18254, 0.24677, 0.25679, 0.11000, 0.23000, 2.5
carbon_intensity_gas = [0.18254, 0.24677, 0.25679, 0.11000, 0.23000, 2.55784, 2.16

# Create a bar chart for carbon intensity comparison
plt.figure(figsize=(12, 6))
plt.bar(years, carbon_intensity_grid, color='skyblue', label='Grid Electricity', wi
plt.bar(years, carbon_intensity_gas, color='lightgreen', label='Natural Gas', widt
plt.xlabel('Year')
plt.ylabel('Carbon Intensity (kg CO2e/kWh or kg CO2e/kWh)')
plt.title('Carbon Intensity Comparison: Grid Electricity vs. Natural Gas')
plt.legend()
plt.grid(True, axis='y')
plt.show()
```



In [95]:

```
import matplotlib.pyplot as plt

# Sample data
years = [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 20
carbon_footprint = [17630, 17631, 17167, 17154, 16019, 14743, 14068, 12546, 11990,
carbon_savings = [10000, 9000, 8000, 7000, 6000, 5000, 4000, 3000, 2000, 1000, 0, 0

# Create a line chart for carbon footprint vs. carbon savings
plt.figure(figsize=(10, 6))
plt.plot(years, carbon_footprint, color='orange', marker='o', label='Carbon Footpri
plt.plot(years, carbon_savings, color='green', marker='s', label='Carbon Savings',
plt.xlabel('Year')
plt.ylabel('Carbon (tCO2e)')
plt.title('Carbon Footprint vs. Carbon Savings')
plt.legend()
plt.grid(True)
plt.show()
```



In [ ]: