In [1]:

```
#A1
class Polygon:
    def __init__(self,sides):
        self.sides=sides
    def display_sides(self):
        print(f"This polygon has {self.sides} sides")
    def find_area(self):
        raise NotImplementedError("Subclasses must implement this method.")

class Triangle(Polygon):
    def __init__(self):
        super().__init__(3)
        self.side1=float(input("Enter the length of side1: "))
        self.side2=float(input("Enter the length of side2: "))
        self.side3=float(input("Enter the length of side3: "))

    def find_area(self):
        s=(self.side1+self.side2+self.side3)/2
        area=(s*(s-self.side1)*(s-self.side2)*(s-self.side3))**0.5
        return area

triangle=Triangle()
triangle.display_sides()
area=triangle.find_area()
print(f" The area of the triangle is:{area}")
```

```
This polygon has 3 sides
 The area of the triangle is:89.97777503361594
```

In [5]:

```
#A3
import numpy as np
def get_student_info():
    name=input("Enter a student name: ")
    height=float(input("Enter a student height: "))
    class_num=int(input("Enter student class:"))
    return name,height,class_num

num_students=int(input("Enter the number of students: "))
dtypes=[('name','U20'),('height',float),('class',int)]
students_array=np.empty(num_students,dtype=dtypes)

for i in range(num_students):
    print(f"\n Enter information for student{i+1}:")
    students_array[i]=get_student_info()
print("\n Original Array:")
print(students_array)

sorted_students_array = np.sort(students_array,order='height')
print("\n Sorted Array based on height:")
print(sorted_students_array)
```

```
 Enter information for student1:

 Enter information for student2:

 Enter information for student3:

 Original Array:
[('Bob', 6.2, 11) ('Alice', 5.5, 10) ('Carol', 5.2, 10)]

 Sorted Array based on height:
[('Carol', 5.2, 10) ('Alice', 5.5, 10) ('Bob', 6.2, 11)]
```

In [13]:

```
#A4
```

```
import pandas as pd
df=pd.read_csv('churn.csv')
print(df)
```

```
     Unnamed: 0  customerID tenure        Contract PaperlessBilling  \
0             1  8260-NGFNY    One  Month-to-month               No   
1             2  2359-QWQUL     39        One year              Yes   
2             3  6598/RFFVI      2        One year               No   
3             4  IXSTS-8780      6  Month-to-month              Yes   
4             5  2674/MIAHT   Four  Month-to-month              Yes   
..          ...         ...    ...             ...              ...   
252         253  9318-NKNFC    One  Month-to-month              Yes   
253         254  9067-SQTNS     44        One year               No   
254         255  9067-SQTNS     44        One year               No   
255         256  9067-SQTNS     44        One year               No   
256         257  9067-SQTNS     44        One year               No   

                 PaymentMethod  MonthlyCharges  TotalCharges  gender  \
0                 Mailed check           25.20         25.20  Female   
1      Credit card (automatic)          104.70       4134.85  Female   
2      Credit card (automatic)           19.30         28.30    Male   
3             Electronic check           90.10        521.30  Female   
4                 Mailed check           80.30        324.20  Female   
..                         ...             ...           ...     ...   
252               Mailed check           18.85         18.85    Male   
253  Bank transfer (automatic)           20.60        926.00    Male   
254  Bank transfer (automatic)           20.60        926.00    Male   
255  Bank transfer (automatic)           20.60        926.00    Male   
256  Bank transfer (automatic)           20.60        926.00    Male   

     SeniorCitizen  ... PhoneService     MultipleLines InternetService  \
0              0.0  ...           No  No phone service             DSL   
1              0.0  ...          Yes                No     Fiber optic   
2              0.0  ...          Yes                No              No   
3              0.0  ...          Yes               Yes     Fiber optic   
4              0.0  ...          Yes               Yes     Fiber optic   
..             ...  ...          ...               ...             ...   
252            0.0  ...          Yes                No              No   
253            0.0  ...          Yes                No              No   
254            0.0  ...          Yes                No              No   
255            0.0  ...          Yes                No              No   
256            0.0  ...          Yes                No              No   

          OnlineSecurity         OnlineBackup     DeviceProtection  \
0                     No                   No                   No   
1                    Yes                   No                  Yes   
2    No internet service  No internet service  No internet service   
3                     No                  Yes                   No   
4                     No                  Yes                   No   
..                   ...                  ...                  ...   
252  No internet service  No internet service  No internet service   
253                  Yes                  Yes  No internet service   
254                  Yes                  Yes  No internet service   
255                  Yes                  Yes  No internet service   
256                  Yes                  Yes  No internet service   

             TechSupport          StreamingTV      StreamingMovies Churn  
0                     No                   No                   No   Yes  
1                    Yes                  Yes                  Yes   Yes  
2    No internet service  No internet service  No internet service   Yes  
3                     No                  Yes                   No   Yes  
4                     No                   No                   No    No  
..                   ...                  ...                  ...   ...  
252  No internet service  No internet service  No internet service   Yes  
253  No internet service                  Yes  No internet service    No  
254  No internet service                  Yes  No internet service    No  
255  No internet service                  Yes  No internet service    No  
256  No internet service                  Yes  No internet service    No  

[257 rows x 22 columns]
```

In [3]:

```
#(i)
duplicate_count=df.duplicated().sum()
print(f"Number of Duplicate Records:{duplicate_count}")
```

```
Number of Duplicate Records:0
```

In [4]:

```
#(ii)
duplicate_customer_id=df['customerID'].duplicated().sum()
print(f"Number of Duplicate Records based on 'customerID' column:{duplicate_customer_id}"
)
```

```
Number of Duplicate Records based on 'customerID' column:7
```

In [5]:

```
#(iii)
missing_values_per_column=df.isnull().sum()
print(f"Number of missing values in each column:{missing_values_per_column}")
```

```
Number of missing values in each column:Unnamed: 0           0
customerID           0
tenure               0
Contract             0
PaperlessBilling     0
PaymentMethod        0
MonthlyCharges      10
TotalCharges        15
gender               0
SeniorCitizen        5
Partner              0
Dependents           0
PhoneService         0
MultipleLines        0
InternetService      0
OnlineSecurity       0
OnlineBackup         0
DeviceProtection     0
TechSupport          0
StreamingTV          0
StreamingMovies      0
Churn                0
dtype: int64
```

In [6]:

```
#(iv)
missing_values_TotalCharges=df['TotalCharges'].isnull().sum()
print(f"Number of missing values in 'Total Charges':{missing_values_TotalCharges}")
```

```
Number of missing values in 'Total Charges':15
```

In [7]:

```
#(v)
average_monthly_charge=df['MonthlyCharges'].mean()
print(f"Average MonthlyCharges:{average_monthly_charge}")
```

```
Average MonthlyCharges:62.47348178137652
```

In [8]:

```
#(vi)
filtered_records=df[df['Dependents']=="1@#"]
print(f"Records with 'Dependents' equal to '1@#':{filtered_records}")

#Alter
filtered_records1=df.query('Dependents =="1@#" ')
print(f"Records with 'Dependents' equal to '1@#':{filtered_records1}")
```

```
Records with 'Dependents' equal to '1@#':     Unnamed: 0  customerID tenure        Contra
ct PaperlessBilling  \
89           90  1754-GKYPY     22  Month-to-month              Yes
125         126  9108-EQPNQ     10        Two year               No
174         175  2640-PMGFL     27  Month-to-month              Yes
220         221  8854-CCVSQ     18  Month-to-month              Yes
234         235  6876-ADESB    One  Month-to-month               No
238         239  1972-XMUWV     65        Two year              Yes

                 PaymentMethod  MonthlyCharges  TotalCharges  gender  \
89   Bank transfer (automatic)           89.75       1938.90    Male
125    Credit card (automatic)           26.10        225.55  Female
174           Electronic check           79.50       2180.55    Male
220           Electronic check           80.65       1451.90    Male
234           Electronic check           48.95         48.95    Male
238    Credit card (automatic)           59.80       3808.20  Female

     SeniorCitizen  ... PhoneService MultipleLines InternetService  \
89             1.0  ...          Yes            No     Fiber optic
125            0.0  ...          Yes           Yes              No
174            0.0  ...          Yes           Yes     Fiber optic
220            0.0  ...          Yes           Yes     Fiber optic
234            0.0  ...          Yes            No             DSL
238            0.0  ...          Yes            No             DSL

          OnlineSecurity         OnlineBackup     DeviceProtection  \
89                    No                   No                   No
125  No internet service  No internet service  No internet service
174                   No                   No                   No
220                   No                  Yes                   No
234                   No                   No                  Yes
238                   No                   No                   No

             TechSupport          StreamingTV      StreamingMovies Churn
89                    No                  Yes                  Yes    No
125  No internet service  No internet service  No internet service    No
174                  Yes                   No                   No   Yes
220                   No                   No                   No   Yes
234                   No                   No                   No   Yes
238                  Yes                  Yes                   No    No

[6 rows x 22 columns]
Records with 'Dependents' equal to '1@#':     Unnamed: 0  customerID tenure        Contra
ct PaperlessBilling  \
89           90  1754-GKYPY     22  Month-to-month              Yes
125         126  9108-EQPNQ     10        Two year               No
174         175  2640-PMGFL     27  Month-to-month              Yes
220         221  8854-CCVSQ     18  Month-to-month              Yes
234         235  6876-ADESB    One  Month-to-month               No
238         239  1972-XMUWV     65        Two year              Yes

                 PaymentMethod  MonthlyCharges  TotalCharges  gender  \
89   Bank transfer (automatic)           89.75       1938.90    Male
125    Credit card (automatic)           26.10        225.55  Female
174           Electronic check           79.50       2180.55    Male
220           Electronic check           80.65       1451.90    Male
234           Electronic check           48.95         48.95    Male
238    Credit card (automatic)           59.80       3808.20  Female

     SeniorCitizen  ... PhoneService MultipleLines InternetService  \
89             1.0  ...          Yes            No     Fiber optic
125            0.0  ...          Yes           Yes              No
174            0.0  ...          Yes           Yes     Fiber optic
220            0.0  ...          Yes           Yes     Fiber optic
234            0.0  ...          Yes            No             DSL
238            0.0  ...          Yes            No             DSL

          OnlineSecurity         OnlineBackup     DeviceProtection  \
89                    No                   No                   No
125  No internet service  No internet service  No internet service
174                   No                   No                   No
220                   No                  Yes                   No
```

```
234                     No                   No                  Yes
238                     No                   No                   No

               TechSupport          StreamingTV      StreamingMovies Churn
89                      No                  Yes                  Yes    No
125  No internet service  No internet service  No internet service    No
174                    Yes                   No                   No   Yes
220                     No                   No                   No   Yes
234                     No                   No                   No   Yes
238                    Yes                  Yes                   No    No

[6 rows x 22 columns]
```

In [9]:

```
#(vii)
new_df=df.fillna(df.median(numeric_only=True))
new_df=new_df.fillna(df.mode().iloc[0])
print(new_df.isnull().sum())
```

```
Unnamed: 0          0
customerID          0
tenure              0
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges        0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
Churn               0
dtype: int64
```

In [10]:

```
print("The original column is:")
print(new_df['MonthlyCharges'])
x=new_df['MonthlyCharges']
print("The replaced column is :")
x.replace(25.20,25)
```

```
The original column is:
0       25.20
1      104.70
2       19.30
3       90.10
4       80.30
        ...
252     18.85
253     20.60
254     20.60
255     20.60
256     20.60
Name: MonthlyCharges, Length: 257, dtype: float64
The replaced column is :
```

Out[10]:

```
0       25.00
1      104.70
2       19.30
3       90.10
```

```
3        90.10
4        80.30
         ...
252      18.85
253      20.60
254      20.60
255      20.60
256      20.60
Name: MonthlyCharges, Length: 257, dtype: float64
```

In [11]:

```
#A2 (a)
import re
text="xyz@gmail.com and 999@99ad.com and abc_987@vvce.ac.in are the mail id's,(897)-012-3
456 ext.23 and 897-0123456x23 are numbers."
emailRegex=re.compile('[a-zA_Z0-9._]+@[a-zA-Z0-9,-]+[.][a-zA-Z]{2,4}')
L=emailRegex.findall(text)
for email in L:
    print(email)
```

```
xyz@gmail.com
999@99ad.com
abc_987@vvce.ac
```

In [12]:

```
#A2 (b)
import re
p=input("Input your password:")
if len(p)>5 and len(p)<17 and re.search('[a-z]',p) and re.search('[A-Z]',p) and re.searc
h('[0-9]',p) and re.search('[$#@_]',p):
    print("Valid password")
else:
    print("Invalid Password")
```

```
Valid password
```

In [5]:

```
#C1
class Employee:
    count = 0
    def __init__(self):
        self.name = None
        self.place = None
        self.department = None
        Employee.count += 1
        self.eid = 'Emp' + str(Employee.count)

    def update(self):
        self.name = input("Enter name: ")
        self.place = input("Enter place: ")
        self.department = input("Enter dept: ")

    def display(self):
        print("Employee ID:", self.eid)
        print("Employee Name:", self.name)
        print("Employee Place:", self.place)
        print("Employee Dept:", self.department)
        print("-" * 30)

n = int(input("Enter total number of Employees: "))
employees = []
for i in range(n):
    emp = Employee()
    emp.update()
    employees.append(emp)
print("\nEmployee Details: \n")
for emp in employees:
    emp.display()
```

```
Employee Details
Employee Details:

Employee ID: Emp1
Employee Name: varun
Employee Place: mys
Employee Dept: IT
------------------------------
Employee ID: Emp2
Employee Name: arun
Employee Place: beng
Employee Dept: IS
------------------------------
```

In [12]:

```
#C2
import pandas as pd
import seaborn as sns
df = pd.read_csv('Automobile_data.csv', na_values=['?'])
print(df.head())
print(df.describe())
```

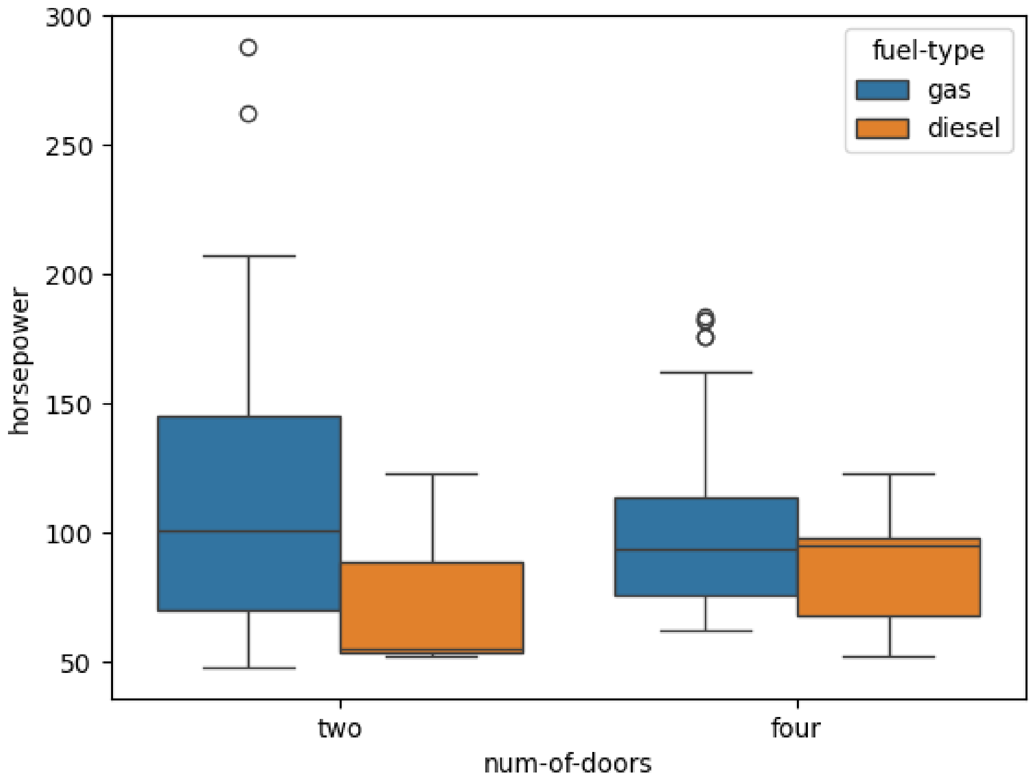
```
   symboling  normalized-losses         make fuel-type aspiration  \
0          3                NaN  alfa-romero       gas        std   
1          3                NaN  alfa-romero       gas        std   
2          1                NaN  alfa-romero       gas        std   
3          2              164.0         audi       gas        std   
4          2              164.0         audi       gas        std   

  num-of-doors   body-style drive-wheels engine-location  wheel-base  ...  \
0          two  convertible          rwd           front        88.6  ...   
1          two  convertible          rwd           front        88.6  ...   
2          two    hatchback          rwd           front        94.5  ...   
3         four        sedan          fwd           front        99.8  ...   
4         four        sedan          4wd           front        99.4  ...   

   engine-size  fuel-system  bore  stroke compression-ratio horsepower  \
0          130         mpfi  3.47    2.68               9.0      111.0   
1          130         mpfi  3.47    2.68               9.0      111.0   
2          152         mpfi  2.68    3.47               9.0      154.0   
3          109         mpfi  3.19    3.40              10.0      102.0   
4          136         mpfi  3.19    3.40               8.0      115.0   

   peak-rpm city-mpg  highway-mpg    price  
0    5000.0       21           27  13495.0  
1    5000.0       21           27  16500.0  
2    5000.0       19           26  16500.0  
3    5500.0       24           30  13950.0  
4    5500.0       18           22  17450.0  

[5 rows x 26 columns]
        symboling  normalized-losses  wheel-base      length       width  \
count  205.000000         164.000000  205.000000  205.000000  205.000000   
mean     0.834146         122.000000   98.756585  174.049268   65.907805   
std      1.245307          35.442168    6.021776   12.337289    2.145204   
min     -2.000000          65.000000   86.600000  141.100000   60.300000   
25%      0.000000          94.000000   94.500000  166.300000   64.100000   
50%      1.000000         115.000000   97.000000  173.200000   65.500000   
75%      2.000000         150.000000  102.400000  183.100000   66.900000   
max      3.000000         256.000000  120.900000  208.100000   72.300000   

           height  curb-weight  engine-size        bore      stroke  \
count  205.000000   205.000000   205.000000  201.000000  201.000000   
mean    53.724878  2555.565854   126.907317    3.329751    3.255423   
std      2.443522   520.680204    41.642693    0.273539    0.316717   
min     47.800000  1488.000000    61.000000    2.540000    2.070000   
25%     52.000000  2145.000000    97.000000    3.150000    3.110000   
50%     54.100000  2414.000000   120.000000    3.310000    3.290000   
75%     55.500000  2935.000000   141.000000    3.590000    3.410000   
max     59.800000  4066.000000   326.000000    3.940000    4.170000   

       compression-ratio  horsepower     peak-rpm    city-mpg  highway-mpg  \
count         205.000000  203.000000   203.000000  205.000000   205.000000
mean           10.142537  104.256158  5125.369458   25.219512    30.751220
std             3.972040   39.714369   479.334560    6.542142     6.886443
min             7.000000   48.000000  4150.000000   13.000000    16.000000
25%             8.600000   70.000000  4800.000000   19.000000    25.000000
50%             9.000000   95.000000  5200.000000   24.000000    30.000000
75%             9.400000  116.000000  5500.000000   30.000000    34.000000
max            23.000000  288.000000  6600.000000   49.000000    54.000000

              price
count    201.000000
mean   13207.129353
std     7947.066342
min     5118.000000
25%     7775.000000
50%    10295.000000
75%    16500.000000
max    45400.000000
```

In [13]:

```
#(i)
sns.boxplot(x=df["num-of-doors"], y=df["horsepower"], hue=df["fuel-type"])
```

Out[13]:

```
<Axes: xlabel='num-of-doors', ylabel='horsepower'>
```

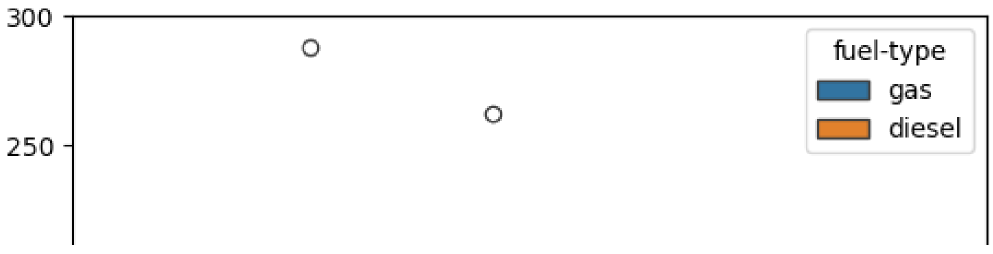


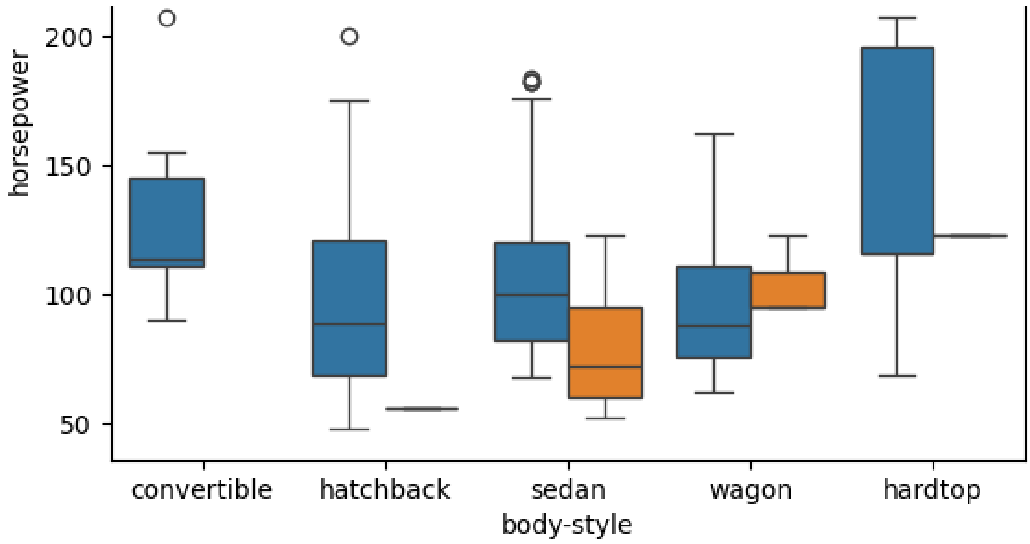
In [14]:

```
#(ii)
# Boxplot: Horsepower vs Body Style, grouped by Fuel Type
sns.boxplot(x=df["body-style"], y=df["horsepower"], hue=df["fuel-type"])
```

Out[14]:

```
<Axes: xlabel='body-style', ylabel='horsepower'>
```



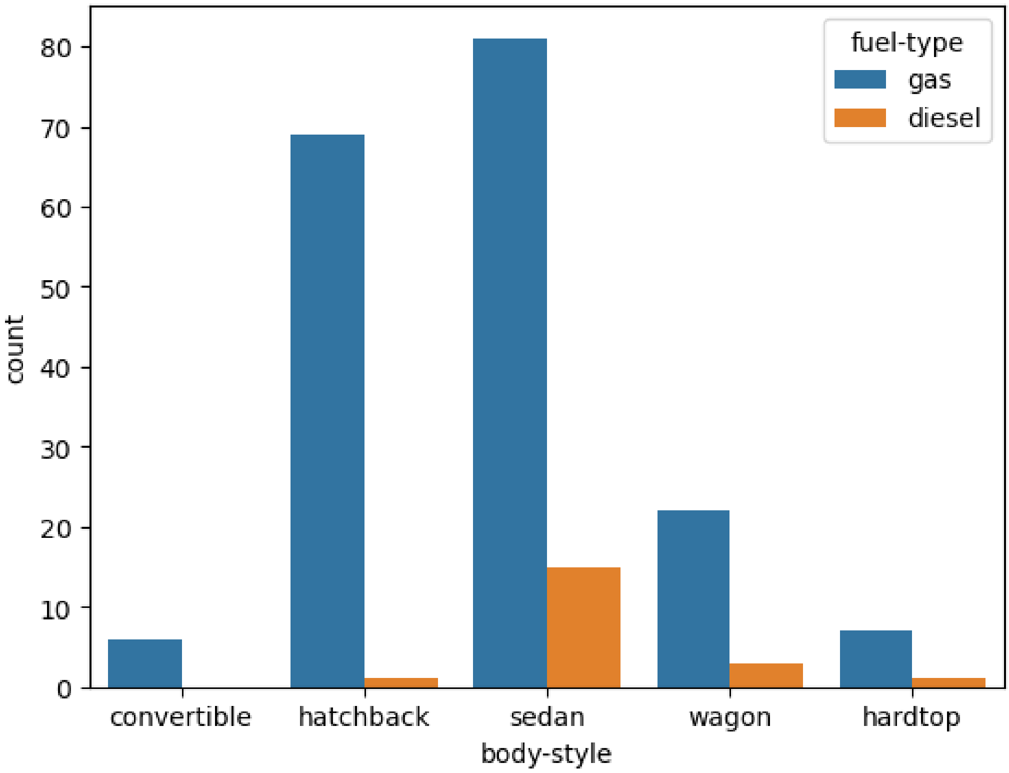


In [15]:

```
#(iii)
sns.countplot(x=df["body-style"], hue=df["fuel-type"])
```

Out[15]:

```
<Axes: xlabel='body-style', ylabel='count'>
```

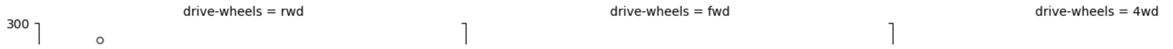


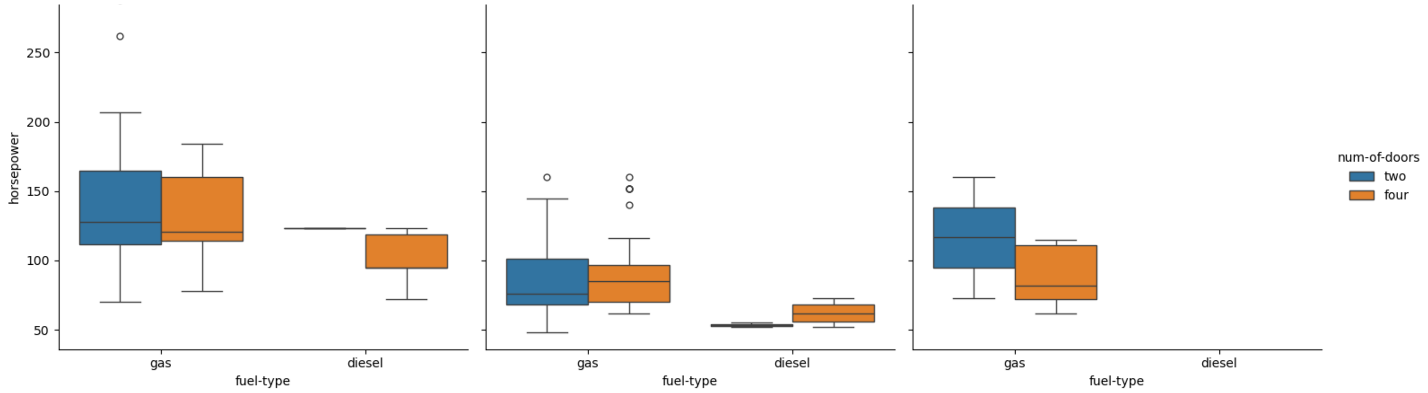
In [18]:

```
#(iv)
# Catplot: Horsepower by Fuel Type, Number of Doors, and Drive Wheels
sns.catplot(
    x="fuel-type",
    y="horsepower",
    hue="num-of-doors",
    col="drive-wheels",
    data=df,
    kind="box"
)
```

Out[18]:

```
<seaborn.axisgrid.FacetGrid at 0x2479ffa8860>
```



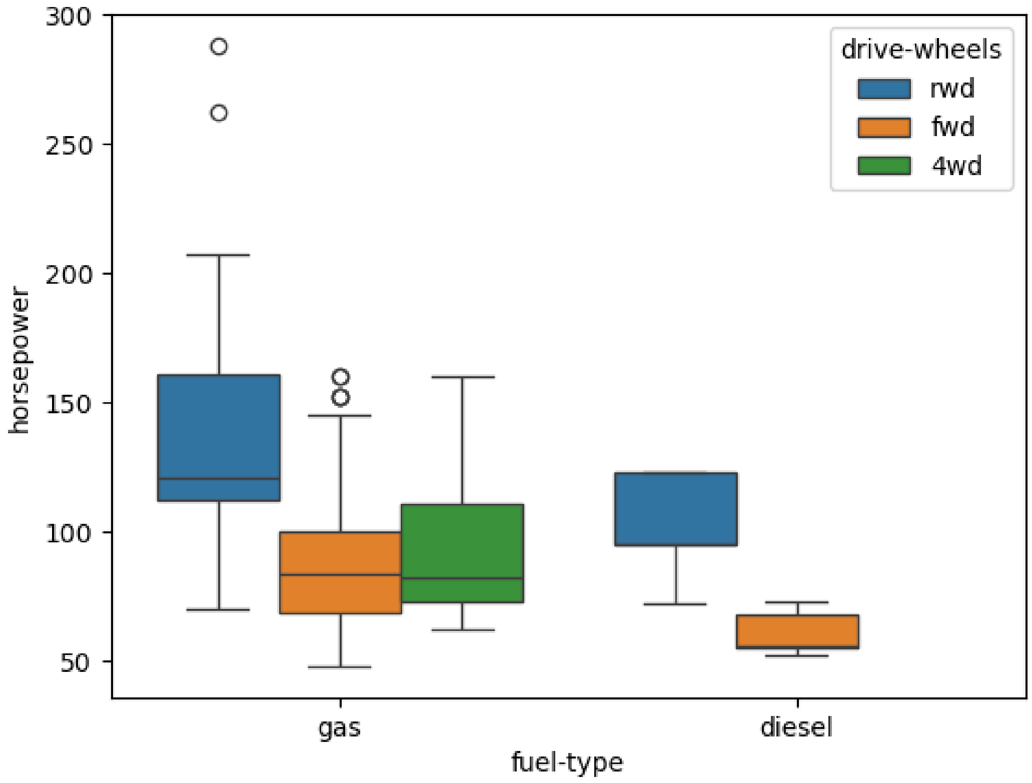


In [19]:

```
#(v)
# Boxplot: Horsepower vs Fuel Type, grouped by Drive Wheels
sns.boxplot(x=df['fuel-type'], y=df['horsepower'], hue=df['drive-wheels'])
```

Out[19]:

```
<Axes: xlabel='fuel-type', ylabel='horsepower'>
```



In [39]:

```
#B1
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('Bridge.csv', index_col='Date', parse_dates=True)
df.columns = ['Total', 'East', 'West']
df.head()
```

Out[39]:

	Total	East	West
Date			
2012-10-02 13:00:00	55	48	7
2012-10-02 14:00:00	130	75	55
2012-10-02 15:00:00	152	71	81
2012-10-02 16:00:00	278	111	167
2012-10-02 17:00:00	563	170	393

2012-10-02 17:00:00	563	170	393
	Total	East	West

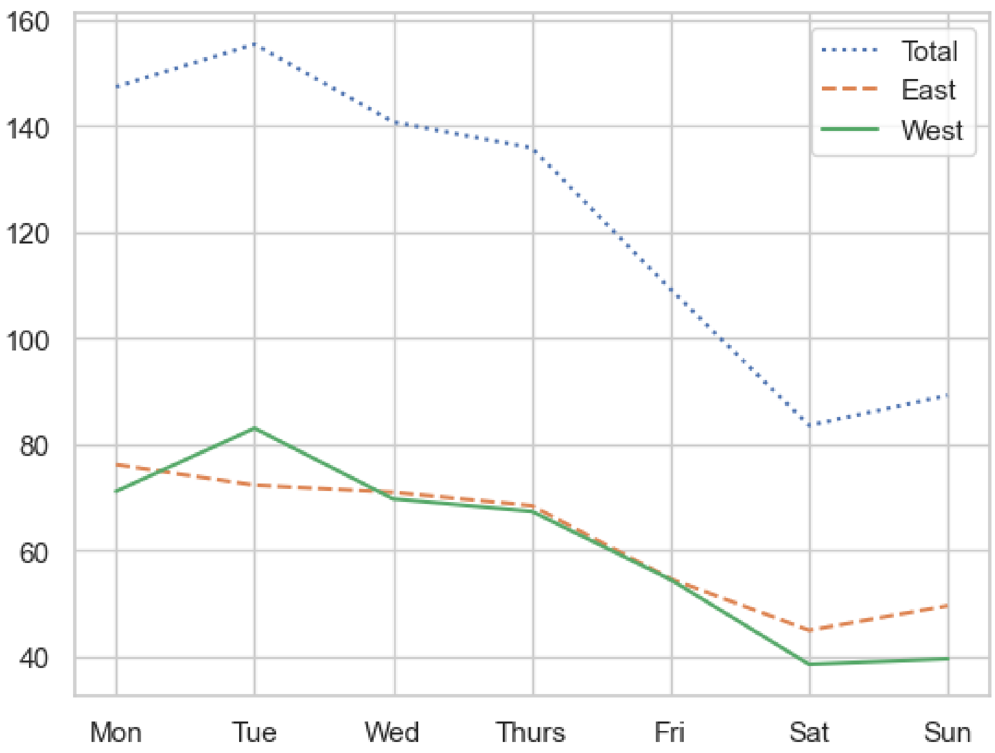
In [40]:

```
#B1 (i)
by_weekday=df.groupby(df.index.dayofweek).mean()
print(by_weekday)
by_weekday.index=['Mon','Tue','Wed','Thurs','Fri','Sat','Sun']
by_weekday.plot(style=[':','--','-'])
```

```
           Total       East       West
Date
0     147.375000  76.208333  71.166667
1     155.400000  72.342857  83.057143
2     140.750000  71.000000  69.750000
3     135.875000  68.479167  67.395833
4     109.255319  54.702128  54.553191
5      83.583333  45.000000  38.583333
6      89.250000  49.625000  39.625000
```

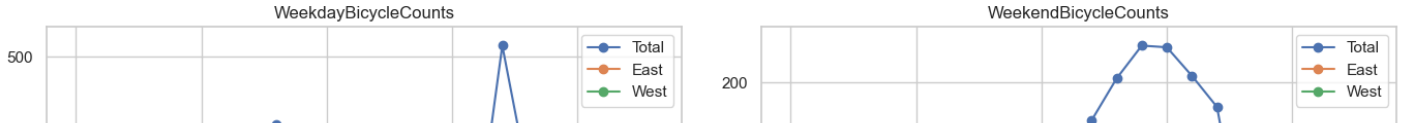
Out[40]:

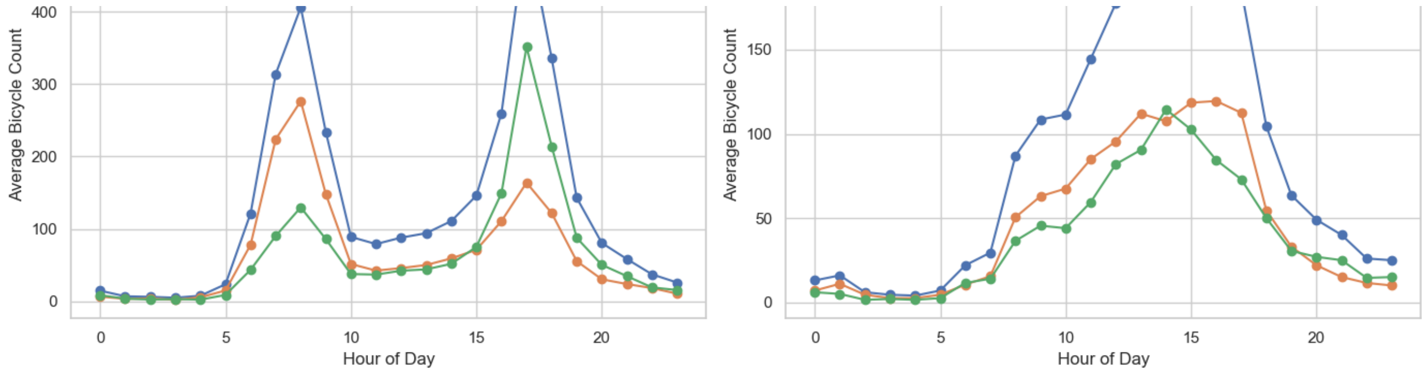
```
<Axes: >
```



In [44]:

```
#B1 (ii)
import numpy as np
import matplotlib.pyplot as plt

weekend_array=np.where(df.index.dayofweek<5,'Weekday','Weekend')
by_time=df.groupby([weekend_array,df.index.hour]).mean()
fig,ax=plt.subplots(1,2,figsize=(14,5))
by_time.loc['Weekday'].plot(ax=ax[0],title='WeekdayBicycleCounts',marker='o')
by_time.loc['Weekend'].plot(ax=ax[1],title='WeekendBicycleCounts',marker='o')

for a in ax:
    a.set_xlabel('Hour of Day')
    a.set_ylabel('Average Bicycle Count')
    a.grid(True)
plt.tight_layout()
plt.show()
```





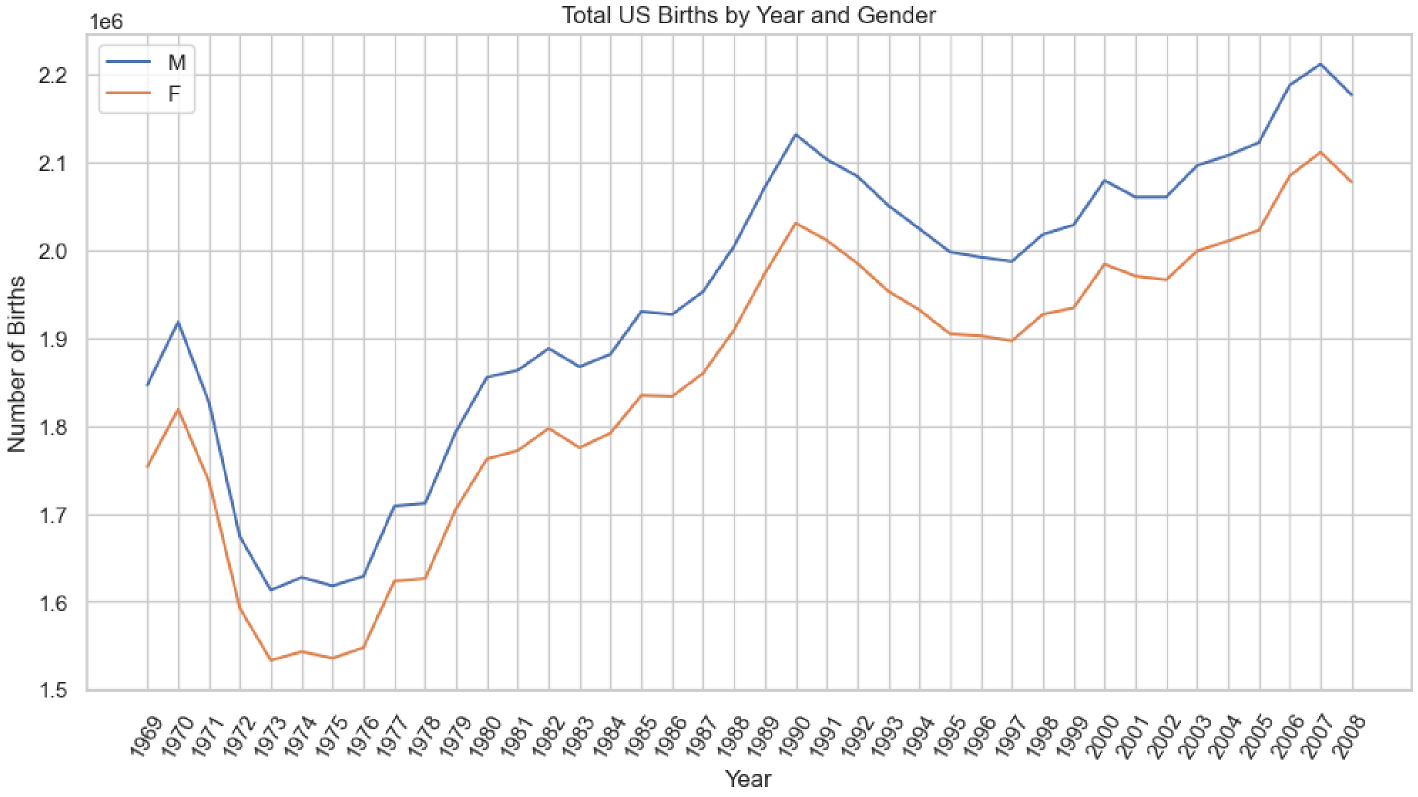
In [53]:

```
#B2
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('births.csv')
df.head()
print(df.columns)
```

```
Index(['year', 'month', 'day', 'gender', 'births'], dtype='object')
```

In [46]:

```
#B2 (i)
total=df.groupby(['year','gender'])['births'].sum()
plt.figure(figsize=(12,6))
for gender in['M','F']:
    plt.plot(total.loc[:,gender].index,total.loc[:,gender].values,label=gender)
    years=total.index.get_level_values('year').unique()
plt.xticks(ticks=years,rotation=60)
plt.title('Total US Births by Year and Gender')
plt.xlabel('Year')
plt.ylabel('Number of Births')
plt.legend()
plt.grid(True)
plt.show()
```



In [47]:

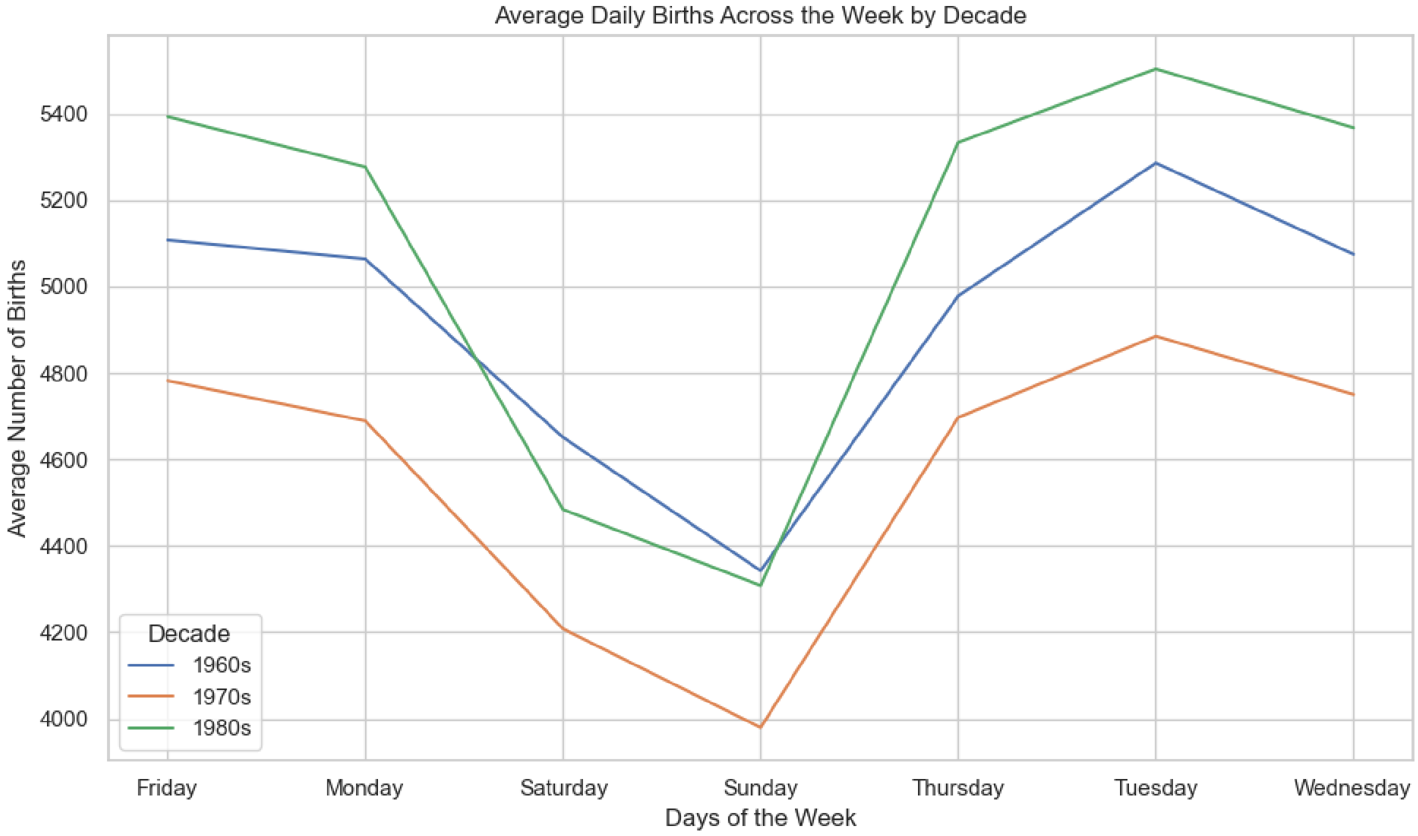
```
#B2 (ii)
# Filter valid dates(Remove Invalid Dates)
```

```
df = df[(df['month'] >= 1) & (df['month'] <= 12) & (df['day'] >= 1) & (df['day'] <= 31)
]

# Convert to datetime and get day of week
df['date'] = pd.to_datetime(df[['year', 'month', 'day']], errors='coerce')
df['day_of_week'] = df['date'].dt.day_name()
df['decade']=(df['year']//10)*10 #data.loc[:,'decade']=(data['year']//10)*10

# Group by decade and day of week to calculate average births
avg_daily = df.groupby(['decade', 'day_of_week'])['births'].mean().unstack()
print(avg_daily)
```

```
day_of_week       Friday       Monday     Saturday       Sunday     Thursday  \
decade
1960         5107.884615  5063.826923  4651.057692  4342.346154  4978.288462
1970         4782.095785  4689.097701  4207.784483  3979.278736  4696.923372
1980         5393.087234  5276.907249  4483.901064  4308.120469  5333.485106

day_of_week      Tuesday    Wednesday
decade
1960         5286.096154  5074.622642
1970         4885.252399  4750.376200
1980         5503.842553  5367.642553
```

In [48]:

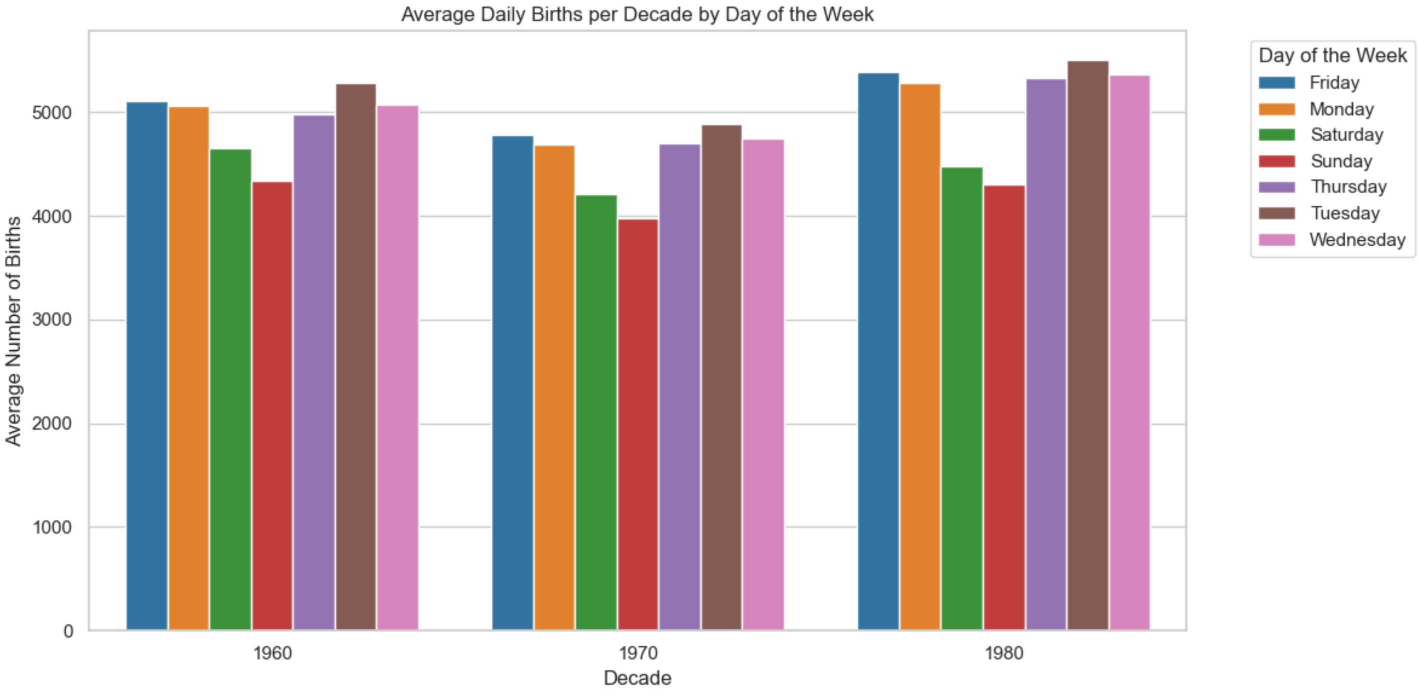
```
plt.figure(figsize=(10, 6))
for decade in avg_daily.index:
    plt.plot(avg_daily.columns, avg_daily.loc[decade], label=f"{decade}s")
plt.title("Average Daily Births Across the Week by Decade")
plt.xlabel("Days of the Week")
plt.ylabel("Average Number of Births")
plt.legend(title="Decade")
plt.grid(True)
plt.tight_layout()
plt.show()
```



In [49]:

```
import seaborn as sns
sns.set(style="whitegrid")
avg_daily_plot = avg_daily.reset_index().melt(id_vars='decade', var_name='day_of_week', v
alue_name='avg_births')
```

```python
plt.figure(figsize=(12, 6))
sns.barplot(
    data=avg_daily_plot,
    x='decade',
    y='avg_births',
    hue='day_of_week',
    palette='tab10'
)

plt.title('Average Daily Births per Decade by Day of the Week')
plt.xlabel('Decade')
plt.ylabel('Average Number of Births')
plt.legend(title='Day of the Week', bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.show()
```


Average Daily Births per Decade by Day of the Week
Average Number of Births
Decade
Day of the Week
Friday
Monday
Saturday
Sunday
Thursday
Tuesday
Wednesday