1 Q1

```
import openpyxl
import pandas as pd
pelican_data = pd.read_excel(r'C:\Users\User\Downloads\
    HW_statistics\PelicanStores6.xlsx')
net_sales_by_customer_type = pelican_data.groupby('Type of
    Customer')['Net Sales'].describe()

pelican_data['Used Proprietary Card'] = pelican_data['Method
     of Payment'] == 'Proprietary Card'
net_sales_by_payment_method = pelican_data.groupby('Used
    Proprietary Card')['Net Sales'].describe()

net_sales_for_age_45_above = pelican_data[pelican_data['Age
    '] >= 45]['Net Sales'].describe()

net_sales_by_customer_type, net_sales_by_payment_method,
    net_sales_for_age_45_above
```

2 Q2

```

def calculate_percentiles_exclusive(data, percentiles):

    sorted_data = data.sort_values()

    ranks = [(p/100) * (len(sorted_data) + 1) for p in
    percentiles]

    percentile_values = [sorted_data.iloc[max(int(rank) - 1,
     0)] for rank in ranks]
    return percentile_values

percentiles_to_calculate = [30, 25, 95, 75]

age_percentiles_exclusive = calculate_percentiles_exclusive(
    pelican_data['Age'], percentiles_to_calculate)

iqr_exclusive = age_percentiles_exclusive[3] -
    age_percentiles_exclusive[1]

age_percentiles_exclusive, iqr_exclusive
```

3 Q3

```
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 6))
sns.boxplot(x='Used Proprietary Card', y='Net Sales', data=
    pelican_data)
plt.title('Net Sales Comparison by Payment Method')
plt.xlabel('Used Proprietary Card')
plt.ylabel('Net Sales ($)')
plt.xticks([0, 1], ['No', 'Yes'])  # Label x-axis with more
    descriptive labels

plt.show()
```

4 Q4

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

correlation_coefficient = pelican_data['Age'].corr(
    pelican_data['Net Sales'])

plt.figure(figsize=(10, 6))
sns.regplot(x='Age', y='Net Sales', data=pelican_data,
    line_kws={"color":"red","alpha":0.7,"lw":2})
plt.title('Scatter Diagram of Age vs Net Sales with
    Trendline')
plt.xlabel('Age')
plt.ylabel('Net Sales ($)')

plt.show(), correlation_coefficient
```

5 Q5

```
contingency_table = pd.crosstab(index=pelican_data['Type of
    Customer'], columns=pelican_data['Marital Status'],
    margins=True)

contingency_table_percentage = contingency_table.div(
    contingency_table['All'], axis=0) * 100
contingency_table_percentage = contingency_table_percentage.
    drop(columns='All').drop('All')

```

```
6 contingency_table_formatted = contingency_table.drop(columns
      ='All').drop('All').astype(str) + " (" +
      contingency_table_percentage.round(1).astype(str) + "%)"
7
8 contingency_table_formatted
```