import numpy as np  
import matplotlib.pyplot as plt  
import pandas as pd  
import seaborn as sns  
  
# Tworzenie tablicy za pomocą biblioteki numpy  
tabela = np.array([  
 [1, 'Anna', 30, 'Kobieta', 'Miasto', 'Palacz'],  
 [2, 'Jan', 35, 'Mężczyzna', 'Wieś', 'Niepalący'],  
 [3, 'Maria', 28, 'Kobieta', 'Miasto', 'Palacz'],  
 [4, 'Piotr', 42, 'Mężczyzna', 'Wieś', 'Palacz'],  
 [5, 'Ewa', 38, 'Kobieta', 'Miasto', 'Niepalący']  
])  
  
# Wyświetlanie tabeli  
print(tabela)  
  
**#zad1.1**  
# ile osób jest w tabelce  
liczba\_osob = len(tabela)  
print(f"Liczba osób w tabelce: {liczba\_osob}")  
  
#zad1.2  
# ile jest kobiet  
kobiety = tabela[:, 3] == 'Kobieta'  
liczba\_kobiet = np.sum(kobiety)  
print(f"Liczba kobiet: {liczba\_kobiet}")  
  
#zad1.3  
# ile jest placych osob ponizej 40 roku zycia  
wiek\_paliacych = tabela[tabela[:, 2].astype(int) < 40][:, 5] == 'Palacz'  
liczba\_paliacych\_mlodszych = np.sum(wiek\_paliacych)  
print(f"Liczba osób poniżej 40 lat palących: {liczba\_paliacych\_mlodszych}")  
  
#zad1.4  
# gdzie mieszka wiekszosc palacych kobiet: na wsi czy w miescie  
palacych\_kobiety = tabela[(tabela[:, 3] == 'Kobieta') & (tabela[:, 5] == 'Palacz')]  
wiejskie\_kobiety = np.sum(palacych\_kobiety[:, 4] == 'Wieś')  
miastowe\_kobiety = np.sum(palacych\_kobiety[:, 4] == 'Miasto')  
  
if wiejskie\_kobiety > miastowe\_kobiety:  
 print("Większość palących kobiet mieszka na wsi.")  
elif wiejskie\_kobiety < miastowe\_kobiety:  
 print("Większość palących kobiet mieszka w mieście.")  
else:  
 print("Liczba palących kobiet jest równomierna w miastach i na wsiach.")  
  
#zad1.5  
# jaki jest sredni wiek osob palacych i niepalacych  
wiek\_palacych = tabela[tabela[:, 5] == 'Palacz'][:, 2].astype(int)  
wiek\_niepalacych = tabela[tabela[:, 5] == 'Niepalący'][:, 2].astype(int)  
  
sredni\_wiek\_palacych = np.mean(wiek\_palacych)  
sredni\_wiek\_niepalacych = np.mean(wiek\_niepalacych)  
  
print(f"Średni wiek osób palących: {sredni\_wiek\_palacych}")  
print(f"Średni wiek osób niepalących: {sredni\_wiek\_niepalacych}")  
  
#zad1.6  
# jaki jest maksymalny wiek osob mieszkajacych na wsi  
wiek\_mieszkajacych\_na\_wsi = tabela[tabela[:, 4] == 'Wieś'][:, 2].astype(int)  
maks\_wiek\_na\_wsi = np.max(wiek\_mieszkajacych\_na\_wsi)  
  
print(f"Maksymalny wiek osób mieszkających na wsi: {maks\_wiek\_na\_wsi}")  
  
**#zad2**  
  
  
# Przygotowanie danych  
x = np.linspace(0, 1, 100) # Przedział [0, 1] podzielony na 100 równoodległych punktów  
y1 = x\*\*2 # Funkcja y = x^2  
y2 = np.sqrt(x) # Funkcja y = √x  
  
# Rysowanie wykresów funkcji y = x^2 i y = √x w jednym oknie  
plt.subplot(2, 1, 1)  
plt.plot(x, y1, 'r-', label='y = x^2')  
plt.title('Wykres funkcji y = x^2')  
plt.xlabel('x')  
plt.ylabel('y')  
plt.legend()  
  
plt.subplot(2, 1, 2)  
plt.plot(x, y2, 'b-', label='y = √x')  
plt.title('Wykres funkcji y = √x')  
plt.xlabel('x')  
plt.ylabel('y')  
plt.legend()  
  
# Wyświetlenie okna z wykresami  
plt.tight\_layout()  
plt.show()  
  
**#zad3**  
  
  
# Wczytanie danych z pliku CSV  
df = pd.read\_csv('titanic.csv')  
  
# Ile kobiet z pierwszej klasy przeżyło?  
female\_survived = df[(df['sex'] == 'female') & (df['pclass'] == 1) & (df['survived'] == 1)]  
num\_female\_survived = len(female\_survived)  
print("Liczba kobiet z pierwszej klasy, które przeżyły:", num\_female\_survived)  
  
# Jaki był średni wiek mężczyzn, którzy nie przeżyli?  
male\_not\_survived = df[(df['sex'] == 'male') & (df['survived'] == 0)]  
mean\_age\_male\_not\_survived = male\_not\_survived['age'].mean()  
print("Średni wiek mężczyzn, którzy nie przeżyli:", mean\_age\_male\_not\_survived)  
  
# Narysuj wykres punktowy zależności opłaty (fare) od wieku pasażera, kolorem zaznacz płeć  
plt.scatter(df['age'], df['fare'], c=df['sex'].map({'male': 'blue', 'female': 'red'}))  
plt.xlabel('Wiek')  
plt.ylabel('Opłata (Fare)')  
plt.title('Zależność opłaty od wieku pasażera (kolorem zaznaczona płeć)')  
plt.show()  
  
# Narysuj wykres słupkowy procentu osób, które przeżyły, w zależności od klasy  
survival\_percent\_by\_class = df.groupby('pclass')['survived'].mean() \* 100  
survival\_percent\_by\_class.plot(kind='bar')  
plt.xlabel('Klasa')  
plt.ylabel('Procent przeżytych')  
plt.title('Procent osób, które przeżyły, w zależności od klasy')  
plt.show()  
  
  
**#zad4**  
  
  
# Wczytanie danych iris  
iris = sns.load\_dataset('iris')  
  
# Narysowanie wykresu punktowego  
sns.scatterplot(x='petal\_length', y='petal\_width', hue='species', size='sepal\_length', sizes=(20, 200),  
 palette='Greens', data=iris)  
  
# Ustawienie tytułu i osi  
plt.title('Zależność szerokości płatka od długości płatka dla gatunków iris')  
plt.xlabel('Długość płatka (petal\_length)')  
plt.ylabel('Szerokość płatka (petal\_width)')  
  
# Wyświetlenie wykresu  
plt.show()