Министерство образования Республики Беларусь

УО «Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №6

По дисциплине: “Алгоритмы и структуры данных”

Тема: «Python. Основы Pandas»Вариант №1

**Выполнил**:

студент 2 курса группы ПО-7 Белев С.В.

**Проверила:**

Дряпко А.В.

Брест 2021

**Цель работы:** ознакомиться с основами библиотеки pandas и научиться строить графики с использованием библиотек matplotlib.pyplot и seaborn.

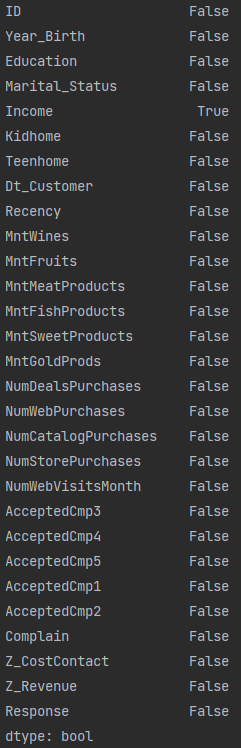
**Постановка задачи:**

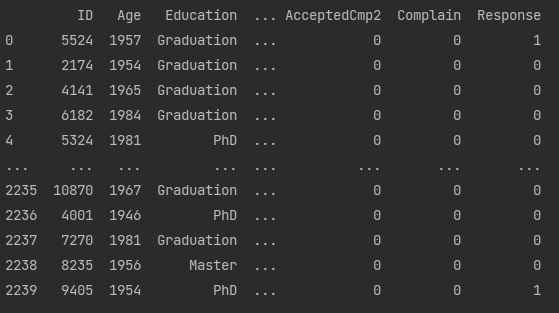
1. Загрузить датасет в pandas и проверить на доступность
2. Вывести общую информацию о датасете
3. Проверка наличия NULL-данных. При их наличии вывести на экран
4. Удалить колонки "Z\_CostContact", "Z\_Revenue"
5. Переименовать колонку "Year\_Birth" в "Age"
6. Оценить состояние колонок "Marital\_Status", "Education". Построить информативные диаграммы и гистограммы для каждой.
7. Создать гистаграмму по колонке "Age" и оценить на распределение по Гауссу.
8. Оценка полей "Kidhome" и "Teenhome", "Response" и "Income" (диаграммы и гистограммы)
9. Построить графики "Response", "Marital\_Status", "Education" и "Kidhome" по образцу
10. Построить heatmap для всех числовых колонок

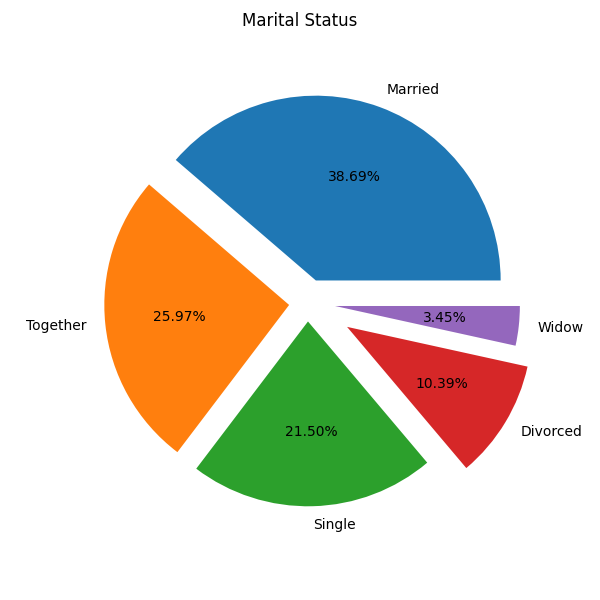
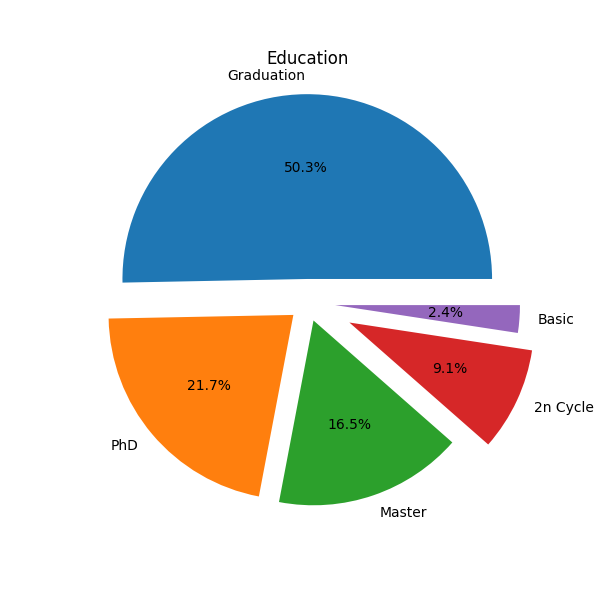
**Код программы:**

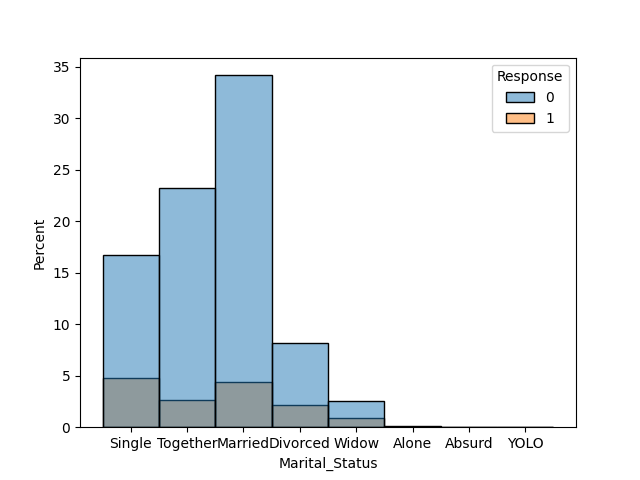
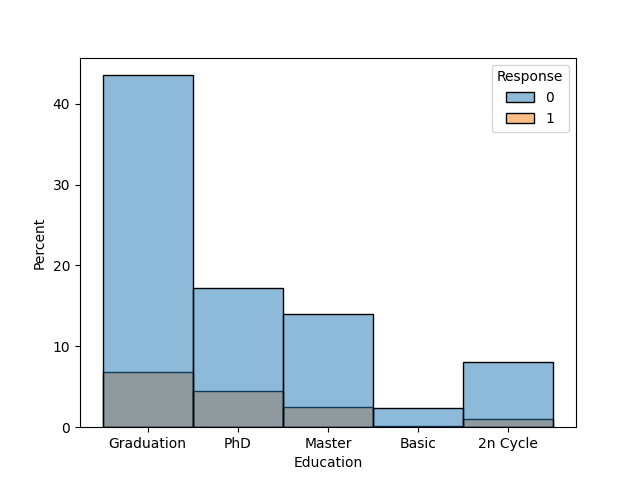
from matplotlib import pyplot as plt  
import pandas as pd  
import seaborn as sns  
  
table = pd.read\_csv('marketing\_campaign.csv', delimiter='\t')  
  
if table.isnull().any().any():  
 print(table.isnull().any())  
  
table.pop('Z\_CostContact')  
table.pop('Z\_Revenue')  
  
table.rename(columns={'Year\_Birth': 'Age'}, inplace=True)  
print(table)  
  
marital\_status\_c = table["Marital\_Status"].value\_counts().to\_frame()  
marital\_status = marital\_status\_c[marital\_status\_c["Marital\_Status"] > 50]  
education = table["Education"].value\_counts().to\_frame()  
  
marital\_status[marital\_status["Marital\_Status"] > 50].plot(  
 kind='pie',  
 autopct='%1.2f%%',  
 legend=None,  
 ylabel="",  
 subplots=True,  
 shadow=False,  
 explode=(0.13, 0.1, 0.1, 0.25, 0.15),  
 title="Marital Status",  
 figsize=(6, 6)  
)  
plt.show()  
  
education.plot(  
 y="Education",  
 kind='pie',  
 autopct='%1.1f%%',  
 legend=None,  
 ylabel="",  
 shadow=False,  
 explode=(0.13, 0.1, 0.1, 0.25, 0.15),  
 title="Education",  
 figsize=(6, 6)  
  
)  
plt.show()  
  
sns.histplot(data=table, x="Marital\_Status", stat='percent', hue="Response")  
plt.show()  
sns.histplot(data=table, x="Education", stat='percent', hue="Response")  
plt.show()  
  
  
def age(born):  
 return 2021 - int(born)  
  
  
table["Age"] = table["Age"].apply(age)  
filtered\_table = table[table["Age"] < 100]  
sns.histplot(x=filtered\_table["Age"], kde=True)  
plt.show()  
  
  
def bar(data\_name: str) -> None:  
 table[data\_name].value\_counts(normalize=True).mul(100).to\_frame().plot(  
 kind='bar',  
 legend=None,  
 xlabel=data\_name,  
 ylabel="Percent"  
 )  
 plt.show()  
  
  
bar("Kidhome")  
bar("Teenhome")  
bar("Response")  
sns.kdeplot(table["Income"], shade=True, clip=(10 \*\* 0, 10 \*\* 5.1))  
plt.show()  
  
  
def grph(hue: str, clip: tuple = (10 \*\* 0, 10 \*\* 5.05)):  
 plt.figure(figsize=(7, 7))  
 sns.kdeplot(data=table, x="Income", hue=hue, shade=True, clip=clip)  
 plt.show()  
  
  
grph("Response")  
grph("Marital\_Status")  
grph("Education")  
grph("Kidhome")  
  
plt.figure(figsize=(20, 20))  
sns.heatmap(table.corr(), cmap='coolwarm', annot=True, fmt='.1g')  
plt.show()

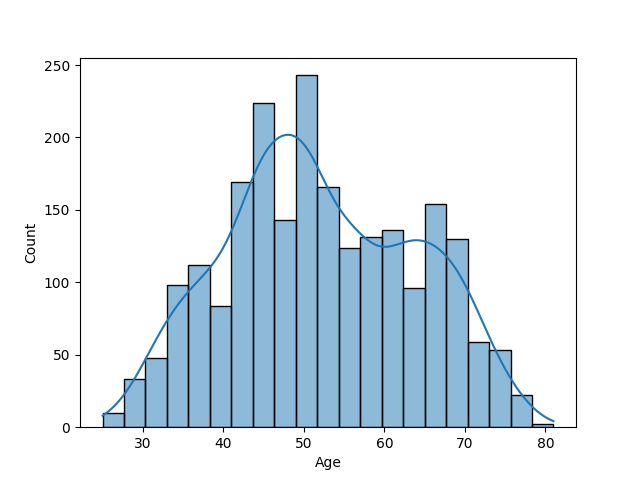
**Результат программы:**

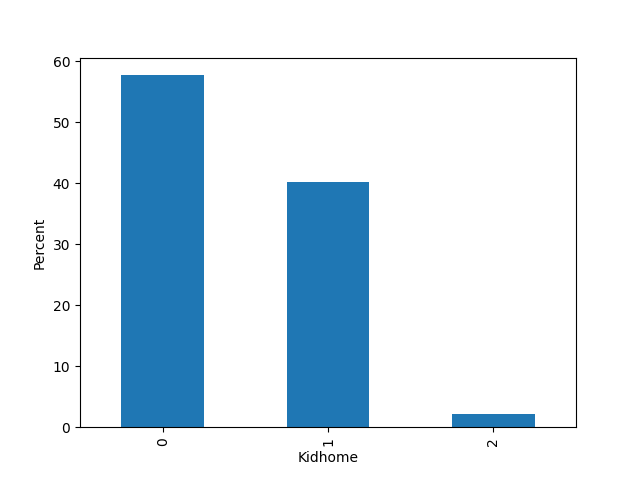
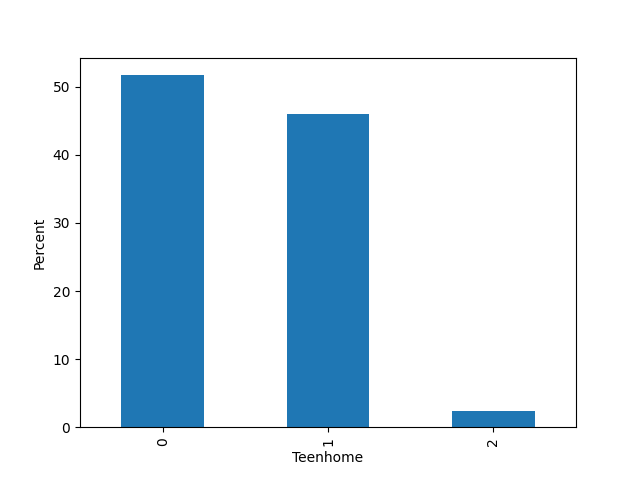


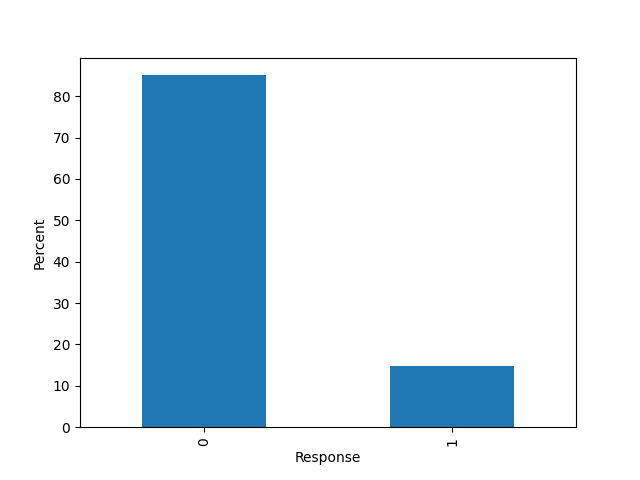
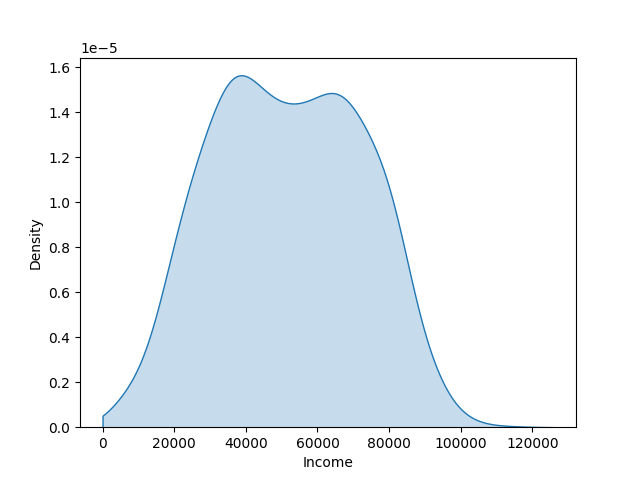


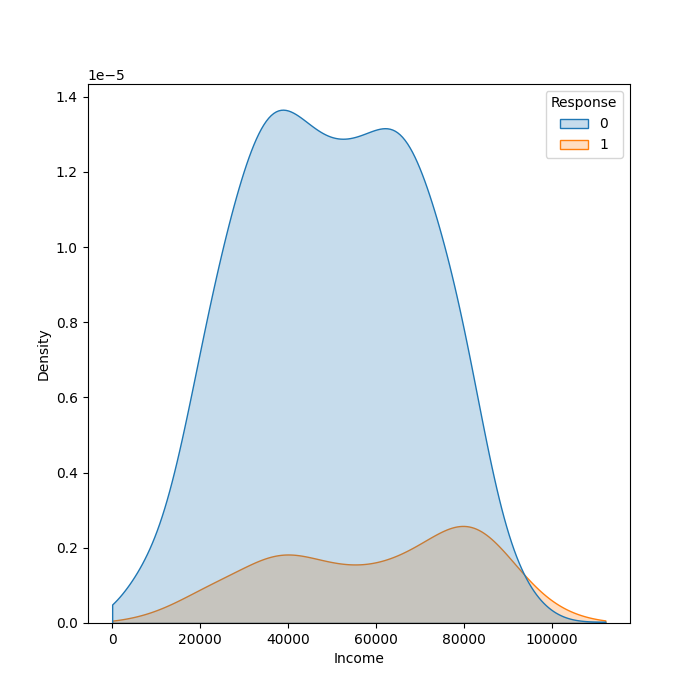
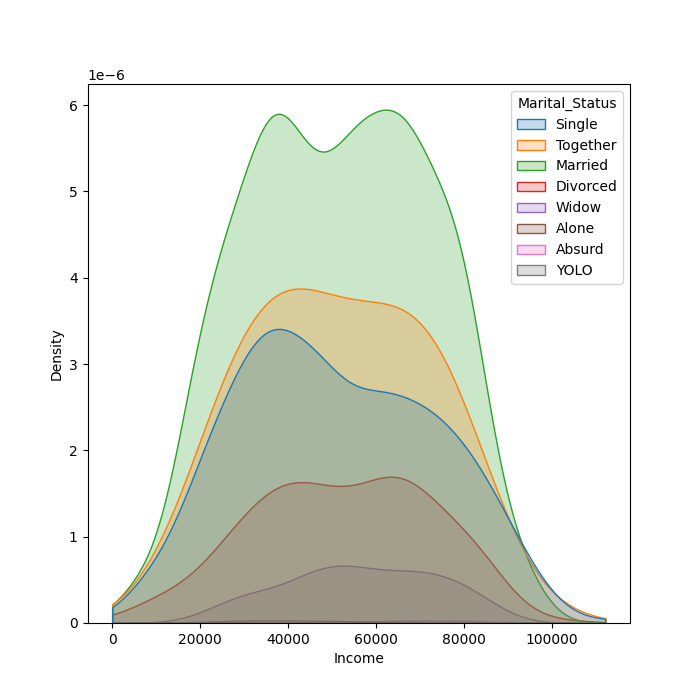
****

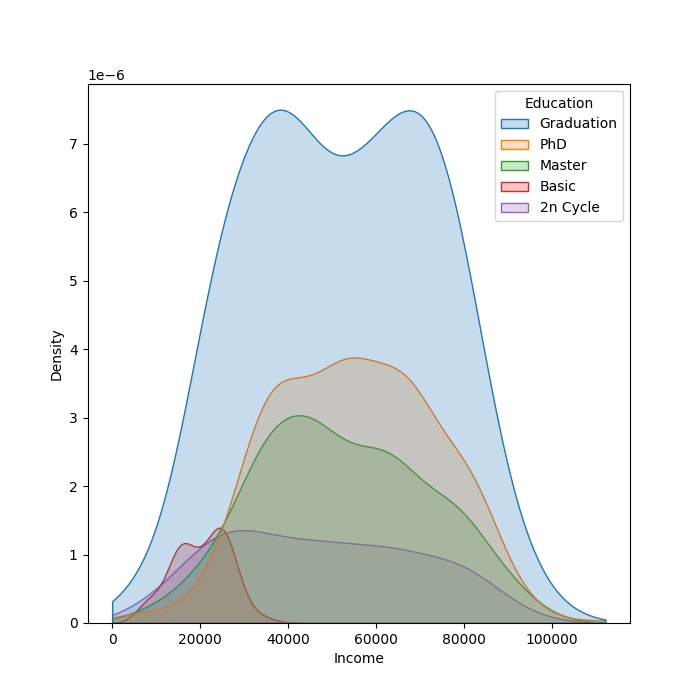
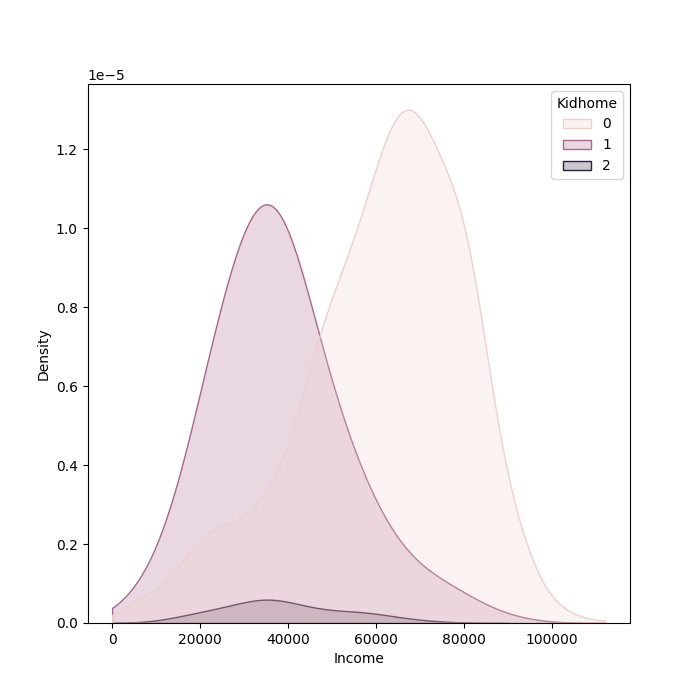
****

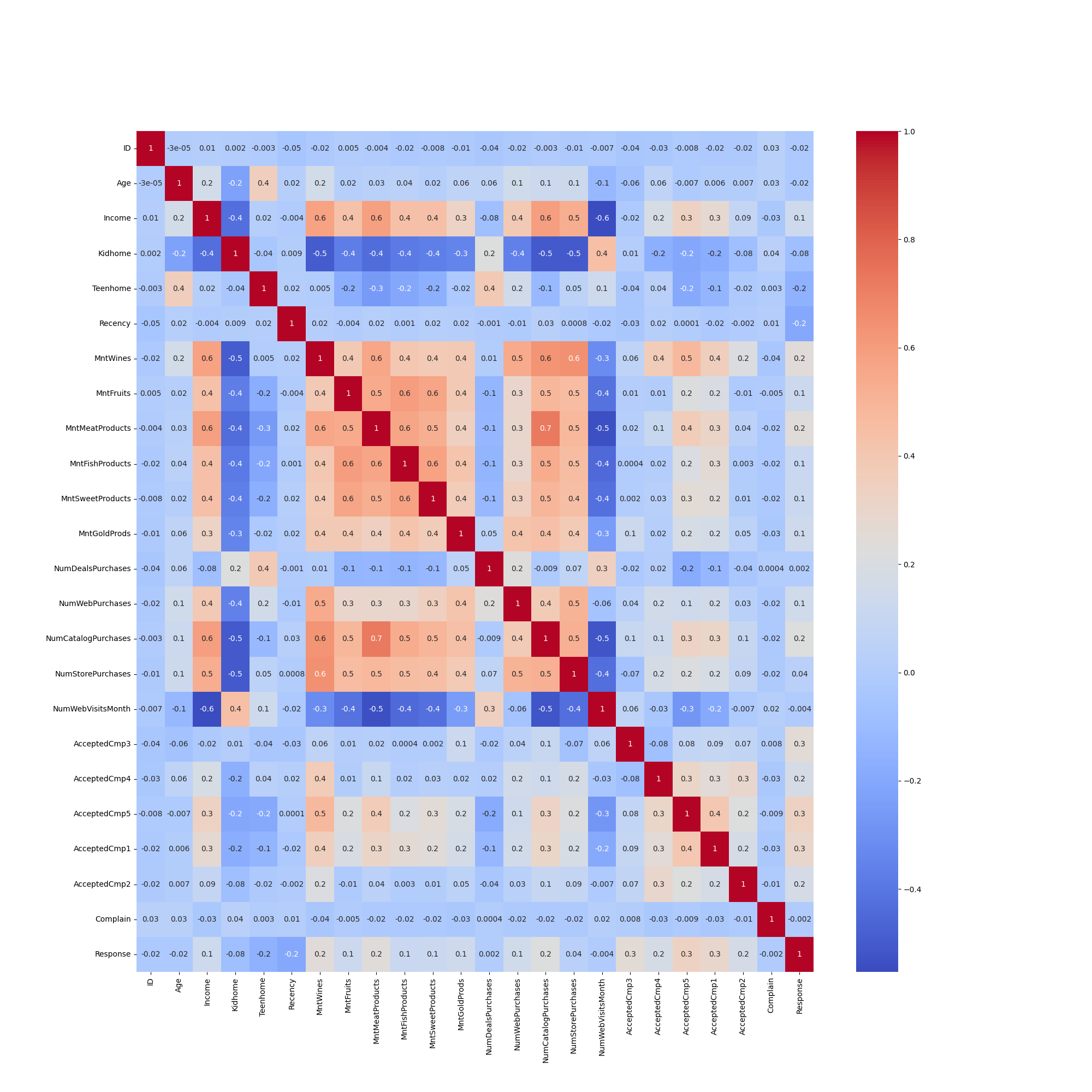
****

****

****

****

****

****

**Вывод:** Ознакомился с основами библиотеки pandas и научиться строить графики с использованием библиотек matplotlib.pyplot и seaborn.