Малахов В.В. ИУ5Ц-83Б | РК1 - вариант №27

Задача №4. Для заданного набора данных постройте основные графики, входящие в этап разведочного анализа данных. В случае наличия пропусков в данных удалите строки или колонки, содержащие пропуски. Какие графики Вы построили и почему? Какие выводы о наборе данных Вы можете сделать на основании построенных графиков?

Датасет № 3. https://www.kaggle.com/carlolepelaars/toy-dataset

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import sys
import warnings
warnings.filterwarnings("ignore")
sys.path
%matplotlib inline
```

Приступим к разведочному анализу

```
data =pd.read_csv('toy_dataset.csv')
data.head(3)
```

```
   Number    City Gender  Age   Income Illness
0       1  Dallas   Male   41  40367.0      No
1       2  Dallas   Male   54  45084.0      No
2       3  Dallas   Male   42  52483.0      No
```

Проверим на пропуски в данных

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150000 entries, 0 to 149999
Data columns (total 6 columns):
 #   Column   Non-Null Count   Dtype  
---  ------   --------------   -----  
 0   Number   150000 non-null  int64  
 1   City     150000 non-null  object 
 2   Gender   150000 non-null  object 
 3   Age      150000 non-null  int64  
 4   Income   150000 non-null  float64
 5   Illness  150000 non-null  object 
dtypes: float64(1), int64(2), object(3)
memory usage: 6.9+ MB
```

```
data.isnull().sum()
```

```
Number     0
City       0
Gender     0
Age        0
Income     0
Illness    0
dtype: int64
```

Описание столбцов

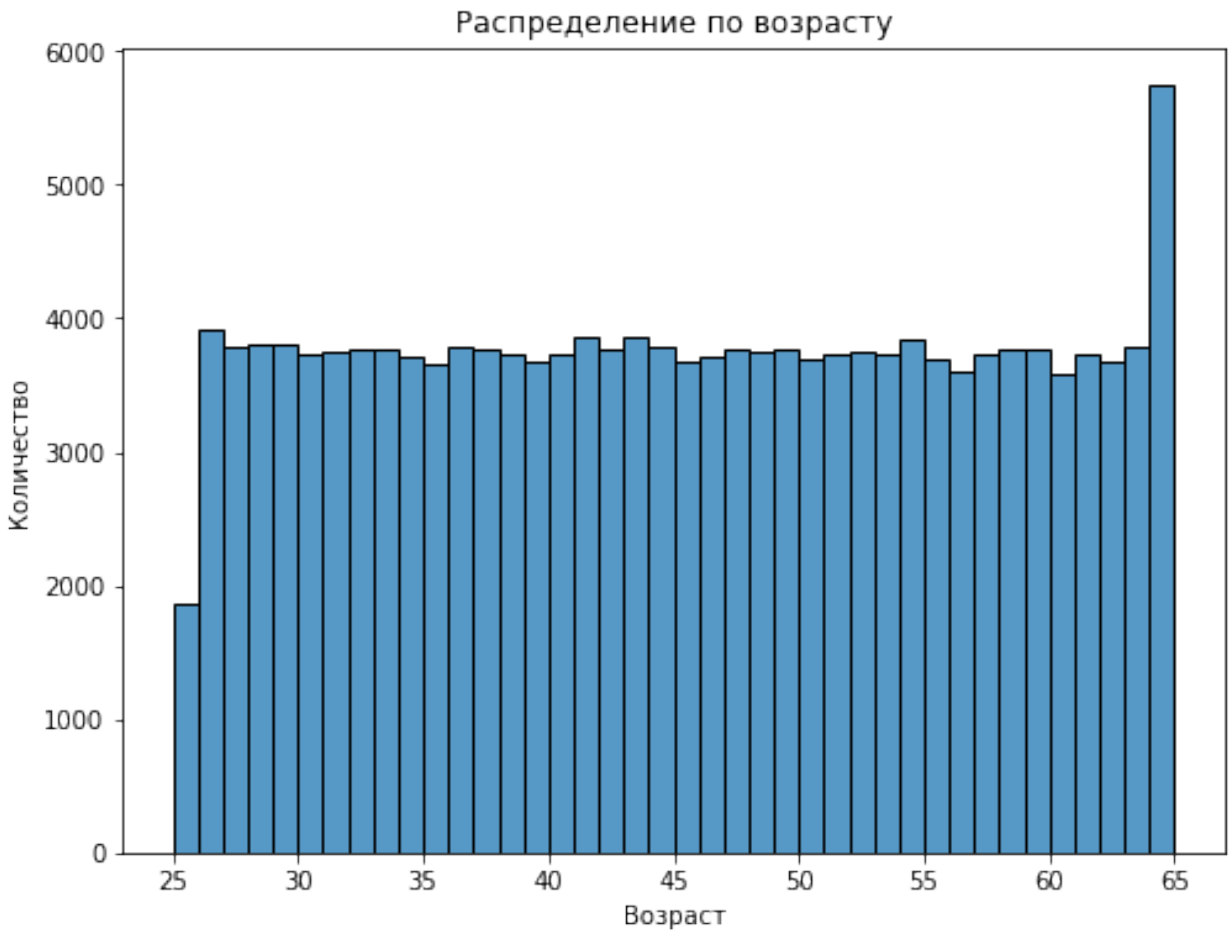
Number - Простой индексный номер для каждого ряда
City - Местонахождение человека
Gender - Пол человека
Age - Возвраст человека
Income - Годовой доход человека
Illnes - Болен ли человек?

```
data.describe()
```

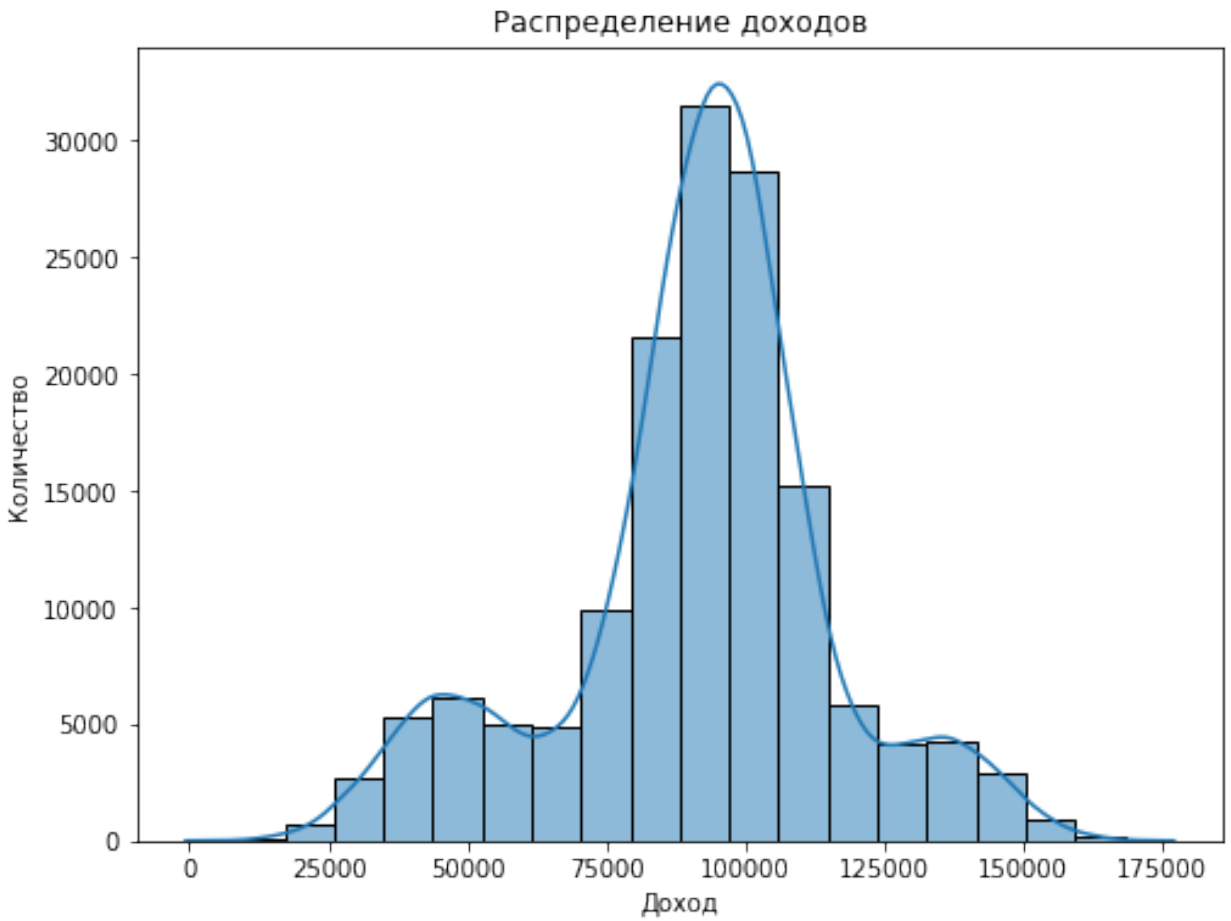
```
              Number             Age          Income
count  150000.000000  150000.000000  150000.000000
mean    75000.500000      44.950200   91252.798273
std     43301.414527      11.572486   24989.500948
min         1.000000      25.000000    -654.000000
25%     37500.750000      35.000000   80867.750000
50%     75000.500000      45.000000   93655.000000
75%    112500.250000      55.000000  104519.000000
max    150000.000000      65.000000  177157.000000
```

Построим диаграммы для каждой переменной в датасете и проанализируем, что там происходит

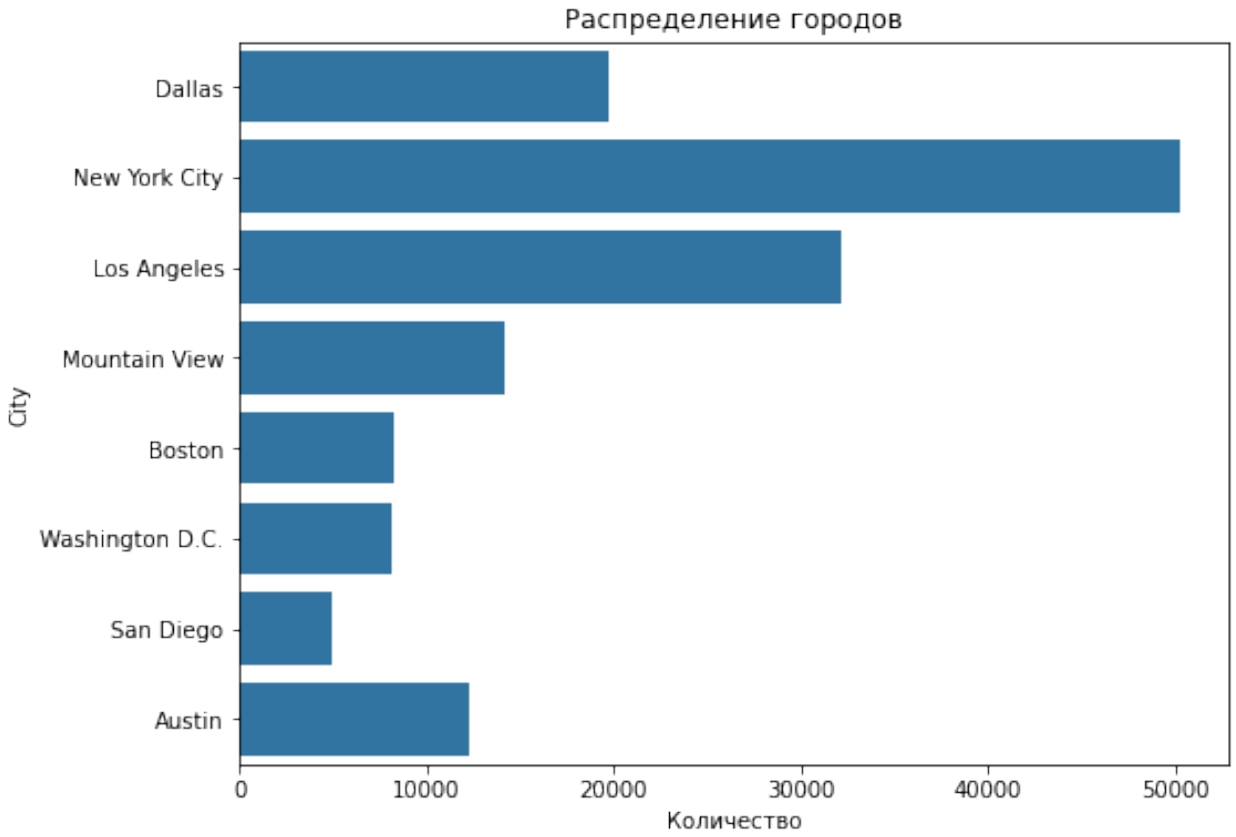
```
# Построение гистограммы для переменной Age
plt.figure(figsize=(8, 6))
sns.histplot(data['Age'], bins=40)
plt.title('Распределение по возрасту')
plt.xlabel('Возраст')
plt.ylabel('Количество')
plt.show()
```



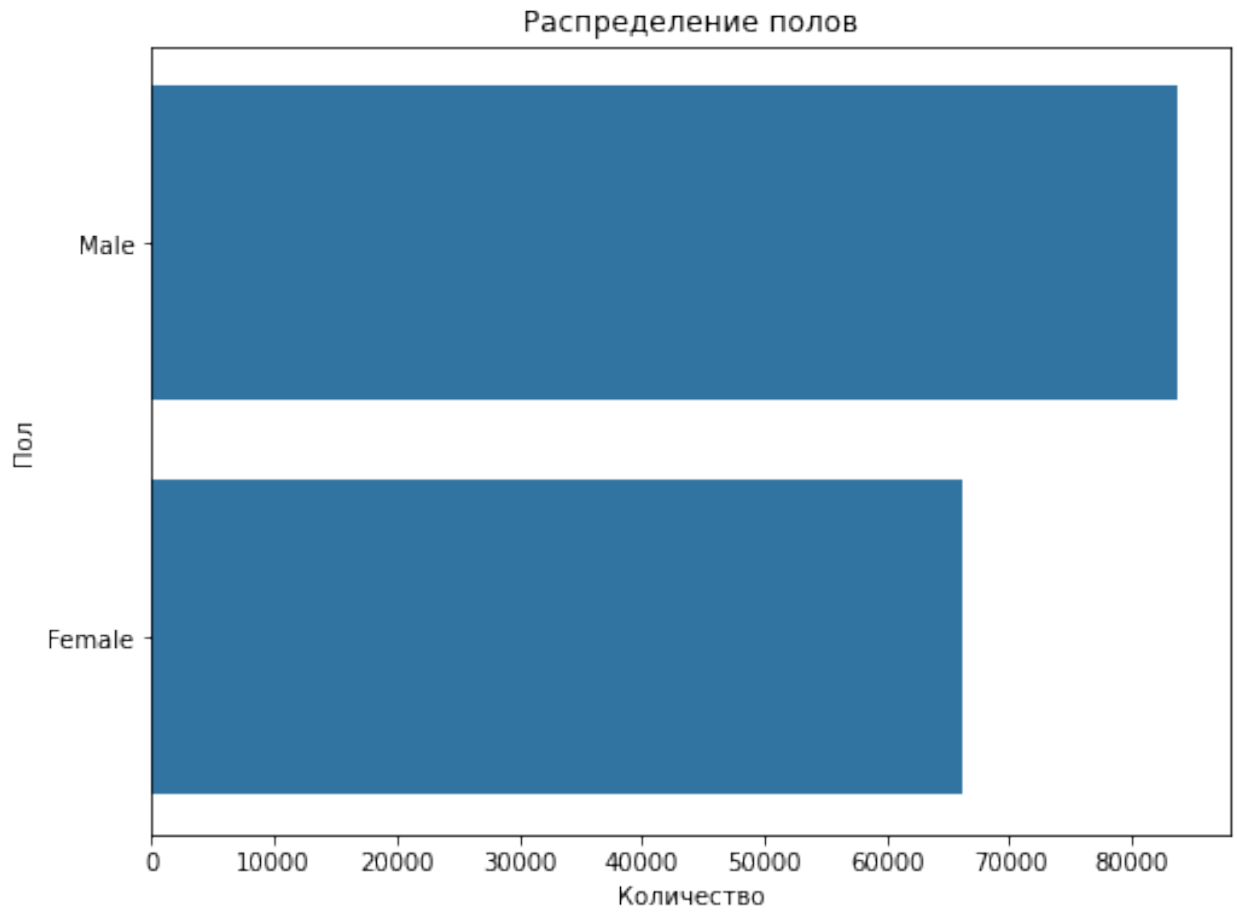

```
# Построение гистограммы для переменной Income
plt.figure(figsize=(8, 6))
sns.histplot(data['Income'], bins=20, kde=True)
plt.title('Распределение доходов')
plt.xlabel('Доход')
plt.ylabel('Количество')
plt.show()
```




```
# Построение столбчатой диаграммы для переменной City
plt.figure(figsize=(8, 6))
sns.countplot(data['City'])
plt.title('Распределение городов')
plt.xlabel('Количество')
plt.show()
```



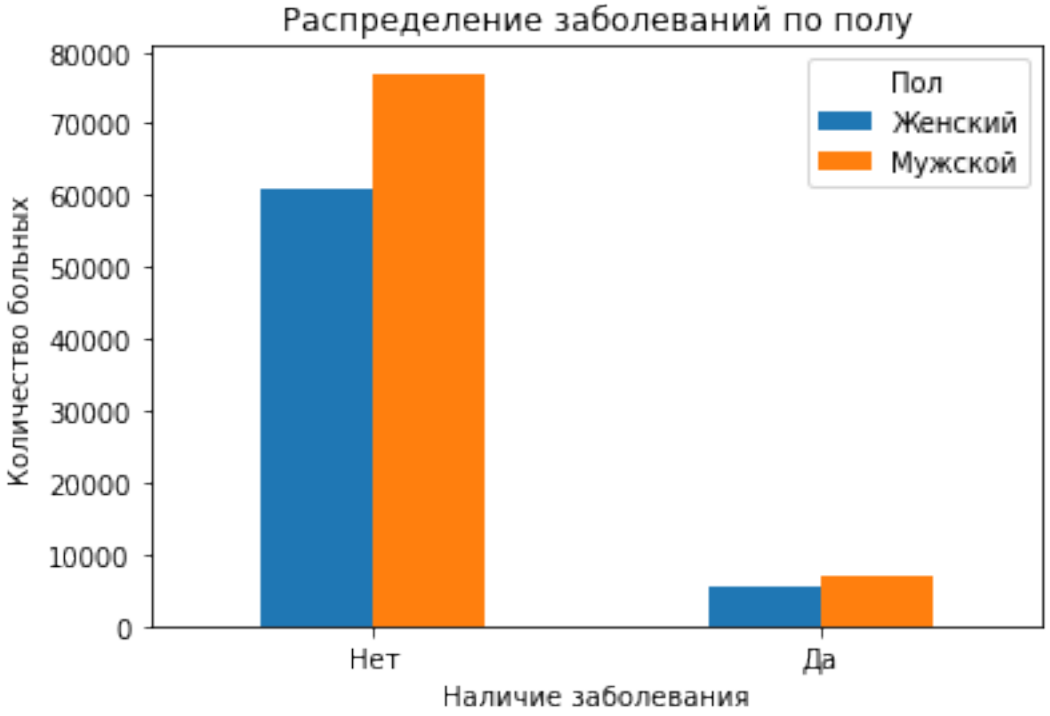

```
# Построение столбчатой диаграммы для переменной Gender
plt.figure(figsize=(8, 6))
sns.countplot(data['Gender'])
plt.title('Распределение полов')
plt.xlabel('Количество')
plt.ylabel('Пол')
plt.show()
```




```
# Построение круговой диаграммы для переменной Illness
plt.figure(figsize=(8, 6))
data['Illness'].value_counts().plot(kind='pie', autopct='%1.1f%%')
plt.title('Распределение больных')
plt.ylabel('')
plt.show()
```

Построим распределение заболевания в зависимости от пола человека

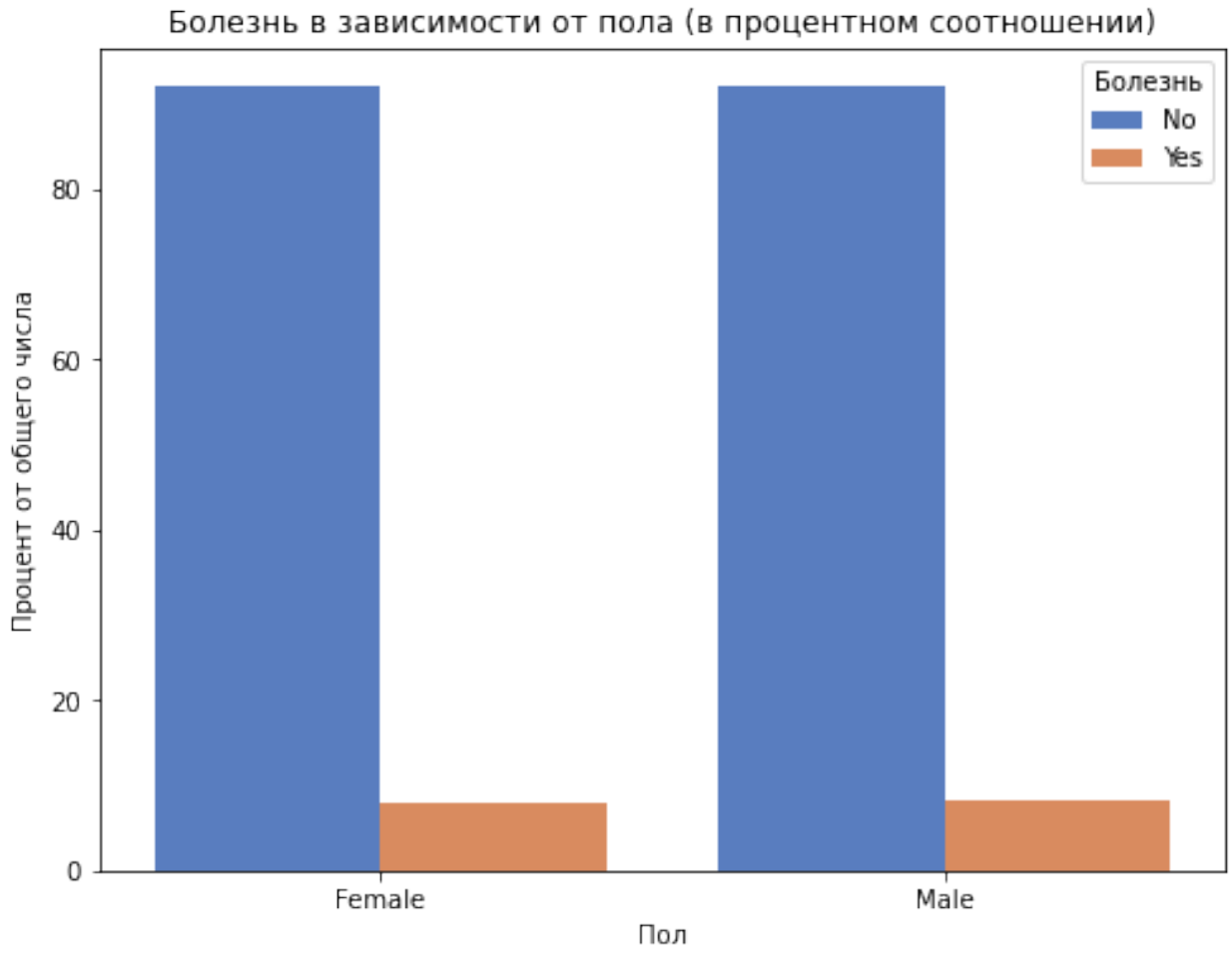
```
sex_counts = data['Gender'].value_counts()

sex_counts.plot(kind='bar', color=['lightblue', 'plum'])

plt.title('Распределение по полу')
plt.xlabel('Пол')
plt.ylabel('Количество')
plt.xticks(ticks=[0, 1], labels=['Мужской', 'Женский'], rotation=0)

plt.show()
pd.crosstab(data.Illness, data.Gender).plot(kind ="bar")
plt.title('Распределение заболеваний по полу')
plt.xlabel('Наличие заболевания')
plt.ylabel('Количество больных')
plt.xticks(ticks=[0, 1], labels=['Нет', 'Да'], rotation=0)
plt.legend(title='Пол', labels=['Женский', 'Мужской'])
plt.show()
```



Как мы видим, мужчины более активно учавствуют в опросах. Мы не можем сказать, что они болеют реже, т.к. нужно сравнивать относительное отношение

```
# Построение столбчатой диаграммы с процентным соотношением
gender_illness_percentage = data.groupby(['Gender',
```

```
'Illness']).size().groupby(level=0).apply(lambda x: 100 * x /
float(x.sum())).reset_index(name='Percentage')
plt.figure(figsize=(8, 6))
sns.barplot(x='Gender', y='Percentage', hue='Illness',
data=gender_illness_percentage, palette='muted')
plt.title('Болезнь в зависимости от пола (в процентном соотношении)')
plt.xlabel('Пол')
plt.ylabel('Процент от общего числа')
plt.legend(title='Болезнь')
plt.show()
```

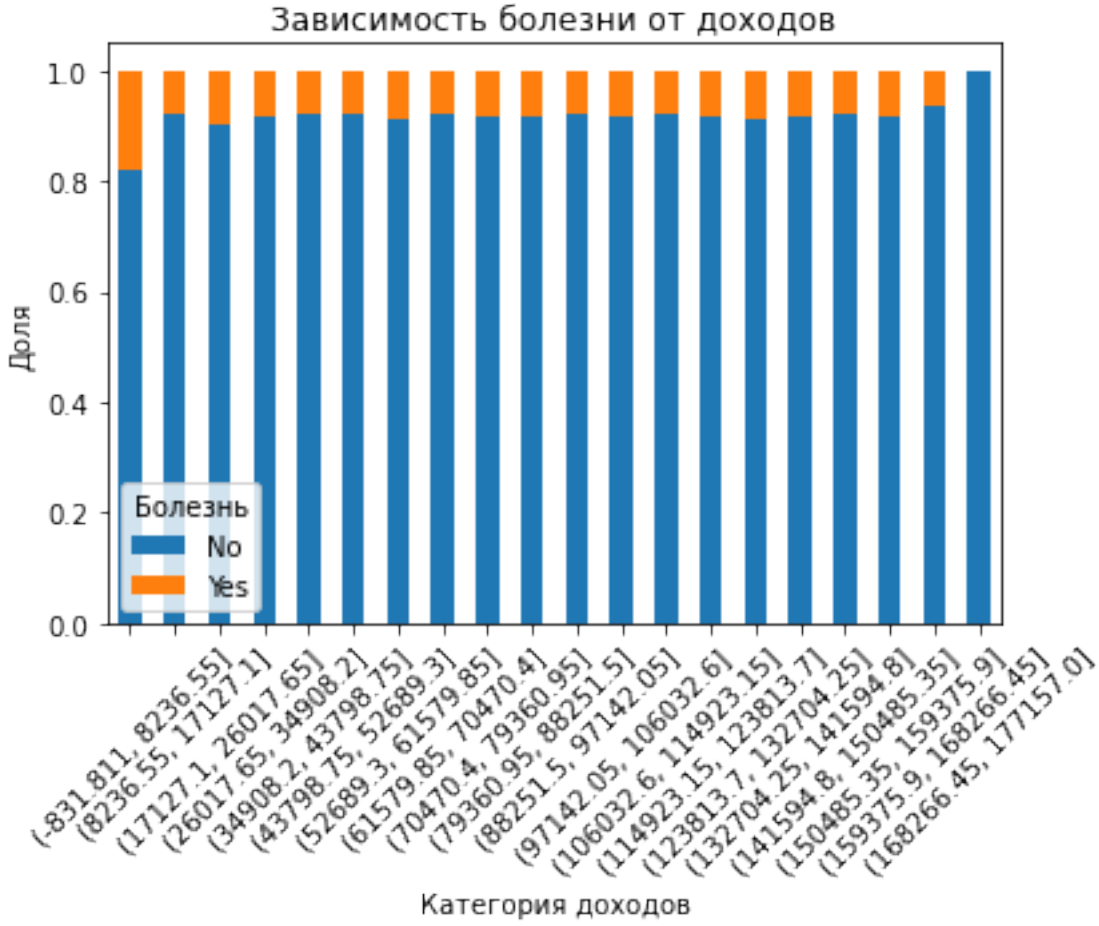


Как видим, оба пола болеют одинаково

```
# Разбивка данных на категории доходов
data['Income_Category'] = pd.cut(data['Income'], bins=20)

# Подсчет доли заболевших и здоровых в каждой категории доходов
illness_ratio = data.groupby('Income_Category')
['Illness'].value_counts(normalize=True).unstack()
```

```
# Построение столбчатой диаграммы
plt.figure(figsize=(10, 6))
illness_ratio.plot(kind='bar', stacked=True)
plt.title('Зависимость болезни от доходов')
plt.xlabel('Категория доходов')
plt.ylabel('Доля')
plt.legend(title='Болезнь')
plt.xticks(rotation=45)
plt.show()
```

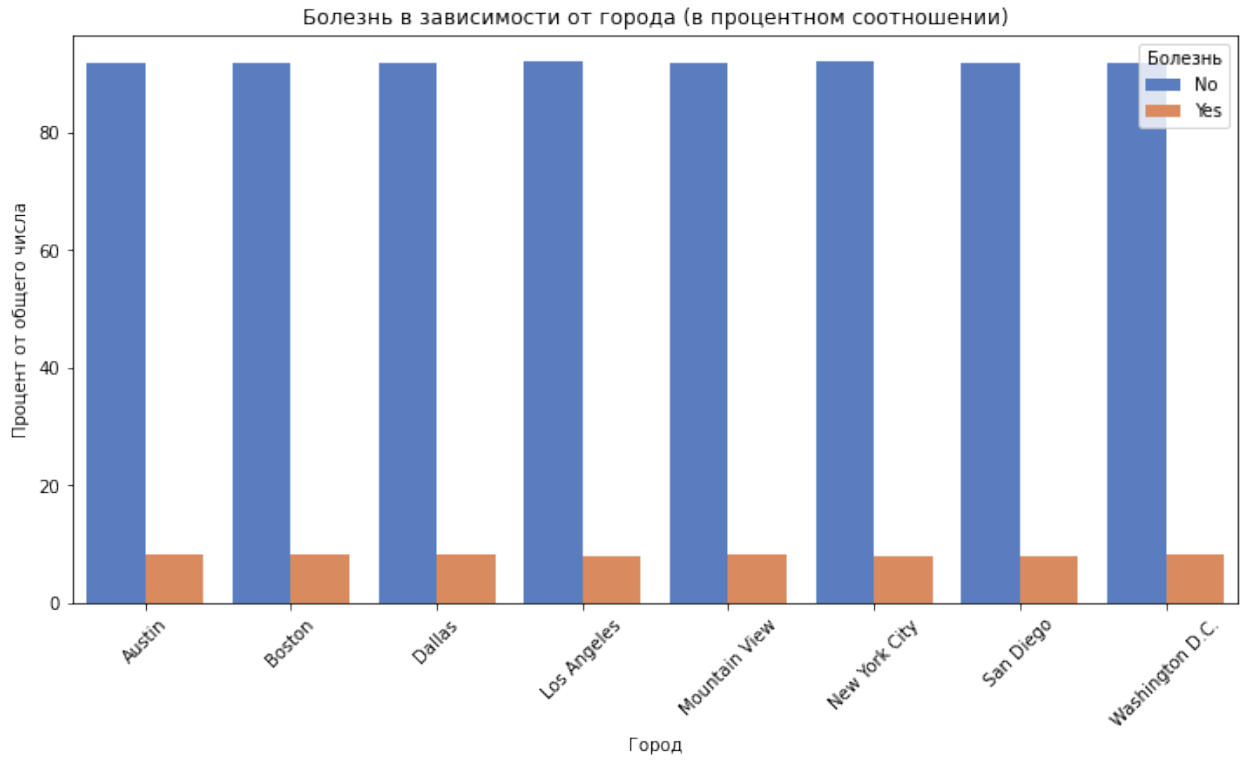
```
<Figure size 720x432 with 0 Axes>
```



Как нетрудно заметить, люди с отрицательным (или низким) доходом болеют намного чаще, чем среднестатистические люди, а люди с самыми высокими доходами не болеют вообще. Также стоит учитывать, что в абсолютном соотношении их сильно меньше остальных групп.

```
# Вычисление процентного соотношения для каждого уровня 'City' и
'Illness'
city_illness_percentage = data.groupby(['City',
'Illness']).size().groupby(level=0).apply(lambda x: 100 * x /
```

```
float(x.sum())).reset_index(name='Percentage')

# Построение столбчатой диаграммы с процентным соотношением
plt.figure(figsize=(12, 6))
sns.barplot(x='City', y='Percentage', hue='Illness',
data=city_illness_percentage, palette='muted')
plt.title('Болезнь в зависимости от города (в процентном
соотношении)')
plt.xlabel('Город')
plt.ylabel('Процент от общего числа')
plt.legend(title='Болезнь')
plt.xticks(rotation=45)
plt.show()
```



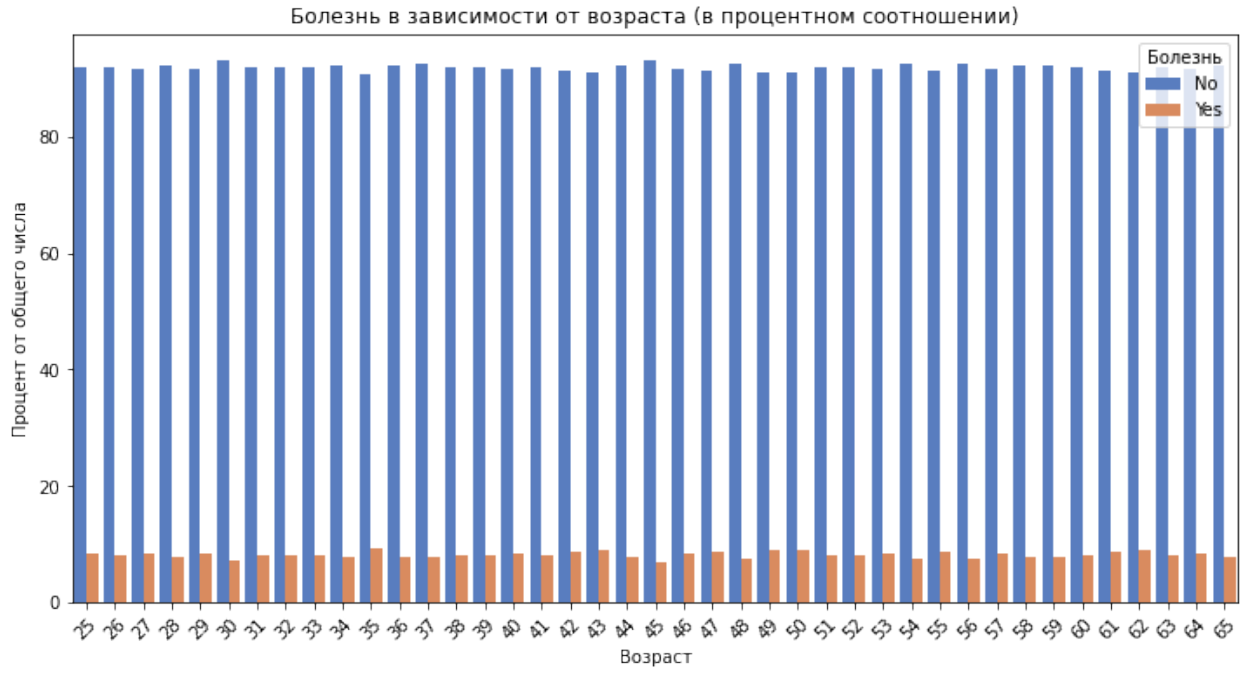

```
data.groupby(['City', 'Illness']).size()
```

```
City             Illness
Austin           No         11281
                 Yes         1011
Boston           No          7615
                 Yes          686
Dallas           No         18094
                 Yes         1613
Los Angeles      No         29605
                 Yes         2568
Mountain View    No         13041
```

```
                   Yes          1178
New York City      No          46286
                   Yes          4021
San Diego          No           4487
                   Yes           394
Washington D.C.    No           7452
                   Yes           668
dtype: int64
```

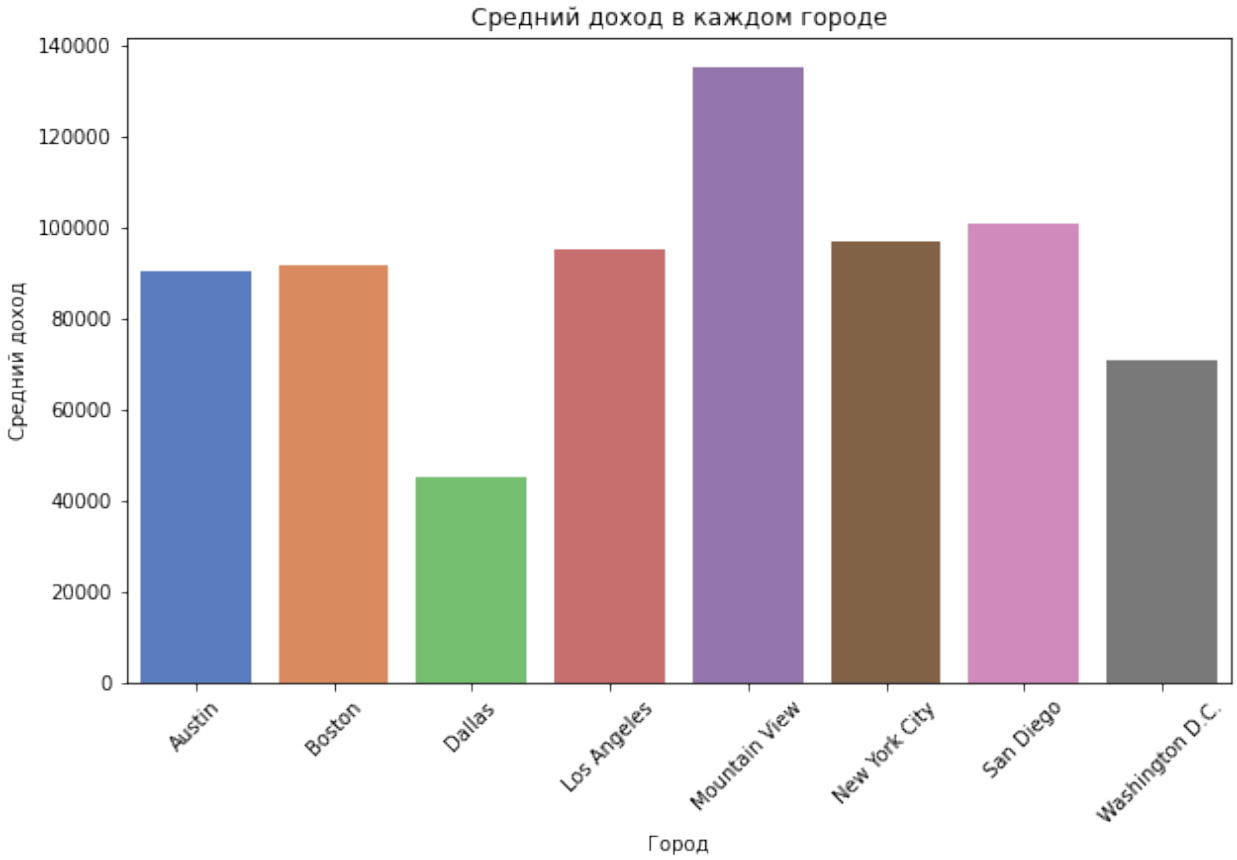
Как мы видим, от города не зависит процентное соотношение больных

```
# Вычисление процентного соотношения для каждого уровня 'Age' и
'Illness'
age_illness_percentage = data.groupby(['Age',
'Illness']).size().groupby(level=0).apply(lambda x: 100 * x /
float(x.sum())).reset_index(name='Percentage')

# Построение столбчатой диаграммы с процентным соотношением
plt.figure(figsize=(12, 6))
sns.barplot(x='Age', y='Percentage', hue='Illness',
data=age_illness_percentage, palette='muted')
plt.title('Болезнь в зависимости от возраста (в процентном
соотношении)')
plt.xlabel('Возраст')
plt.ylabel('Процент от общего числа')
plt.legend(title='Болезнь')
plt.xticks(rotation=45)
plt.show()
```



Как видно из графика, возраст тоже не влияет на болезненность

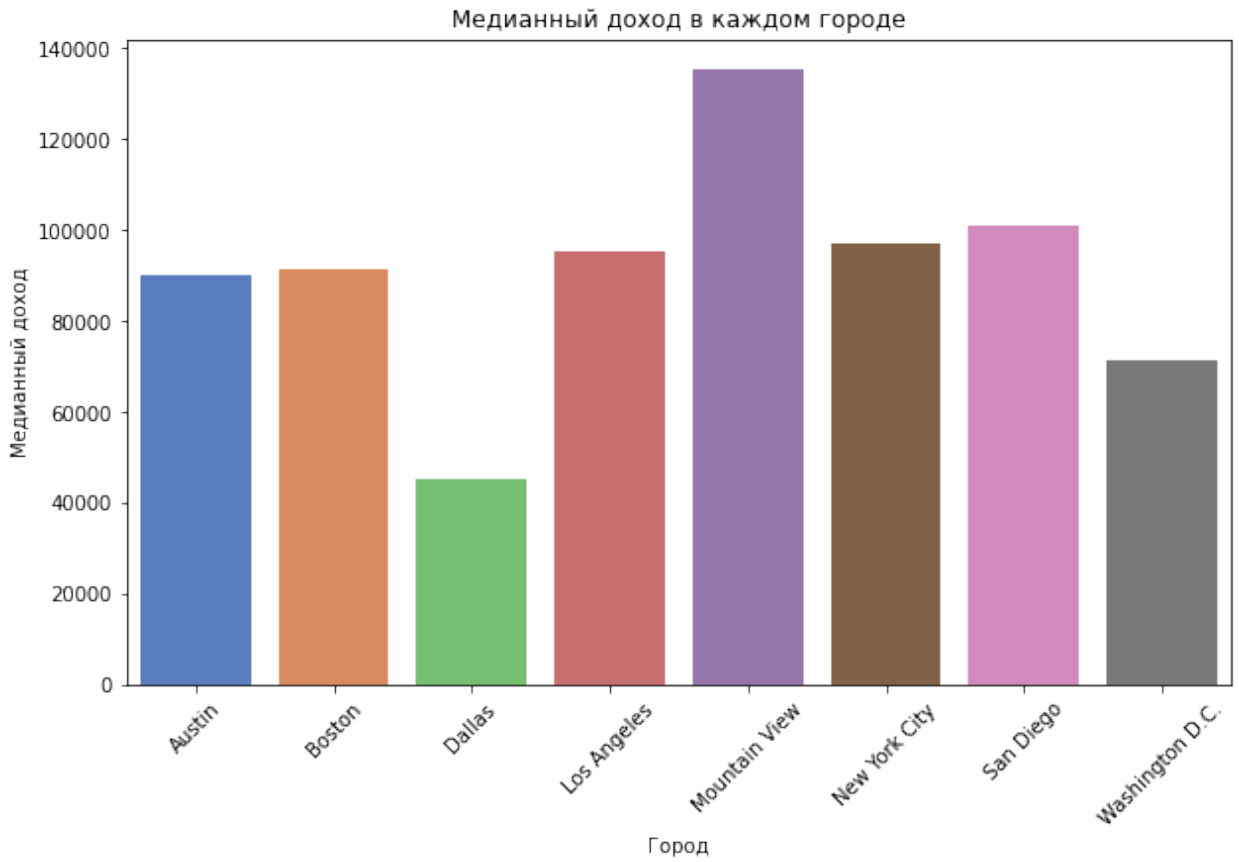
```
# Вычисление среднего дохода по каждому городу
city_income_mean = data.groupby('City')['Income'].mean().reset_index()

# Построение столбчатой диаграммы
plt.figure(figsize=(10, 6))
sns.barplot(x='City', y='Income', data=city_income_mean,
palette='muted')
plt.title('Средний доход в каждом городе')
plt.xlabel('Город')
plt.ylabel('Средний доход')
plt.xticks(rotation=45)
plt.show()
```



Как видно из графика, доходы в разных городах - разные. Самый прибыльный город - это Mountain View, а самый неприбыльный - это Dallas. Между ними расположился город Washington D.C.; Остальные города имеют примерно одинаковых средний доход.

```
# Вычисление медианного дохода по каждому городу
city_income_median = data.groupby('City')
['Income'].median().reset_index()
```

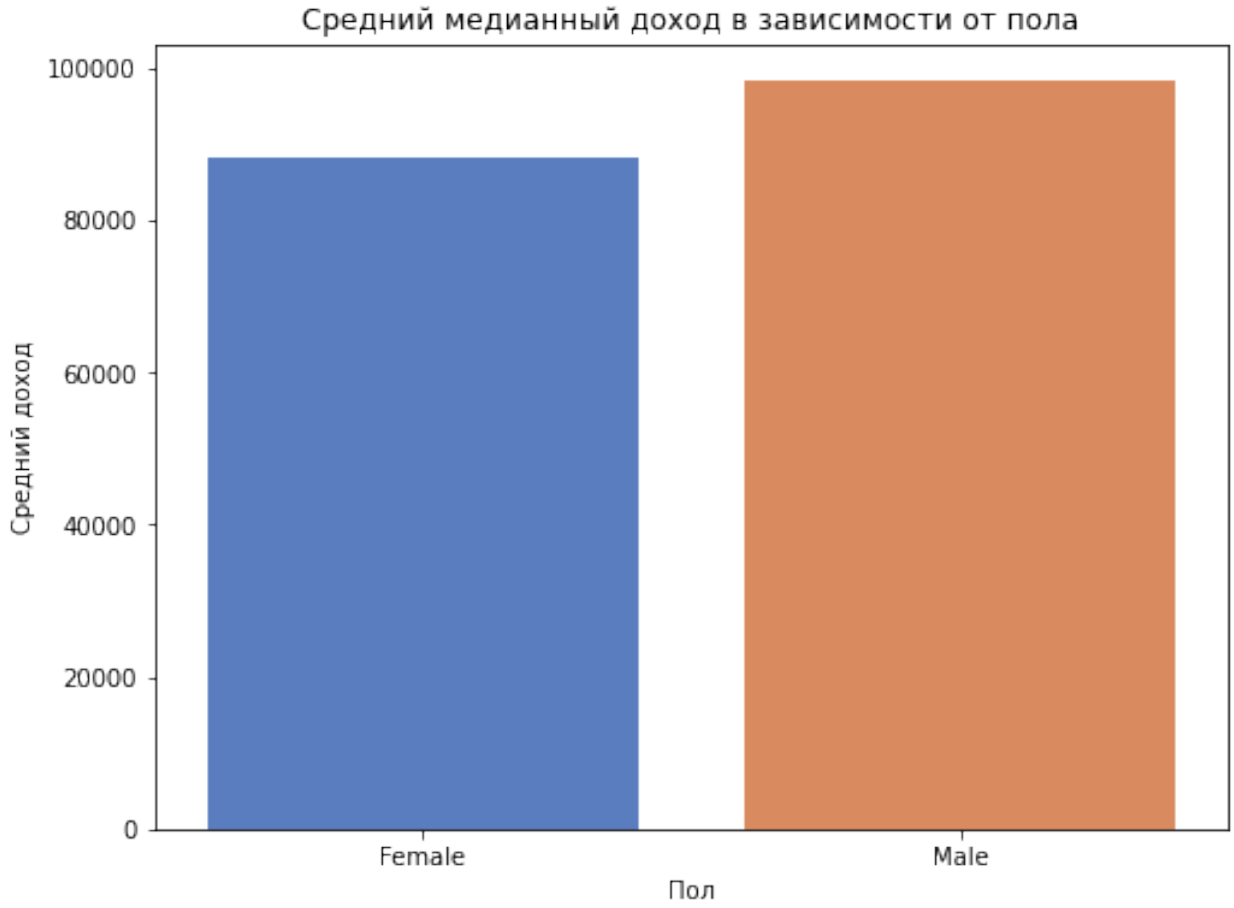
```
# Построение столбчатой диаграммы
plt.figure(figsize=(10, 6))
sns.barplot(x='City', y='Income', data=city_income_median,
palette='muted')
plt.title('Медианный доход в каждом городе')
plt.xlabel('Город')
plt.ylabel('Медианный доход')
plt.xticks(rotation=45)
plt.show()
```



Медианный доход оставил выводы те же.

```
# Вычисление среднего дохода по каждому полу
gender_income_mean = data.groupby('Gender')
['Income'].median().reset_index()

# Построение столбчатой диаграммы
plt.figure(figsize=(8, 6))
sns.barplot(x='Gender', y='Income', data=gender_income_mean,
palette='muted')
plt.title('Средний медианный доход в зависимости от пола')
plt.xlabel('Пол')
```

```
plt.ylabel('Средний доход')
plt.show()
```

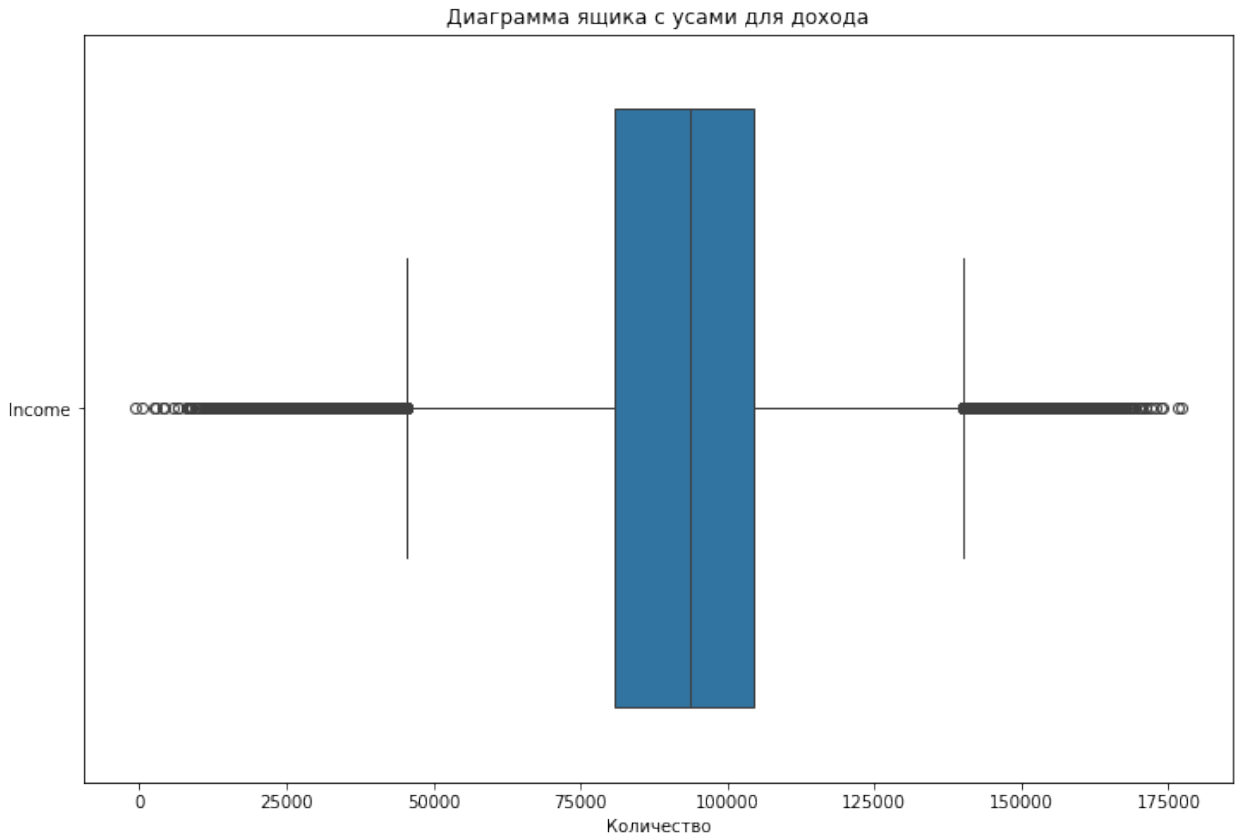


Мужчины немного больше зарабатывают в год, чем женщины

```
# Группировка данных по возрасту и подсчет количества заболеваний в
каждой возрастной группе
age_illness_counts = data.groupby('Age')['Illness'].apply(lambda x: (x
== 'Yes').sum())

# Построение графика зависимости количества заболеваний от возраста
plt.figure(figsize=(12, 6))
plt.plot(age_illness_counts.index, age_illness_counts.values,
marker='o', linestyle='-')
plt.title('Зависимость количества заболеваний от возраста')
plt.xlabel('Возраст')
plt.ylabel('Количество заболеваний')
plt.xticks(rotation=0)
plt.grid(True)
plt.show()
```

В каждой возрастной категории от 25 до 65 невключительно количество заболеваний лежит в пределах от 250 до 350 случаев

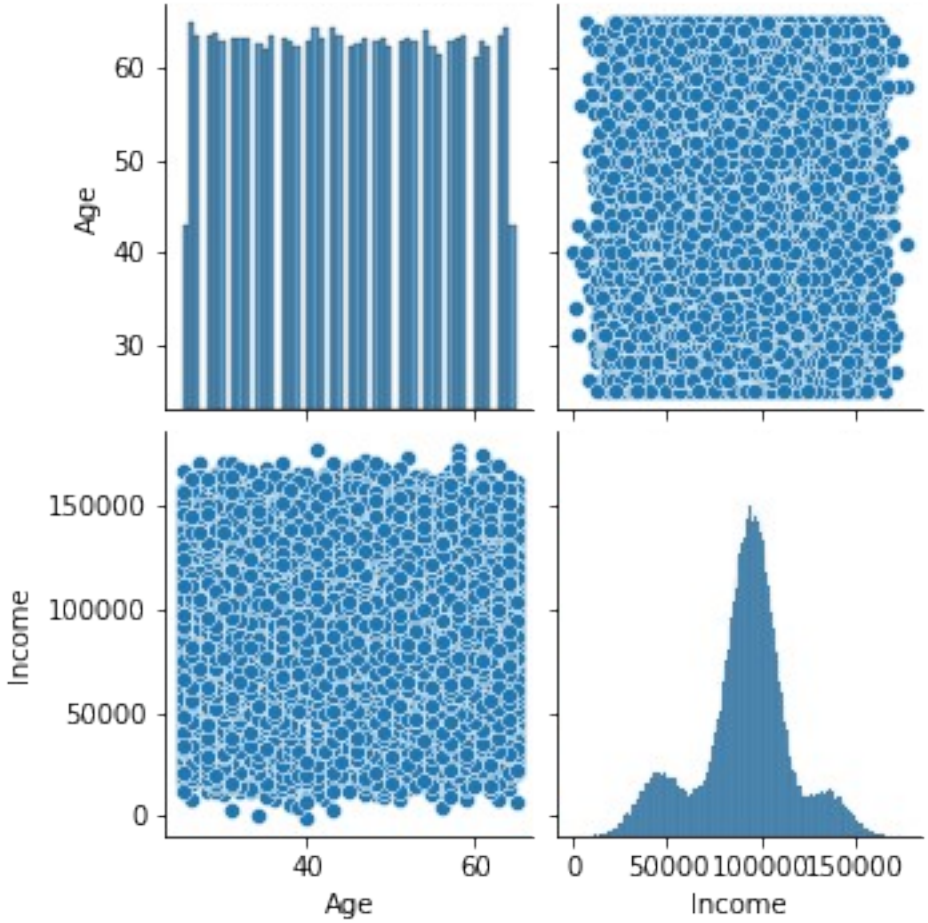
```
plt.figure(figsize=(12, 8))
sns.boxplot(data=data[['Income']], orient='h')
plt.title('Диаграмма ящика с усами для дохода')
plt.xlabel('Количество')
plt.show()
```



Большая часть дохода лежит в районе 90000-110000 в год

```
sns.pairplot(data[['Age', 'Income']])
plt.suptitle('Матрица диаграмм рассеяния для переменных возраста и дохода', y=1.02)
plt.show()
```

Матрица диаграмм рассеяния для переменных возраста и дохода

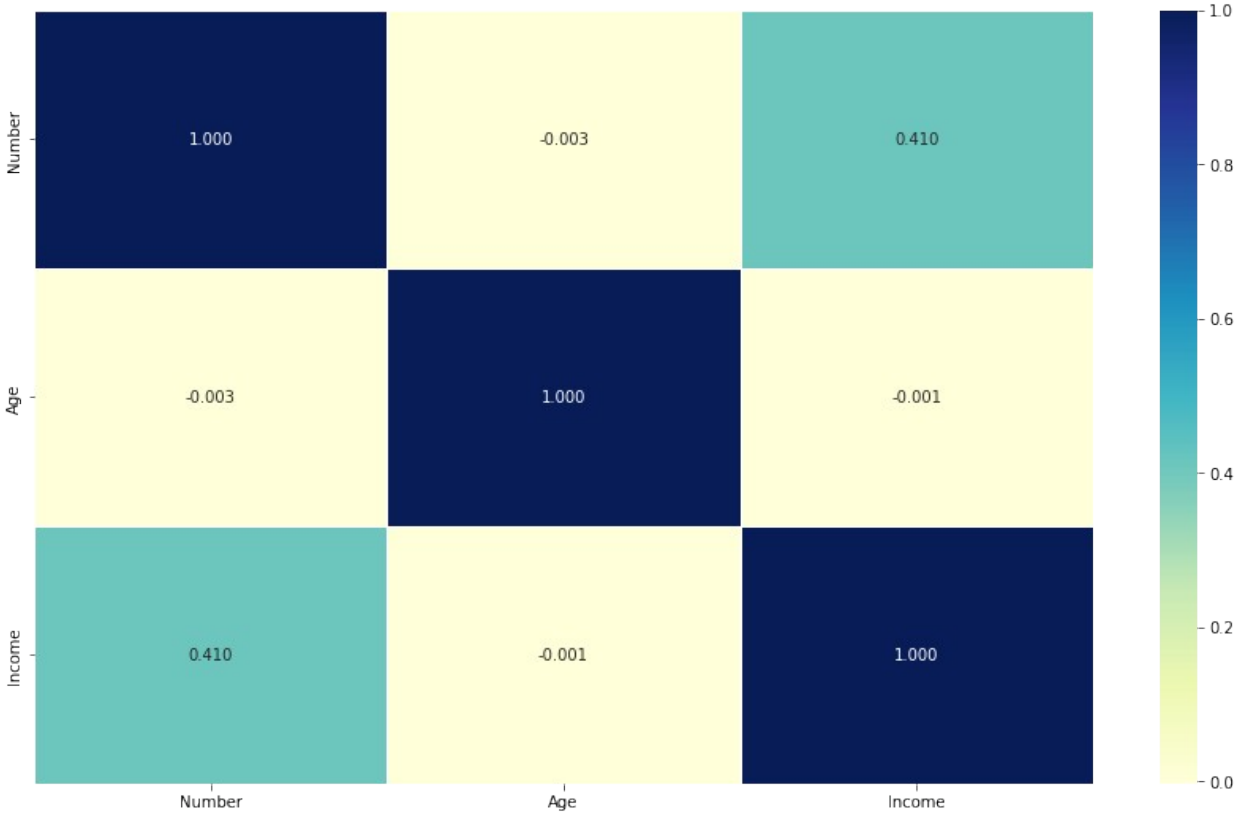



```
cor_matrix = data[['Age', 'Income']].corr()
cor_matrix
```

```
              Age    Income
Age      1.000000 -0.001318
Income -0.001318  1.000000
```

```
cor_matrix = data.corr()
plt.figure(figsize = (15,9))
sns.heatmap(cor_matrix,
            annot = True,
            linewidth = 1,
             fmt= ".3f",
             cmap="YlGnBu"
            )
```

```
<AxesSubplot: >
```



Как мы видим, возраст не коррелирует с доходом, однако доход удивительным образом коррелирует с номером опрошенного, из чего можно сделать вывод, что сначала опрашивали респондентов из бедных слоев населения, затем стали опрашивать уже в более благополучных районах.

```
plt.figure(figsize=(8, 6))
plt.scatter(data['Age'], data['Income'], alpha=0.5)
plt.title('Диаграмма рассеяния между возрастом и доходом')
plt.xlabel('Возраст')
plt.ylabel('Доход')
plt.grid(True)
plt.show()
```

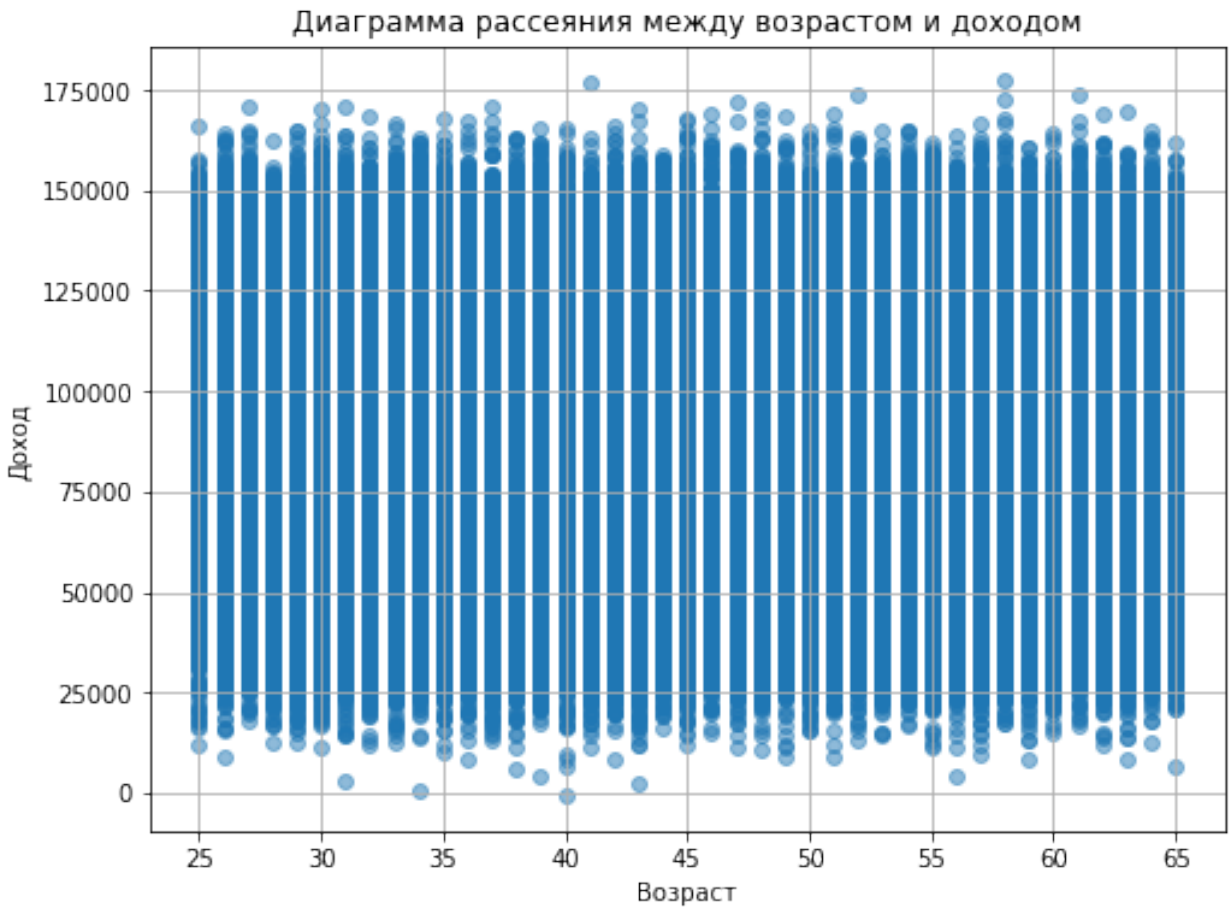


Диаграмма рассеяния (дополнительное требование)