

Исследование целевой аудитории вебтунов

```
In [1]: import pandas as pd
import numpy as np
pd.options.mode.chained_assignment = None

import matplotlib.pyplot as plt
from matplotlib.pyplot import MultipleLocator

from IPython.core.pylabtools import figsize

import seaborn as sns
sns.set_theme(style="darkgrid")
colors = sns.color_palette('pastel')

from collections import Counter
from pandas.core.common import flatten

from sklearn.preprocessing import LabelEncoder
import matplotlib.ticker as mtick
```

```
In [2]: #Загружаем данные
data = pd.read_csv('Answers.csv')
```

```
In [3]: # Смотрим типы данных столбцов и не пропущенные значения
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31 entries, 0 to 30
Data columns (total 24 columns):
 #   Column                                                                 Non-Null Count  Dtype
---  -
 0   Отметка времени                                                       31 non-null    object
 1   Ваш возраст                                                            31 non-null    int64
 2   Ваш пол                                                                31 non-null    object
 3   Нравится ли вам читать художественную литературу?                  31 non-null    object
 4   Книга или вебтун?                                                     23 non-null    object
 5   Ваши любимые жанры                                                    31 non-null    object
 6   А теперь по вашим любимым тегам                                       31 non-null    object
 7   Покупали ли вы главы?                                                31 non-null    object
 8   Сколько вы тратите денег на главы в месяц?                           9 non-null     object
 9   Сколько вы зарабатываете в месяц?                                     31 non-null    object
10   Населённый пункт и регион                                             31 non-null    object
11   Где читаете?                                                          31 non-null    object
12   На что вы обращаете больше внимания при выборе проекта?            31 non-null    object
13   Сколько часов в день вы читаете?                                     31 non-null    object
14   Укажите свою почту для результатов                                   21 non-null    object
15   Unnamed: 15                                                            18 non-null    object
16   Unnamed: 16                                                            16 non-null    object
17   Unnamed: 17                                                            11 non-null    object
18   Unnamed: 18                                                            27 non-null    object
19   Unnamed: 19                                                            23 non-null    object
20   Unnamed: 20                                                            9 non-null     object
21   Unnamed: 21                                                            27 non-null    object
22   Unnamed: 22                                                            11 non-null    object
23   Unnamed: 23                                                            12 non-null    object
dtypes: int64(1), object(23)
memory usage: 5.9+ KB
```

```
In [4]: #Убираем плохо обрабатываемые критерии и с большим количеством пропусков
data = data.drop([
    'Отметка времени', 'Сколько вы тратите денег на главы в месяц?',
    'Укажите свою почту для результатов', 'Населённый пункт и регион',
    'Unnamed: 15', 'Unnamed: 16', 'Unnamed: 17', 'Unnamed: 18',
    'Unnamed: 19', 'Unnamed: 20', 'Unnamed: 21', 'Unnamed: 22',
    'Unnamed: 23'], axis=1)
```

```
In [5]: #Заполняем оставшиеся пропуски
data = data.fillna('Нет ответа')

#Фиксим 15-летнего гения
data['Сколько вы зарабатываете в месяц?'][26] = 'Не работаю'
```

График разделения по полу

```
In [6]: data.head(10)
```

Out[6]:

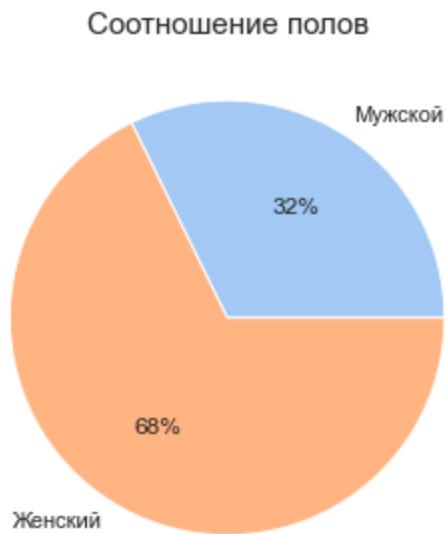
	Ваш возраст	Ваш пол	Нравится ли вам читать художественную литературу?	Книга или вебтун?	Ваши любимые жанры	А теперь лю
0	16	Мужской	Да	Нет ответа	Сёнэн;Боевик;Детектив;История;Приключения;Псих...	Насилие / и
1	18	Мужской	Нет	Нет ответа	Боевик;Драма;Детектив;Исекай;Приключения;Роман...	мужчина;М / жестоко
2	13	Женский	Да	Нет ответа	Гарем;Повседневность;Романтика;Психология;Ужас...	Г мужчина;Др
3	18	Женский	Да	Нет ответа	Сёнэн;Дзёсэй;Боевик;Детектив;Приключения;Повсе...	Насилие / ум
4	17	Женский	Да	Нет ответа	Дзёсэй;Исекай;Повседневность;Романтика;Психоло...	Насилие / ум
5	16	Женский	Да	Нет ответа	Сёнэн;Драма;Психология;Фэнтези	жесток элемен
6	14	Мужской	Да	Нет ответа	Повседневность;Романтика;Школа	Г ум
7	19	Женский	Да	Вебтун	Сёнэн;Детектив;История;Психология	
8	32	Женский	Нет	Нет ответа	Дзёсэй;Детектив;Исекай;Научная фантастика;Прик...	Навыки / ст умны
9	17	Женский	Да	Книга	Сёнэн;История;Научная фантастика;Приключения;П...	ГГ мужч способн

```
In [7]: #Считаем значения, выход - словарь
genders = dict(Counter([gender for gender in data['Ваш пол']]))

#Разбиваем словарь на ключи и значения, формируем круговую диаграмму
keys_genders, values_genders = zip(*genders.items())
```

```
sns.set(rc={'figure.figsize':(5,5)})
plt.pie(values_genders, labels=keys_genders, colors=colors, autopct='%.0f%%')

#Настраиваем названия
plt.title('Соотношение полов', {'fontsize': 15})
plt.show()
```



Разбивка по возрасту

```
In [8]: #Сортируем по категориям, меняя значения в data
for i in range(len(data)):
    if data['Ваш возраст'][i] < 16:
        data['Ваш возраст'][i] = 'Менее 16 лет'
    elif data['Ваш возраст'][i] < 18:
        data['Ваш возраст'][i] = '16 - 17'
    elif data['Ваш возраст'][i] < 25:
        data['Ваш возраст'][i] = '18 - 24'
    elif data['Ваш возраст'][i] < 35:
        data['Ваш возраст'][i] = '25 - 35'
    else:
        data['Ваш возраст'][i] = 'Более 35 лет'
```

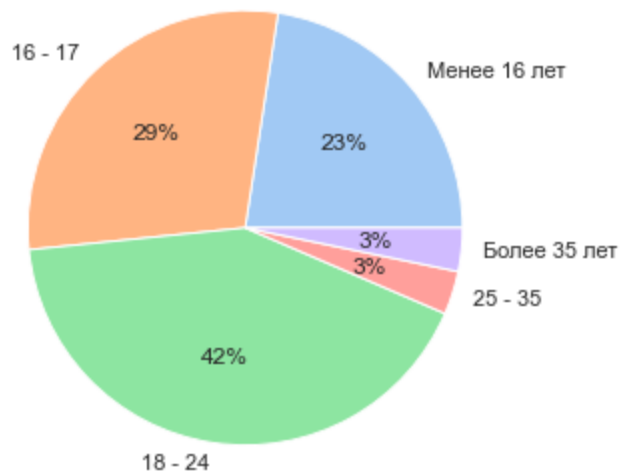
```
In [9]: #Считаем значения, выход - словарь
timely_ages = dict(Counter([age for age in data['Ваш возраст']]))
ages = {'Менее 16 лет': timely_ages.pop('Менее 16 лет')}
ages.update(timely_ages)

#Разбиваем словарь на ключи и значения, формируем круговую диаграмму
keys_ages, values_ages = zip(*ages.items())

sns.set(rc={'figure.figsize':(5,5)})
plt.pie(values_ages, labels=keys_ages, colors=colors, autopct='%.0f%%')

#Настраиваем названия
plt.title('Возраст аудитории', {'fontsize': 15})
plt.show()
```

Возраст аудитории



В основном читателям до 25 лет

Выбор формата

```
In [10]: #Считаем значения, выход - словарь
books = dict(Counter([book for book in data['Книга или вебтун?']]))

#Разбиваем словарь на ключи и значения, формируем круговую диаграмму
keys_books, values_books = zip(*books.items())

sns.set(rc={'figure.figsize': (5,5)})
plt.pie(values_books, labels=keys_books, colors=colors, autopct='%0f%%')

#Настраиваем названия
plt.title('Книга или вебтун?', {'fontsize': 15})
plt.show()
```



Из-за того, что опросник продолжал улучшаться во время действия (никогда не делайте так!), появилась большая категория "Нет ответа"

График вариативности выбора формата в зависимости от отношения к художественной литературе

```
In [11]: def procentage(graphic, axis='x', long=len(data)):
        '''Создаёт процентаж у оси'''
        if axis == 'x':
            graphic.xaxis.set_major_formatter(mtick.PercentFormatter(long))
            x_major_locator=MultipleLocator(long*0.1)
            graphic.xaxis.set_major_locator(x_major_locator)
            graphic.set(xlim=(0, long+long*0.05))
        elif axis == 'y':
            graphic.yaxis.set_major_formatter(mtick.PercentFormatter(long))
            y_major_locator=MultipleLocator(long*0.1)
            graphic.yaxis.set_major_locator(y_major_locator)
            graphic.set(ylim=(0, long+long*0.05))
```

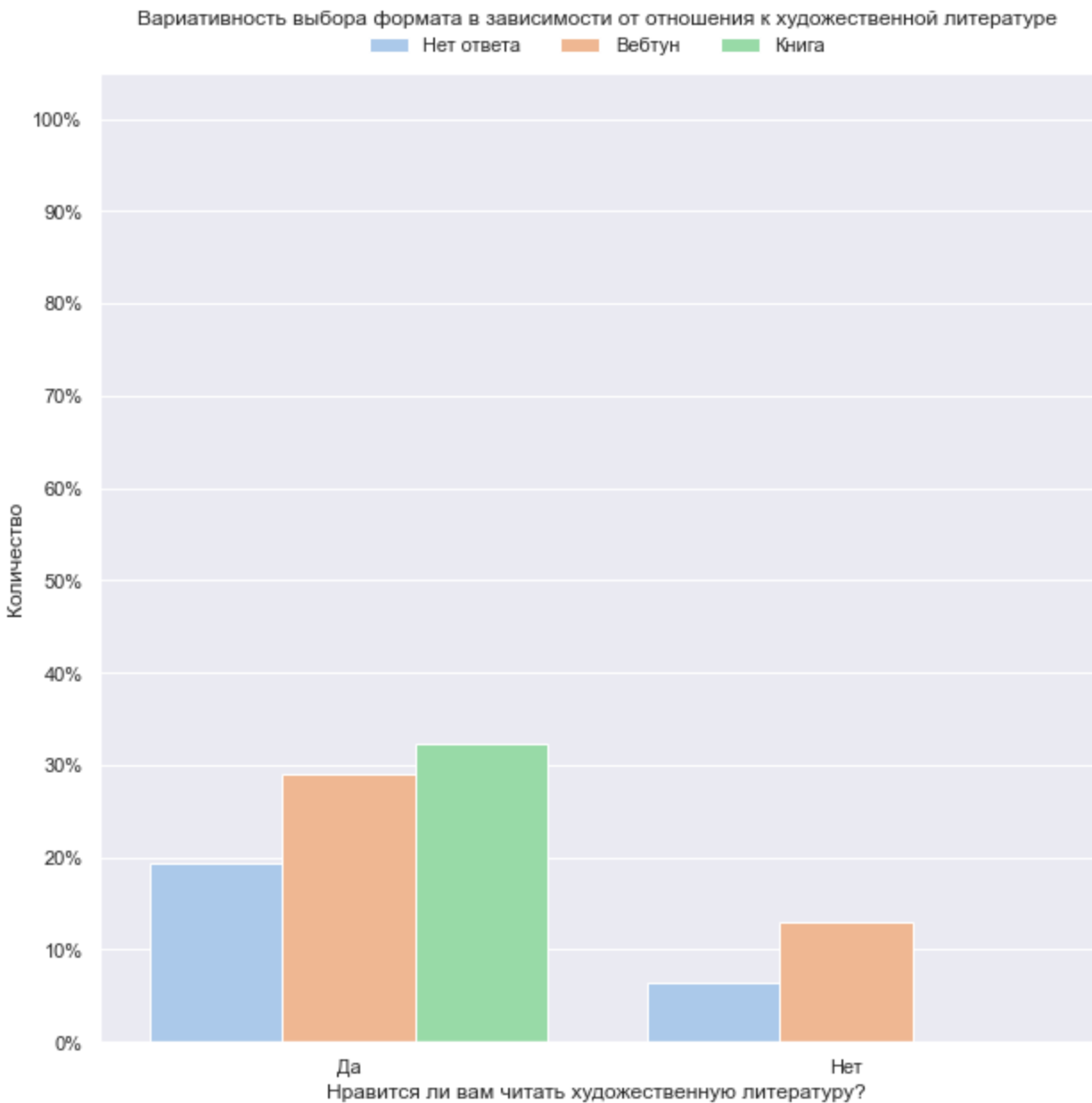
```
In [12]: sns.set(rc={'figure.figsize':(10,10)})
variability_format = sns.countplot(
    x='Нравится ли вам читать художественную литературу?', hue='Книга или вебтун?', data
)

#Настраииваем процентаж
procentage(variability_format, 'y')

#Перемещаем легенду с названием наверх
sns.move_legend(
    variability_format, "lower center",
    bbox_to_anchor=(.5, 1), ncol=3,
    title='Вариативность выбора формата в зависимости от отношения к художественной лите
)

#Настраииваем названия
plt.ylabel('Количество')
```

```
Out[12]: Text(0, 0.5, 'Количество')
```



Есть достаточно большой запрос на чтение художественной литературы в виде вебтунов, при этом можно потерять около 35% читателей, предпочитающих вебтун книгам, так как у них такого запроса нет.

График наиболее читаемых жанров

```
In [13]: def data_sort(old_data, buy=False):
'''
Сливает всё в один список, создаёт из него dataframe,
устанавливает порядок по возрастанию.
'''
new_data = []
if buy:
    #Создаем единый список для каждого ответа, если этот человек покупал главы
    count, buyers = 0, 0
    for category in old_data:
        if data['Покупали ли вы главы?'][count] == 'Да':
            answer = category.split(';')
            new_data.append(answer)
            buyers += 1
        count += 1
    else:
        #Создаем единый список для каждого ответа
```

```

        for category in old_data:
            answer = category.split(';')
            new_data.append(answer)

#Исправляем повторение в анкете, убирая ответ "ГГ умный" и заменяя на "Умный ГГ"
for member in new_data:
    if 'Умный ГГ' and 'ГГ умный' in member:
        member.remove('ГГ умный')
    elif 'ГГ умный' in member:
        member.remove('ГГ умный')
        member.append('Умный ГГ')

#Соединяем все ответы в один список, создаём dataframe
new_data = list(flatten(new_data))
new_data = pd.DataFrame({'Категории':new_data})

#Сортируем ответы по значениям
order = Counter([step for step in new_data['Категории']])
order = dict(sorted(order.items(), key=lambda item: item[1], reverse=True))

if buy:
    return new_data, order, buyers
return new_data, order

```

```

In [14]: #Создаём новый dataframe и порядок для него
data_genres, order_genres = data_sort(data['Ваши любимые жанры'])

color_for_many = sns.cubehelix_palette(31, start=2, rot=5, dark=0.4, light=0.8)
sns.set(rc={'figure.figsize':(10,10)})
genres_graphic = sns.countplot(y='Категории', data=data_genres, order=order_genres, pale

#Настраиваем процентаж
procentage(genres_graphic)

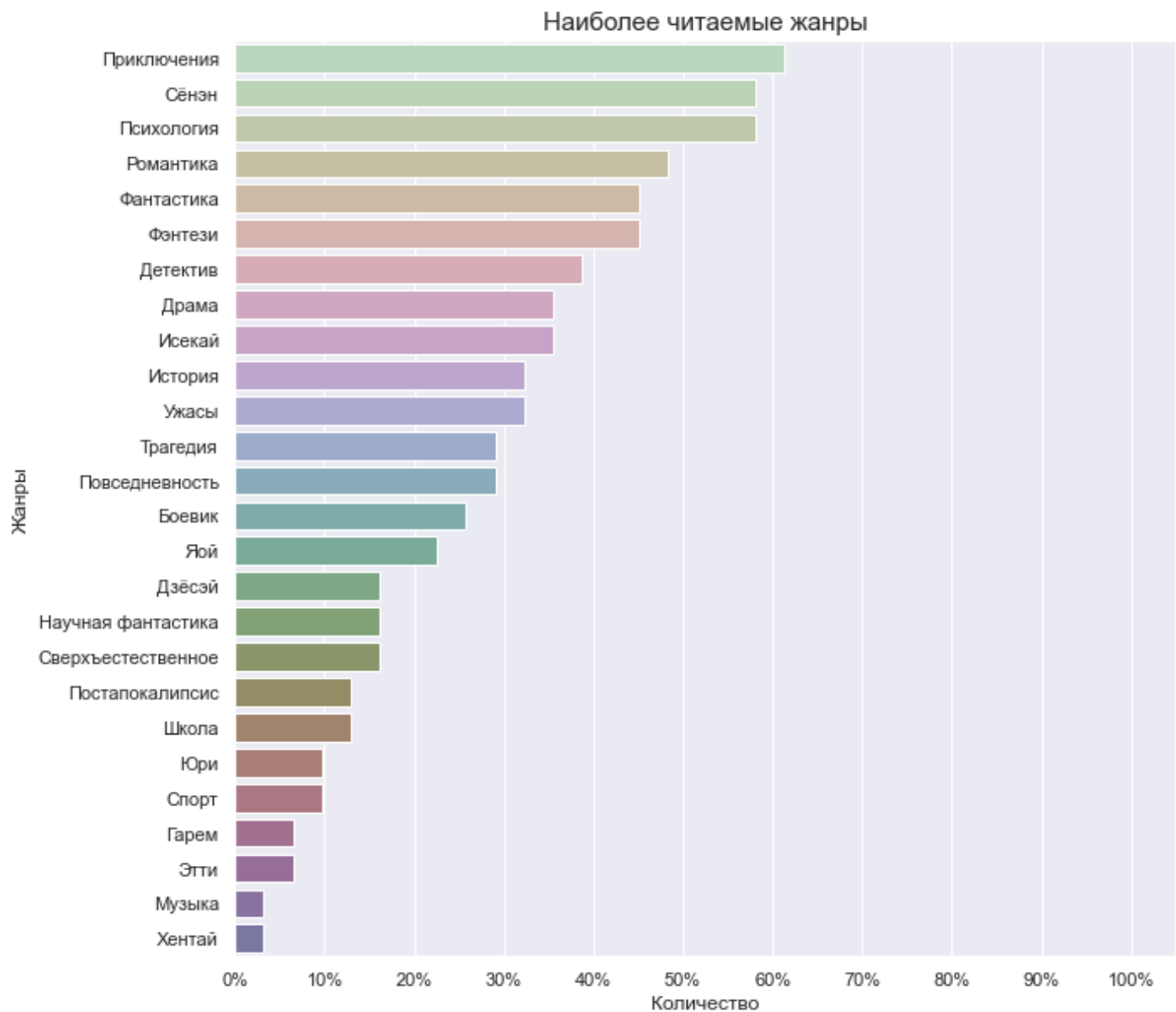
#Настраиваем названия
plt.ylabel("Жанры")
plt.xlabel('Количество')
plt.title('Наиболее читаемые жанры', {'fontsize': 15})

```

```

Out[14]: Text(0.5, 1.0, 'Наиболее читаемые жанры')

```



Несмотря на 68% женской аудитории, в тройке самых читаемых жанров присутствует сёнэн.

Самые популярные жанры: Приключения, Сёнэн, Психология, Романтика, Фантастика, Фэнтези.

Непопулярные жанры: Хентай, Музыка, Эти, Гарем (под сомнением), Спорт.

Наиболее читаемые жанры среди покупателей

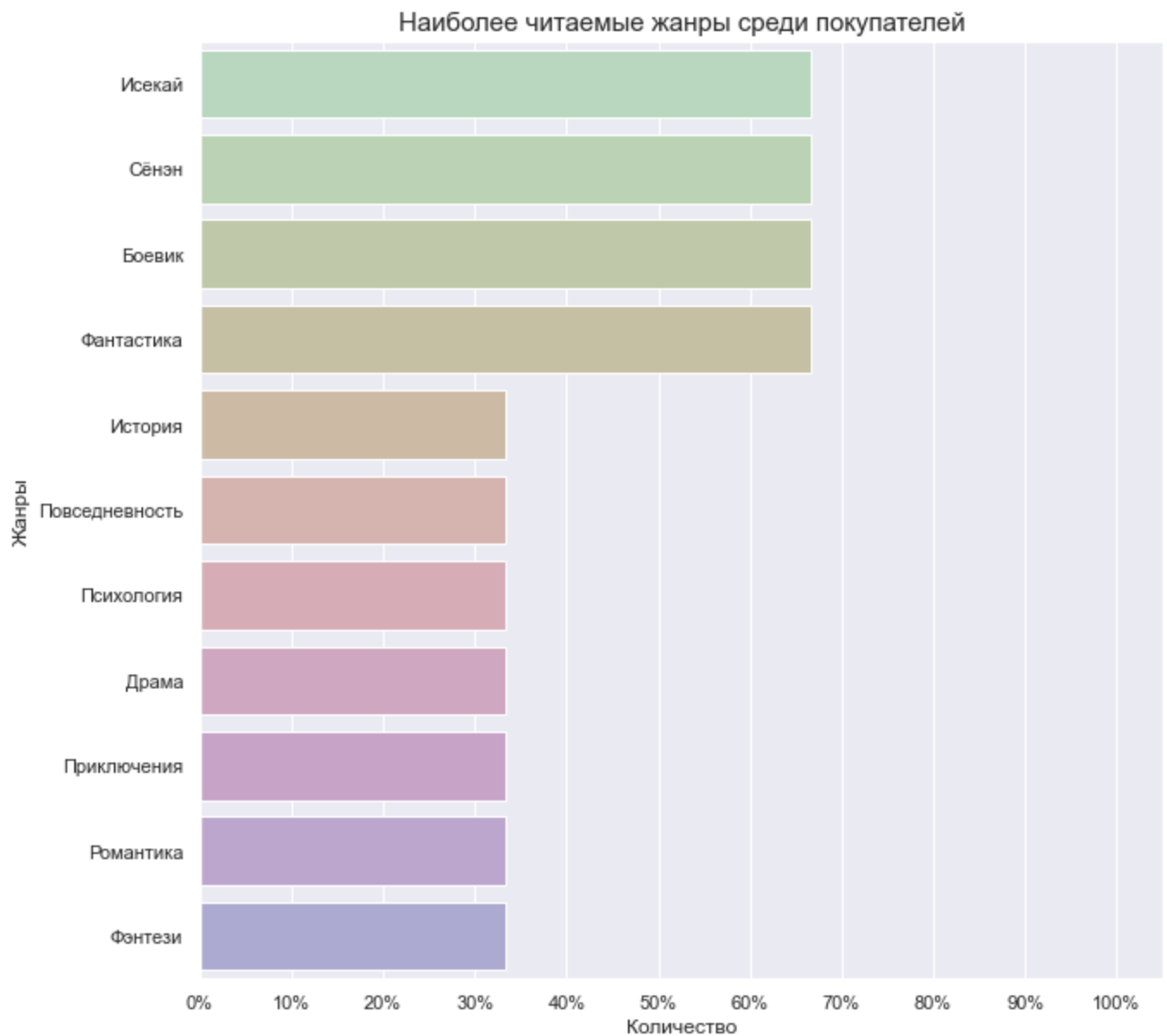
```
In [15]: #Создаём новый dataframe и порядок для него
data_genres_buyer, order_genres_buyer, buyers = data_sort(data['Ваши любимые жанры'], Tr

sns.set(rc={'figure.figsize':(10,10)})
genres_buyer_graphic = sns.countplot(y='Категории', data=data_genres_buyer, order=order_

#Настраиваем процентаж
procentage(genres_buyer_graphic, long=buyers)

#Настраиваем названия
plt.ylabel('Жанры')
plt.xlabel('Количество')
plt.title('Наиболее читаемые жанры среди покупателей', {'fontsize': 15})
```

```
Out[15]: Text(0.5, 1.0, 'Наиболее читаемые жанры среди покупателей')
```

Покупатели - ядро ЦА.

Ядро ЦА предпочитает жанры: Исекай, Сёнэн, Боевик, Фантастика.

График наиболее читаемых тегов

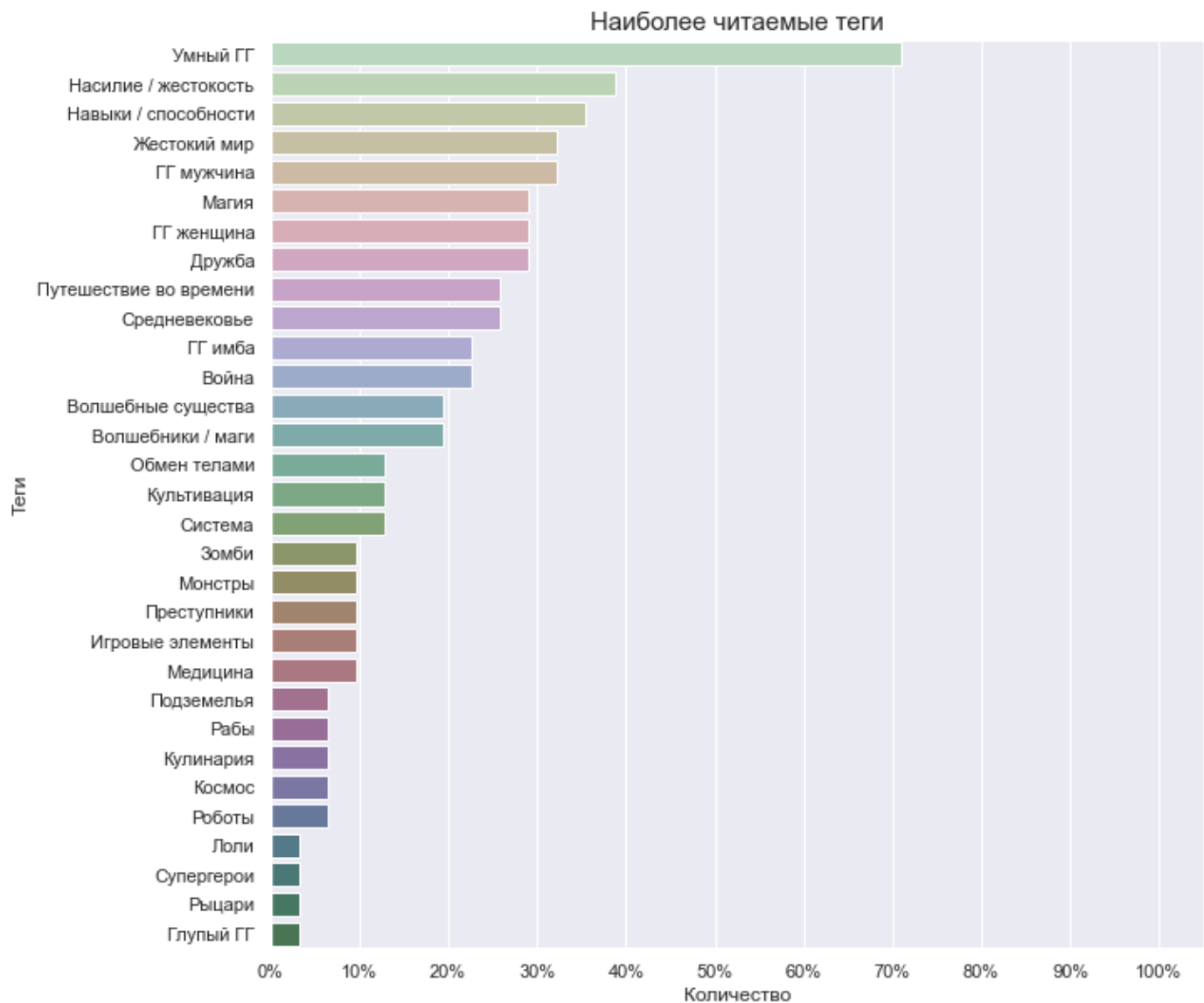
```
In [16]: #Создаём новый dataframe и порядок для него
data_tags, order_tags = data_sort(data['А теперь по вашим любимым тегам'])

sns.set(rc={'figure.figsize': (10,10)})
tags_graphic = sns.countplot(y='Категории', data=data_tags, order=order_tags, palette=co

#Настраиваем процент
percentage(tags_graphic)

#Настраиваем названия
plt.ylabel('Теги')
plt.xlabel('Количество')
plt.title('Наиболее читаемые теги', {'fontsize': 15})

Out[16]: Text(0.5, 1.0, 'Наиболее читаемые теги')
```



Теги "ГГ мужчина" и "ГГ женщина" стоят рядом.

Самые популярные теги: Умный ГГ, Насилие/жестокость, Навыки/способности, Жестокий мир.

Непопулярные теги: Глупый ГГ, Рыцари, Супергерои, Лоли.

Наиболее читаемые теги среди покупателей

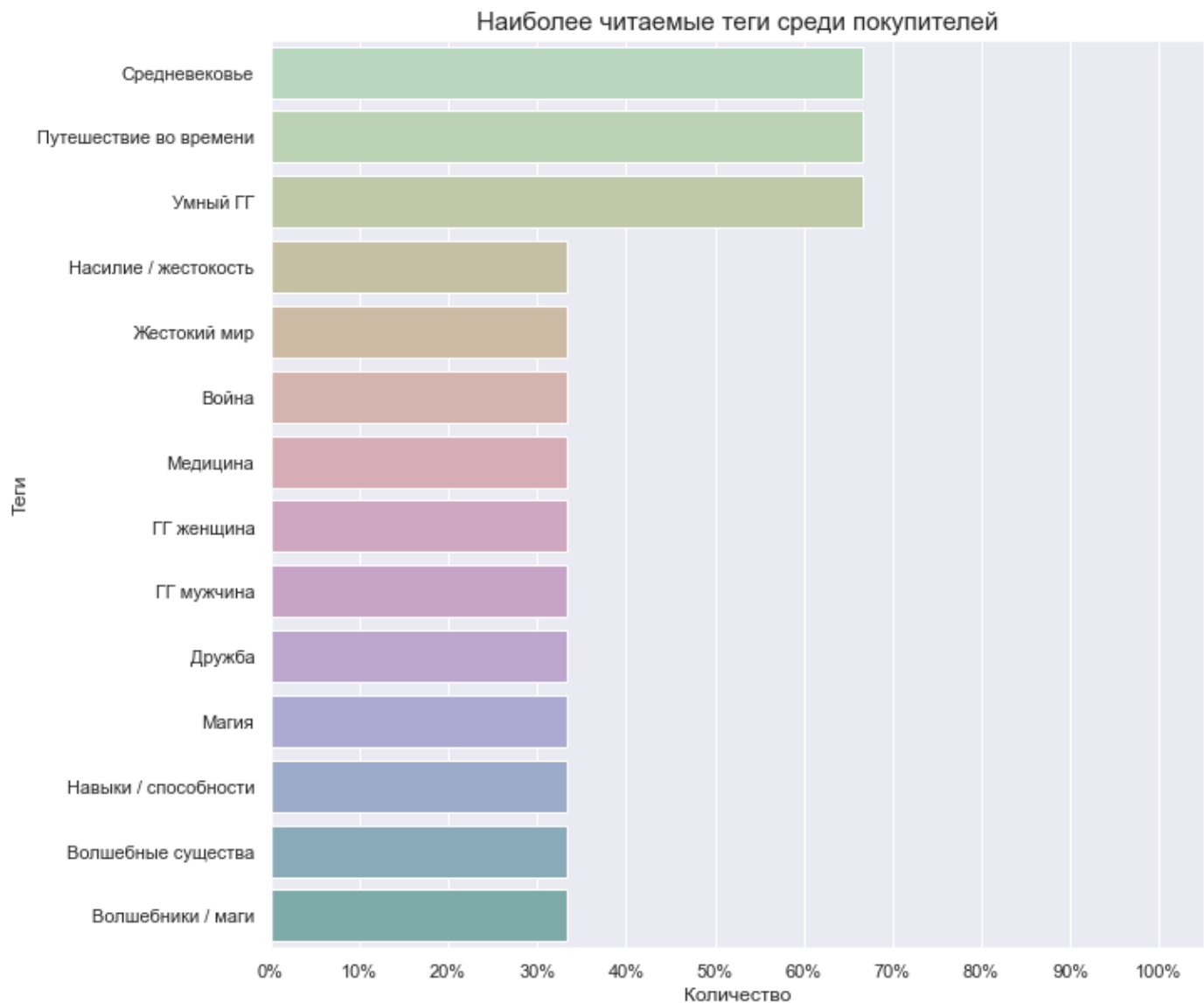
```
In [17]: #Создаём новый dataframe и порядок для него
data_tags_buyer, order_tags_buyer, buyers = data_sort(data['А теперь по вашим любимым т

sns.set(rc={'figure.figsize': (10,10)})
tags_buyer_graphic = sns.countplot(y='Категории', data=data_tags_buyer, order=order_tags

#Настраиваем процент
procentage(tags_buyer_graphic, long=buyers)

#Настраиваем названия
plt.ylabel('Теги')
plt.xlabel('Количество')
plt.title('Наиболее читаемые теги среди покупателей', {'fontsize': 15})
```

```
Out[17]: Text(0.5, 1.0, 'Наиболее читаемые теги среди покупателей')
```



Ядро ЦА предпочитает теги: Средневековые, Путешествие во времени, Умный ГГ.

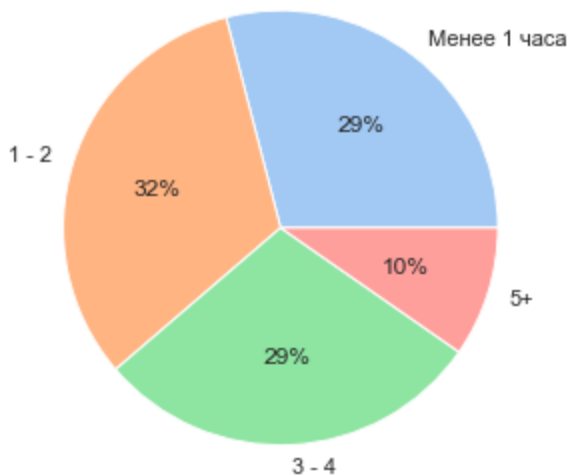
Графики продолжительности чтения

```
In [18]: #Считаем значения, выход - словарь
timely_read_times = dict(Counter([read_time for read_time in data['Сколько часов в день
read_times = {'Менее 1 часа': timely_read_times.pop('Менее 1 часа'), '1 - 2': timely_rea
read_times.update(timely_read_times)

#Разбиваем словарь на ключи и значения, формируем круговую диаграмму
keys_read_times, values_read_times = zip(*read_times.items())
sns.set(rc={'figure.figsize': (5,5)})
plt.pie(values_read_times, labels=keys_read_times, colors=colors, autopct='%.0f%%')

#Настраиваем названия
plt.title('Продолжительность чтения', {'fontsize': 15})
plt.show()
```

Продолжительность чтения



Подавляющая часть опрошенных читают более часа.

График зависимости продолжительности чтения от отношения к художественной литературе

```
In [19]: #Порядок
order_read_time = ['Менее 1 часа', '1 - 2', '3 - 4', '5+']

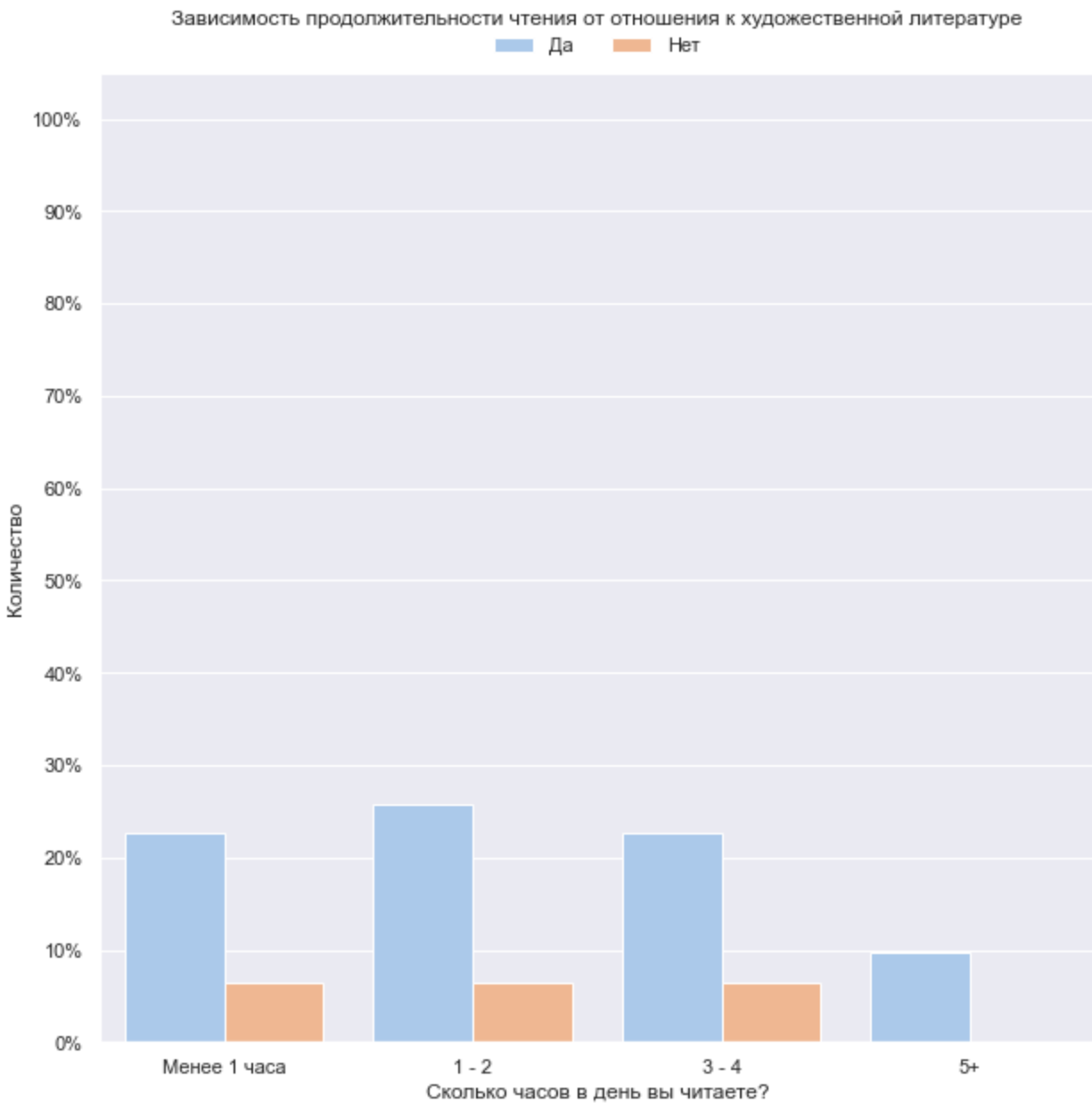
sns.set(rc={'figure.figsize':(10,10)})
read_time_by_lit_graphic = sns.countplot(
    x='Сколько часов в день вы читаете?',
    data=data, hue='Нравится ли вам читать художественную литературу?',
    order=order_read_time, palette='pastel'
)

#Перемещаем легенду с названием наверх
sns.move_legend(
    read_time_by_lit_graphic, "lower center",
    bbox_to_anchor=(.5, 1), ncol=3,
    title='Зависимость продолжительности чтения от отношения к художественной литературе'
)

#Настраиваем процентаж
procentage(read_time_by_lit_graphic, 'y')

#Настраиваем названия
plt.ylabel('Количество')
```

```
Out[19]: Text(0, 0.5, 'Количество')
```



Отношение к художественной литературе мало влияет на продолжительность чтения. Из всех опрошенных только любители художественной литературы читают более 5-ти часов.

График платежеспособности аудитории

In [20]:

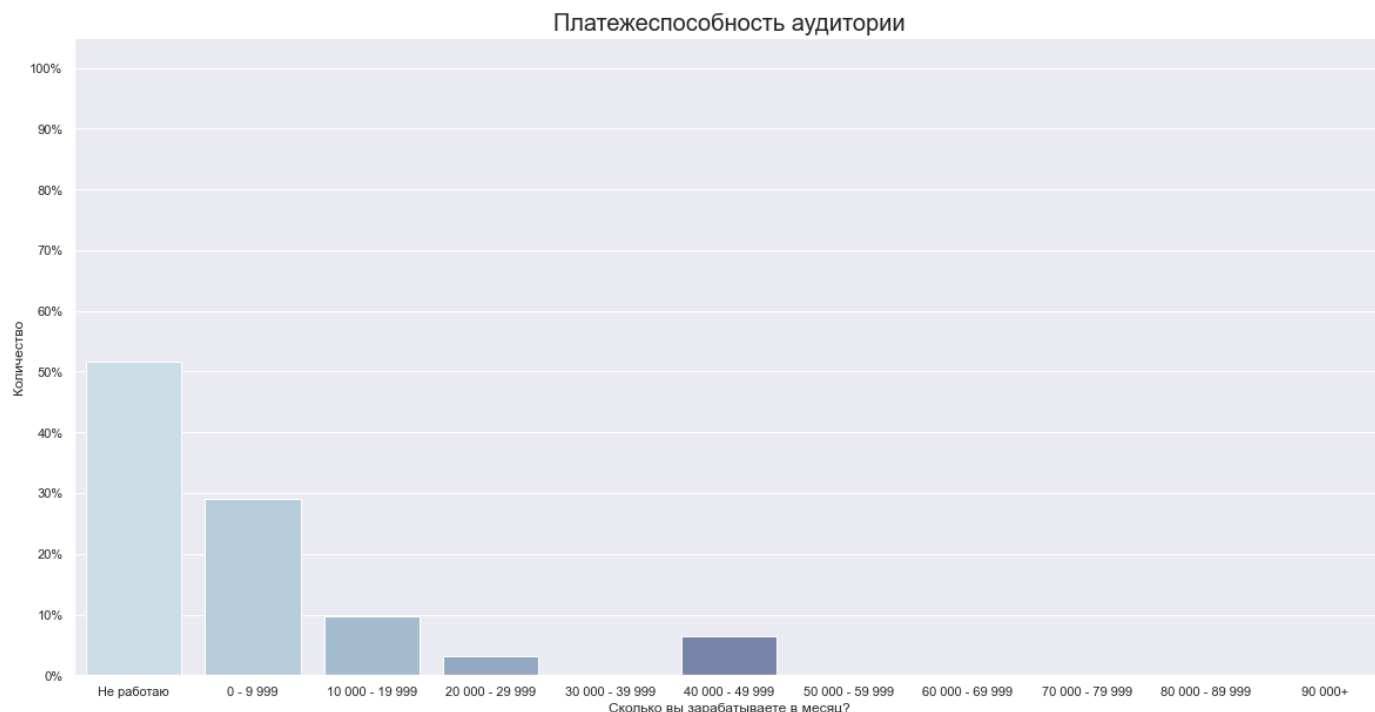
```
#Порядок
order_solvenсy = [
    'Не работаю', '0 - 9 999', '10 000 - 19 999', '20 000 - 29 999',
    '30 000 - 39 999', '40 000 - 49 999', '50 000 - 59 999',
    '60 000 - 69 999', '70 000 - 79 999', '80 000 - 89 999', '90 000+'
]

sns.set(rc={'figure.figsize': (20,10)})
solvenсy_graphic = sns.countplot(
    x='Сколько вы зарабатываете в месяц?', data=data, order=order_solvenсy, palette="ch:
)

#Настраиваем процентаж
procentage(solvenсy_graphic, 'y')

#Настраиваем названия
plt.ylabel('Количество')
plt.title('Платежеспособность аудитории', {'fontsize': 20})
```

Out[20]: Text(0.5, 1.0, 'Платежеспособность аудитории')



Подовляющая часть опрошенных не работает или получает до 10 тысяч (скорее всего подработка).

График читалок

```
In [21]: #Создаём новый dataframe и порядок для него
data_reader, order_reader = data_sort(data['Где читаете?'])

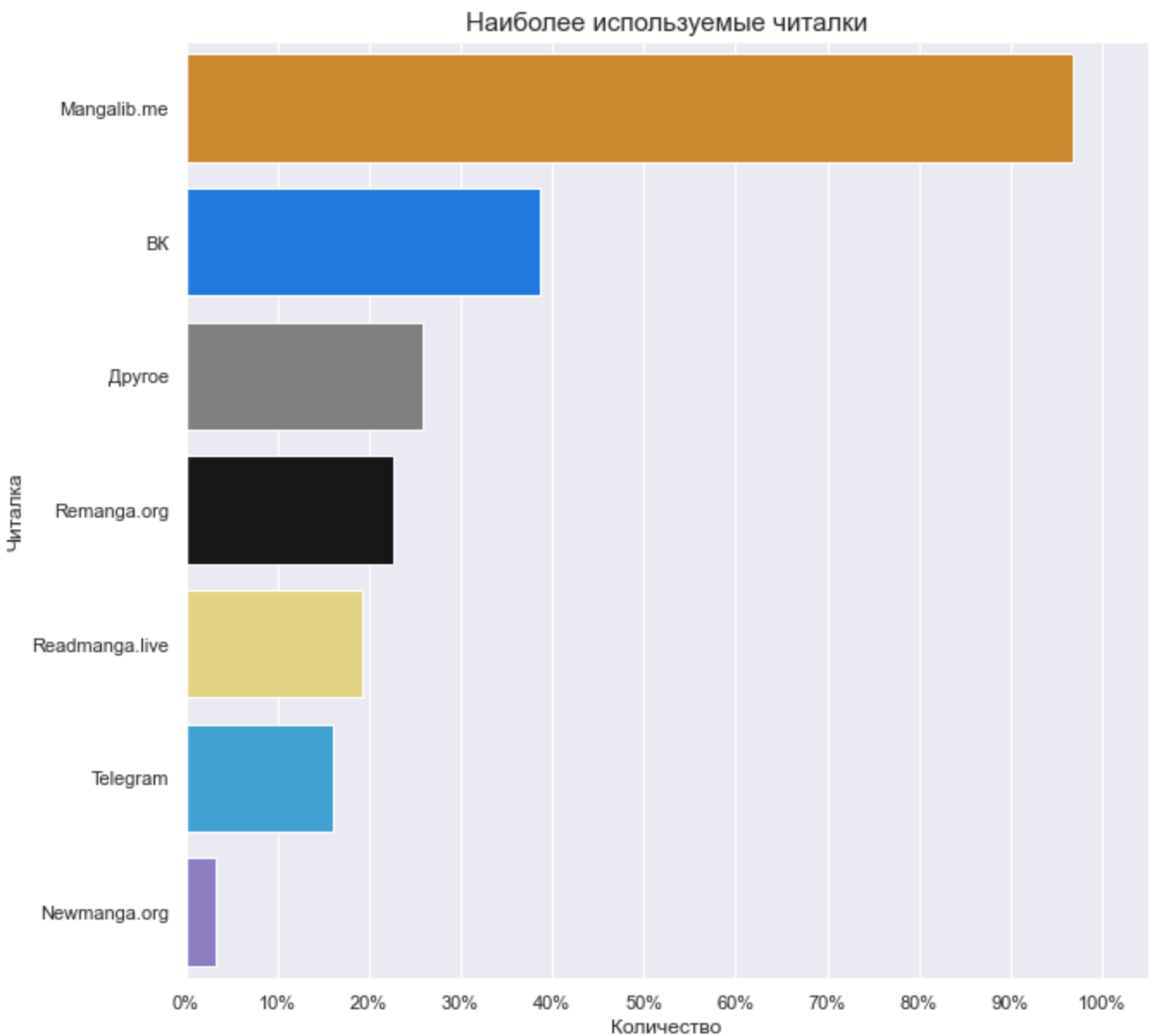
colors_for_reader = {
    'Mangalib.me': '#e48f13', 'BK': '#0077fe', 'Remanga.org': '#171717',
    'Newmanga.org': '#8573cb', 'Telegram': '#28a8ea', 'Другое': '#808080',
    'Readmanga.live': '#f2dd75'
}

sns.set(rc={'figure.figsize': (10,10)})
reader_graphic = sns.countplot(y='Категории', data=data_reader, order=order_reader, palette=colors_for_reader)

#Настраииваем процентаж
procentage(reader_graphic)

#Настраииваем названия
plt.ylabel('Читалка')
plt.xlabel('Количество')
plt.title('Наиболее используемые читалки', {'fontsize': 15})
```

Out[21]: Text(0.5, 1.0, 'Наиболее используемые читалки')



Наиболее популярная читалка - MangaLib.me

Это коррелирует с местом проведения опроса (большинство опрошенных с форума). Интересно, что читатели MangaLib.me предпочитают сайт VK читалке Remanga.org

На что читатели обращают внимание при выборе тайтла

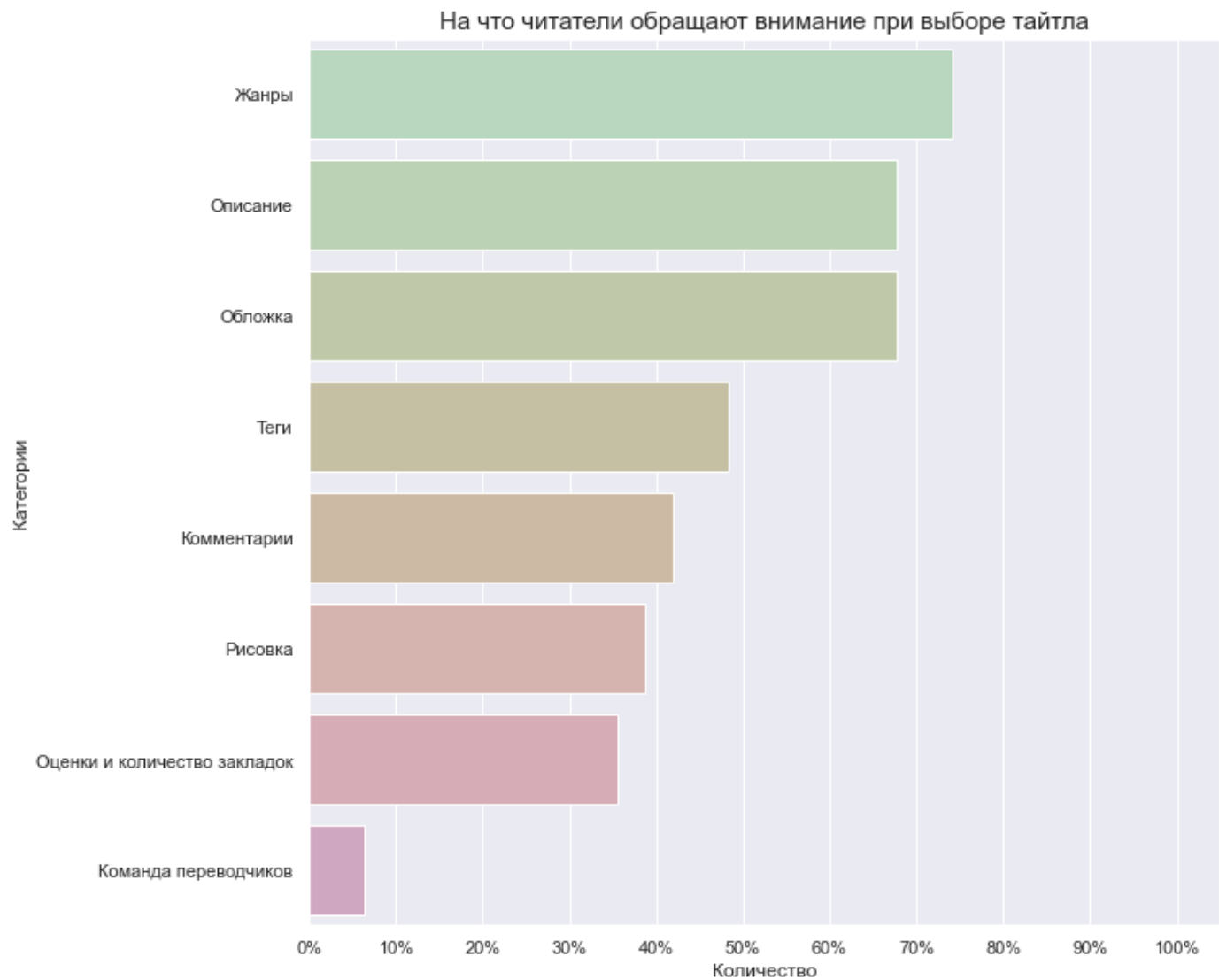
```
In [22]: #Создаём новый dataframe и порядок для него
data_attention, order_attention = data_sort(data['На что вы обращаете больше внимания при выборе тайтла'])

sns.set(rc={'figure.figsize': (10,10)})
attention_graphic = sns.countplot(y='Категории', data=data_attention, order=order_attention)

#Настраиваем процентаж
procentage(attention_graphic)

#Настраиваем названия
plt.xlabel('Количество')
plt.title('На что читатели обращают внимание при выборе тайтла', {'fontsize': 15})
```

Out[22]: Text(0.5, 1.0, 'На что читатели обращают внимание при выборе тайтла')



Особенно нужно уделить внимание жанрам, описанию (лонгриду) и работать над обложкой.

Спасибо за участие в опросе

В скором времени я сделаю работу над ошибками и запущу новое исследование. Основная проблема заключалась в малой выборке (помогайте исследованиям распространяться)!