

```
In [1]: import pandas as pd # Библиотека Pandas для работы с табличными данными
from google.colab import files # чтобы загружать файлы в облако через проводник

import numpy as np # библиотека Numpy для операций линейной алгебры и прочего
import matplotlib.pyplot as plt # библиотека Matplotlib для визуализации
import seaborn as sns; sns.set() # библиотека Seaborn для визуализации данных из
import json
```

```
In [2]: pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
```

```
In [3]: uploaded = files.upload() #просто запускаем и выбираем файл в проводнике
```

No file chosen

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving CleanSubmissionsWithForms.json to CleanSubmissionsWithForms.json

```
In [4]: df = pd.read_json('/content/CleanSubmissionsWithForms.json')
```

Dataset cleanup

```
In [5]: df.head(15)
```

Out[5]:

	id	SubmissionLikes	Category	Description	Participation	Money	A
0	7	67	Общественные пространства, природа и экология	Благоустроить Силинский лес: освещение, дорожк...	Нет	Пока не знаю	4
1	8	61	Общественные пространства, природа и экология	Благоустроить Силинский лес	Нет	Пока не знаю	4
2	9	109	Дороги и транспорт	Здание автовокзала требует колоссальной реконс...	Нет	Бюджетные	2
3	10	44	Общественные пространства, природа и экология	Ремонтируют парк очень плохо, страшно коляской...	Нет	Бюджетные	1
4	11	185	Здоровье, образование, спорт и культура	Реконструкция ДК Судостроитель	Нет	Бюджетные	2
5	12	116	Общественные пространства, природа и экология	привести в современное состояние парки города...	Да	Пока не знаю	4
6	13	123	Дороги и транспорт	Возрождение муниципального транспорта- автобусы...	Нет	Бюджетные	4
7	14	152	Дороги и транспорт	Восстановить нормальную работу Комсомольского ...	Нет	Бюджетные	4
8	15	101	Здоровье, образование, спорт и культура	Предлагаю провести реконструкцию одной из глав...	Нет	Бюджетные	2
9	16	137	Дороги и транспорт	Учитывая, что у нас в городе с очисткой дорог...	Да	Бюджетные	4
10	17	68	Общественные пространства, природа и экология	Набережная зимой не очищается от снега.	Нет	Бюджетные	4
11	18	75	Общественные пространства,	Привести в порядок заброшенный	Да	Бюджетные	2

id		SubmissionLikes	Category	Description	Participation	Money	A
			природа и экология	парк. Восстанов...			
12	19	47	Общественные пространства, природа и экология	Привести в порядок заброшенный парк	Да	Бюджетные	4
13	20	80	Дороги и транспорт	Восстановить трамвайную линию и пустить маршруты	Нет	Пока не знаю	4
14	21	83	Здоровье, образование, спорт и культура	Реконструкция дома культуры "Строитель"	Да	Пока не знаю	2

Total Likes

```
In [6]: df['SubmissionLikes'].sum()
```

Out[6]: 4514

```
In [7]: df['Age'].value_counts()
```

Out[7]: 25-40 76
40-65 56
До 25 11
Более 65 2
Name: Age, dtype: int64

```
In [8]: def get_mean_age(age_str):  
        if age_str == 'До 25':  
            return 24  
        if age_str == 'Более 65':  
            return 66  
        if age_str == '25-40':  
            return 32.5  
        if age_str == '40-65':  
            return 52.5
```

```
In [9]: df['MeanAge'] = df['Age'].apply(get_mean_age)
```

```
In [10]: df['MeanAge'].value_counts()
```

Out[10]: 32.5 76
52.5 56
24.0 11
66.0 2
Name: MeanAge, dtype: int64

Mean age of survey - 40.04

```
In [11]: df['MeanAge'].mean()
```

```
Out[11]: 40.04137931034483
```

```
In [12]: def convert_age_to_int(age_str):  
    if age_str == 'До 25':  
        return (None, 24)  
    if age_str == 'Более 65':  
        return (66, None)  
    bounds = age_str.split('-')  
    return (int(bounds[0]), int(bounds[1]))
```

```
In [13]: df[['LowerAge', 'UpperAge']] = df['Age'].apply(convert_age_to_int).apply(pd.Series)
```

```
In [14]: df['LiveTime'].value_counts()
```

```
Out[14]: С рождения          108  
Более 5 лет              21  
Жил, но уехал           11  
Менее 5 лет              3  
Я здесь временно         2  
Name: LiveTime, dtype: int64
```

```
In [15]: def map_livetime(value, default):  
    if value == 'С рождения':  
        return default  
    if value == 'Более 5 лет':  
        return 10  
    if value == 'Жил, но уехал':  
        return 1  
    if value == 'Менее 5 лет':  
        return 5  
    if value == 'Я здесь временно':  
        return 1
```

```
In [16]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145 entries, 0 to 144
Data columns (total 18 columns):
#   Column                Non-Null Count  Dtype
---  -
0   id                     145 non-null    int64
1   SubmissionLikes        145 non-null    int64
2   Category               145 non-null    object
3   Description             145 non-null    object
4   Participation           145 non-null    object
5   Money                  145 non-null    object
6   Age                    145 non-null    object
7   ContactName            145 non-null    object
8   LiveTime               145 non-null    object
9   FeatureType            145 non-null    object
10  GeometryType           145 non-null    object
11  CoordinatesLongitude    145 non-null    float64
12  CoordinatesLatitude     145 non-null    float64
13  CreatedAt              145 non-null    object
14  Gender                  143 non-null    object
15  MeanAge                 145 non-null    float64
16  LowerAge                134 non-null    float64
17  UpperAge                143 non-null    float64
dtypes: float64(5), int64(2), object(11)
memory usage: 20.5+ KB
```

```
In [17]: # df['LiveTimeInt'] = df['LiveTime'].apply(map_livetime)
df['LiveTimeInt'] = df.apply(lambda row: map_livetime(row['LiveTime'], row['Mean
```

```
In [18]: df['LiveTimeInt'].value_counts()
```

```
Out[18]: 32.5    62
         52.5    36
         10.0    21
          1.0    13
         24.0     9
          5.0     3
         66.0     1
         Name: LiveTimeInt, dtype: int64
```

```
In [19]: def convert_participation_to_bool(value):
         if value == "Да":
             return True
         return False
```

```
In [20]: df['Participation'] = df['Participation'].apply(convert_participation_to_bool)
```

```
In [21]: df.head()
```

Out[21]:

	id	Submission	Likes	Category	Description	Participation	Money	Age
0	7		67	Общественные пространства, природа и экология	Благоустроить Силинский лес: освещение, дорожк...	False	Пока не знаю	40-65
1	8		61	Общественные пространства, природа и экология	Благоустроить Силинский лес	False	Пока не знаю	40-65
2	9		109	Дороги и транспорт	Здание автовокзала требует колоссальной реконс...	False	Бюджетные	25-40
3	10		44	Общественные пространства, природа и экология	Ремонтируют парк очень плохо, страшно коляской...	False	Бюджетные	До 25
4	11		185	Здоровье, образование, спорт и культура	Реконструкция ДК Судостроитель	False	Бюджетные	25-40

check if data was not lost

In [22]: df.shape

Out[22]: (145, 19)

In [23]: !pip install transformers

```

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting transformers
  Downloading transformers-4.28.1-py3-none-any.whl (7.0 MB)
    _____ 7.0/7.0 MB 56.4 MB/s eta 0:00:00
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.9/dist-packages (from transformers) (2022.10.31)
Collecting tokenizers!=0.11.3,<0.14,>=0.11.1
  Downloading tokenizers-0.13.3-cp39-cp39-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (7.8 MB)
    _____ 7.8/7.8 MB 102.8 MB/s eta 0:00:00
Requirement already satisfied: filelock in /usr/local/lib/python3.9/dist-packages (from transformers) (3.11.0)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.9/dist-packages (from transformers) (23.1)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.9/dist-packages (from transformers) (6.0)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.9/dist-packages (from transformers) (4.65.0)
Collecting huggingface-hub<1.0,>=0.11.0
  Downloading huggingface_hub-0.13.4-py3-none-any.whl (200 kB)
    _____ 200.1/200.1 kB 23.8 MB/s eta 0:00:00
Requirement already satisfied: requests in /usr/local/lib/python3.9/dist-packages (from transformers) (2.27.1)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.9/dist-packages (from transformers) (1.22.4)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.9/dist-packages (from huggingface-hub<1.0,>=0.11.0->transformers) (4.5.0)
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.9/dist-packages (from requests->transformers) (2.0.12)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.9/dist-packages (from requests->transformers) (2022.12.7)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.9/dist-packages (from requests->transformers) (1.26.15)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.9/dist-packages (from requests->transformers) (3.4)
Installing collected packages: tokenizers, huggingface-hub, transformers
Successfully installed huggingface-hub-0.13.4 tokenizers-0.13.3 transformers-4.28.1

```

```

In [24]: import nltk
        from transformers import pipeline

        # Download the necessary NLTK packages
        nltk.download('punkt')
        nltk.download('stopwords')

        # Load the pre-trained sentiment analysis model
        MODEL = pipeline('sentiment-analysis', model='nlptown/bert-base-multilingual-uncased')

        # Analyze a Russian comment
        # comment = "Замечательный фильм! Очень трогательный и увлекательный!"
        # result = model(comment)[0]
        # print(f"Comment: {comment}\nSentiment: {result['label']} ({result['score']})")

```

```

[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Unzipping corpora/stopwords.zip.

```

```

Downloading (...)lve/main/config.json: 0% | 0.00/953 [00:00<?, ?B/s]

```

```

Downloading pytorch_model.bin: 0%|          | 0.00/669M [00:00<?, ?B/s]
Downloading (...)_tokenizer_config.json: 0%|          | 0.00/39.0 [00:00<?, ?B/s]
Downloading (...)_vocab.txt: 0%|          | 0.00/872k [00:00<?, ?B/s]
Downloading (...)_vocab.json: 0%|          | 0.00/112 [00:00<?, ?B/s]

```

```

In [25]: def calculate_happy(input):
        try:
            result = MODEL(input)[0]
        except Exception as e:
            # just put average if something goes wrong
            return 3
        return int(result['label'].split(' ')[0])

```

```

In [26]: df['Happy'] = df['Description'].apply(calculate_happy)

```

Token indices sequence length is longer than the specified maximum sequence length for this model (634 > 512). Running this sequence through the model will result in indexing errors

```

In [27]: import seaborn as sns

```

How happy are the people in counts

```

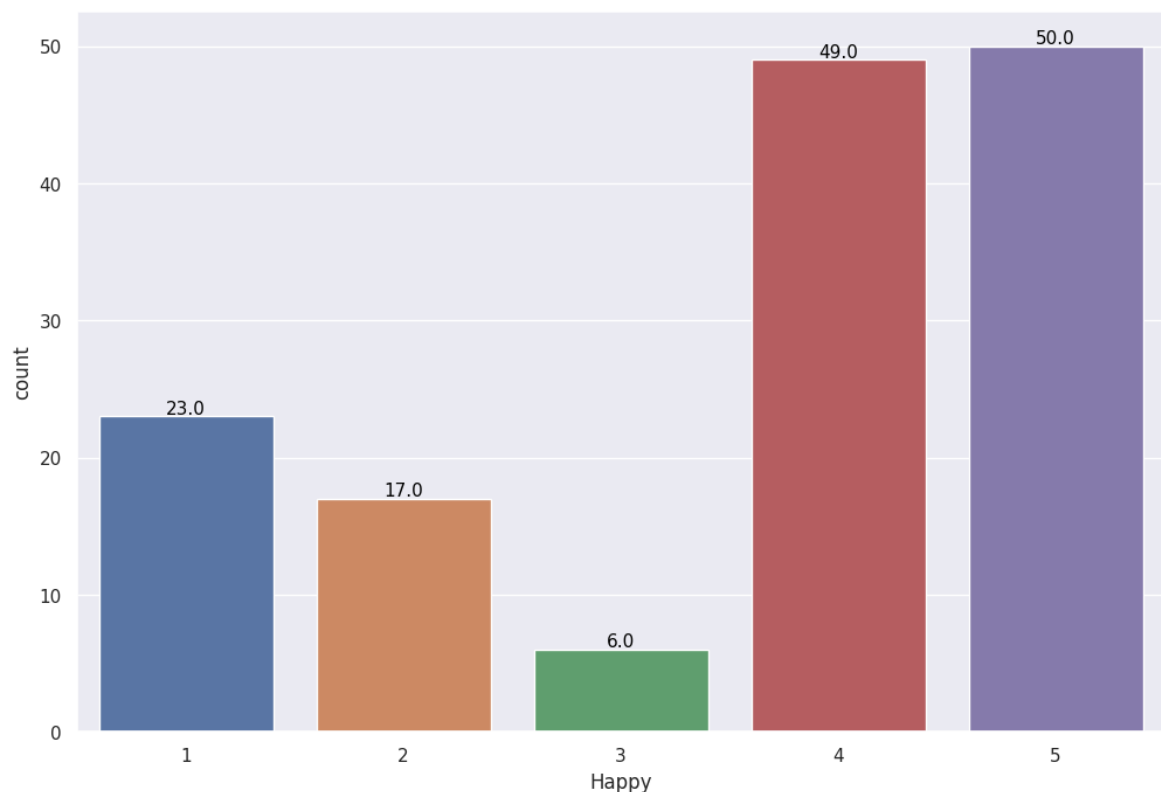
In [28]: plt.figure(figsize=(12, 8))

ax = sns.countplot(x=df['Happy'])

for p in ax.patches:
    ax.annotate(str(p.get_height()), (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', fontsize=11, color='black', xytext=(0,

plt.show()

```



How happy are the people in percentages

```
In [29]: percentile_happy_str = df.Happy.value_counts(normalize=True).mul(100).round(1).a
percentile_happy_int = df.Happy.value_counts(normalize=True).mul(100).round(1).a
display(percentile_happy_str)
```

5 34.5%

4 33.8%

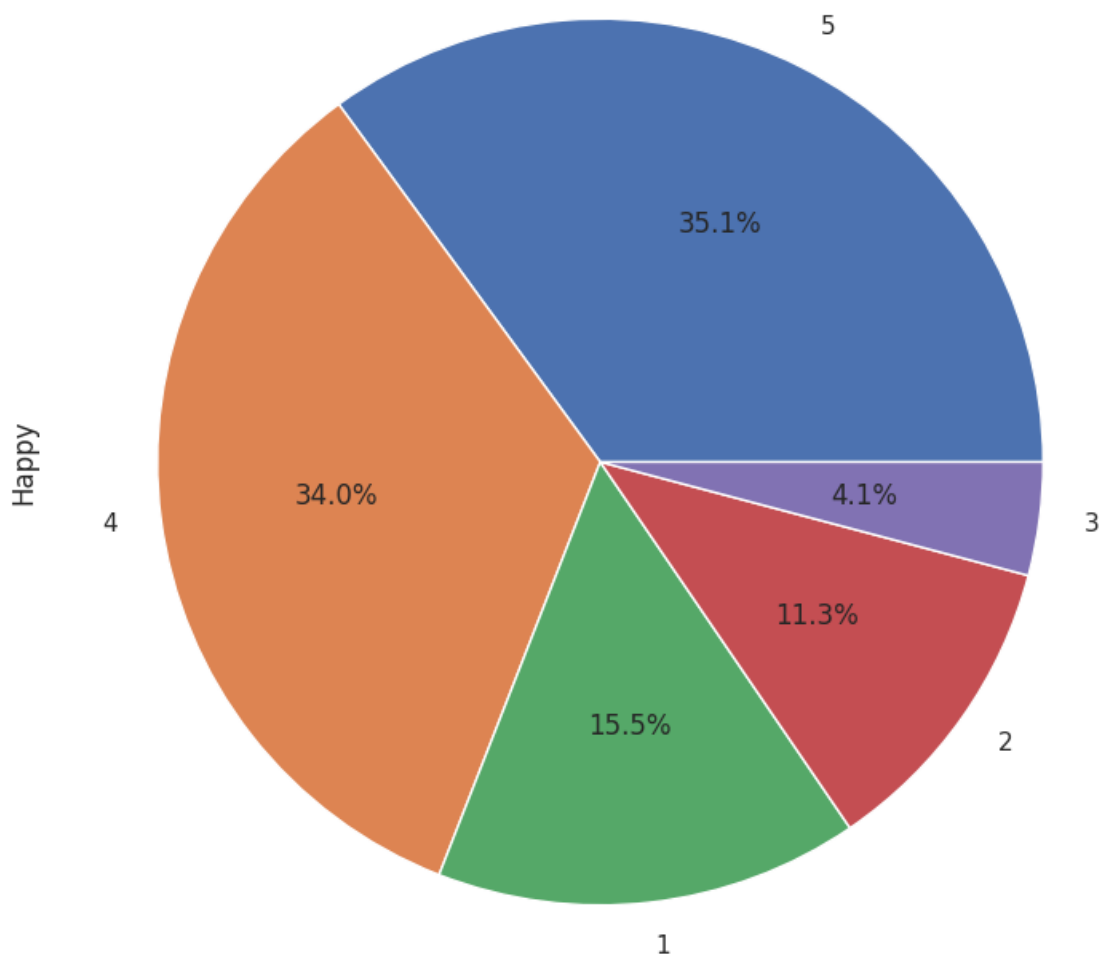
1 15.9%

2 11.7%

3 4.1%

Name: Happy, dtype: object

```
In [30]: plt.figure(figsize=(15, 9))
percentile_happy_int.plot(kind='pie', autopct='%1.1f%%')
plt.show()
```



Satisfaction percentage is 71.86%

```
In [31]: # average satisfaction of the people is 3.59 about/proportional that is 71.8%
df['Happy'].mean() * 20
```

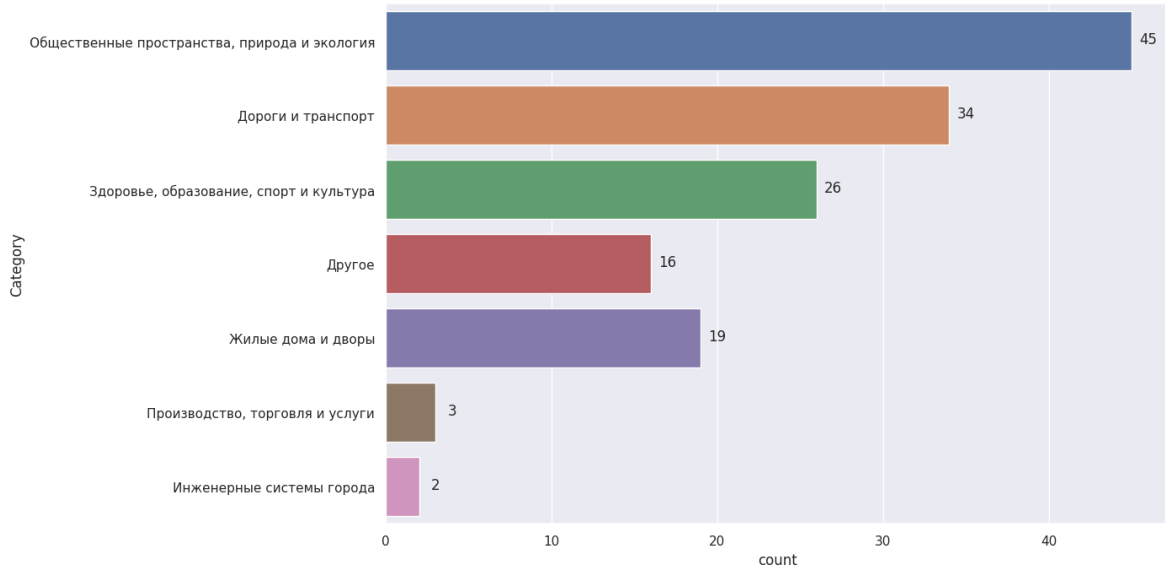
Out[31]: 71.86206896551724

Category analysis

```
In [32]: df['Category'].value_counts().sort_values(ascending=False)
```

```
Out[32]: Общественные пространства, природа и экология    45
Дороги и транспорт    34
Здоровье, образование, спорт и культура    26
Жилые дома и дворы    19
Другое    16
Производство, торговля и услуги    3
Инженерные системы города    2
Name: Category, dtype: int64
```

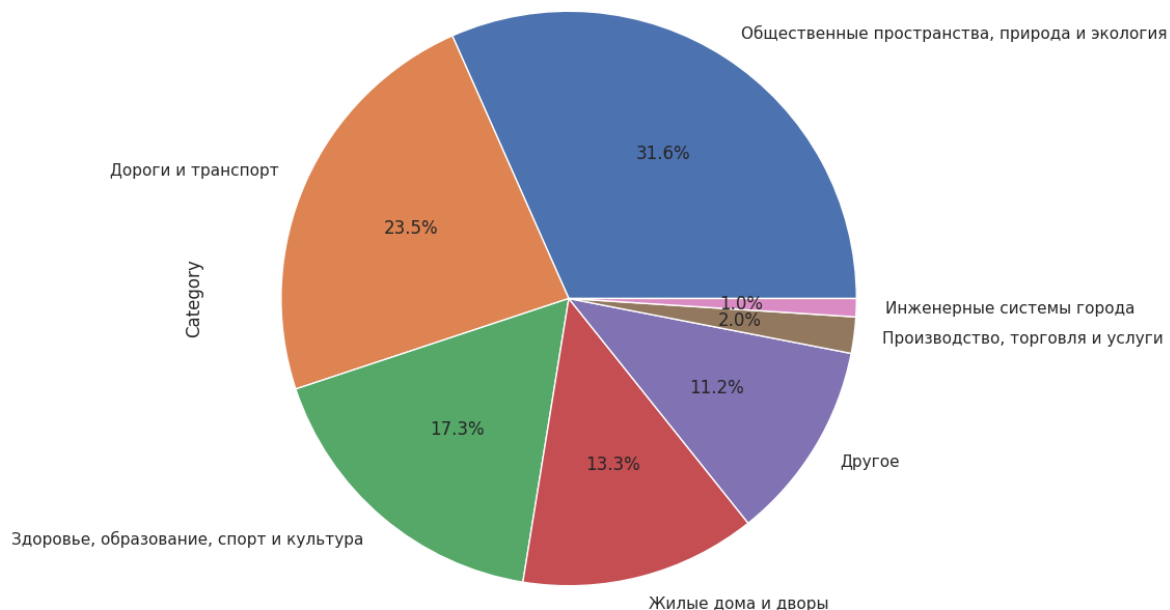
```
In [33]: plt.figure(figsize=(12,8))
ax = sns.countplot(y="Category", data=df)
for p in ax.patches:
    ax.annotate(
        format(p.get_width(), '.0f'),
        (p.get_width() + 1, p.get_y() + p.get_height() / 1.5),
        ha = 'center',
        va = 'center',
        xytext = (0, 10),
        textcoords = 'offset points'
    )
plt.show()
```



```
In [34]: percentile_category_str = df.Category.value_counts(normalize=True).mul(100).round(1)
percentile_category_int = df.Category.value_counts(normalize=True).mul(100).round(1)
display(percentile_category_str)
```

```
Общественные пространства, природа и экология    31.0%
Дороги и транспорт    23.4%
Здоровье, образование, спорт и культура    17.9%
Жилые дома и дворы    13.1%
Другое    11.0%
Производство, торговля и услуги    2.1%
Инженерные системы города    1.4%
Name: Category, dtype: object
```

```
In [35]: plt.figure(figsize=(15, 9))
percentile_category_int.plot(kind='pie', autopct='%1.1f%%')
plt.show()
```



10 Most relevant comments

```
In [36]: ten_most_popular = df.nlargest(10, 'SubmissionLikes')
```

```
In [37]: ten_most_popular.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 10 entries, 4 to 8
Data columns (total 20 columns):
#   Column                Non-Null Count  Dtype
---  -
0   id                    10 non-null    int64
1   SubmissionLikes       10 non-null    int64
2   Category              10 non-null    object
3   Description           10 non-null    object
4   Participation         10 non-null    bool
5   Money                 10 non-null    object
6   Age                   10 non-null    object
7   ContactName          10 non-null    object
8   LiveTime              10 non-null    object
9   FeatureType           10 non-null    object
10  GeometryType          10 non-null    object
11  CoordinatesLongitude   10 non-null    float64
12  CoordinatesLatitude    10 non-null    float64
13  CreatedAt             10 non-null    object
14  Gender                10 non-null    object
15  MeanAge               10 non-null    float64
16  LowerAge              9 non-null     float64
17  UpperAge              10 non-null    float64
18  LiveTimeInt           10 non-null    float64
19  Happy                 10 non-null    int64
dtypes: bool(1), float64(6), int64(3), object(10)
memory usage: 1.6+ KB
```

```
In [38]: display(ten_most_popular[['Description', 'SubmissionLikes', 'Category', 'Money'],
```

	Description	SubmissionLikes	Category	Money	Happy
4	Реконструкция ДК Судостроитель	185	Здоровье, образование, спорт и культура	Бюджетные	5
17	Предлагаем в ваш проект включить автоспортивны...	166	Здоровье, образование, спорт и культура	Частные	4
7	Восстановить нормальную работу Комсомольского ...	152	Дороги и транспорт	Бюджетные	5
9	Учитывая, что у нас в городе с очисткой дорог...	137	Дороги и транспорт	Бюджетные	2
36	Я хочу и даже требую, чтобы наш город "Президе...	131	Здоровье, образование, спорт и культура	Бюджетные	2
6	Возрождение муниципального транспорта-автобусы...	123	Дороги и транспорт	Бюджетные	5
5	привести в современное состояние парки города...	116	Общественные пространства, природа и экология	Пока не знаю	5
2	Здание автовокзала требует колоссальной реконс...	109	Дороги и транспорт	Бюджетные	4
20	Штрафовать за выгул собак по всему городу, вес...	108	Общественные пространства, природа и экология	Бюджетные	1
8	Предлагаю провести реконструкцию одной из глав...	101	Здоровье, образование, спорт и культура	Бюджетные	2

```
In [39]: from IPython.display import display, HTML

columns = ['Description', 'SubmissionLikes', 'Category', 'Money', 'Happy']

ten_most_popular[columns].iloc[0]
for i in range(10):
    l = ten_most_popular[columns].iloc[i].tolist()
    comment_content = l[0]
    comment_likes = l[1]
    comment_category = l[2]
    comment_money = l[3]
    comment_happy = l[4]
    if not all([comment_content, comment_likes, comment_category, comment_money,
               comment_happy]):
        continue
    display(HTML(f"<h2>{i+1}. Likes: {comment_likes} Category: {comment_category}"))
    display(HTML(f"<p style='font-size:16px'>{comment_content}<br><br>"))
```

1. Likes: 185 Category: Здоровье, образование, спорт и культура Funding: Бюджетные Happy: 5

Реконструкция ДК Судостроитель

2. Likes: 166 Category: Здоровье, образование, спорт и культура Funding: Частные Happy: 4

Предлагаем в ваш проект включить автоспортивный комплекс для технических видов спорта, а так же как ещё одна зона отдыха для жителей Комсомольска на Амуре, есть презентация с описанием возможных видов направления развития данной территории, в настоящий момент администрацией города данный участок передан нам под данное направление развития, но мы не можем найти фирму которая нам поможет для начала нарисовать внешний вид проекта для поиска инвесторов через агентство инвестиций Хабаровского края. В данном комплексе планируется гоночная трасса, картинг трасса, мотокросс трасса (уже реализована) и многие другие направления. Единственный комплекс на базе картинг трассы закрыли в приморье и передали для авторынка, в настоящий момент на дальнем востоке полностью отсутствует альтернатива для проведения гоночных заездов, обучения экстремальному вождению, просто любительских заездов и т.д. данный комплекс мог бы стать мощной точкой притяжения со всего ДВ а так же стран Азии т.к. есть отличная возможность проведения тестовых испытаний автомобилей автоконцернами благодаря своей погоде летом до +40 зимой до -45 есть отличная возможность проверять концепты и пред серийные образцы. Так же картинг трасса поможет возродить потенциал молодых спортсменов в данном направлении. Для примера из нашего города очень много талантливых спортсменов уехало на запад одним из пунктов переезда стало отсутствие инфраструктуры для повышения своего опыта вождения и оттачивания навыков. Надеюсь на сотрудничество, С уважением руководитель НКО "СТА АВТОМИР" Шанин Дмитрий Олегович, тел. 89144096857

3. Likes: 152 Category: Дороги и транспорт Funding: Бюджетные Happy: 5

Восстановить нормальную работу Комсомольского аэропорта

4. Likes: 137 Category: Дороги и транспорт Funding: Бюджетные Happy: 2

Учитывая, что у нас в городе с очисткой дорог от снега полный хаос, нужен контрольный инструмент с функцией обратной связи от жителей города. Нужен специальный программный продукт (портал) с общим доступом зарегистрированных на нем граждан, в котором будет вестись интерактивный отчет городских служб о фактически выполненных работах по уборке снега с конкретной привязкой (на карте города) объектов очистки и количестве привлекаемой техники (и какой техники). Жители города смогут зайти на этот ресурс и дать свои комментарии - действительно ли есть реально работа по уборке снега, прикрепить фото, что это совсем не так или наоборот подтвердить. В нашей ситуации - информация это оружие, оружие против очковтирательства со стороны городской администрации, что они занимаются уборкой и очисткой улиц, остановок, от снега. Можно оборудовать всю технику задействованную на уборке системой мониторинга БСМТС (бортовые системы мониторинга транспортных средств) и будет понятно где, когда эта техника работала, так как есть во всех таких системах архив (история движения). Эти данные должны быть наложены и интегрированы с картой вышеуказанного ресурса. Это позволит увидеть жителям города, что властям не плевать на население и они (власти) действительно заинтересованы в благоустройстве. Тем более данные технологии уже обкатаны - есть система контроля продвижения общественного транспорта, в ГБЛД можно оставить фото интернета. Дело за малым - принять решение и далее уже реализовать.

5. Likes: 131 Category: Здоровье, образование, спорт и культура Funding: Бюджетные Happy: 2

Я хочу и даже требую, чтобы наш город "Президентского внимания" был обеспечен системой непрерывного мониторинга он же Либра. Для детей инвалидов, страдающих сахарным диабетом, в нашем городе к сожалению есть дети, начиная с 2х летнего возраста которые болеют диабетом, колоть по 10 раз пальчики в сутки это очень больно, а так же не всегда уследишь что падает или растёт сахар, что может привести к коме.. очень жаль, что соседние города обеспечивают детей и так же людей с сахарным диабетом 1 типа, за счёт федерального бюджета, а в нашем городе это невозможно.

6. Likes: 123 Category: Дороги и транспорт Funding: Бюджетные Happy: 5

Возрождение муниципального транспорта-автобусы, трамваи

7. Likes: 116 Category: Общественные пространства, природа и экология Funding: Пока не знаю Happy: 5

привести в современное состояние парки города, где жители смогут отдохнуть от ежедневной суеты, заняться активными видами спорта и с пользой провести время

8. Likes: 109 Category: Дороги и транспорт Funding: Бюджетные Happy: 4

Здание автовокзала требует колоссальной реконструкции, или же замены!

9. Likes: 108 Category: Общественные пространства, природа и экология Funding: Бюджетные Happy: 1

Штрафовать за выгул собак по всему городу, весь город завален собачим дерьмом. На стадионе школы 23 постоянно гуляют с собаками.

10. Likes: 101 Category: Здоровье, образование, спорт и культура Funding: Бюджетные Happy: 2

Предлагаю провести реконструкцию одной из главных достопримечательностей города - дворца культуры "Судостроитель", который расположен на пересечении главных транспортных и пешеходных маршрутов и является центром притяжения жителей и гостей города. К сожалению, данное здание в 2003 году практически полностью было уничтожено пожаром.

Money

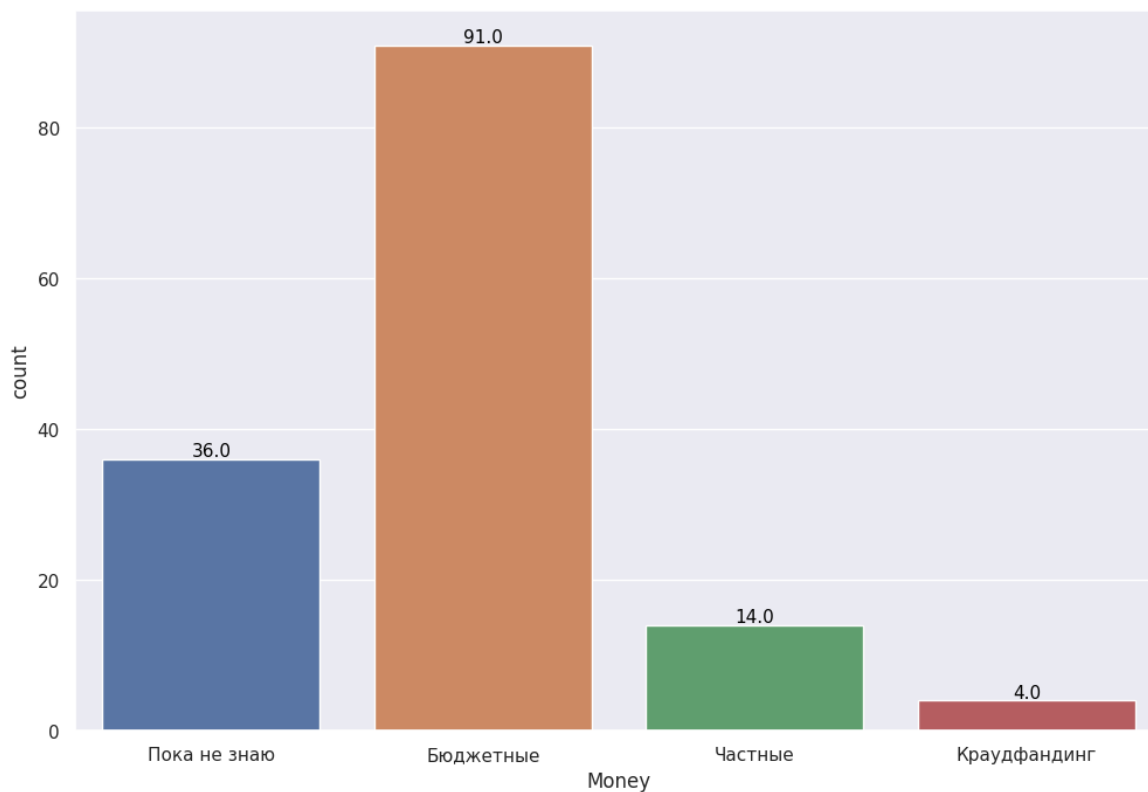
```
In [40]: plt.figure(figsize=(12, 8))  
ax = sns.countplot(x=df['Money'])
```



```

for p in ax.patches:
    ax.annotate(str(p.get_height()), (p.get_x() + p.get_width() / 2., p.get_height() / 2.,
                                     ha='center', va='center', fontsize=11, color='black', xytext=(0,
plt.show()

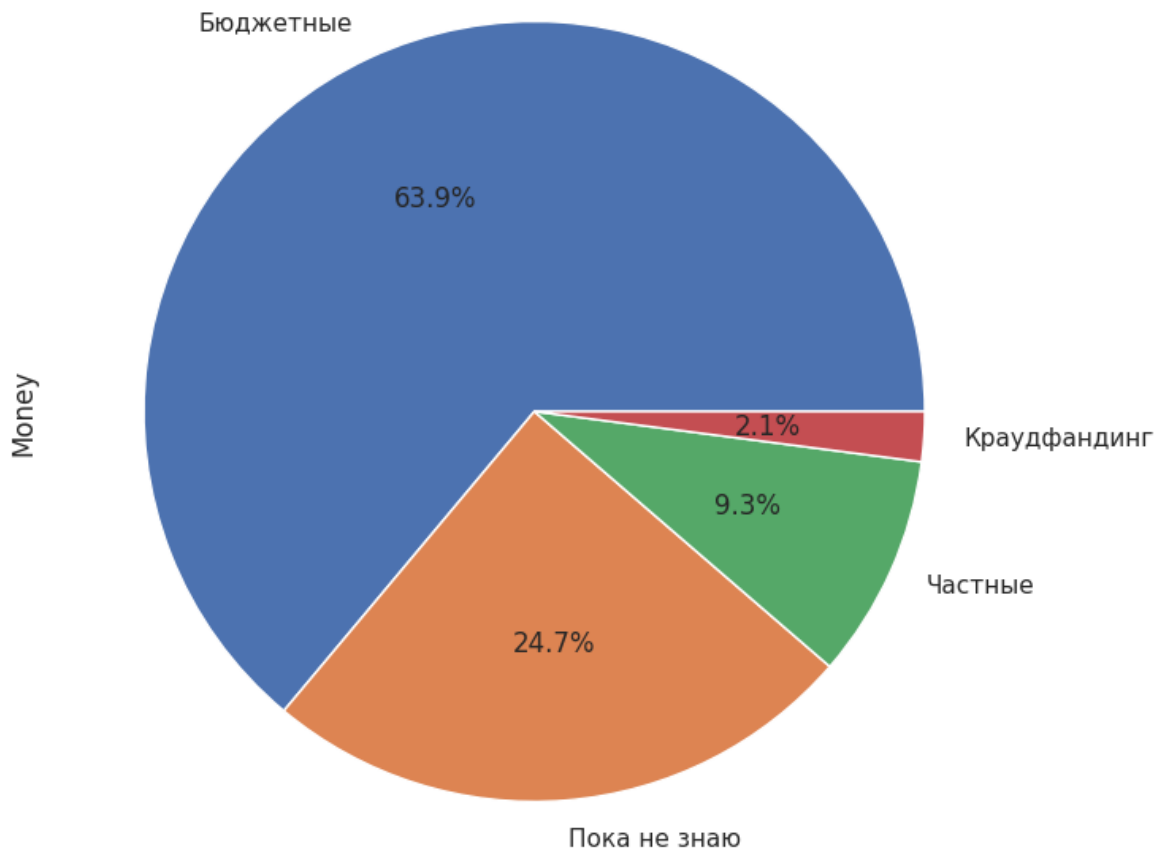
```



```

In [41]: percentile_money_int = df.Money.value_counts(normalize=True).mul(100).round(1).a
plt.figure(figsize=(12, 8))
percentile_money_int.plot(kind='pie', autopct='%1.1f%%')
plt.show()

```

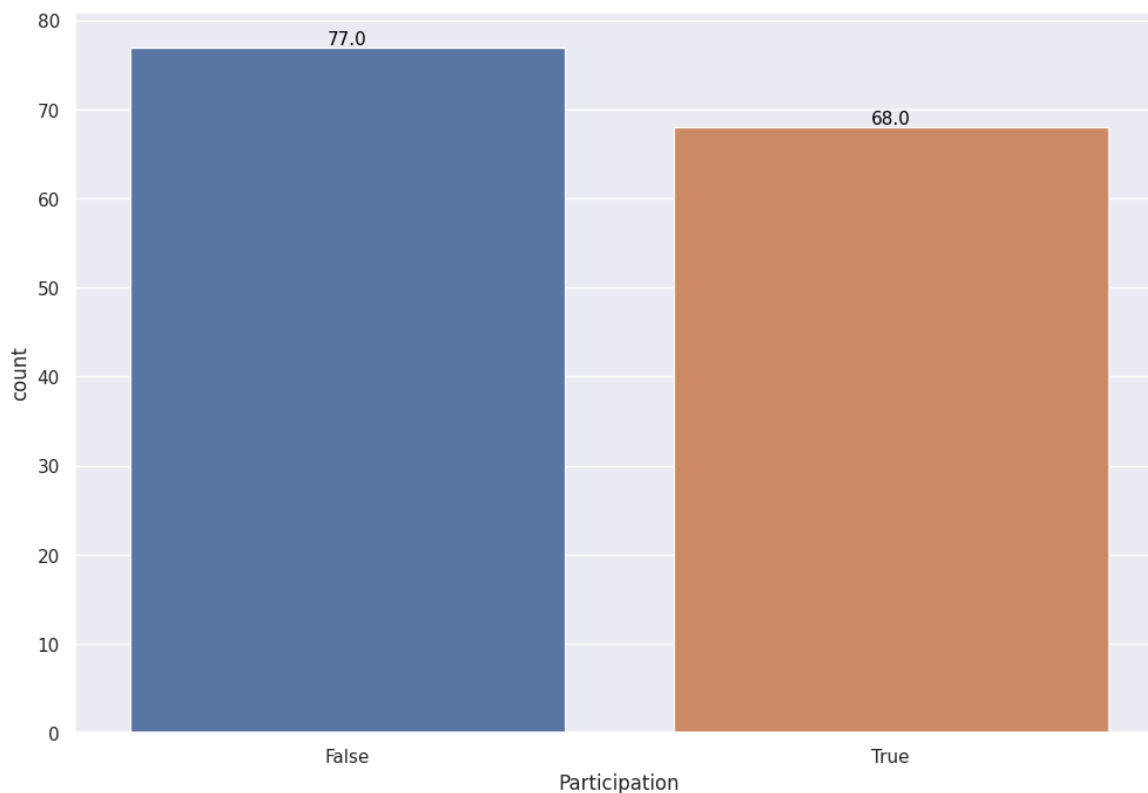


Participation

```
In [42]: plt.figure(figsize=(12, 8))

ax = sns.countplot(x=df['Participation'])

for p in ax.patches:
    ax.annotate(str(p.get_height()), (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', fontsize=11, color='black', xytext=(0,
plt.show()
```



In [43]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145 entries, 0 to 144
Data columns (total 20 columns):
#   Column                Non-Null Count  Dtype
---  -
0   id                     145 non-null   int64
1   SubmissionLikes        145 non-null   int64
2   Category               145 non-null   object
3   Description             145 non-null   object
4   Participation           145 non-null   bool
5   Money                  145 non-null   object
6   Age                    145 non-null   object
7   ContactName            145 non-null   object
8   LiveTime               145 non-null   object
9   FeatureType            145 non-null   object
10  GeometryType           145 non-null   object
11  CoordinatesLongitude    145 non-null   float64
12  CoordinatesLatitude     145 non-null   float64
13  CreatedAt              145 non-null   object
14  Gender                  143 non-null   object
15  MeanAge                 145 non-null   float64
16  LowerAge                134 non-null   float64
17  UpperAge                143 non-null   float64
18  LiveTimeInt             145 non-null   float64
19  Happy                  145 non-null   int64
dtypes: bool(1), float64(6), int64(3), object(10)
memory usage: 21.8+ KB
```

In [44]: `df.at[3, 'Gender'] = 'f'`
`df.at[31, 'Gender'] = 'm'`
`df.at[82, 'Gender'] = 'f'`
`df.at[83, 'Gender'] = 'm'`
`df.at[92, 'Gender'] = 'm'`

```
In [45]: df['Gender'].value_counts()
```

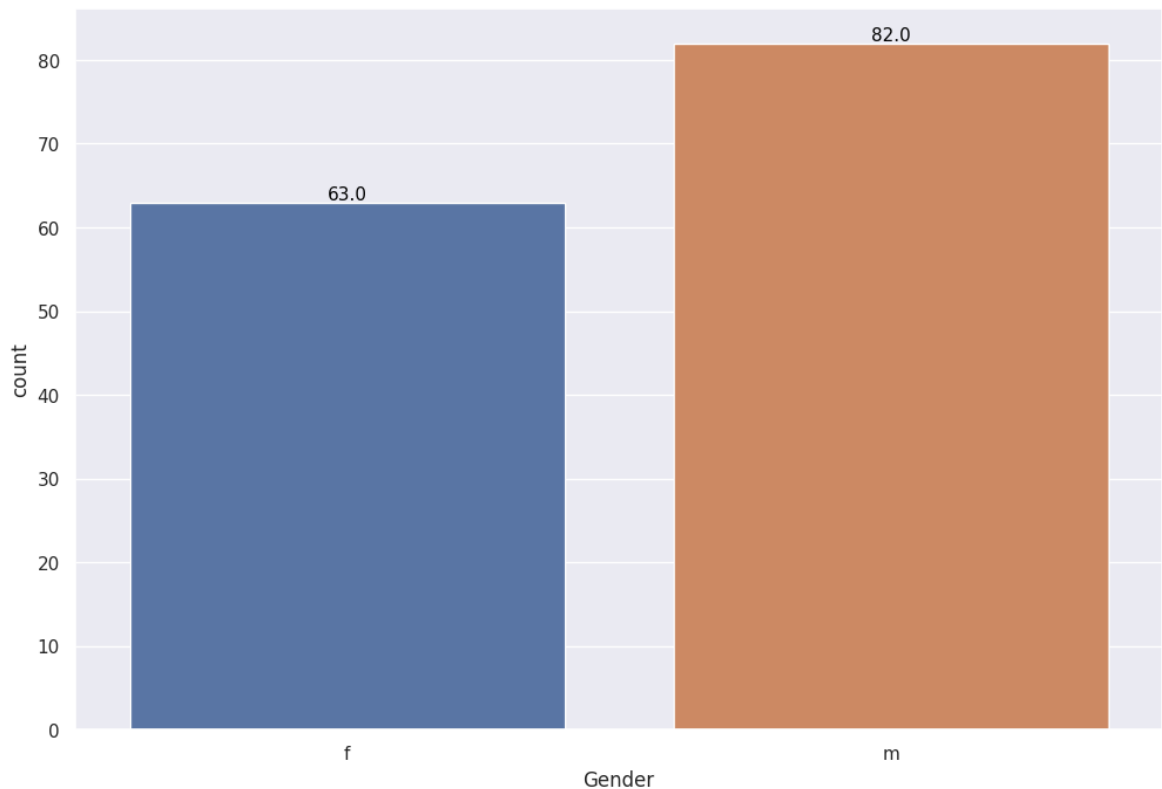
```
Out[45]: m    82
         f    63
         Name: Gender, dtype: int64
```

```
In [46]: plt.figure(figsize=(12, 8))

         ax = sns.countplot(x=df['Gender'])

         for p in ax.patches:
             ax.annotate(str(p.get_height()), (p.get_x() + p.get_width() / 2., p.get_height()),
                           ha='center', va='center', fontsize=11, color='black', xytext=(0,
                                                                                       p.get_height() - 5))

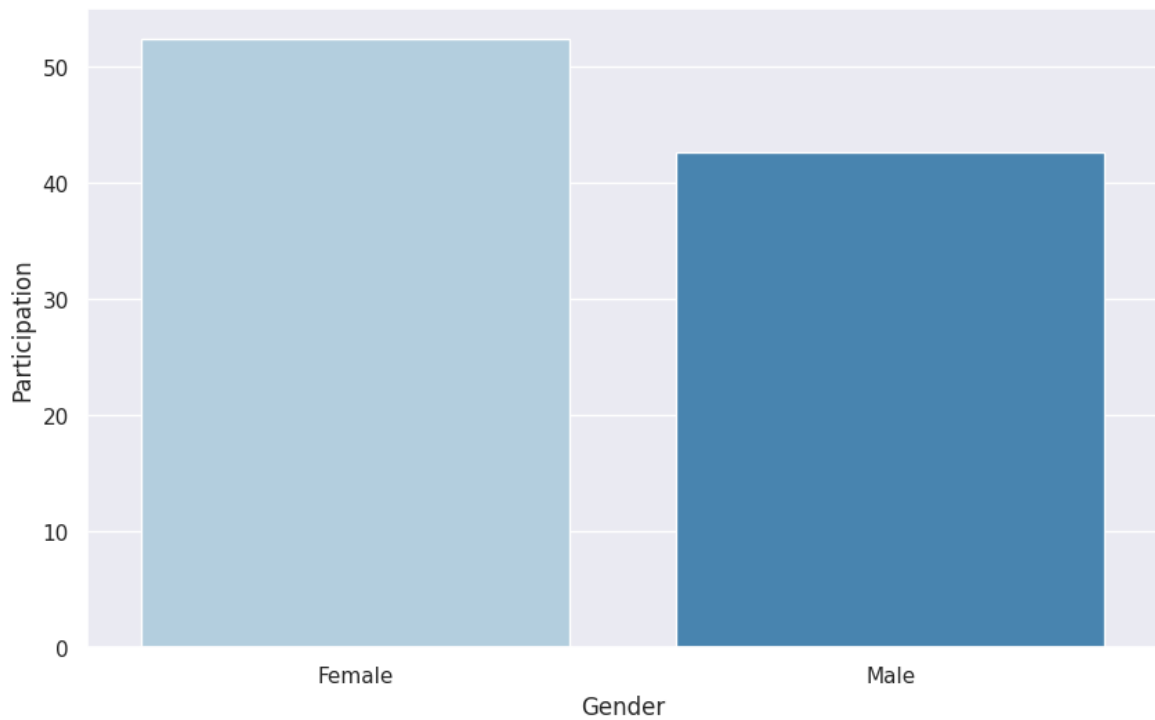
         plt.show()
```



```
In [47]: df['Gender'] = df['Gender'].apply(lambda x: 'Female' if x == 'f' else 'Male')
```

```
In [48]: plt.figure(figsize=(10, 6))
         result = df.groupby('Gender')['Participation'].mean() * 100
         display(result)
         sns.barplot(x=result.index, y=result, palette='Blues')
         plt.show()
```

```
Gender
Female    52.380952
Male      42.682927
         Name: Participation, dtype: float64
```

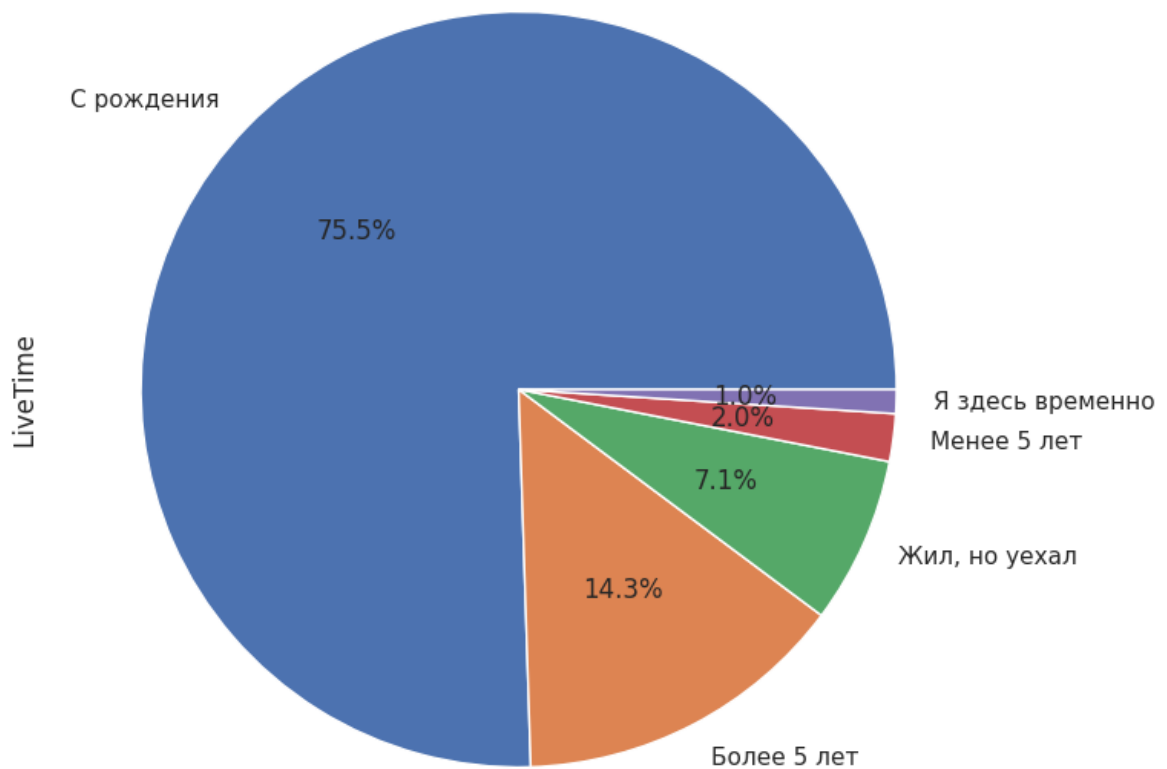


```
In [49]: df.groupby('Gender')['Participation'].value_counts()
```

```
Out[49]: Gender  Participation
Female  True           33
        False          30
Male    False          47
        True           35
Name: Participation, dtype: int64
```

Native

```
In [50]: percentile_live_int = df.LiveTime.value_counts(normalize=True).mul(100).round(1)
plt.figure(figsize=(12, 8))
percentile_live_int.plot(kind='pie', autopct='%1.1f%%')
plt.show()
```



Correlation between volunteer and living

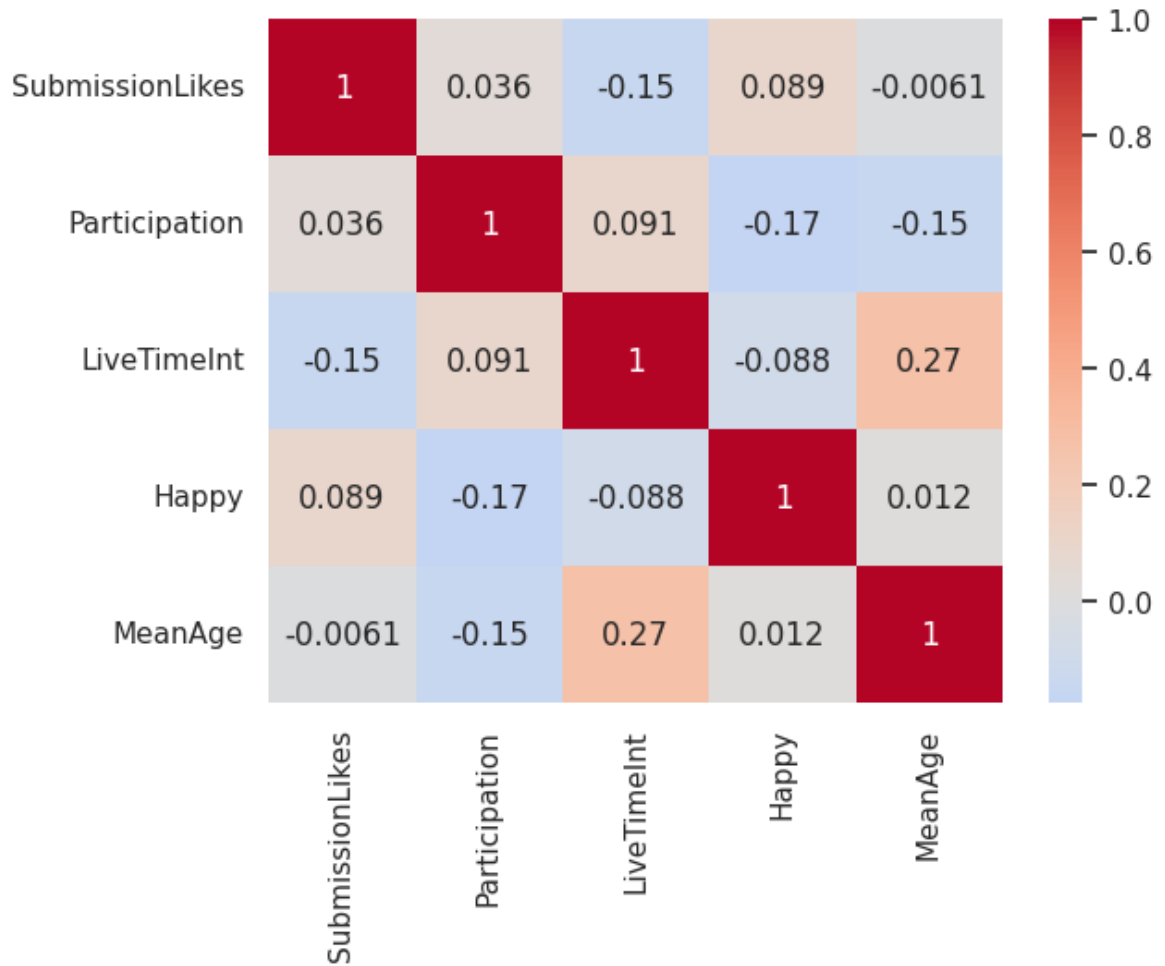
```
In [51]: corr = df['Participation'].corr(df['LiveTimeInt'])
corr
```

```
Out[51]: 0.09064063376564829
```

Correlation matrix

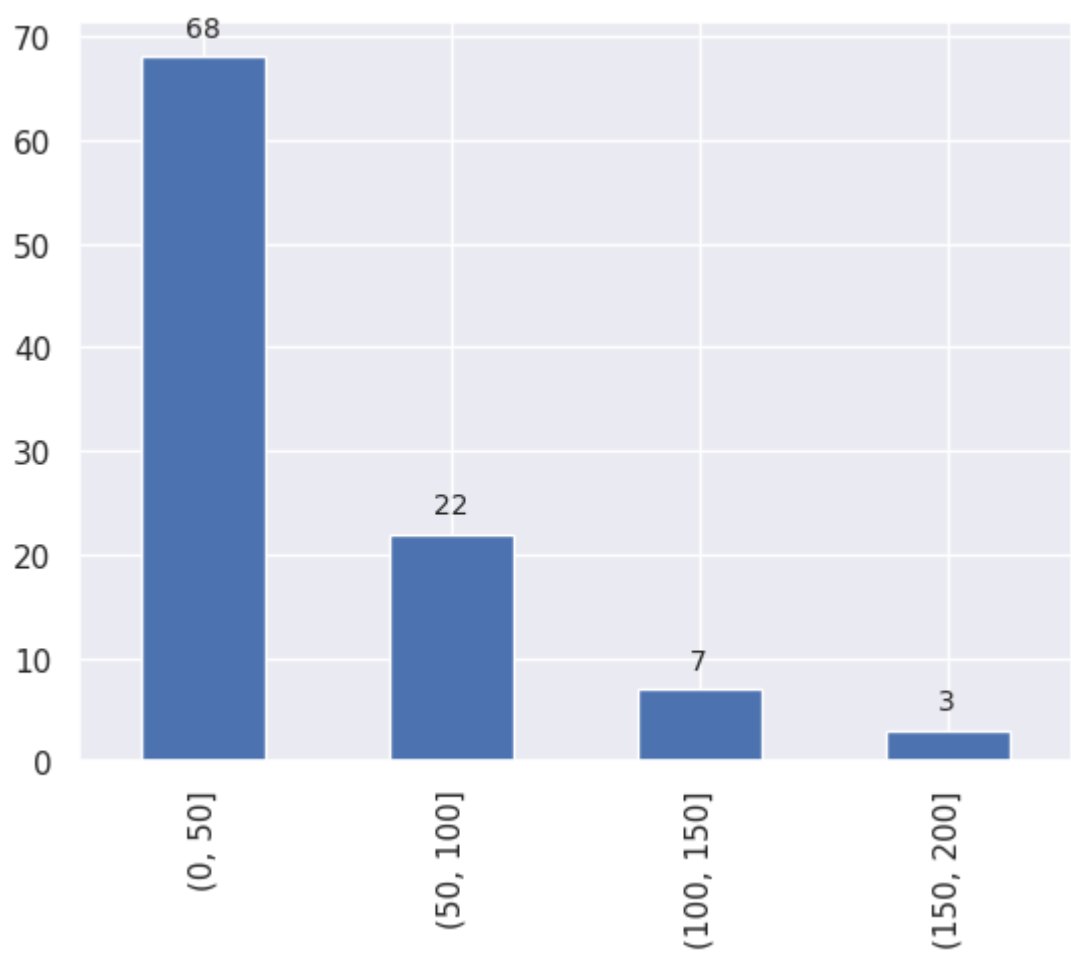
```
In [52]: df_subset = df[['SubmissionLikes', 'Participation', 'LiveTimeInt', 'Happy', 'Mean']]
corr_matrix = df_subset.corr()

sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', center=0)
plt.show()
```



Likes bins

```
In [53]: bins = [0, 50, 100, 150, 200]
df['LikeBins'] = pd.cut(df['SubmissionLikes'], bins=bins)
counts = df['LikeBins'].value_counts().sort_index()
ax = counts.plot(kind='bar')
for i, v in enumerate(counts):
    ax.text(i, v + 2, str(v), ha='center', fontsize=10)
# display(df['LikeBins'])
```



```
In [54]: df.head()
```


Out[54]:

	id	SubmissionLikes	Category	Description	Participation	Money	Age
0	7	67	Общественные пространства, природа и экология	Благоустроить Силинский лес: освещение, дорожк...	False	Пока не знаю	40-65
1	8	61	Общественные пространства, природа и экология	Благоустроить Силинский лес	False	Пока не знаю	40-65
2	9	109	Дороги и транспорт	Здание автовокзала требует колоссальной реконс...	False	Бюджетные	25-40
3	10	44	Общественные пространства, природа и экология	Ремонтируют парк очень плохо, страшно коляской...	False	Бюджетные	До 25
4	11	185	Здоровье, образование, спорт и культура	Реконструкция ДК Судостроитель	False	Бюджетные	25-40

