РК №1

Исполнитель: ИУ5-24М Оганесян Рубен Рубенович

In [3]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

In [4]:

```
data = pd.read_csv('googleplaystore.csv', sep=",")
```

In [5]:

```
data.head()
```

Out[5]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone

In [6]:

```
data.shape
```

Out[6]:

```
(10841, 13)
```

In [7]:

```
data.columns
```

Out[7]:

```
Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs',
'Type',
       'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current
Ver',
       'Android Ver'],
      dtype='object')
```

In [8]:

```
for col in data.columns:

    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
App - 0
Category - 0
Rating - 1474
Reviews - 0
Size - 0
Installs - 0
Type - 1
Price - 0
Content Rating - 1
Genres - 0
Last Updated - 0
Current Ver - 8
Android Ver - 3
```

In [9]:

```
data = data.dropna(axis='columns')
```

In [10]:

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

Out[10]:

```
App             0
Category        0
Reviews         0
Size            0
Installs        0
Price           0
Genres          0
Last Updated    0
dtype: int64
```

In [11]:

```
data.drop_duplicates(subset='App', inplace=True)
```

In [12]:

```
print('Number of apps in the dataset : ' , len(data))
data.sample(7)
```

```
Number of apps in the dataset :  9660
```

Out[12]:

	App	Category	Reviews	Size	Installs	Price	Genres
9970	Bird - Enjoy The Ride	TRAVEL_AND_LOCAL	2649	25M	500,000+	0	Travel & Local
5015	AE Garage	AUTO_AND_VEHICLES	64	66M	1,000+	0	Auto & Vehicles
6494	BM speed test	TOOLS	1	3.7M	10+	0	Tools
1378	Fooducate Healthy Weight Loss & Calorie Counter	HEALTH_AND_FITNESS	14402	Varies with device	1,000,000+	0	Health & Fitness
7601	Surely You Quest - Magiswords	FAMILY	43314	83M	1,000,000+	0	Role Playing;Action & Adventure
132	Eyeliner step by step 2018	BEAUTY	18	3.2M	5,000+	0	Beauty
8282	D.C. Driving/Walking Tours	TRAVEL_AND_LOCAL	0	32M	50+	$4.99	Travel & Local

Cleaning data

In [13]:

```
data = data[data['Installs'] != 'Free']
data = data[data['Installs'] != 'Paid']


data['Installs'] = data['Installs'].apply(lambda x: x.replace('+', '') if '+' in
str(x) else x)
data['Installs'] = data['Installs'].apply(lambda x: x.replace(',', '') if ',' in
str(x) else x)
data['Installs'] = data['Installs'].apply(lambda x: int(x))
```

In [14]:

```
data['Size'] = data['Size'].apply(lambda x: str(x).replace('Varies with device',
'NaN') if 'Varies with device' in str(x) else x)

data['Size'] = data['Size'].apply(lambda x: str(x).replace('M', '') if 'M' in st
r(x) else x)
data['Size'] = data['Size'].apply(lambda x: str(x).replace(',', '') if 'M' in st
r(x) else x)
data['Size'] = data['Size'].apply(lambda x: float(str(x).replace('k', '')) / 100
0 if 'k' in str(x) else x)


data['Size'] = data['Size'].apply(lambda x: float(x))
data['Installs'] = data['Installs'].apply(lambda x: float(x))

data['Price'] = data['Price'].apply(lambda x: str(x).replace('$', '') if '$' in
str(x) else str(x))
data['Price'] = data['Price'].apply(lambda x: float(x))

data['Reviews'] = data['Reviews'].apply(lambda x: int(x))
```

In [15]:

```
data.head()
```

Out[15]:

	App	Category	Reviews	Size	Installs	Price	Genres	Last Updated
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	159	19.0	10000.0	0.0	Art & Design	January 7, 2018
1	Coloring book moana	ART_AND_DESIGN	967	14.0	500000.0	0.0	Art & Design;Pretend Play	January 15, 2018
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	87510	8.7	5000000.0	0.0	Art & Design	August 1, 2018
3	Sketch - Draw & Paint	ART_AND_DESIGN	215644	25.0	50000000.0	0.0	Art & Design	June 8, 2018
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	967	2.8	100000.0	0.0	Art & Design;Creativity	June 20, 2018

In [16]:

```
data.dtypes
```

Out[16]:

```
App               object
Category          object
Reviews            int64
Size             float64
Installs         float64
Price            float64
Genres            object
Last Updated      object
dtype: object
```

In [17]:

```
# Основные статистические характеристки набора данных
data.describe()
```

Out[17]:

	Reviews	Size	Installs	Price
count	9.659000e+03	8432.000000	9.659000e+03	9659.000000
mean	2.165926e+05	20.395289	7.777507e+06	1.099299
std	1.831320e+06	21.827542	5.375828e+07	16.852152
min	0.000000e+00	0.008500	0.000000e+00	0.000000
25%	2.500000e+01	4.600000	1.000000e+03	0.000000
50%	9.670000e+02	12.000000	1.000000e+05	0.000000
75%	2.940100e+04	28.000000	1.000000e+06	0.000000
max	7.815831e+07	100.000000	1.000000e+09	400.000000

Диаграмма рассеяния

Позволяет построить распределение двух колонок данных и визуально обнаружить наличие зависимости. Не предполагается, что значения упорядочены (например, по времени).

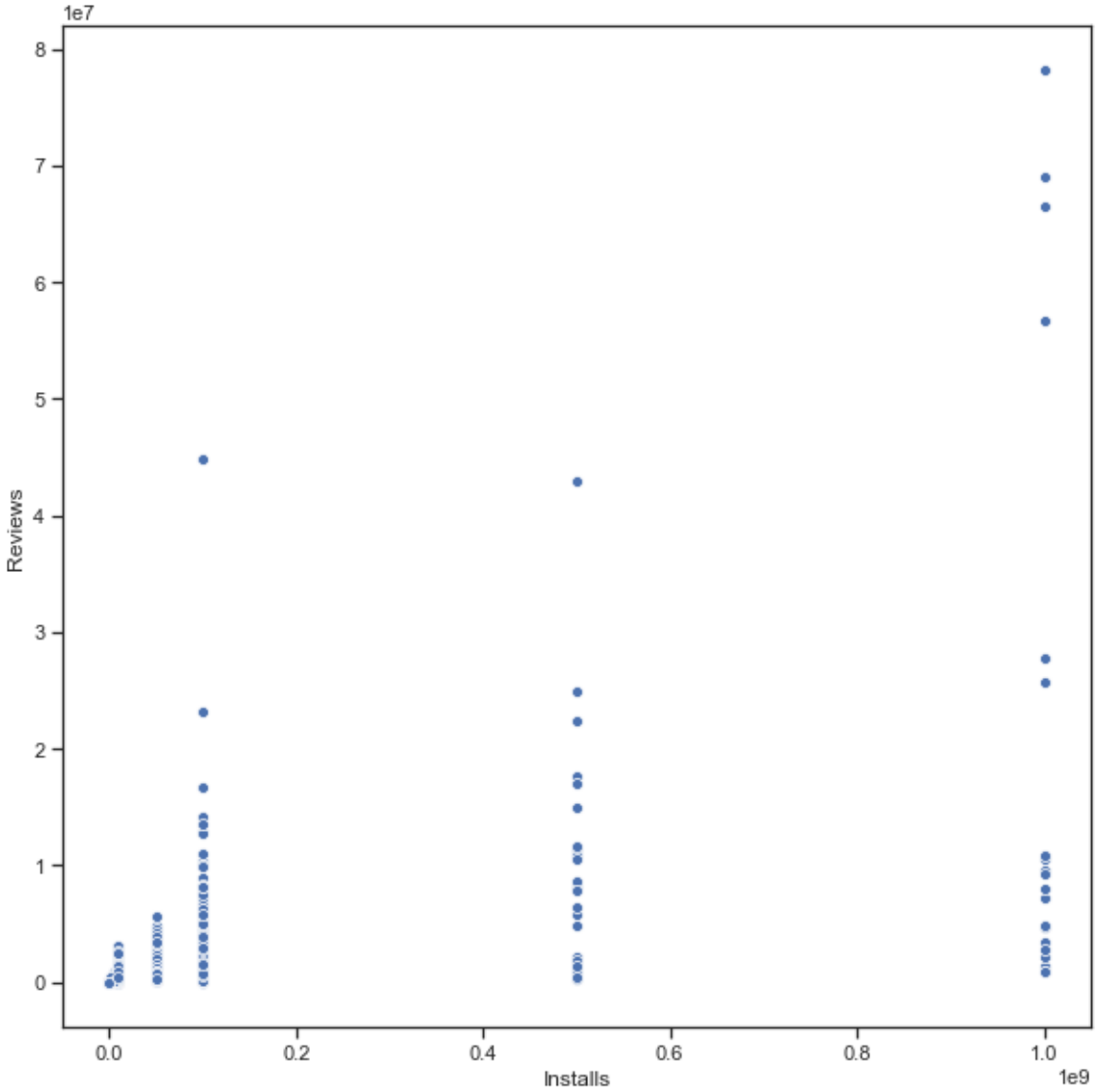
Построим диаграмму рассеяния для двух признаков - Installs и Reviews. По графику видно, что количество отзывов о приложении влияет на число установок. Но также распространены случаи, когда большое число установок приходится на приложения с небольшим количеством отзывов.

In [18]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='Installs', y='Reviews', data=data)
```

Out[18]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1100cf978>
```

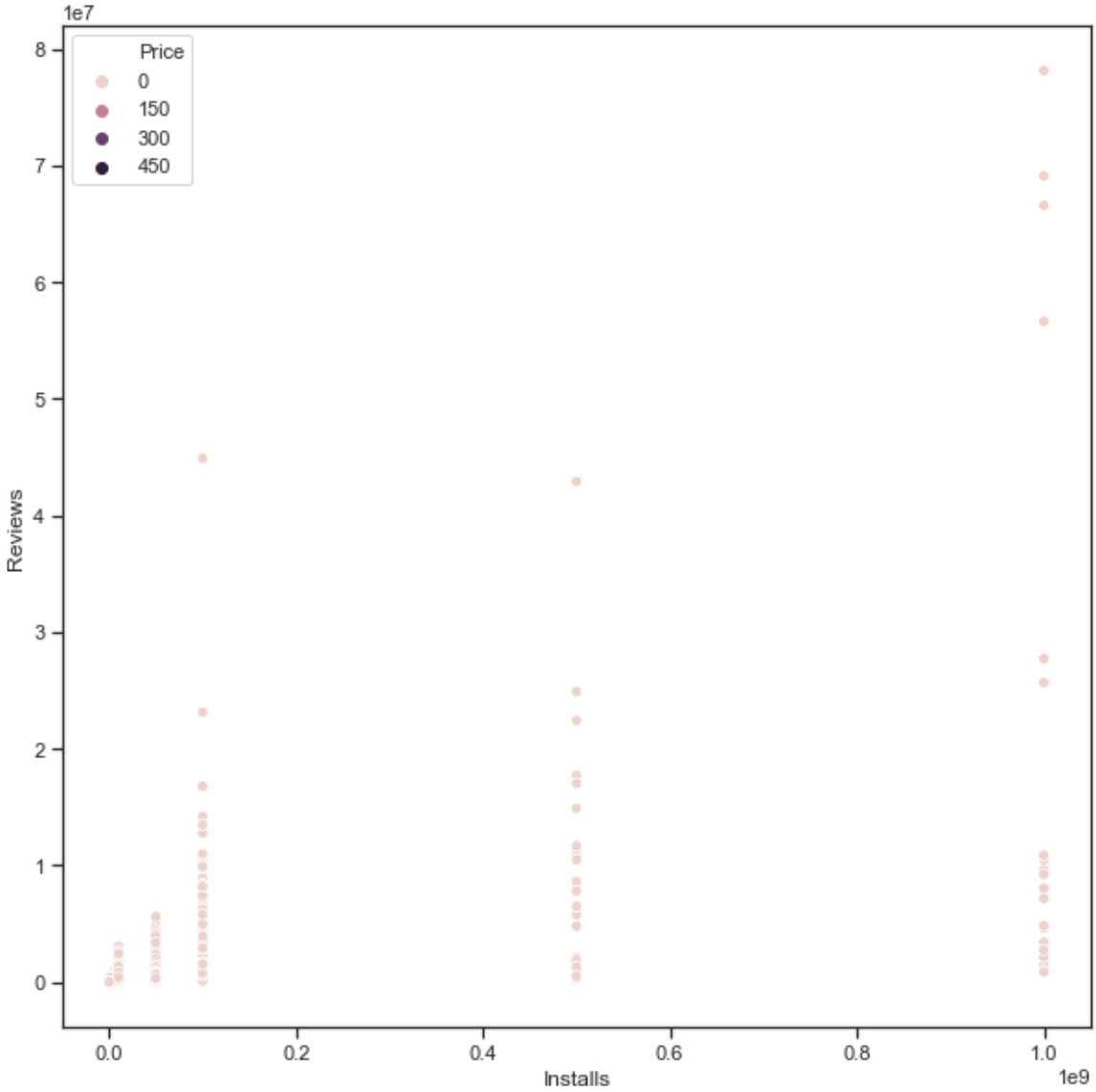


In [19]:

```
# Насколько на эту зависимость влияет целевой признак.
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='Installs', y='Reviews', data=data, hue='Price')
```

Out[19]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x123bede10>
```



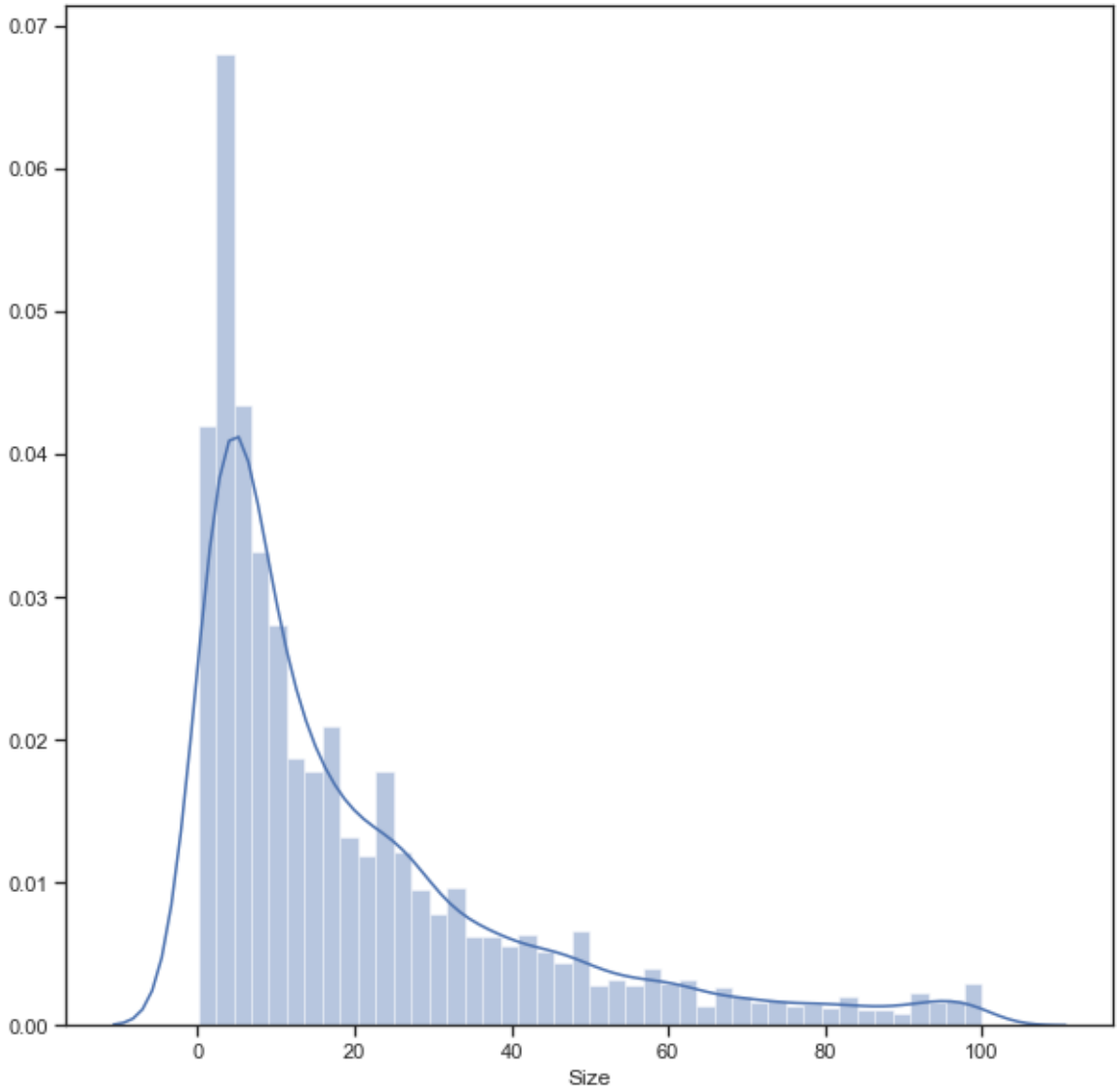
Гистограмма

In [20]:

```
# Определение вероятного значения size
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(data['Size'])
```

Out[20]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1242a5a20>
```



In [21]:

```
data['Size'].median()
```

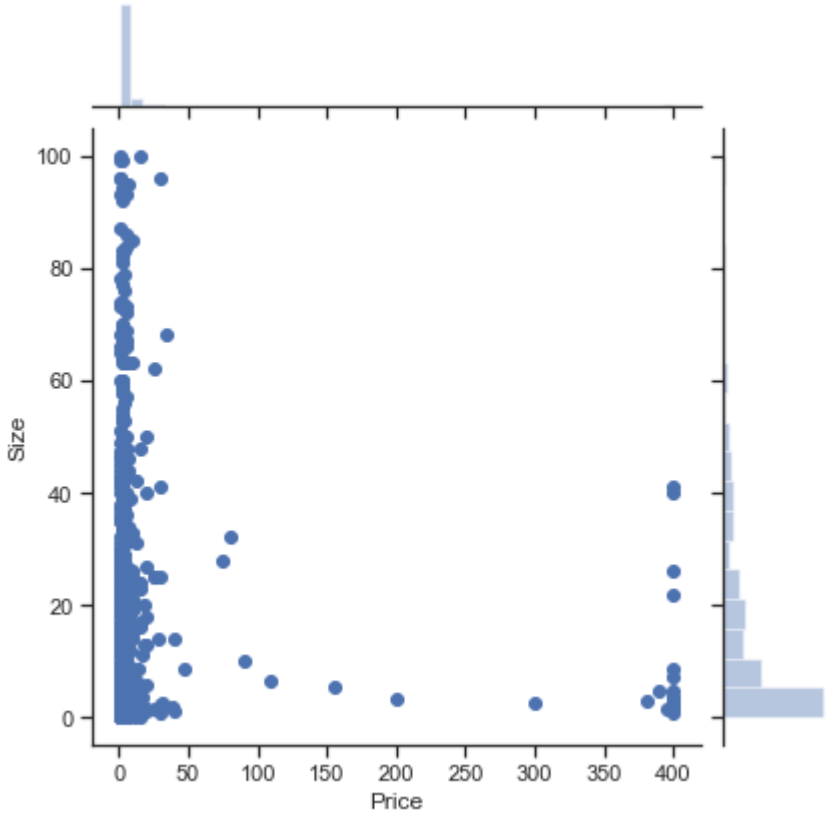
Out[21]:

```
12.0
```

Joinplot

In [22]:

```
# joinplot отображает зависимость цены от размера приложения
paid_apps = data[data.Price > 0]
p = sns.jointplot( "Price", "Size", paid_apps)
```



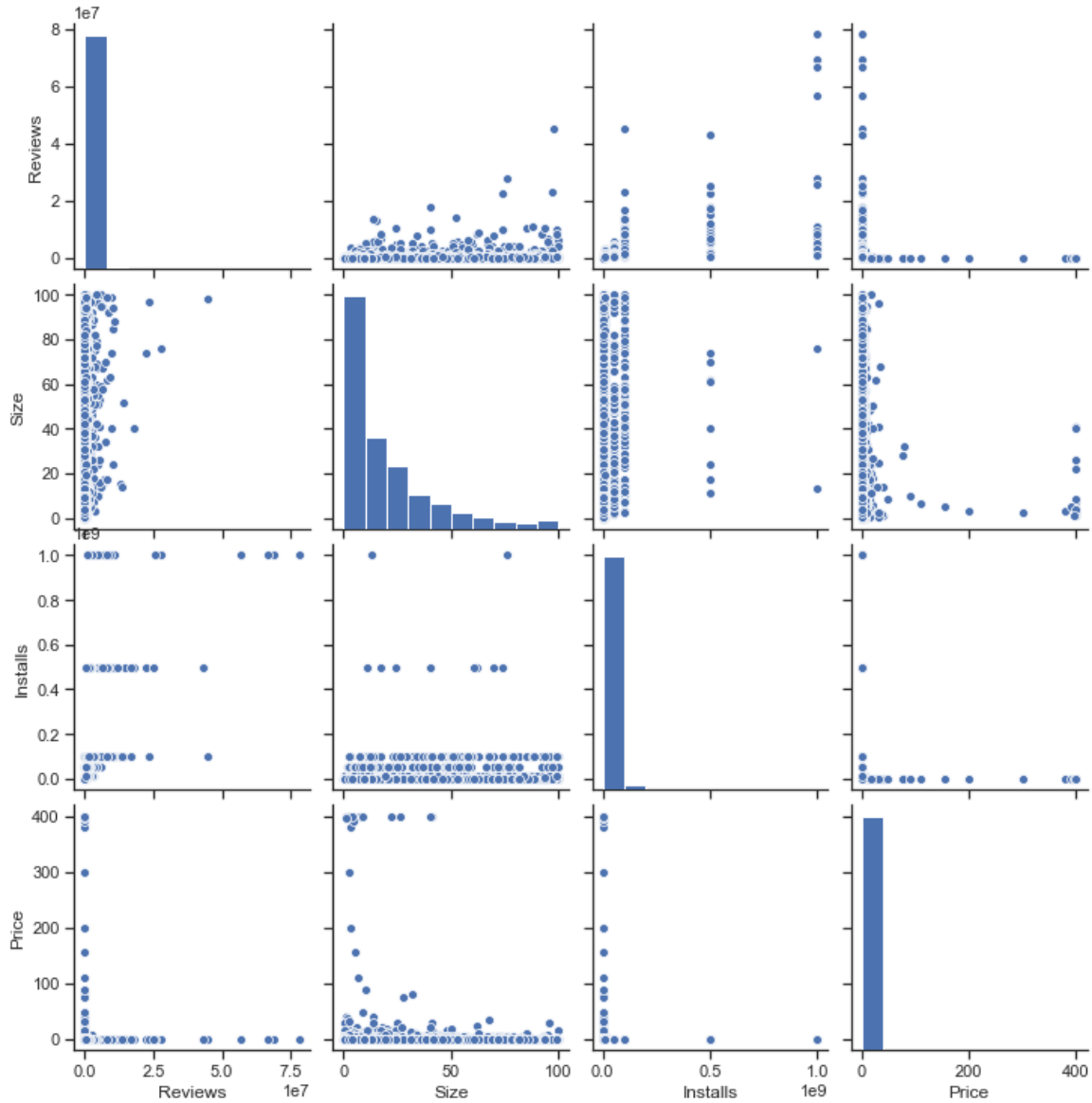
Парные диаграммы

In [23]:

```
# Парные диаграммы по признакам датасета
sns.pairplot(data)
```

Out[23]:

```
<seaborn.axisgrid.PairGrid at 0x124809278>
```



In [ ]:

```
# Группировка по значениям признака Price
sns.pairplot(data, hue="Price")
```

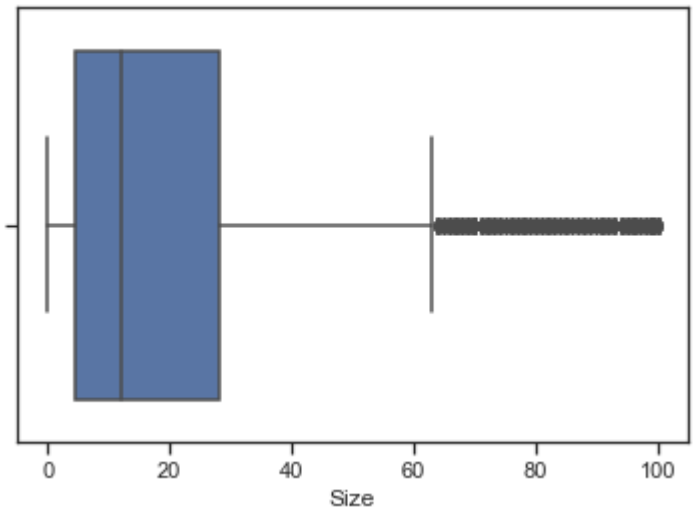
Ящик с усами

In [25]:

```
# Одномерное распределение вероятности
sns.boxplot(x=data['Size'])
```

Out[25]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x12ae69438>
```

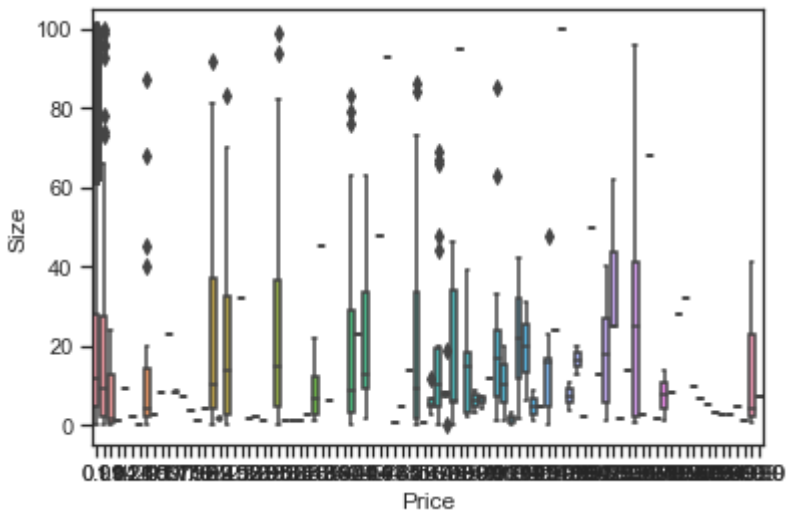


In [26]:

```
# Распределение параметра Size сгруппированные по Price.
sns.boxplot(x='Price', y='Size', data=data)
```

Out[26]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x12b094e80>
```



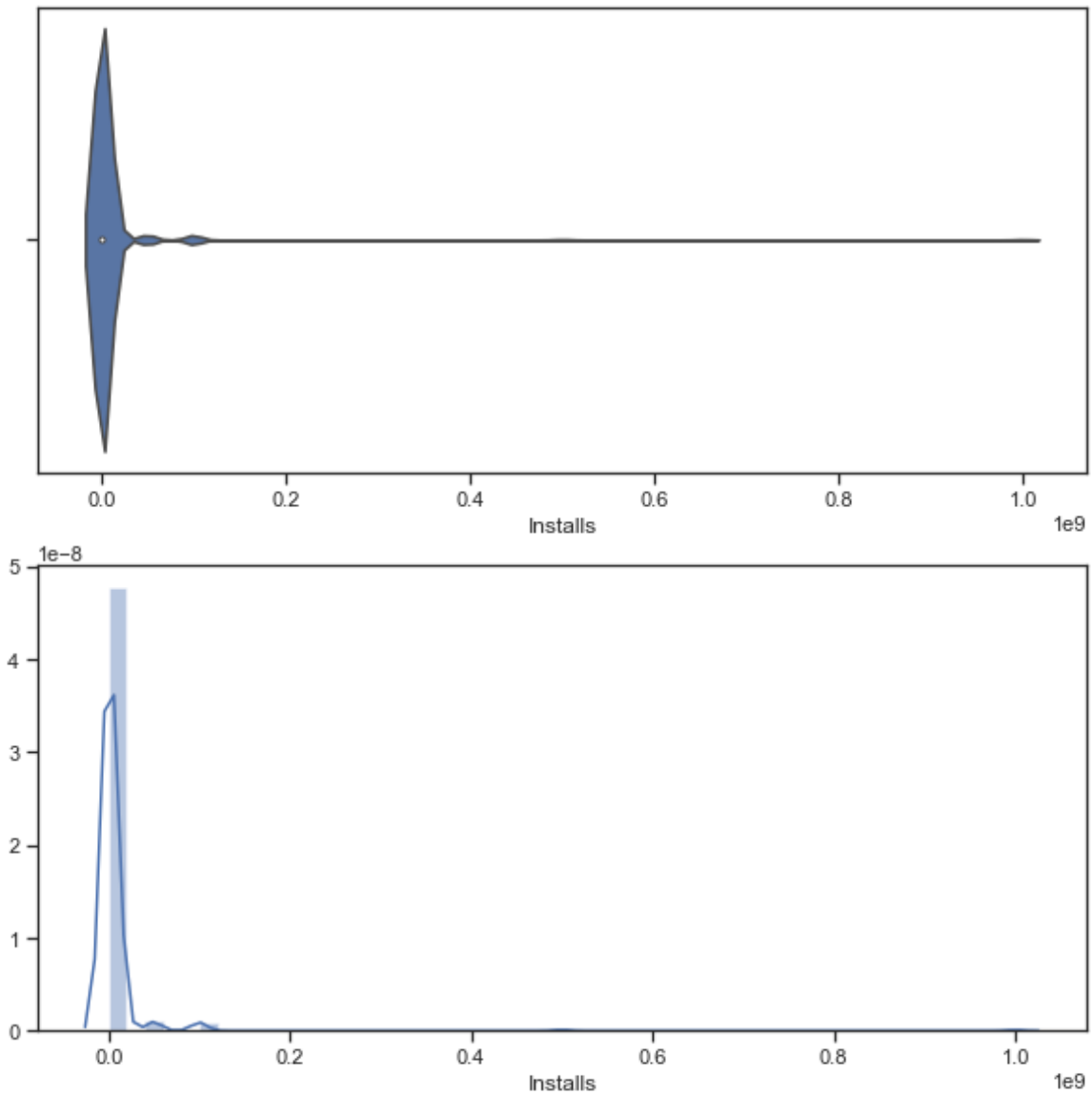
Violin plot

In [27]:

```
# По краям графика отображаются распределения плотности признака Installs
fig, ax = plt.subplots(2, 1, figsize=(10,10))
sns.violinplot(ax=ax[0], x=data['Installs'])
sns.distplot(data['Installs'], ax=ax[1])
```

Out[27]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x12b475eb8>
```

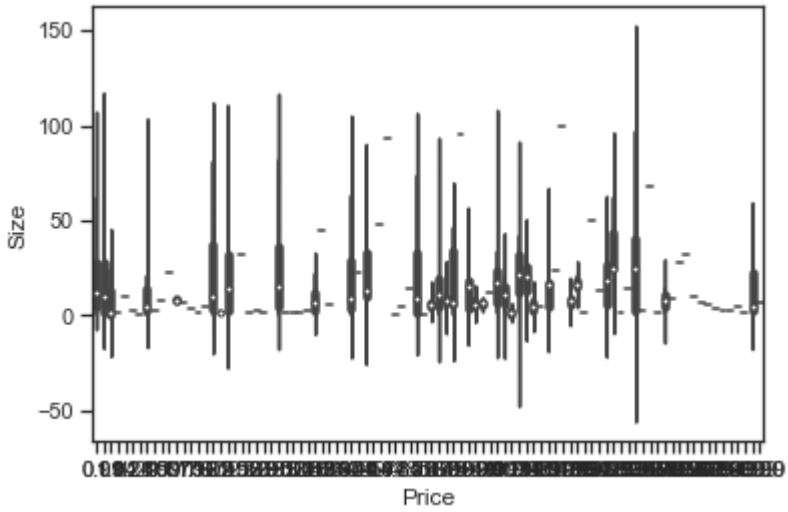


In [28]:

```
# Распределение параметра Size сгруппированные по Price.
sns.violinplot(x='Price', y='Size', data=data)
```

Out[28]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x12b1f8f28>
```



Корреляционный анализ

In [29]:

```
data.corr()
```

Out[29]:

	Reviews	Size	Installs	Price
Reviews	1.000000	0.179321	0.625165	-0.007598
Size	0.179321	1.000000	0.134291	-0.022439
Installs	0.625165	0.134291	1.000000	-0.009405
Price	-0.007598	-0.022439	-0.009405	1.000000

In [30]:

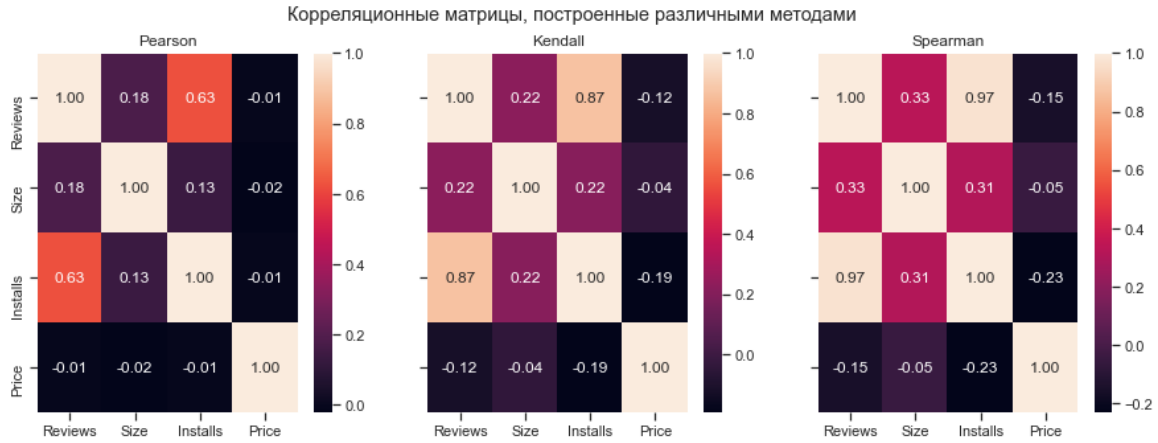
```
data.corr(method='spearman')
```

Out[30]:

	Reviews	Size	Installs	Price
Reviews	1.000000	0.329949	0.967707	-0.150713
Size	0.329949	1.000000	0.310458	-0.049035
Installs	0.967707	0.310458	1.000000	-0.232029
Price	-0.150713	-0.049035	-0.232029	1.000000

In [31]:

```
fig, ax = plt.subplots(1, 3, sharex='col', sharey='row', figsize=(15,5))
sns.heatmap(data.corr(method='pearson'), ax=ax[0], annot=True, fmt='.2f')
sns.heatmap(data.corr(method='kendall'), ax=ax[1], annot=True, fmt='.2f')
sns.heatmap(data.corr(method='spearman'), ax=ax[2], annot=True, fmt='.2f')
fig.suptitle('Корреляционные матрицы, построенные различными методами')
ax[0].title.set_text('Pearson')
ax[1].title.set_text('Kendall')
ax[2].title.set_text('Spearman')
```



Вывод

Корреляционная матрица содержит коэффициенты корреляции между всеми парами признаков.

На основе корреляционной матрицы можно сделать следующие выводы:

1. Целевой признак Price слишком слабо коррелирует с остальными признаками. Больше всего с признаком Size(-0.05 Spearman), что достаточно логично, так как на цену влияет сложность приложения. Данный признак можно оставить в модели.
2. Признаки Installs и Reviews коррелируют друг с другом. Можно оставить в модели один из этих признаков, к примеру, Reviews. Этот признак сильнее коррелирован с целевым признаком.

В ходе выполнения РК1 был проведен разведочный анализ данных приложений с Google Play Store. Были исследованы основные характеристики датасета, а также проведено визуальное исследование данных в результате которого были построены графики: диаграмма рассеяния, гистограммы распределения, joinplot(Комбинация гистограмм и диаграмм рассеивания), парные диаграммы, диаграмма "ящик с усами" и графики violin plot.

Диаграмма рассеивания позволяет построить распределение двух колонок данных и визуально обнаружить наличие зависимости. В данном случае исследовалась взаимосвязь между признаками - Installs и Reviews, чтобы определить влияние числа отзывов на количество установок приложения.

Гистограмма распределения позволяет оценить плотность вероятности распределения данных. При помощи гистограммы было исследовано распределение признака Size(размер приложения). По гистограмме частот можно предположить, что признак описывается законом, близким к нормальному, и имеет наиболее вероятное значение, лежащее в пределах 12-20MB.

Joinplot - комбинация гистограмм и диаграмм рассеивания. С помощью этой гистограммы исследовалась взаимосвязь между признаками - Installs и Reviews. По графику видно, что количество отзывов о приложении влияет на число установок. Но также распространены случаи, когда большое число установок приходится на приложения с небольшим количеством отзывов.

Парные диаграммы представляют комбинацию гистограмм и диаграмм рассеивания для всего набора данных. Вывод содержит множество диаграмм рассеивания и гистограмм распределения по каждой паре признаков. Таким образом парная диаграмма обобщает все ранее построенные графики.

Диаграмма "ящик с усами" показывает одномерное распределение вероятности. Построен график по признаку Size(размер приложения). На графике показаны наблюдаемый минимум - 0MB, максимум - 60MB, нижний квартиль - примерно 5MB, верхний квартиль - 25MB, медиану - 12MB и выбросы - более 60MB.

На графиках violin plot по краям отображаются распределения плотности. При помощи данного вида графиков исследовался признак Installs(число установок). Вместе с гистограммой график показывает, что наибольшее значение вероятоности приходится примерно на 0,1 * 10^6 установок.

На основании построенных графиков можно сделать вывод о том, что пользователи отдают предпочтение наиболее легким приложениям(требующего небольшого объема памяти - Size). Чем ниже цена, тем более чаще приложение скачивают пользователи. Приложения с низкой стоимостью(либо бесплатные) имеют больше отзывов. Одними из самых дорогих приложений являются приложения категории Medical и Family, однако присутствуют выбросы, где цена достаточно высокая для приложений категории Game, Lifestyle, Finance. Также размер приложения влияет на цену, что обусловлено сложностью реализации.

In [ ]:

```

```

In [ ]:

```

```

In [ ]:

```

```