Исследование данных интернет-магазина «Стримчик»

Есть исторические данные по продажах игр, оценки пользователей и экспертов, жанры и платформы. Компании хочет провести анализ данных, найти закономерности и выявить факторы определяющие успешность будущей игры.

Содержание

Описание данных

- Name — название игры
- Platform — платформа
- Year_of_Release — год выпуска
- Genre — жанр игры
- NA_sales — продажи в Северной Америке (миллионы проданных копий)
- EU_sales — продажи в Европе (миллионы проданных копий)
- JP_sales — продажи в Японии (миллионы проданных копий)
- Other_sales — продажи в других странах (миллионы проданных копий)
- Critic_Score — оценка критиков (максимум 100)
- User_Score — оценка пользователей (максимум 10)
- Rating — рейтинг от организации ESRB (англ. Entertainment Software Rating Board). Эта ассоциация определяет рейтинг компьютерных игр и присваивает им подходящую возрастную категорию.

Чтение данных

In [2]:
```
import pandas as pd
import matplotlib.pyplot as plt
from scipy import stats as st
import numpy as np
import seaborn
from IPython.display import display
```

In [3]:
```
# Убрать лимит на изображаемое количество столбцов
pd.options.display.max_columns = None

# Показывать только 5 цйфр после запятой
pd.options.display.precision = 5  # избавиться от научной нотации
```

In [4]:
```
data = pd.read_csv('/datasets/games.csv')
```

In [5]:
```
data.head(10)
```

Out[5]:

	Name	Platform	Year_of_Release	Genre	NA_sales	EU_sales	JP_sales	Othe
0	Wii Sports	Wii	2006.0	Sports	41.36	28.96	3.77	
1	Super Mario Bros.	NES	1985.0	Platform	29.08	3.58	6.81	
2	Mario Kart Wii	Wii	2008.0	Racing	15.68	12.76	3.79	
3	Wii Sports Resort	Wii	2009.0	Sports	15.61	10.93	3.28	
4	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	11.27	8.89	10.22	
5	Tetris	GB	1989.0	Puzzle	23.20	2.26	4.22	
6	New Super Mario Bros.	DS	2006.0	Platform	11.28	9.14	6.50	
7	Wii Play	Wii	2006.0	Misc	13.96	9.18	2.93	
8	New Super Mario Bros. Wii	Wii	2009.0	Platform	14.44	6.94	4.70	
9	Duck Hunt	NES	1984.0	Shooter	26.93	0.63	0.28	

In [6]:
```
data.columns
```

Out[6]:
```
Index(['Name', 'Platform', 'Year_of_Release', 'Genre', 'NA_sales', 'EU_sales',
       'JP_sales', 'Other_sales', 'Critic_Score', 'User_Score', 'Rating'],
      dtype='object')
```

In [7]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16715 entries, 0 to 16714
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Name             16713 non-null  object 
 1   Platform         16715 non-null  object 
 2   Year_of_Release  16446 non-null  float64
 3   Genre            16713 non-null  object 
 4   NA_sales         16715 non-null  float64
 5   EU_sales         16715 non-null  float64
 6   JP_sales         16715 non-null  float64
 7   Other_sales      16715 non-null  float64
 8   Critic_Score     8137 non-null   float64
 9   User_Score       10014 non-null  object 
 10  Rating           9949 non-null   object 
dtypes: float64(6), object(5)
memory usage: 1.4+ MB
```

In [8]:
```
data.describe(include='all')
```

Out[8]:

	Name	Platform	Year_of_Release	Genre	NA_sales	EU_sales	JP_sal
count	16713	16715	16446.00000	16713	16715.00000	16715.00000	16715.000
unique	11559	31	NaN	12	NaN	NaN	Na
top	Need for Speed: Most Wanted	PS2	NaN	Action	NaN	NaN	Na
freq	12	2161	NaN	3369	NaN	NaN	Na
mean	NaN	NaN	2006.48462	NaN	0.26338	0.14506	0.077
std	NaN	NaN	5.87705	NaN	0.81360	0.50334	0.308
min	NaN	NaN	1980.00000	NaN	0.00000	0.00000	0.000
25%	NaN	NaN	2003.00000	NaN	0.00000	0.00000	0.000
50%	NaN	NaN	2007.00000	NaN	0.08000	0.02000	0.000
75%	NaN	NaN	2010.00000	NaN	0.24000	0.11000	0.040
max	NaN	NaN	2016.00000	NaN	41.36000	28.96000	10.220

In [9]:
```
data.shape
```

Out[9]:
```
(16715, 11)
```

Количество строк 16715, в столбцах есть неполные данные, также часть колонок с неверными типами данных. Нужно посмотреть детально.

Предобработка данных

Переименование столбцов

In [10]:
```
data.columns = data.columns.str.lower()
```

In [11]:
```
data.columns
```

Out[11]:
```
Index(['name', 'platform', 'year_of_release', 'genre', 'na_sales', 'eu_sales',
       'jp_sales', 'other_sales', 'critic_score', 'user_score', 'rating'],
      dtype='object')
```

Переименовали столбцы в нижний регистр

Обработка типов

In [12]:
```
data = data.convert_dtypes()
```

In [13]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16715 entries, 0 to 16714
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   name             16713 non-null  string
 1   platform         16715 non-null  string
 2   year_of_release  16446 non-null  Int64
 3   genre            16713 non-null  string
 4   na_sales         16715 non-null  Float64
 5   eu_sales         16715 non-null  Float64
 6   jp_sales         16715 non-null  Float64
 7   other_sales      16715 non-null  Float64
 8   critic_score     8137 non-null   Int64
 9   user_score       10014 non-null  string
 10  rating           9949 non-null   string
dtypes: Float64(4), Int64(2), string(5)
memory usage: 1.5 MB
```

In [14]:
```
for i in ['na_sales','eu_sales','jp_sales','other_sales','critic_score']:
    data[i] = data[i].astype('float64')
```

In [15]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16715 entries, 0 to 16714
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   name             16713 non-null  string
 1   platform         16715 non-null  string
 2   year_of_release  16446 non-null  Int64
 3   genre            16713 non-null  string
 4   na_sales         16715 non-null  float64
 5   eu_sales         16715 non-null  float64
 6   jp_sales         16715 non-null  float64
 7   other_sales      16715 non-null  float64
 8   critic_score     8137 non-null   float64
 9   user_score       10014 non-null  string
 10  rating           9949 non-null   string
dtypes: Int64(1), float64(5), string(5)
memory usage: 1.4 MB
```

Вывод: Изменели тип данных автоматическим методом `convert_dtypes()` `user_score` , содержит уникальное значение tbd, поэтому он преобразовался в строкое значение это надо исправить. Половина данных отсутствуют в рейтингах.

Проверка пропусков

name

In [16]:
```
data.isna().sum()
```

Out[16]:
```
name                 2
platform             0
year_of_release    269
genre                2
na_sales             0
eu_sales             0
jp_sales             0
other_sales          0
critic_score      8578
user_score        6701
rating            6766
dtype: int64
```

In [17]:
```
data.isna().sum()/data.shape[0] * 100
```

Out[17]:
```
name                0.01197
platform            0.00000
year_of_release     1.60933
genre               0.01197
na_sales            0.00000
eu_sales            0.00000
jp_sales            0.00000
other_sales         0.00000
critic_score       51.31917
user_score         40.08974
rating             40.47861
dtype: float64
```

In [18]:
```
data[data['name'].isna()]
```

Out[18]:

	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	other_sales
659	<NA>	GEN	1993	<NA>	1.78	0.53	0.00	0.08
14244	<NA>	GEN	1993	<NA>	0.00	0.00	0.03	0.00

In [19]:
```
data = data.dropna(subset=['name'])
```

In [20]:
```
data[data['name'].isna()]
```

Out[20]:

	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	other_sales	criti

Вывод: Удалили пустые данные в колонке `name`

year

In [21]:
```
data[data['year_of_release'].isna()]
```

Out[21]:

	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales
183	Madden NFL 2004	PS2	<NA>	Sports	4.26	0.26	0.01
377	FIFA Soccer 2004	PS2	<NA>	Sports	0.59	2.36	0.04
456	LEGO Batman: The Videogame	Wii	<NA>	Action	1.80	0.97	0.00
475	wwe Smackdown vs. Raw 2006	PS2	<NA>	Fighting	1.57	1.02	0.00
609	Space Invaders	2600	<NA>	Shooter	2.36	0.14	0.00
...	...	...	...	...	...	...	...
16373	PDC World Championship Darts 2008	PSP	<NA>	Sports	0.01	0.00	0.00
16405	Freaky Flyers	GC	<NA>	Racing	0.01	0.00	0.00
16448	Inversion	PC	<NA>	Shooter	0.01	0.00	0.00
16458	Hakuouki: Shinsengumi Kitan	PS3	<NA>	Adventure	0.01	0.00	0.00
16522	Virtua Quest	GC	<NA>	Role-Playing	0.01	0.00	0.00

269 rows × 11 columns

In [22]:
```
data['year_of_release'].value_counts(dropna=False).sort_index() #провекра на про
```

Out[22]:
```
1980       9
1981      46
1982      36
1983      17
1984      14
1985      14
1986      21
1987      16
1988      15
1989      17
1990      16
1991      41
1992      43
1993      60
1994     121
1995     219
1996     263
1997     289
1998     379
1999     338
2000     350
2001     482
2002     829
2003     775
2004     762
2005     939
2006    1006
2007    1197
2008    1427
2009    1426
2010    1255
2011    1136
2012     653
2013     544
2014     581
2015     606
2016     502
NaN      269
Name: year_of_release, dtype: Int64
```

In [23]:
```
data = data.dropna(subset=['year_of_release'])
```

In [24]:
```
data['year_of_release'].isna().sum()
```

Out[24]:
```
0
```

Вывод: В колонке year_of_release в 269 строк отсутсвуют данные, можно удалить данные, они занимают меньше 2%

user_score

In [25]:
```
data.query('user_score == "tbd"')
```

Out[25]:

	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	o
119	Zumba Fitness	Wii	2010	Sports	3.45	2.59	0.0	
301	Namco Museum: 50th Anniversary	PS2	2005	Misc	2.08	1.35	0.0	
520	Zumba Fitness 2	Wii	2011	Sports	1.51	1.03	0.0	
645	uDraw Studio	Wii	2010	Misc	1.65	0.57	0.0	
718	Just Dance Kids	Wii	2010	Misc	1.52	0.54	0.0	
...	...	...	...	...	...	...	...	...
16695	Planet Monsters	GBA	2001	Action	0.01	0.00	0.0	
16697	Bust-A-Move 3000	GC	2003	Puzzle	0.01	0.00	0.0	
16698	Mega Brain Boost	DS	2008	Puzzle	0.01	0.00	0.0	
16704	Plushees	DS	2008	Simulation	0.01	0.00	0.0	
16706	Men in Black II: Alien Escape	GC	2003	Shooter	0.01	0.00	0.0	

2376 rows × 11 columns

tbd = to be determined, что переводится будет определено, возможно связано с сайтом, откуда получали данные. Предлагаю обнулить `critic_score`, `user_score` все отсутствующие данные

In [26]:
```
data['user_score'] = pd.to_numeric(data['user_score'], errors='coerce')
```

In [27]:
```
data['user_score'].head(3)
```

Out[27]:
```
0    8.0
1    NaN
2    8.3
Name: user_score, dtype: float64
```

rating

In [28]:
```
data['rating'].isna().sum()
```

Out[28]:
```
6676
```

In [29]:
```
data['rating'] = data['rating'].fillna('unknown')
```

Вывод Организации ESRB осуществляет деятельность в США и Канаде с этим связаны пропуски в данных рейтинга, а не с ошибкой заполнения данных

Суммарные продажи

In [30]:
```
data['total_sales'] = data['na_sales'] + data['eu_sales'] + data['jp_sales'] + d
```

In [31]:
```
data['total_sales']
```

Out[31]:
```
0        82.54
1        40.24
2        35.52
3        32.77
4        31.38
         ...
16710     0.01
16711     0.01
16712     0.01
16713     0.01
16714     0.01
Name: total_sales, Length: 16444, dtype: float64
```

Проверка на дубликаты

In [32]:
```
data[data.duplicated()]
```

Out[32]:

name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	other_sales	criti

In [33]:
```
data.nunique()
```

Out[33]:
```
name               11426
platform              31
year_of_release       37
genre                 12
na_sales             401
eu_sales             307
jp_sales             244
other_sales          155
critic_score          81
user_score            95
rating                 9
total_sales         1004
dtype: int64
```

In [34]:
```
data[data[['name', 'platform','year_of_release']].duplicated(keep=False)]
```

Out[34]:

	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	other_sal
604	Madden NFL 13	PS3	2012	Sports	2.11	0.22	0.0	0.
16230	Madden NFL 13	PS3	2012	Sports	0.00	0.01	0.0	0.

In [35]:
```
data = data.drop(data[data[['name', 'platform','year_of_release']].duplicated(ke
```

In [36]:
```
data[data[['name', 'platform','year_of_release']].duplicated(keep=False)]
```

Out[36]:

	index	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	other_sale

In [37]:
```
data.query('name=="Madden NFL 13"')
```

Out[37]:

	index	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	ot
503	507	Madden NFL 13	X360	2012	Sports	2.53	0.15	0.0	
600	604	Madden NFL 13	PS3	2012	Sports	2.11	0.22	0.0	
3933	3986	Madden NFL 13	Wii	2012	Sports	0.47	0.00	0.0	
5800	5887	Madden NFL 13	PSV	2012	Sports	0.28	0.00	0.0	
6956	7066	Madden NFL 13	WiiU	2012	Sports	0.21	0.00	0.0	

Убрали неявный дубликат Madden NFL 13

Промежуточный вывод

- Преобразовали типы данных
- Удалили пустые строки в `name`
- Удалили менее 2% отсутствующих данных `year`
- Дубликаты явные
- Удалили неявный дубликат Madden NFL 13
- Заменили значение *tbd* `, user_score` на NaN
- `rating` заменили пустые строки на `unknown`
- Организации ESRB осуществляет деятельность в США и Канаде с этим связаны пропуски в данных `rating`, а не с ошибкой заполнения данных
- Добавили колонку total_sales сумарное значение

Исследование данных

Выпуск игр в разные годы

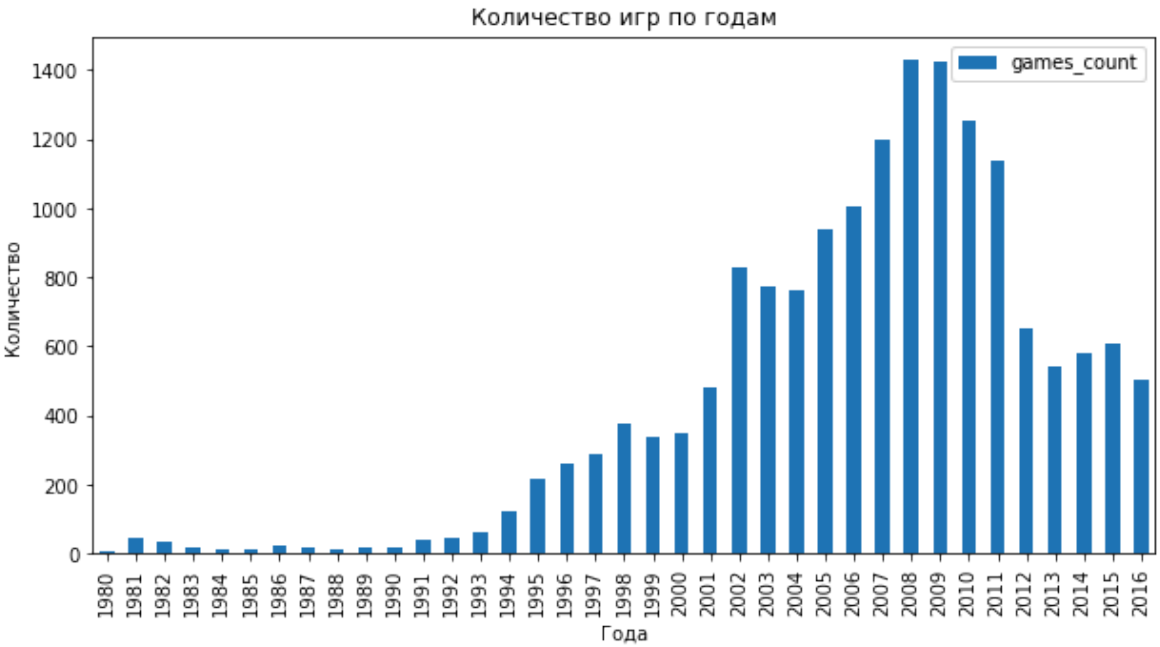
In [39]:
```
games_count = data.groupby(['year_of_release']).agg(games_count=('name', 'count'
games_count
```

Out[39]:

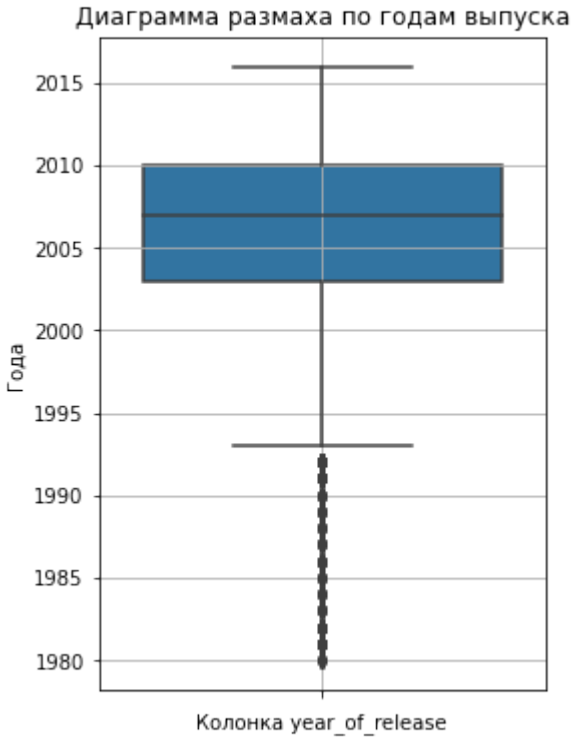
year_of_release	games_count
1980	9
1981	46
1982	36
1983	17
1984	14
1985	14
1986	21
1987	16
1988	15
1989	17
1990	16
1991	41
1992	43
1993	60
1994	121
1995	219
1996	263
1997	289
1998	379
1999	338
2000	350
2001	482
2002	829
2003	775
2004	762
2005	939
2006	1006
2007	1197
2008	1427
2009	1426
2010	1255
2011	1136

year_of_release	games_count
2012	652
2013	544
2014	581
2015	606
2016	502

In [40]:
```
games_count.plot.bar(figsize=(10, 5))
plt.title('Количество игр по годам')
plt.xlabel('Года')
plt.ylabel('Количество')
plt.show()
```



In [41]:
```
plt.figure(figsize=(4, 6))
seaborn.boxplot(y='year_of_release', data=data)
plt.grid(True)
plt.xlabel('Колонка year_of_release')
plt.ylabel('Года')
plt.title('Диаграмма размаха по годам выпуска')
plt.show()
```



In [42]:
```
games_count.describe()
```

Out[42]:

	games_count
count	37.00000
mean	444.40541
std	451.59153
min	9.00000
25%	36.00000
50%	338.00000
75%	762.00000
max	1427.00000

In [43]:
```
good_stat =  data.query('year_of_release >=1995')
```

In [44]:
```
good_stat['year_of_release'].unique()
```

Out[44]:
```
<IntegerArray>
[2006, 2008, 2009, 1996, 2005, 1999, 2007, 2010, 2013, 2004, 2002, 2001, 2011,
 1998, 2015, 2012, 2014, 1997, 2016, 2003, 2000, 1995]
Length: 22, dtype: Int64
```

Вывод Как мы видим, производство игр росло экспоненциально и достигло пиков 2008, 2009 годов, затем резко упало, скорее всего связано с экономическим кризисом 2008 года. Имеем небольшое число выпущенных игр до 1995 можно от них избавится.

Продажи по платформам

In [45]:
```
platform_ = good_stat.groupby('platform').agg({'total_sales':'sum'}).sort_values
platform_.head(5)
```

Out[45]:

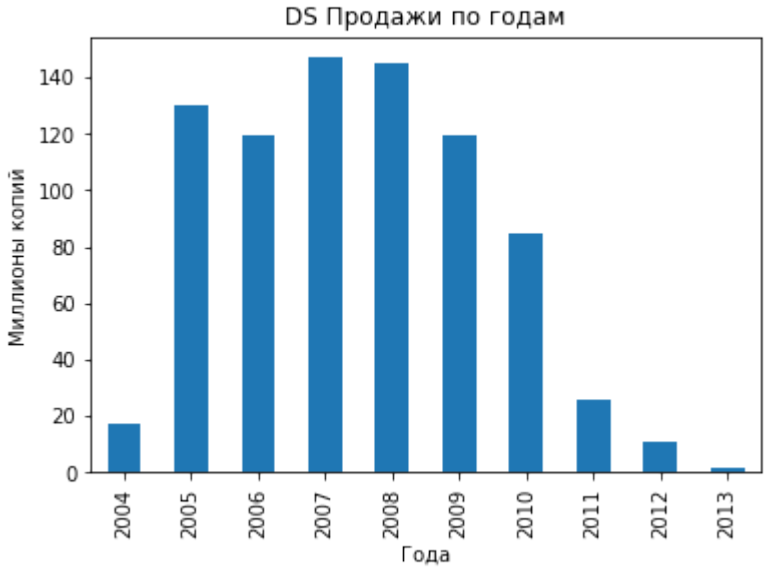
platform	total_sales
PS2	1233.56
X360	961.24
PS3	931.33
Wii	891.18
DS	802.76

In [46]:
```
for i in platform_.index[:5]:
    try:
        pl = good_stat.query('platform == @i')
        pl.groupby(['year_of_release'])['total_sales'].sum().plot.bar()
        plt.title(f'{i} Продажи по годам')
        plt.xlabel('Года')
        plt.ylabel('Миллионы копий')
        plt.show()
    except:
        print('smt wront', i)
```


X360 Продажи по годам
Миллионы копий
0
20
40
60
80
100
120
140
160
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
Года


PS3 Продажи по годам
Миллионы копий
0
20
40
60
80
100
120
140
160
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
Года


Wii Продажи по годам
Миллионы копий
0
25
50
75
100
125
150
175
200
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
Года



In [47]:
```
age_ = good_stat.groupby('platform')['year_of_release'].agg(['min', 'max'])
(age_[min]-age_[max] + 1).median()
```

Out[47]: 7.0

Вывод

- Наибольшие суммарные продажи среди платформ

1. PS2 1233.56
2. X360 961.24
3. PS3 931.34
4. Wii 891.18
5. DS 802.76

- Пики продаж по годам

1. PS2 2004г 200млн
2. X360 2010г 160млн
3. PS3 2011 160млн
4. Wii 2009 200млн
5. DS 2007 140млн

- В среднем характерный срок жизни платформы 7 лет

Актуальный период

In [49]:
```
actual_data = good_stat.query('year_of_release >=2013')
```

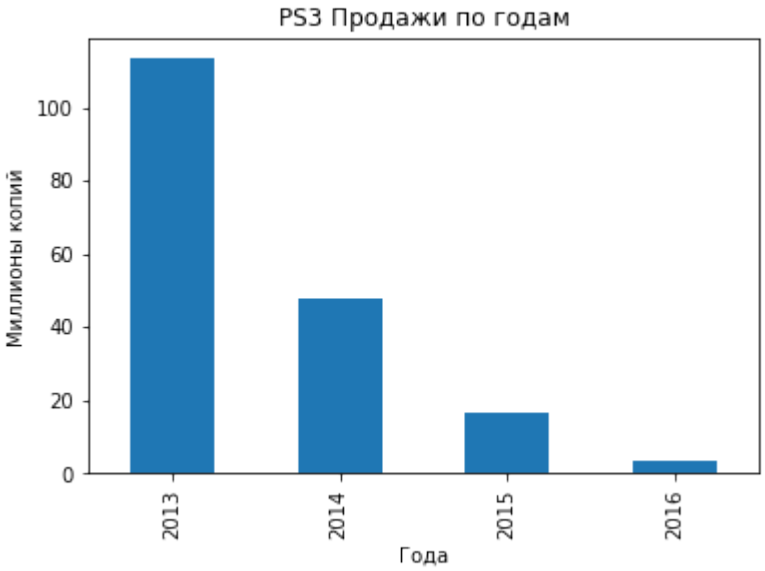
In [50]:
```
actual_data['year_of_release'].value_counts()
```

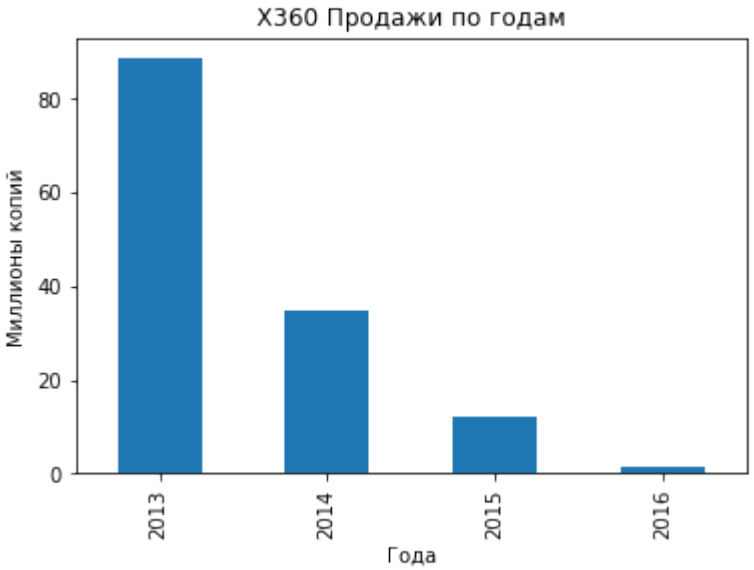
Out[50]:
```
2015    606
2014    581
2013    544
2016    502
Name: year_of_release, dtype: Int64
```

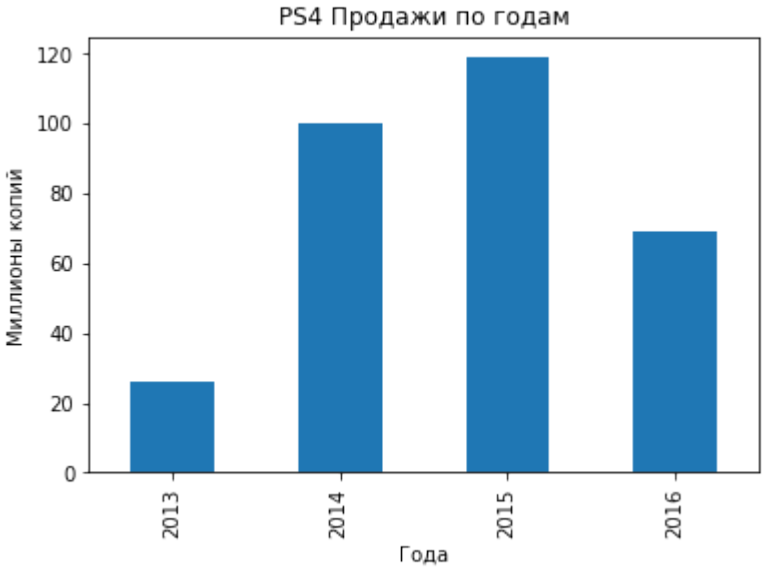
Вывод В ходе исследования жизни платформ разумно взять актуальный период 3 года включительно, именно эти данные помогут построить прогноз на следующий год

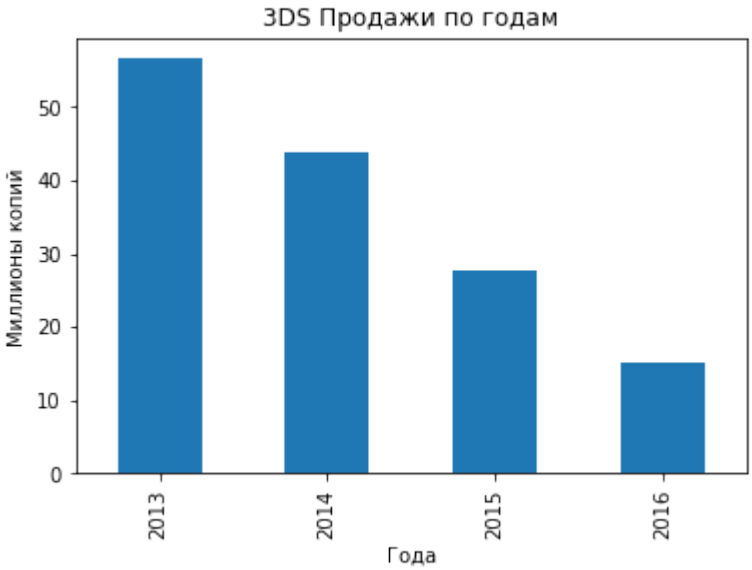
Актуальные платформы по продажам

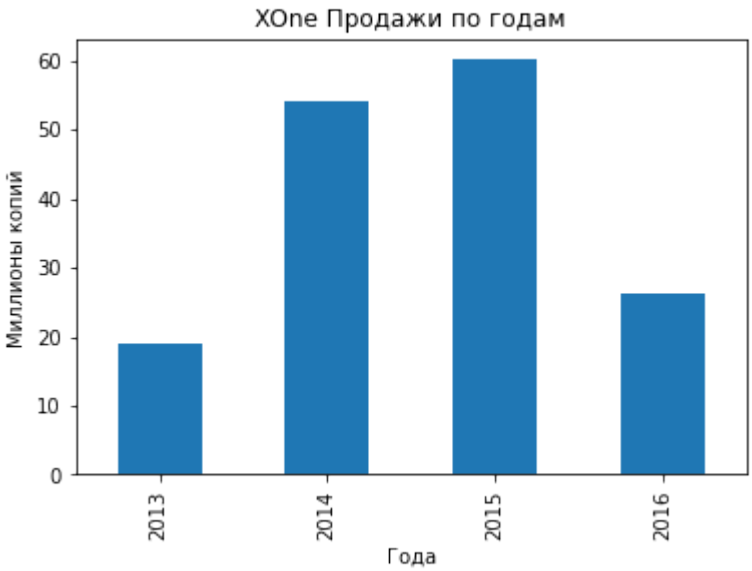
In [51]:
```
for i in actual_data['platform'].unique():
    try:
        pl = actual_data.query('platform == @i')
        pl.groupby(['year_of_release'])['total_sales'].sum().plot.bar()
        plt.title(f'{i} Продажи по годам')
        plt.xlabel('Года')
        plt.ylabel('Миллионы копий')
        plt.show()
    except:
        print('smt wrong', {i})
```

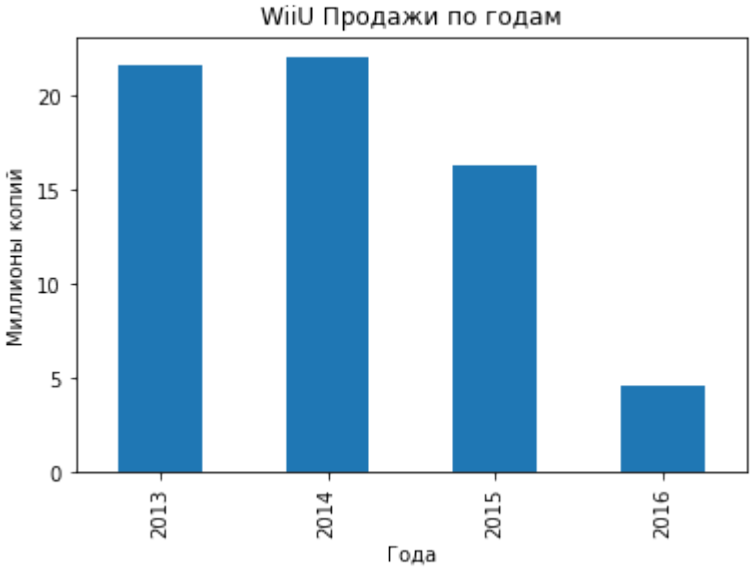


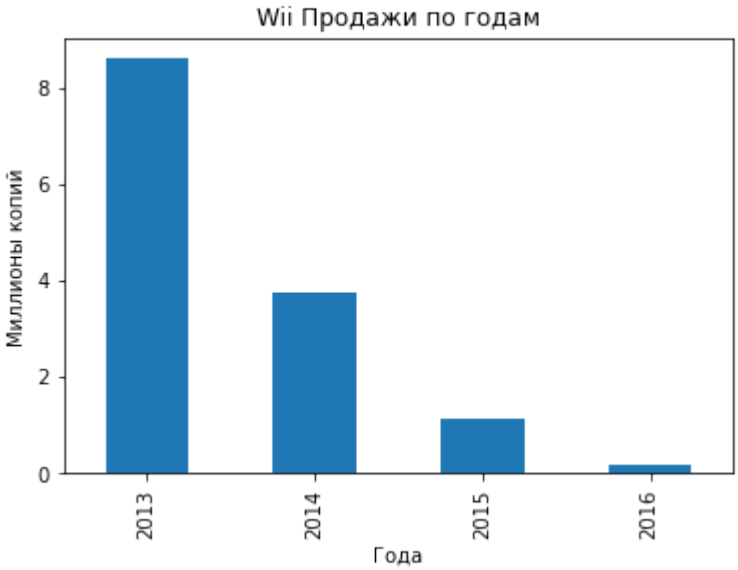


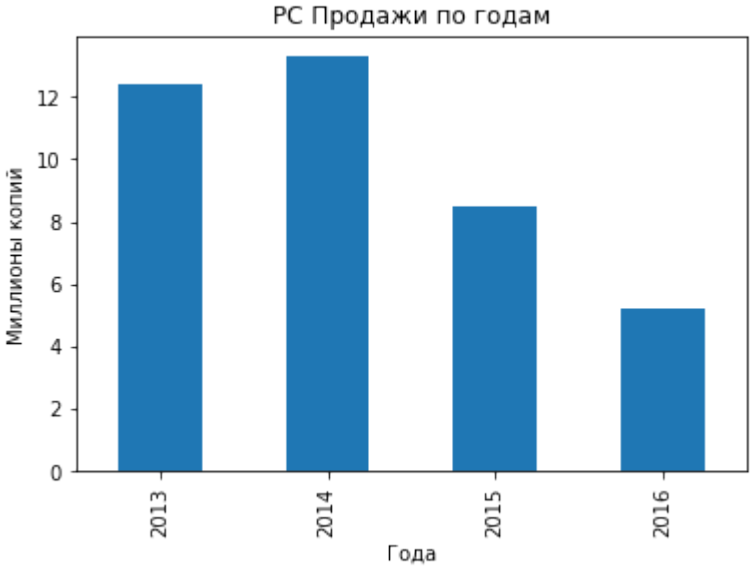

PS4 Продажи по годам
Миллионы копий
0
20
40
60
80
100
120
2013
2014
2015
2016
Года

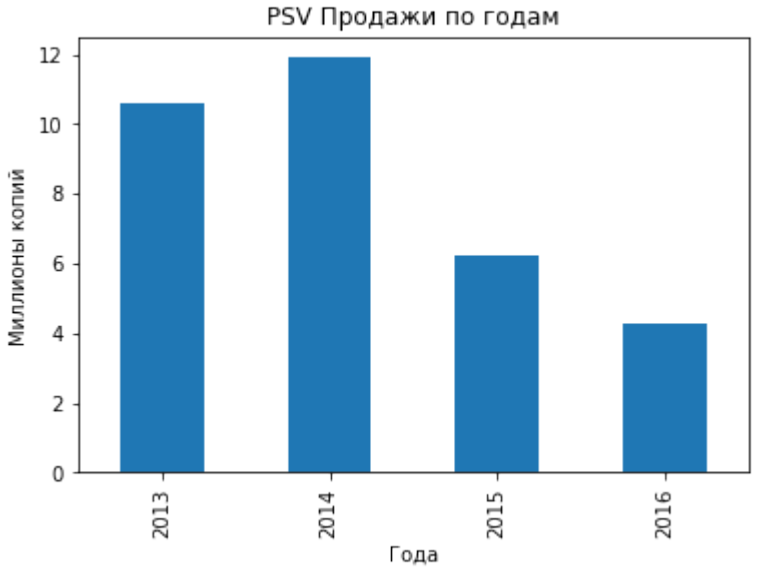

3DS Продажи по годам
Миллионы копий
0
10
20
30
40
50
2013
2014
2015
2016
Года

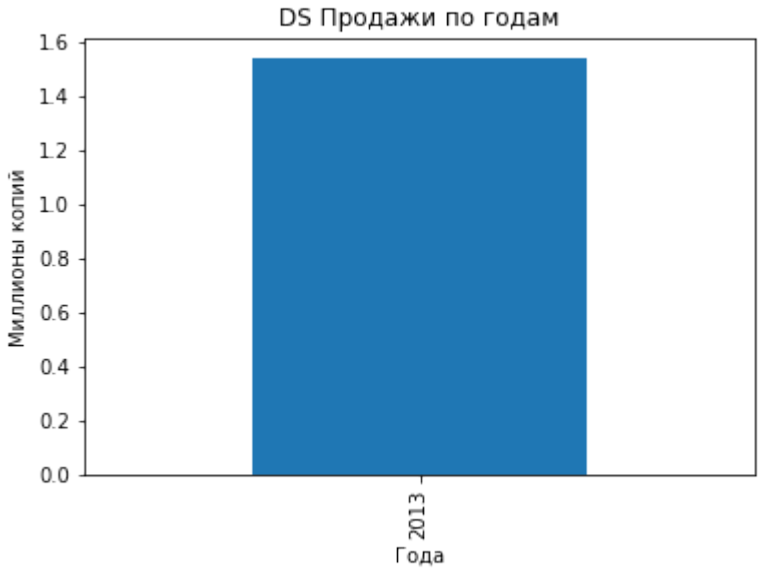

XOne Продажи по годам
Миллионы копий
0
10
20
30
40
50
60
2013
2014
2015
2016
Года


WiiU Продажи по годам
Миллионы копий
0
5
10
15
20
2013
2014
2015
2016
Года


Wii Продажи по годам
Миллионы копий
0
2
4
6
8
2013
2014
2015
2016
Года


PC Продажи по годам
Миллионы копий
0
2
4
6
8
10
12
2013
2014
2015
2016
Года





In [52]:
```
pivot = actual_data.pivot_table(index=['year_of_release'],columns='platform',val
pivot.style.background_gradient(
    subset=['PS4', '3DS', 'XOne', 'WiiU', 'PS3', 'X360', 'PC', 'Wii', 'PSV', 'PS
)
```

Out[52]:

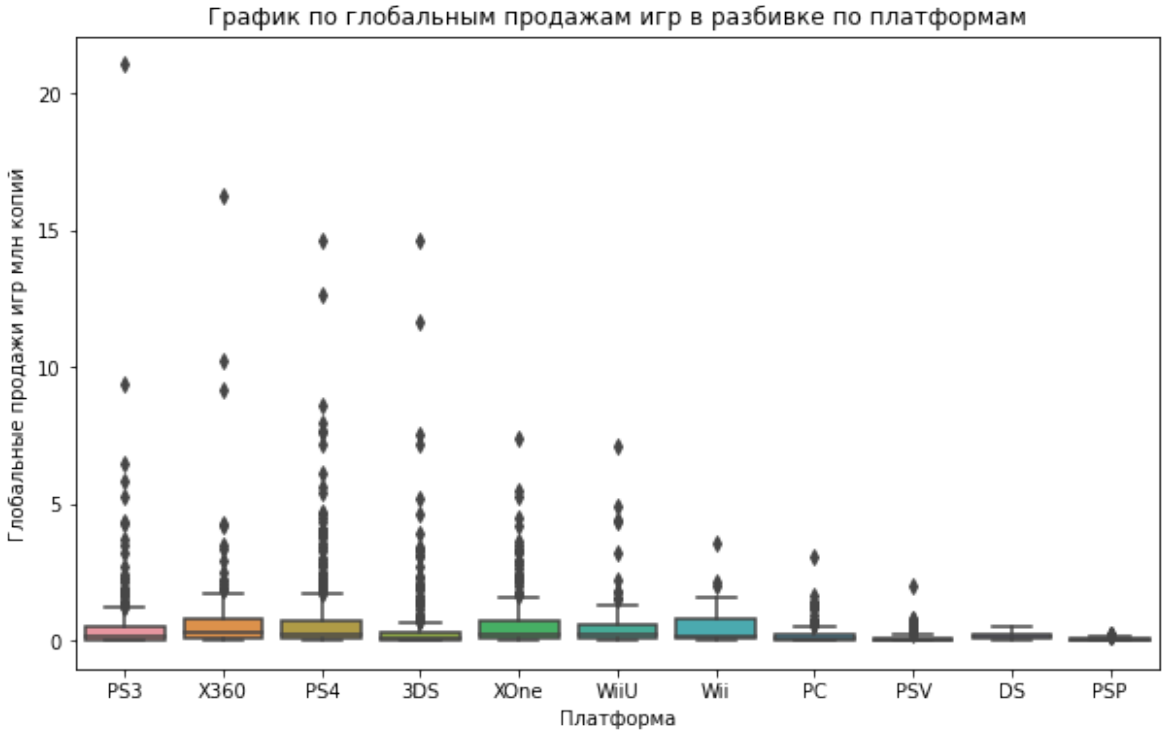
platform	3DS	DS	PC	PS3	PS4	PSP	PSV	
year_of_release								
2013	56.57000	1.54000	12.38000	113.25000	25.99000	3.14000	10.59000	8.
2014	43.76000	0.00000	13.28000	47.76000	100.00000	0.24000	11.90000	3.
2015	27.78000	0.00000	8.52000	16.82000	118.90000	0.12000	6.25000	1.
2016	15.14000	0.00000	5.25000	3.60000	69.25000	0.00000	4.25000	0.

Вывод

- Потенциальные лидерами по продажам за последние годы PS4 **118.9** млн и XOne **60,14** млн копий
- Тенденция к снижению 3DS, PC, PSV, WiiU, X360, PS3
- Поддержка psp прекращены в 2014

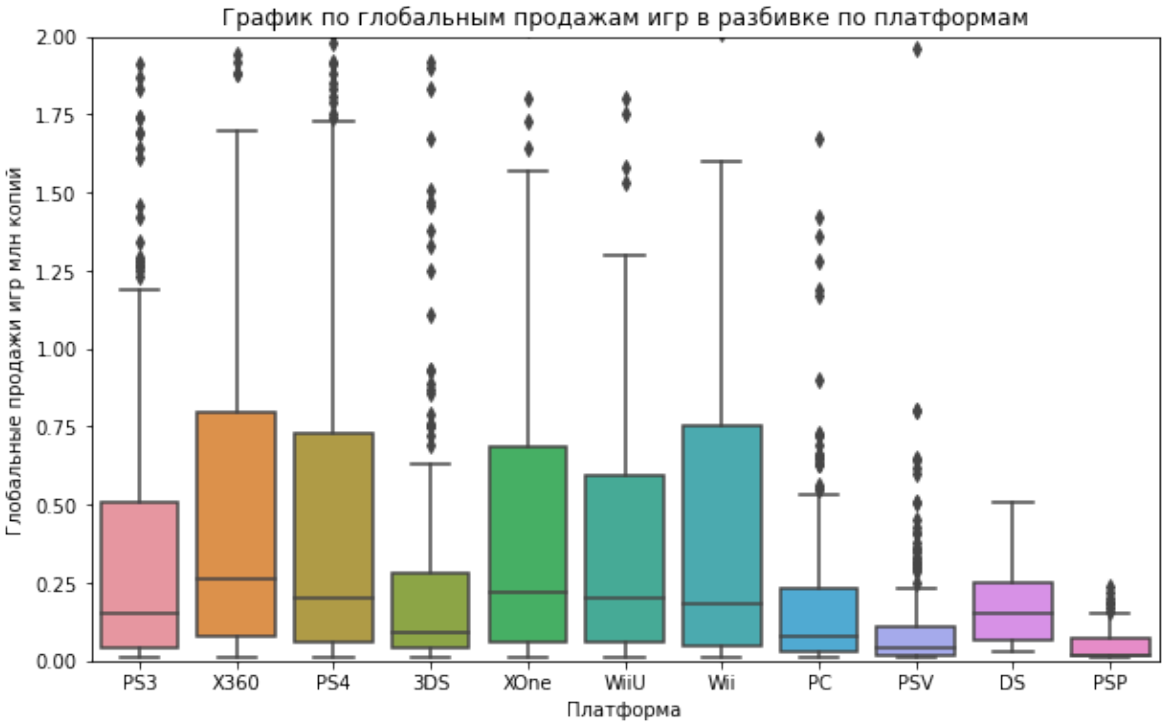
Диаграмма размаха по платформам

In [53]:
```
plt.figure(figsize=(10, 6))
seaborn.boxplot(x='platform', y='total_sales', data=actual_data)

plt.xlabel('Платформа')
plt.ylabel('Глобальные продажи игр млн копий')
plt.title('График по глобальным продажам игр в разбивке по платформам')
plt.show()
```



In [54]:
```
plt.figure(figsize=(10, 6))
seaborn.boxplot(x='platform', y='total_sales', data=actual_data)
plt.ylim(0,2)
plt.xlabel('Платформа')
plt.ylabel('Глобальные продажи игр млн копий')
```

```
plt.title('График по глобальным продажам игр в разбивке по платформам')
plt.show()
```



In [55]:
```
pivot.describe()
```

Out[55]:

platform	3DS	DS	PC	PS3	PS4	PSP	PSV	Wii	
count	4.00000	4.000	4.0000	4.00000	4.00000	4.00000	4.00000	4.00000	4
mean	35.81250	0.385	9.8575	45.35750	78.53500	0.87500	8.24750	3.41500	16
std	18.12834	0.770	3.7011	48.89868	40.56792	1.51318	3.59621	3.76536	8
min	15.14000	0.000	5.2500	3.60000	25.99000	0.00000	4.25000	0.18000	4
25%	24.62000	0.000	7.7025	13.51500	58.43500	0.09000	5.75000	0.90000	13
50%	35.77000	0.000	10.4500	32.29000	84.62500	0.18000	8.42000	2.44500	19
75%	46.96250	0.385	12.6050	64.13250	104.72500	0.96500	10.91750	4.96000	21
max	56.57000	1.540	13.2800	113.25000	118.90000	3.14000	11.90000	8.59000	22

Вывод

- Медианные лидеры по продажам игр у PS4 XOne, с медианной 100 и 54 млн соответственно
- Максимальное количество проданной игры у PS4 118.9 млн
- Среднее значение у PS4 96.05 млн, XOne 46.78 млн
- Медианное значение значение млн копий за последние 3 годы
- 3DS = 27.78
- PC = 8.5
- PS3 = 16.82
- PS4 = 100

- PSV = 6.25
- Wii = 1.14
- WiiU = 16.35
- X360 = 11.96
- XOne = 54.07

Влияние отзывов на продажи платформы PS4

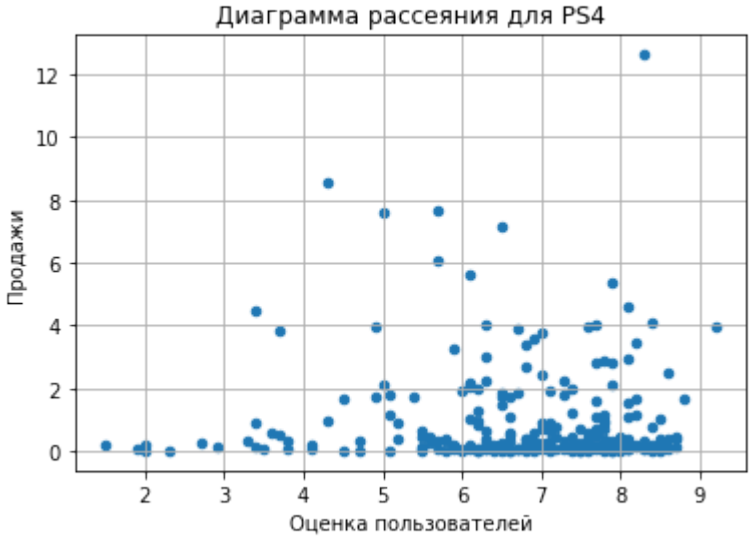
In [56]:
```
platform_ps4 = actual_data.copy()
platform_ps4 = actual_data[actual_data['platform']=='PS4']
```

In [57]:
```
platform_ps4.isna().sum()
```

Out[57]:
```
index                0
name                 0
platform             0
year_of_release      0
genre                0
na_sales             0
eu_sales             0
jp_sales             0
other_sales          0
critic_score       140
user_score         135
rating               0
total_sales          0
dtype: int64
```

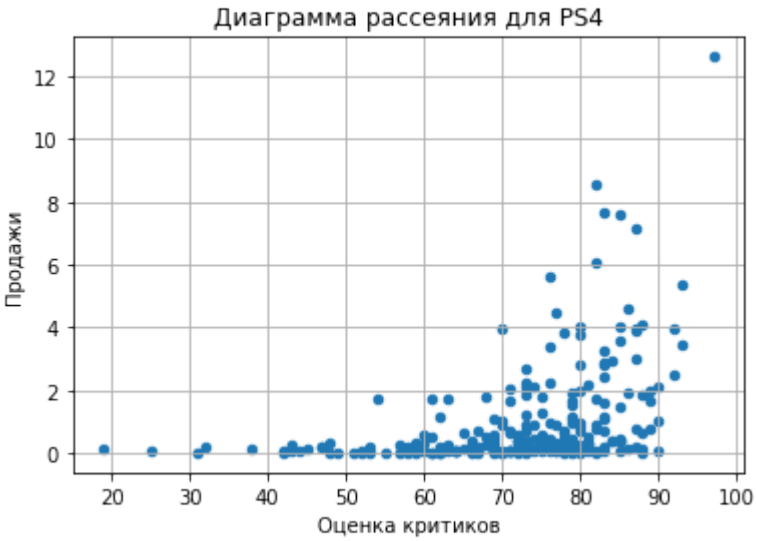
In [58]:
```
platform_ps4 = platform_ps4.dropna()
```

In [59]:
```
platform_ps4.plot(kind='scatter', x='user_score', y='total_sales')
plt.title(f'Диаграмма рассеяния для PS4')
plt.xlabel('Оценка пользователей')
plt.ylabel('Продажи')
plt.grid(True)
plt.show()
```



In [60]:
```
platform_ps4.plot(kind='scatter', x='critic_score', y='total_sales')
plt.title(f'Диаграмма рассеяния для PS4')
```

```
plt.xlabel('Оценка критиков')
plt.ylabel('Продажи')
plt.grid(True)
plt.show()
```



In [61]:
```
platform_ps4['total_sales'] = platform_ps4['total_sales'].astype('float64')
platform_ps4['critic_score'] = platform_ps4['critic_score'].astype('float64')
platform_ps4.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 249 entries, 42 to 16258
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   index            249 non-null    int64
 1   name             249 non-null    string
 2   platform         249 non-null    string
 3   year_of_release  249 non-null    Int64
 4   genre            249 non-null    string
 5   na_sales         249 non-null    float64
 6   eu_sales         249 non-null    float64
 7   jp_sales         249 non-null    float64
 8   other_sales      249 non-null    float64
 9   critic_score     249 non-null    float64
 10  user_score       249 non-null    float64
 11  rating           249 non-null    string
 12  total_sales      249 non-null    float64
dtypes: Int64(1), float64(7), int64(1), string(4)
memory usage: 27.5 KB
```

In [62]:
```
display(platform_ps4['critic_score'].corr(platform_ps4['total_sales']))
display(platform_ps4['user_score'].corr(platform_ps4['total_sales']))
```

```
0.40589480145836687
-0.03362497596528878
```

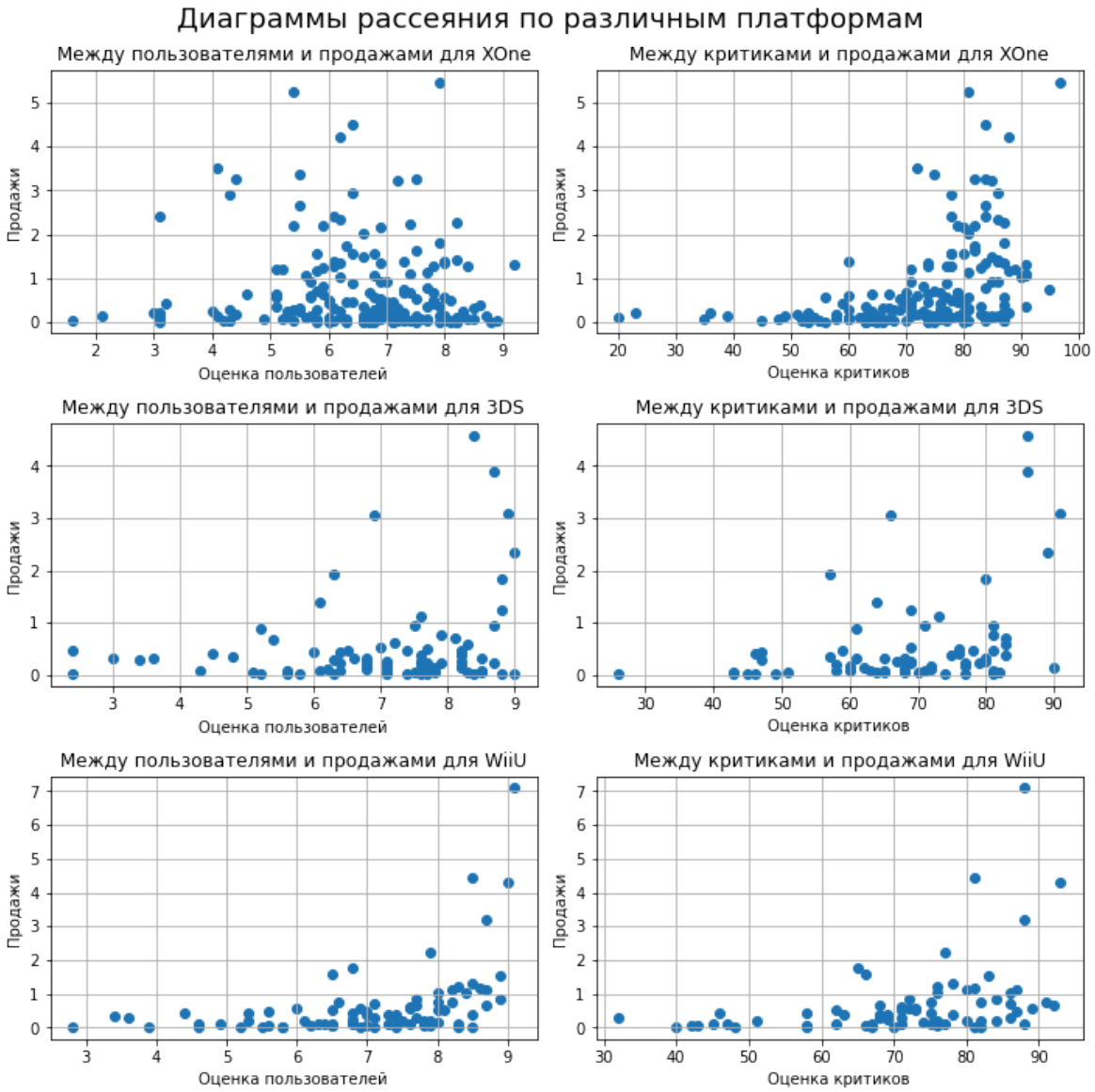
Вывод Провели анализ корреляции влияние отзывов пользователей и критиков на продажи у платформы PS3.

- Присутствует средняя корреляция между отывами критиков и продажами
- Отсутствует корреляция между отзывами пользователей и продажами

Возможно разница связана с тем, что критикам предоставляется ранний доступ и они охватывают больше аудиторию

Соотнести выводы с продажами игр на других платформах.

In [63]:
```
fig, axes = plt.subplots(3, 2, figsize=(10, 10))
pls = ['XOne','3DS','WiiU']
for i,pl in enumerate(pls):
    platform_ = actual_data.query('platform == @pl')

    axes[i,0].scatter(x='user_score', y='total_sales',data=platform_)
    axes[i,1].scatter(x='critic_score', y='total_sales',data=platform_)

    axes[i,0].set_title(f'Между пользователями и продажами для {pl}')
    axes[i,1].set_title(f'Между критиками и продажами для {pl}')

    axes[i,0].set_xlabel(f'Оценка пользователей')
    axes[i,0].set_ylabel(f'Продажи')

    axes[i,1].set_xlabel(f'Оценка критиков')
    axes[i,1].set_ylabel(f'Продажи')

    axes[i,0].grid(True)
    axes[i,1].grid(True)

plt.suptitle("Диаграммы рассеяния по различным платформам", fontsize=18)
plt.tight_layout()
plt.show()
```



In [64]:
```
for i in pls:
    platform_ = actual_data.query('platform == @i')
    cr = platform_['critic_score'].corr(platform_['total_sales'])
    display(f'Корреляция продаж к оценкам критиков {i}, {cr}')
for i in pls:
    platform_ = actual_data.query('platform == @i')
    us = platform_['user_score'].corr(platform_['total_sales'])
    display(f'Корреляция продаж к оценкам пользователей {i}, {us}')
```

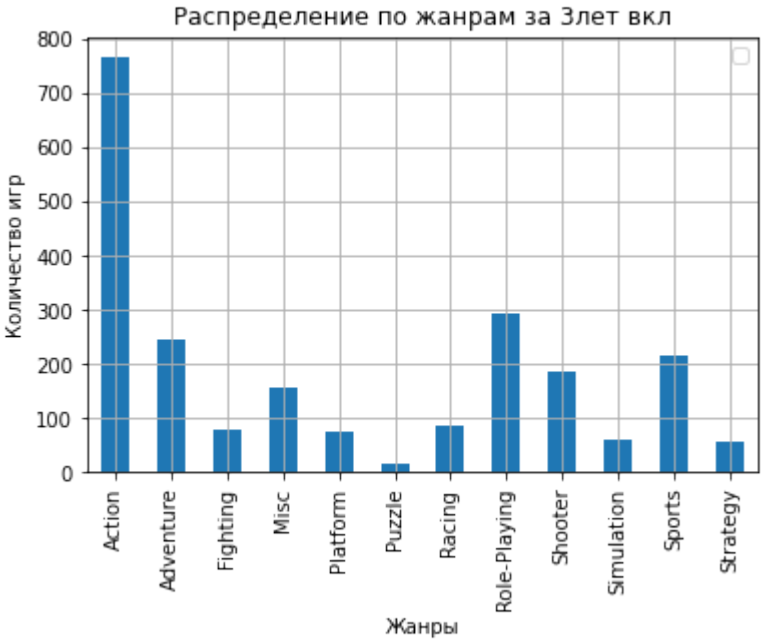
```
'Корреляция продаж к оценкам критиков XOne, 0.4169983280084017'
'Корреляция продаж к оценкам критиков 3DS, 0.3570566142288103'
'Корреляция продаж к оценкам критиков WiiU, 0.3764149065423912'
'Корреляция продаж к оценкам пользователей XOne, -0.06892505328279414'
'Корреляция продаж к оценкам пользователей 3DS, 0.24150411773563016'
'Корреляция продаж к оценкам пользователей WiiU, 0.4193304819266187'
```

Вывод:

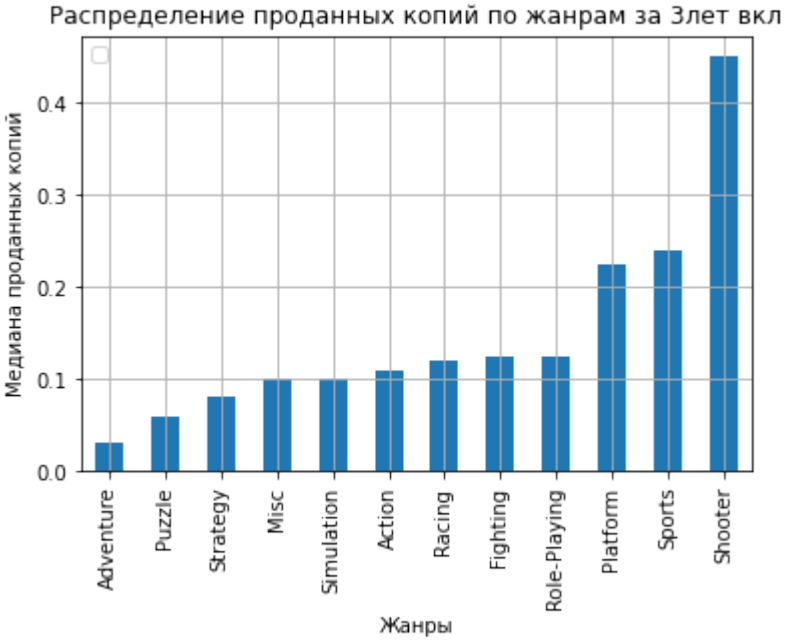
- Связь между отзывам критиков и продажами, коррелируют с популярными платформами PS4,XOne,WiiU,3DS. Люди присматривают к отзывам критиков
- На платформа WiiU, 3DS есть средняя связь между продажами и оценкой пользователей, но не коррелируют между XOne и PS4. Возможно, это связано откуда мы получали информацию или спецификой платформы

Распределение игр по жанрам

In [65]:
```
actual_data.groupby('genre').agg({'name':'count'}).plot.bar()
plt.title('Распределение по жанрам за 3лет вкл')
plt.ylabel('Количество игр')
plt.xlabel('Жанры')
plt.legend('')
plt.grid(True)
plt.show()
```



In [82]:
```
actual_data.groupby('genre').agg({'total_sales':'median'}).sort_values(by='total
plt.title('Распределение проданных копий по жанрам за 3лет вкл')
plt.ylabel('Медиана проданных копий')
plt.xlabel('Жанры')
plt.legend('')
plt.grid(True)
plt.show()
```



Вывод

- Наибольшее количество созданных и проданных игр за 3 года включительно принадлежит жанрам Action
- Adventure, RPG, Misc, Sports в среднем вышло одинаковое количество
- Хорошие медианные продажи показывают жанры
- Shooter количество проданных копий 500т
- Platform, Sports проданных копий 250т
- Остальные жанры иммеют около 100т
- Самые низкие продажи принадлежат жанрам Adventure, Puzzle, Strategy
- Выделяется жанр Adventure, где коэффициент количество выпущенных на проданных намного ниже остальных жанров

Промежуточный вывод

- Количество выпускаемых росло экспоненциально и достигло пиков 2008, 2009 годов, затем резко упало, скорее всего связано с экономическим кризисом 2008 года. Имеем небольшое число выпущенных игр до 1995 можно от них избавится.
- Наибольшие суммарные продажи среди платформ
- PS2 1233.56
- X360 961.24
- PS3 931.34
- Пики продаж по годам
- PS2 2004г 200млн
- X360 2010г 160млн
- PS3 2011 160млн
- В среднем характерный срок жизни платформы 7 лет
- Потенциальные лидерами по продажам за последние годы PS4 118.9 млн и XOne 60,14 млн копий
- Тенденция к снижению 3DS, PC, PSV, WiiU, X360, PS3
- В ходе исследования жизни платформ разумно взять актуальный период **3 года** включительно, именно эти данные помогут построить прогноз на следующий год
- Медианные лидеры по продажам игр у PS4 XOne, с медианной 100 и 54 млн соответственно
- Максимальное количество проданной игры у PS4 118.9 млн
- Среднее значение у PS4 96.05 млн, XOne 46.78 млн
- Медианное значение значение млн копий за последние 3 годы
- 3DS = 27.78
- PC = 8.5
- PS3 = 16.82
- PS4 = 100
- PSV = 6.25
- Wii = 1.14
- WiiU = 16.35

- X360 = 11.96
- XOne = 54.07
- У DC, PSP, Wii есть тайтлы, которые имеют сильный выброс
- Провели анализ корреляции влияние отзывов пользователей и критиков на продажи у платформ.
- Связь между отзывам критиков и продажами, коррелируют с популярными платформами PS4,XOne,WiiU,3DS. Люди присматривают к отзывам критиков
- На платформа WiiU, 3DS есть средняя связь между продажами и оценкой пользователей, но не коррелируют между XOne и PS4. Возможно, это связано откуда мы получали информацию.
- Наибольшее количество созданных и проданных игр за 3 года включительно принадлежит жанрам Action
- Adventure, RPG, Misc, Sports в среднем вышло одинаковое количество
- Хорошие медианные продажи показывают жанры
- Shooter количество проданных копий 500т
- Platform, Sports проданных копий 250т
- Остальные жанры иммеют около 100т
- Самые низкие продажи принадлежат жанрам Adventure, Puzzle, Strategy
- Выделяется жанр Adventure, где коэффициент количество выпущенных на проданных намного ниже остальных жанров

Портрет пользователя каждого региона

In [67]:
```
def portrait(region):
    print('Топ 5 платформ в регионе \n',actual_data.groupby('platform').agg({reg
          .sort_values(by=region ,ascending=False)
          .head(5))
    print()
    print('Топ 5 жанров в регионе \n',actual_data.groupby('genre').agg({region:'
          .sort_values(by=region ,ascending=False)
          .head(5))
    print()
    k = actual_data.copy()
    k['rating'] = pd.factorize(k['rating'])[0]
    print('Корреляция между рейтингом и регионом',k['rating'].corr(k[region]))
```

In [68]:
```
portrait('na_sales')
```

```
Топ 5 платформ в регионе
          na_sales
platform
PS4         108.74
XOne         93.12
X360         81.66
PS3          63.50
3DS          38.20

Топ 5 жанров в регионе
          na_sales
genre
Shooter      0.200
Platform     0.090
Sports       0.080
Fighting     0.045
Racing       0.030

Корреляция между рейтингом и регионом -0.05887207432616984
```

In [69]:
```
portrait('eu_sales')
```

```
Топ 5 платформ в регионе
          eu_sales
platform
PS4         141.09
PS3          67.81
XOne         51.59
X360         42.52
3DS          30.96

Топ 5 жанров в регионе
             eu_sales
genre
Shooter         0.190
Platform        0.080
Racing          0.060
Sports          0.050
Simulation      0.035

Корреляция между рейтингом и регионом -0.06189288443818195
```

In [70]:
```
portrait('jp_sales')
```

```
Топ 5 платформ в регионе
          jp_sales
platform
3DS          67.81
PS3          23.35
PSV          18.59
PS4          15.96
WiiU         10.88

Топ 5 жанров в регионе
              jp_sales
genre
Role-Playing      0.05
Fighting          0.03
Misc              0.02
Puzzle            0.02
Action            0.01

Корреляция между рейтингом и регионом -0.03934934136624959
```

In [71]:
```
fig, axes = plt.subplots(1, 3, figsize=(15, 5))
sal = ['na_sales', 'eu_sales', 'jp_sales']
titles = ['NA', 'EU', 'JP']
for i, region in enumerate(sal):
    r = actual_data.groupby('platform').agg({region:'sum'})\
        .sort_values(by=region ,ascending=False)\
        .head(5)
    r[region] = r[region]/r[region].sum()*100
    r[region].plot(kind='pie', ax=axes[i], title = titles[i],autopct='%1.1f%%',
    axes[i].set_ylabel('')

fig.suptitle('Топ 5 продаж платформ', fontsize=16)
plt.tight_layout()
plt.show()
```

In [72]:
```
fig, axes = plt.subplots(1, 3, figsize=(15, 5))
sal = ['na_sales', 'eu_sales', 'jp_sales']
titles = ['NA', 'EU', 'JP']
for i, region in enumerate(sal):
    r = actual_data.groupby('genre').agg({region:'sum'})\
        .sort_values(by=region ,ascending=False)
    top_5 = r.copy()
    top_5 = top_5.head(5)
    others = r.tail(len(r) - 5)
    top_5.loc['others'] = others[region].sum()
    top_5[region] = top_5[region]/top_5[region].sum()*100
    top_5[region].plot(kind='pie', ax=axes[i], title = titles[i], autopct='%1.1f
```

```
    axes[i].set_ylabel('')

fig.suptitle('Топ 5 продаж жанров', fontsize=16)
plt.tight_layout()
plt.show()
```

In [73]:
```
fig, axes = plt.subplots(1, 3, figsize=(15, 5))
sal = ['na_sales', 'eu_sales', 'jp_sales']
titles = ['NA', 'EU', 'JP']
for i, region in enumerate(sal):
    r = actual_data.groupby('rating').agg({region:'sum'})\
        .sort_values(by=region ,ascending=False)
    r[region] = r[region]/r[region].sum()*100
    r[region].plot(kind='pie', ax=axes[i], title = titles[i],autopct='%1.1f%%',
    axes[i].set_ylabel('')

fig.suptitle('Доля продаж по рейтингам', fontsize=16)
plt.tight_layout()
plt.show()
```

Вывод

1. Топ 5 платформ по регионам

- Наибольшую долю занимает PS4 в Европе и Америке
- В Японии большая доля принадлежит 3DS
- Второй по популярности в Америке XOne, в Европе и Японии PS3
- Разница связана с тем, что на PS4 тайтлы выходят качественее
- 3DS владеет японская компания Нинтендо, поэтому такая высокая доля продаж в Японии

2. Топ 5 жанров по регионам

- Вкусы у Америки и Европы совпадают жанрово, различаются только rpg и misc
- В Японии больше предпочитают rpg
- Возможно разница связана с тем, что крупный тайтлы больше популярны в западных странах, чем в Японии, так же связано с рынком самих платформ, rgp чаще выходят и более популярны там на 3DS

3. Наибольшая доля продаж в Европе и Америке принадлежит Mature, затем E

- Так как ESRB Американская ассоциация, по Японии отсутствуют данные

Проверка гипотез

Средние пользовательские рейтинги

Гипотеза H0: Средние пользовательские рейтинги платформ Xbox One и PC одинаковые

Альтернативная H1: Средние пользовательские рейтинги платформ Xbox One разные

In [74]:
```
user_x1 = actual_data[actual_data['platform']=='XOne']['user_score'].dropna()
user_pc = actual_data[actual_data['platform']=='PC']['user_score'].dropna()
alpha = 0.05
```

In [75]:
```
# так как мы сравниваем две разные выборки используем метод ttest_ind
result = st.ttest_ind(user_x1,user_pc)
print(f'p-value: {result.pvalue}')
```

```
p-value: 0.14012658403611647
```

In [76]:
```
if result.pvalue < alpha:
    print('Отвергаем нулевую гипотезу')
else:
    print('Нет оснований отвергнуть нулевую гипотезу')
```

```
Нет оснований отвергнуть нулевую гипотезу
```

In [77]:
```
display(user_x1.mean())
display(user_pc.mean())
```

```
6.521428571428572
6.2696774193548395
```

Вывод: Значение pvalue 0.14 Нет оснований отвергнуть нулевую гипотезу, что средний пользовательский рейтинг одинаков.

Средний пользовательский рейтинг выборки XOne = 6.5

Средний пользовательский рейтинг выборки PC = 6.3

Выбор статистического критерия Для проверки этих гипотез мы можем использовать `t-тест` для независимых выборок (если у нас есть два независимых набора данных, например, рейтинги для двух платформ). Этот тест позволяет

проверить, различаются ли средние значения двух групп (в данном случае — пользовательских рейтингов для разных платформ).

Средние пользовательские рейтинги жанров Action

Гипотез H0: Средние пользовательские рейтинги жанров Action (англ. «действие», экшен-игры) и Sports (англ. «спортивные соревнования») одинаковые.

Альтернативня H1: Средние пользовательские рейтинги разные

In [78]:
```
action_g = actual_data[actual_data['genre']=='Action']['user_score'].dropna()
sports_g = actual_data[actual_data['genre']=='Sports']['user_score'].dropna()
alpha = 0.05
```

In [79]:
```
# так как мы сравниваем две разные выборки используем метод ttest_ind
result = st.ttest_ind(user_x1,user_pc)
print(f'p-value: {result.pvalue}')
```

```
p-value: 0.14012658403611647
```

In [80]:
```
if result.pvalue < alpha:
    print('Отвергаем нулевую гипотезу')
else:
    print('Нет оснований отвергнуть нулевую гипотезу')
```

```
Нет оснований отвергнуть нулевую гипотезу
```

In [81]:
```
display(action_g.mean())
display(sports_g.mean())
```

```
6.837532133676092
5.238124999999999
```

Вывод: Нет оснований отвергнуть нулевую гипотезу рейтинги жанров Action и Sports `pvalue 0.9` значимо одинаково Средний пользовательский рейтинг выборки Action = 6.8

Средний пользовательский рейтинг выборки Sports = 5.2

Выбор статистического критерия Для проверки этих гипотез мы можем использовать `t-тест` для независимых выборок (если у нас есть два независимых набора данных, например, рейтинги для двух жанров). Этот тест позволяет проверить, различаются ли средние значения двух групп (в данном случае — пользовательских рейтингов для разных жанров).

Формирование гипотез основывается на предложении, что разницы нет. Это стандартная практика при статистическом исследовании выборок. Альтернативную гипотезу взяли такую, чтобы проверит различие между выборками. Использовали параметр двусторонней гипотезы, чтобы проверить выходит ли значения дальше уровня значимости

Итоговый вывод

Предобработка данных

- Переименовали столбцы в нижний регистр
- Убрали `tbd` в колонке `user_score`
- Удалил пустые значения в `name` и `year`
- Преобразовали типы данных
- Дубликаты явные
- Удалили неявный дубликат Madden NFL 13
- `rating` заменили пустые строки на `unknown`
- Организации ESRB осуществляет деятельность в США и Канаде с этим связаны пропуски в данных rating, а не с ошибкой заполнения данных
- Добавили колонку `total_sales` сумарное продажи

Исследование данных

- Количество выпускаемых росло экспоненциально и достигло пиков 2008, 2009 годов, затем резко упало, скорее всего связано с экономическим кризисом 2008 года.

- Наибольшие суммарные продажи среди платформ | PS2| X360| PS3| | :----: | :---: |:------:| | 1233.56 | 961.24 | 931.34 |

- Пики продаж по годам | PS2 | X360 | PS3| | :----: | :---: |:------:| | 2004г | 2010г | 2011 | | 200млн | 160млн| 160млн |

- **В среднем характерный срок жизни платформы 7 лет**

- Потенциальные лидерами по продажам за последние годы PS4 118.9 млн и XOne 60,14 млн копий

- Тенденция к снижению 3DS, PC, PSV, WiiU, X360, PS3

- В ходе исследования жизни платформ разумно взять актуальный период **`3 года`** включительно, именно эти данные помогут построить прогноз на следующий год

- Медианные лидеры по продажам игр у PS4 XOne, с медианной 100 и 54 млн соответственно

- Максимальное количество проданной игры у PS4 118.9 млн

- Среднее значение у PS4 96.05 млн, XOne 46.78 млн

- Медианное значение значение млн копий за последние 3 годы

- 3DS = 27.78

- PC = 8.5

- PS3 = 16.82

- PS4 = 100
- PSV = 6.25
- Wii = 1.14
- WiiU = 16.35
- X360 = 11.96
- XOne = 54.07
- У DC, PSP, Wii есть тайтлы, которые имеют сильный выброс
- Провели анализ корреляции влияние отзывов пользователей и критиков на продажи у платформ.
- Связь между отзывам критиков и продажами, коррелируют с популярными платформами PS4,XOne,WiiU,3DS. Люди присматривают к отзывам критиков
- На платформа WiiU, 3DS есть средняя связь между продажами и оценкой пользователей, но не коррелируют между XOne и PS4. Возможно, это связано откуда мы получали информацию или спецификой платформы
- Наибольшее количество созданных и проданных игр за 3 года включительно принадлежит жанрам Action
- Adventure, RPG, Misc, Sports в среднем вышло одинаковое количество
- Хорошие медианные продажи показывают жанры
- Shooter количество проданных копий 500т
- Platform, Sports проданных копий 250т
- Остальные жанры иммеют около 100т
- Самые низкие продажи принадлежат жанрам Adventure, Puzzle, Strategy
- Выделяется жанр Adventure, где коэффициент количество выпущенных на проданных намного ниже остальных жанров

Портрет пользователя каждого региона

1. Топ 5 платформ по регионам

- Наибольшую долю занимает PS4 в Европе и Америке
- В Японии большая доля принадлежит 3DS
- Второй по популярности в Америке XOne, в Европе и Японии PS3
- Разница связана с тем, что на PS4 тайтлы выходят качественее
- 3DS владеет японская компания Нинтендо, поэтому такая высокая доля продаж в Японии

2. Топ 5 жанров по регионам

- Вкусы у Америки и Европы совпадают жанрово, различаются только rpg и misc
- В Японии больше предпочитают rpg
- Возможно разница связана с тем, что крупный тайтлы больше популярны в западных странах, чем в Японии, так же связано с рынком самих платформ, rgp чаще выходят и более популярны там на 3DS

3. Наибольшая доля продаж в Европе и Америке принадлежит Mature, затем E

- Так как ESRB Американская ассоциация, по Японии отсутствуют данные

Проверка гипотез

1. Гипотеза: пользовательские рейтинги платформ Xbox One и PC одинаковые Вывод Значение pvalue 0.14 Нет оснований отвергнуть нулевую гипотезу, средний пользовательский рейтинг возможно одинаков.

   - Средний пользовательский рейтинг выборки XOne = 6.51
   - Средний пользовательский рейтинг выборки PC = 6.3

   Выбор статистического критерия Для проверки этих гипотез мы можем использовать `t-тест` для независимых выборок (если у нас есть два независимых набора данных, например, рейтинги для двух платформ). Этот тест позволяет проверить, различаются ли средние значения двух групп (в данном случае — пользовательских рейтингов для разных платформ).

2. Гипотеза о различиях рейтинги жанров Action и Sports Вывод: Нет оснований отвергнуть нулевую гипотезу, рейтинги жанров Action и Sports `pvalue 0.9` значимо одинаково

   - Средний пользовательский рейтинг выборки Action = 6.8
   - Средний пользовательский рейтинг выборки Sports = 5.2

   Выбор статистического критерия Для проверки этих гипотез мы можем использовать `t-тест` для независимых выборок (если у нас есть два независимых набора данных, например, рейтинги для двух жанров). Этот тест позволяет проверить, различаются ли средние значения двух групп (в данном случае — пользовательских рейтингов для разных жанров).

Общий вывод

В данном иследовании провели ретроспективный анализ исторических данных по продажах игр, оценки пользователей и экспертов, жанры и платформы. Нашли закономерности и выявить факторы определяющие успешность будущей игры. Выявили, что время жизни платформ около 3 лет и следует планировать около этого горизонта времени. Мнение критиков влияют на продажи игр. В много общего между Америкой и Европой в жанрах, но разные в предпочтении платформ. Наиоблее актуальные платформы 3DS и PS4, XOne