# Исследование закономерностей, определяющих популярность компьютерных игр.

## 1  Описание проекта.

Я работаю в интернет-магазине «Стримчик», который продаёт по всему миру компьютерные игры. Из открытых источников доступны исторические данные о продажах игр, оценки пользователей и экспертов, жанры и платформы (например, Xbox или PlayStation). Нужно выявить определяющие успешность игры закономерности. Это позволит сделать ставку на потенциально популярный продукт и спланировать рекламные кампании. Перед нами данные до 2016 года. Представим, что сейчас декабрь 2016 г., и мы планируете кампанию на 2017-й. Нужно отработать принцип работы с данными. Неважно, прогнозируем ли мы продажи на 2017 год по данным 2016-го или же 2027-й — по данным 2026 года. В наборе данных попадается аббревиатура ESRB (Entertainment Software Rating Board) — это ассоциация, определяющая возрастной рейтинг компьютерных игр. ESRB оценивает игровой контент и присваивает ему подходящую возрастную категорию, например, «Для взрослых», «Для детей младшего возраста» или «Для подростков».

## 2  Описание данных.

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.

Данные за 2016 год могут быть неполными.

## 3  Гипотезы к проверке.

• Средние пользовательские рейтинги платформ Xbox One и PC одинаковые;

• Средние пользовательские рейтинги жанров Action (англ. «действие», экшен-игры) и Sports (англ. «спортивные соревнования») разные.

# Подготовка данных.

Ввод [1]:

**import** pandas **as** pd

**import** matplotlib.pyplot **as** plt

**import** numpy **as** np

**from** scipy.stats **import** ttest\_ind

Загрузка данных.

Ввод [2]:

df **=** pd.read\_csv('/datasets/games.csv')

Ввод [3]:

df.head(10)

Out[3]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Wii Sports | Wii | 2006.0 | Sports | 41.36 | 28.96 | 3.77 | 8.45 | 76.0 | 8 | E |
| **1** | Super Mario Bros. | NES | 1985.0 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | NaN | NaN | NaN |
| **2** | Mario Kart Wii | Wii | 2008.0 | Racing | 15.68 | 12.76 | 3.79 | 3.29 | 82.0 | 8.3 | E |
| **3** | Wii Sports Resort | Wii | 2009.0 | Sports | 15.61 | 10.93 | 3.28 | 2.95 | 80.0 | 8 | E |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996.0 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | NaN |
| **5** | Tetris | GB | 1989.0 | Puzzle | 23.20 | 2.26 | 4.22 | 0.58 | NaN | NaN | NaN |
| **6** | New Super Mario Bros. | DS | 2006.0 | Platform | 11.28 | 9.14 | 6.50 | 2.88 | 89.0 | 8.5 | E |
| **7** | Wii Play | Wii | 2006.0 | Misc | 13.96 | 9.18 | 2.93 | 2.84 | 58.0 | 6.6 | E |
| **8** | New Super Mario Bros. Wii | Wii | 2009.0 | Platform | 14.44 | 6.94 | 4.70 | 2.24 | 87.0 | 8.4 | E |
| **9** | Duck Hunt | NES | 1984.0 | Shooter | 26.93 | 0.63 | 0.28 | 0.47 | NaN | NaN | NaN |

Ввод [4]:

df.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

Ввод [5]:

df.isna().sum()

Out[5]:

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

Ввод [6]:

nan\_rows **=** df[df['Name'].isna()]

nan\_rows

Out[6]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **659** | NaN | GEN | 1993.0 | NaN | 1.78 | 0.53 | 0.00 | 0.08 | NaN | NaN | NaN |
| **14244** | NaN | GEN | 1993.0 | NaN | 0.00 | 0.00 | 0.03 | 0.00 | NaN | NaN | NaN |

Вижу всего пару пропусков в столбце name - возможно не внесены данные по невнимательности ли утеряны при трансформации данных/их пересылке.

Ввод [7]:

nan\_rows **=** df[df['Genre'].isna()]

nan\_rows

Out[7]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **659** | NaN | GEN | 1993.0 | NaN | 1.78 | 0.53 | 0.00 | 0.08 | NaN | NaN | NaN |
| **14244** | NaN | GEN | 1993.0 | NaN | 0.00 | 0.00 | 0.03 | 0.00 | NaN | NaN | NaN |

Вижу всего пару пропусков в столбце name - возможно не внесены данные по невнимательности ли утеряны при трансформации данных/их пересылке.

Ввод [8]:

nan\_rows **=** df[df['Year\_of\_Release'].isna()]

nan\_rows

Out[8]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **183** | Madden NFL 2004 | PS2 | NaN | Sports | 4.26 | 0.26 | 0.01 | 0.71 | 94.0 | 8.5 | E |
| **377** | FIFA Soccer 2004 | PS2 | NaN | Sports | 0.59 | 2.36 | 0.04 | 0.51 | 84.0 | 6.4 | E |
| **456** | LEGO Batman: The Videogame | Wii | NaN | Action | 1.80 | 0.97 | 0.00 | 0.29 | 74.0 | 7.9 | E10+ |
| **475** | wwe Smackdown vs. Raw 2006 | PS2 | NaN | Fighting | 1.57 | 1.02 | 0.00 | 0.41 | NaN | NaN | NaN |
| **609** | Space Invaders | 2600 | NaN | Shooter | 2.36 | 0.14 | 0.00 | 0.03 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16373** | PDC World Championship Darts 2008 | PSP | NaN | Sports | 0.01 | 0.00 | 0.00 | 0.00 | 43.0 | tbd | E10+ |
| **16405** | Freaky Flyers | GC | NaN | Racing | 0.01 | 0.00 | 0.00 | 0.00 | 69.0 | 6.5 | T |
| **16448** | Inversion | PC | NaN | Shooter | 0.01 | 0.00 | 0.00 | 0.00 | 59.0 | 6.7 | M |
| **16458** | Hakuouki: Shinsengumi Kitan | PS3 | NaN | Adventure | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16522** | Virtua Quest | GC | NaN | Role-Playing | 0.01 | 0.00 | 0.00 | 0.00 | 55.0 | 5.5 | T |

269 rows × 11 columns

Пропуски в данных по году выпуска игры могут иметь ту же этимологию, что и в предыдущих рассмотреных мною столбцах, дополнительно хочу сказать, что данные могли быть утеряны при техническом сбое (то есть существует не только человеческий фактор).

Ввод [9]:

nan\_rows **=** df[df['Critic\_Score'].isna()]

nan\_rows

Out[9]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | Super Mario Bros. | NES | 1985.0 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | NaN | NaN | NaN |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996.0 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | NaN |
| **5** | Tetris | GB | 1989.0 | Puzzle | 23.20 | 2.26 | 4.22 | 0.58 | NaN | NaN | NaN |
| **9** | Duck Hunt | NES | 1984.0 | Shooter | 26.93 | 0.63 | 0.28 | 0.47 | NaN | NaN | NaN |
| **10** | Nintendogs | DS | 2005.0 | Simulation | 9.05 | 10.95 | 1.93 | 2.74 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.00 | 0.01 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |

8578 rows × 11 columns

Пропуски в столбце с оценками критиков могут быть по всем вышеописанным причинам, а так же потому что, например игры вышедшие до 1992 г не оценивались по данным интернет-источников.

Ввод [10]:

nan\_rows **=** df[df['User\_Score'].isna()]

nan\_rows

Out[10]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | Super Mario Bros. | NES | 1985.0 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | NaN | NaN | NaN |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996.0 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | NaN |
| **5** | Tetris | GB | 1989.0 | Puzzle | 23.20 | 2.26 | 4.22 | 0.58 | NaN | NaN | NaN |
| **9** | Duck Hunt | NES | 1984.0 | Shooter | 26.93 | 0.63 | 0.28 | 0.47 | NaN | NaN | NaN |
| **10** | Nintendogs | DS | 2005.0 | Simulation | 9.05 | 10.95 | 1.93 | 2.74 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.00 | 0.01 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |

6701 rows × 11 columns

Ввод [11]:

nan\_rows **=** df[df['Rating'].isna()]

nan\_rows

Out[11]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | Super Mario Bros. | NES | 1985.0 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | NaN | NaN | NaN |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996.0 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | NaN |
| **5** | Tetris | GB | 1989.0 | Puzzle | 23.20 | 2.26 | 4.22 | 0.58 | NaN | NaN | NaN |
| **9** | Duck Hunt | NES | 1984.0 | Shooter | 26.93 | 0.63 | 0.28 | 0.47 | NaN | NaN | NaN |
| **10** | Nintendogs | DS | 2005.0 | Simulation | 9.05 | 10.95 | 1.93 | 2.74 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.00 | 0.01 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |

6766 rows × 11 columns

Дополнительно оценка игроков может отсутствовать и не заполняться, тк возможно пользователь приобрел игру, но еще не играл в нее и не смог на момент заполнения данных как-то оценить свои впечатления.

Ввод [12]:

old\_games **=** df[df['Year\_of\_Release'] **<=** 1992]

old\_games

Out[12]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | Super Mario Bros. | NES | 1985.0 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | NaN | NaN | NaN |
| **5** | Tetris | GB | 1989.0 | Puzzle | 23.20 | 2.26 | 4.22 | 0.58 | NaN | NaN | NaN |
| **9** | Duck Hunt | NES | 1984.0 | Shooter | 26.93 | 0.63 | 0.28 | 0.47 | NaN | NaN | NaN |
| **18** | Super Mario World | SNES | 1990.0 | Platform | 12.78 | 3.75 | 3.54 | 0.55 | NaN | NaN | NaN |
| **21** | Super Mario Land | GB | 1989.0 | Platform | 10.83 | 2.71 | 4.18 | 0.42 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **14470** | Alter Ego | PC | 1985.0 | Simulation | 0.00 | 0.03 | 0.00 | 0.01 | 59.0 | 5.8 | T |
| **14610** | Doom | PC | 1992.0 | Shooter | 0.02 | 0.00 | 0.00 | 0.00 | 85.0 | 8.2 | M |
| **14621** | SimCity | PC | 1988.0 | Simulation | 0.00 | 0.02 | 0.00 | 0.01 | 64.0 | 2.2 | E10+ |
| **15692** | Empire Deluxe | PC | 1992.0 | Strategy | 0.02 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **15957** | Strongest Tokyo University Shogi DS | DS | 1985.0 | Action | 0.00 | 0.00 | 0.02 | 0.00 | NaN | NaN | NaN |

305 rows × 11 columns

Думаю пропуски в данных с годом релиза игры имеют больше технический характер, тк обычно данная информация по продукту чаще всего известна.

Ввод [13]:

low\_sales\_games **=** df[df['NA\_sales'] **==** 0.0]

low\_sales\_games

Out[13]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **215** | Monster Hunter Freedom 3 | PSP | 2010.0 | Role-Playing | 0.0 | 0.00 | 4.87 | 0.00 | NaN | NaN | NaN |
| **340** | Friend Collection | DS | 2009.0 | Misc | 0.0 | 0.00 | 3.67 | 0.00 | NaN | NaN | NaN |
| **385** | Monster Hunter 4 | 3DS | 2013.0 | Role-Playing | 0.0 | 0.00 | 3.44 | 0.00 | NaN | NaN | NaN |
| **403** | English Training: Have Fun Improving Your Skills! | DS | 2006.0 | Misc | 0.0 | 0.99 | 2.32 | 0.02 | NaN | NaN | NaN |
| **428** | Dragon Quest VI: Maboroshi no Daichi | SNES | 1995.0 | Role-Playing | 0.0 | 0.00 | 3.19 | 0.00 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16709** | SCORE International Baja 1000: The Official Game | PS2 | 2008.0 | Racing | 0.0 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.0 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.0 | 0.01 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.0 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.0 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |

4508 rows × 11 columns

Ввод [14]:

low\_sales\_games **=** df[df['EU\_sales'] **==** 0.0]

low\_sales\_games

Out[14]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **148** | Final Fantasy XII | PS2 | 2006.0 | Role-Playing | 1.88 | 0.0 | 2.33 | 1.74 | 92.0 | 7.6 | T |
| **215** | Monster Hunter Freedom 3 | PSP | 2010.0 | Role-Playing | 0.00 | 0.0 | 4.87 | 0.00 | NaN | NaN | NaN |
| **268** | Metal Gear Solid 3: Snake Eater | PS2 | 2004.0 | Action | 1.46 | 0.0 | 0.83 | 1.93 | 91.0 | 9.3 | M |
| **284** | Half-Life | PC | 1997.0 | Shooter | 4.03 | 0.0 | 0.09 | 0.00 | 96.0 | 9.1 | M |
| **316** | Dragon Warrior III | NES | 1988.0 | Role-Playing | 0.10 | 0.0 | 3.77 | 0.00 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16709** | SCORE International Baja 1000: The Official Game | PS2 | 2008.0 | Racing | 0.00 | 0.0 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.00 | 0.0 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.00 | 0.0 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.0 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.00 | 0.0 | 0.01 | 0.00 | NaN | NaN | NaN |

5870 rows × 11 columns

Ввод [15]:

low\_sales\_games **=** df[df['JP\_sales'] **==** 0.0]

low\_sales\_games

Out[15]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **61** | Just Dance 3 | Wii | 2011.0 | Misc | 5.95 | 3.11 | 0.0 | 1.06 | 74.0 | 7.8 | E10+ |
| **85** | The Sims 3 | PC | 2009.0 | Simulation | 0.99 | 6.42 | 0.0 | 0.60 | 86.0 | 7.6 | T |
| **89** | Pac-Man | 2600 | 1982.0 | Puzzle | 7.28 | 0.45 | 0.0 | 0.08 | NaN | NaN | NaN |
| **100** | Call of Duty: World at War | X360 | 2008.0 | Shooter | 4.81 | 1.88 | 0.0 | 0.69 | 84.0 | 7.6 | M |
| **103** | Just Dance | Wii | 2009.0 | Misc | 3.48 | 2.99 | 0.0 | 0.73 | 49.0 | 8 | E10+ |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16706** | Men in Black II: Alien Escape | GC | 2003.0 | Shooter | 0.01 | 0.00 | 0.0 | 0.00 | NaN | tbd | T |
| **16708** | Woody Woodpecker in Crazy Castle 5 | GBA | 2002.0 | Platform | 0.01 | 0.00 | 0.0 | 0.00 | NaN | NaN | NaN |
| **16709** | SCORE International Baja 1000: The Official Game | PS2 | 2008.0 | Racing | 0.00 | 0.00 | 0.0 | 0.00 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.00 | 0.01 | 0.0 | 0.00 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.00 | 0.0 | 0.00 | NaN | NaN | NaN |

10514 rows × 11 columns

Ввод [16]:

low\_sales\_games **=** df[df['Other\_sales'] **==** 0.0]

low\_sales\_games

Out[16]:

|  | **Name** | **Platform** | **Year\_of\_Release** | **Genre** | **NA\_sales** | **EU\_sales** | **JP\_sales** | **Other\_sales** | **Critic\_Score** | **User\_Score** | **Rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **138** | World of Warcraft | PC | 2004.0 | Role-Playing | 0.08 | 6.21 | 0.00 | 0.0 | 93.0 | 7.3 | T |
| **215** | Monster Hunter Freedom 3 | PSP | 2010.0 | Role-Playing | 0.00 | 0.00 | 4.87 | 0.0 | NaN | NaN | NaN |
| **284** | Half-Life | PC | 1997.0 | Shooter | 4.03 | 0.00 | 0.09 | 0.0 | 96.0 | 9.1 | M |
| **288** | World of Warcraft: The Burning Crusade | PC | 2007.0 | Role-Playing | 2.57 | 1.52 | 0.00 | 0.0 | 91.0 | 7.9 | T |
| **316** | Dragon Warrior III | NES | 1988.0 | Role-Playing | 0.10 | 0.00 | 3.77 | 0.0 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.00 | 0.00 | 0.01 | 0.0 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.00 | 0.01 | 0.00 | 0.0 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.00 | 0.00 | 0.01 | 0.0 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.00 | 0.00 | 0.0 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.00 | 0.00 | 0.01 | 0.0 | NaN | NaN | NaN |

6601 rows × 11 columns

Так же хочу предположить, что оценки отсутствуют у игр, по которым не было продано ни одной копии, для Чего вывожу срезы по продажам в Северной Америке/Европе/Японии выше, на которых мы можем наблюдать пропуски оценок в столбцах с оценками критиков/пользователей и отсутствие маркировки рейтинга ESRB.

Ввод [17]:

df['Year\_of\_Release'].min(), df['Year\_of\_Release'].max()

Out[17]:

(1980.0, 2016.0)

При беглом осмотре, можно сказать, что в данных 16715 строк и есть пропуски в столбцах (перечислю по возрастанию количества пропусков): Name, Genre, Year\_of\_Release, User\_Score, Rating, Critic\_Score. Пропуски в именах и жанрах допускает при заведении данных платформа GEN (SEGA Mega Drive), но в незначительном кол-ве, удаление которых скорее всего не повлияет на дальнейший анализ данных; пропуски в оценках (критиков/пользователей/ESRB) от части объясняются тем, что до 1992 г игры вообще не оценивались (первая рейтинговая система появилась в Японии в 1992 году, а на западе необходимость формирования возрастных рейтингов назрела в девяностых с выходом игр, содержащих жестокие сцены, — Death Race, Mortal Kombat, Doom, Night Trap) - 308 строк; так же полностью не оценивались игры провалившиеся в прокате в США - 4508 строк. В данных собраны игры за последние 36 лет (на дату предоставления данных).

Ввод [18]:

df.columns **=** df.columns.str.lower()

df

Out[18]:

|  | **name** | **platform** | **year\_of\_release** | **genre** | **na\_sales** | **eu\_sales** | **jp\_sales** | **other\_sales** | **critic\_score** | **user\_score** | **rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Wii Sports | Wii | 2006.0 | Sports | 41.36 | 28.96 | 3.77 | 8.45 | 76.0 | 8 | E |
| **1** | Super Mario Bros. | NES | 1985.0 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | NaN | NaN | NaN |
| **2** | Mario Kart Wii | Wii | 2008.0 | Racing | 15.68 | 12.76 | 3.79 | 3.29 | 82.0 | 8.3 | E |
| **3** | Wii Sports Resort | Wii | 2009.0 | Sports | 15.61 | 10.93 | 3.28 | 2.95 | 80.0 | 8 | E |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996.0 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16710** | Samurai Warriors: Sanada Maru | PS3 | 2016.0 | Action | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16711** | LMA Manager 2007 | X360 | 2006.0 | Sports | 0.00 | 0.01 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16712** | Haitaka no Psychedelica | PSV | 2016.0 | Adventure | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |
| **16713** | Spirits & Spells | GBA | 2003.0 | Platform | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | NaN |
| **16714** | Winning Post 8 2016 | PSV | 2016.0 | Simulation | 0.00 | 0.00 | 0.01 | 0.00 | NaN | NaN | NaN |

16715 rows × 11 columns

Привожу к нижнему регистру названия столбцов.

Ввод [19]:

df **=** df.dropna(subset**=**['name']).reset\_index(drop**=True**)

df **=** df.dropna(subset**=**['year\_of\_release']).reset\_index(drop**=True**)

df.isna().sum()

Out[19]:

name 0

platform 0

year\_of\_release 0

genre 0

na\_sales 0

eu\_sales 0

jp\_sales 0

other\_sales 0

critic\_score 8461

user\_score 6605

rating 6676

dtype: int64

Удаляю пропуски в данных там, где их незначительное количество.

Ввод [20]:

df.dtypes

Out[20]:

name object

platform object

year\_of\_release float64

genre object

na\_sales float64

eu\_sales float64

jp\_sales float64

other\_sales float64

critic\_score float64

user\_score object

rating object

dtype: object

Ввод [21]:

*#год выпуска сохранен с плавающей точкой, хотя никаких дробных частей в годах нет, приведем их к целочисленному типу.*

df['year\_of\_release'] **=** df['year\_of\_release'].astype(int)

​

df['user\_score'] **=** pd.to\_numeric(df['user\_score'], errors**=**'coerce', downcast**=**'float')

df['user\_score'] **=** df['user\_score'].astype('float64')

​

df.info()

​

​

​

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 16444 entries, 0 to 16443

Data columns (total 11 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 name 16444 non-null object

1 platform 16444 non-null object

2 year\_of\_release 16444 non-null int64

3 genre 16444 non-null object

4 na\_sales 16444 non-null float64

5 eu\_sales 16444 non-null float64

6 jp\_sales 16444 non-null float64

7 other\_sales 16444 non-null float64

8 critic\_score 7983 non-null float64

9 user\_score 7463 non-null float64

10 rating 9768 non-null object

dtypes: float64(6), int64(1), object(4)

memory usage: 1.4+ MB

TBD в столбце user\_score - аббревиатура от английского To Be Determined (будет определено) или To Be Decided (будет решено). Используется, если какая-то информация еще не определена или решение по вопросу не принято. Принимаю значение за пропуск, тк оценка пользователем еще не проставлена и меняю тип данных на числовой для дальнейших рассчетов.

Ввод [22]:

df['rating']**=**df['rating'].fillna('RP')

​

df.loc[df['year\_of\_release'] **<=** 1992, 'critic\_score'] **=** df.loc[df['year\_of\_release'] **<=** 1992, 'critic\_score'].fillna(0)

​

df.loc[df['na\_sales'] **==** 0.0, 'critic\_score'] **=** df.loc[df['na\_sales'] **==** 0.0, 'critic\_score'].fillna(0)

df.loc[df['na\_sales'] **==** 0.0, 'user\_score'] **=** df.loc[df['na\_sales'] **==** 0.0, 'user\_score'].fillna(0)

​

df.isna().sum()

​

Out[22]:

name 0

platform 0

year\_of\_release 0

genre 0

na\_sales 0

eu\_sales 0

jp\_sales 0

other\_sales 0

critic\_score 4399

user\_score 5211

rating 0

dtype: int64

Ввод [23]:

df

Out[23]:

|  | **name** | **platform** | **year\_of\_release** | **genre** | **na\_sales** | **eu\_sales** | **jp\_sales** | **other\_sales** | **critic\_score** | **user\_score** | **rating** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Wii Sports | Wii | 2006 | Sports | 41.36 | 28.96 | 3.77 | 8.45 | 76.0 | 8.0 | E |
| **1** | Super Mario Bros. | NES | 1985 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | 0.0 | NaN | RP |
| **2** | Mario Kart Wii | Wii | 2008 | Racing | 15.68 | 12.76 | 3.79 | 3.29 | 82.0 | 8.3 | E |
| **3** | Wii Sports Resort | Wii | 2009 | Sports | 15.61 | 10.93 | 3.28 | 2.95 | 80.0 | 8.0 | E |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | RP |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16439** | Samurai Warriors: Sanada Maru | PS3 | 2016 | Action | 0.00 | 0.00 | 0.01 | 0.00 | 0.0 | 0.0 | RP |
| **16440** | LMA Manager 2007 | X360 | 2006 | Sports | 0.00 | 0.01 | 0.00 | 0.00 | 0.0 | 0.0 | RP |
| **16441** | Haitaka no Psychedelica | PSV | 2016 | Adventure | 0.00 | 0.00 | 0.01 | 0.00 | 0.0 | 0.0 | RP |
| **16442** | Spirits & Spells | GBA | 2003 | Platform | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | RP |
| **16443** | Winning Post 8 2016 | PSV | 2016 | Simulation | 0.00 | 0.00 | 0.01 | 0.00 | 0.0 | 0.0 | RP |

16444 rows × 11 columns

Предполагаю, что в столбце rating пропуски там, где категория еще не определена и заменяю их на соответствующее значение из системы ESRB - RP (категория ещё не присвоена). Так же ранее мною было выяснено, что до 1992 г игры не оценивались критиками, поэтому заменяю пропуски по ним в столбце critic\_score на ноль. Заменяю на ноль пропуски в рейтингах игр, которые провалились в прокате (либо не было продано ни одной копии по иным причинам)и не оценивались ни критиками ни пользователями. Остальные пропуски оставляю, тк не было выявлено иной связи или их объяснения, а удаление такого большого количества строк может повлиять на дальнейшее исследование.

Ввод [24]:

df['total\_sales'] **=** df['na\_sales'] **+** df['eu\_sales'] **+** df['jp\_sales'] **+** df['other\_sales']

df

Out[24]:

|  | **name** | **platform** | **year\_of\_release** | **genre** | **na\_sales** | **eu\_sales** | **jp\_sales** | **other\_sales** | **critic\_score** | **user\_score** | **rating** | **total\_sales** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Wii Sports | Wii | 2006 | Sports | 41.36 | 28.96 | 3.77 | 8.45 | 76.0 | 8.0 | E | 82.54 |
| **1** | Super Mario Bros. | NES | 1985 | Platform | 29.08 | 3.58 | 6.81 | 0.77 | 0.0 | NaN | RP | 40.24 |
| **2** | Mario Kart Wii | Wii | 2008 | Racing | 15.68 | 12.76 | 3.79 | 3.29 | 82.0 | 8.3 | E | 35.52 |
| **3** | Wii Sports Resort | Wii | 2009 | Sports | 15.61 | 10.93 | 3.28 | 2.95 | 80.0 | 8.0 | E | 32.77 |
| **4** | Pokemon Red/Pokemon Blue | GB | 1996 | Role-Playing | 11.27 | 8.89 | 10.22 | 1.00 | NaN | NaN | RP | 31.38 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **16439** | Samurai Warriors: Sanada Maru | PS3 | 2016 | Action | 0.00 | 0.00 | 0.01 | 0.00 | 0.0 | 0.0 | RP | 0.01 |
| **16440** | LMA Manager 2007 | X360 | 2006 | Sports | 0.00 | 0.01 | 0.00 | 0.00 | 0.0 | 0.0 | RP | 0.01 |
| **16441** | Haitaka no Psychedelica | PSV | 2016 | Adventure | 0.00 | 0.00 | 0.01 | 0.00 | 0.0 | 0.0 | RP | 0.01 |
| **16442** | Spirits & Spells | GBA | 2003 | Platform | 0.01 | 0.00 | 0.00 | 0.00 | NaN | NaN | RP | 0.01 |
| **16443** | Winning Post 8 2016 | PSV | 2016 | Simulation | 0.00 | 0.00 | 0.01 | 0.00 | 0.0 | 0.0 | RP | 0.01 |

16444 rows × 12 columns

Считаю суммарные продажи во всех регионах и записываю их в отдельный столбец total\_sales.

Ввод [25]:

df.duplicated().sum()

Out[25]:

0

Явные дубликаты в данных стандартным методом не найдены.

Ввод [26]:

df[df[['name', 'platform','year\_of\_release']].duplicated(keep**=False**)]

​

Out[26]:

|  | **name** | **platform** | **year\_of\_release** | **genre** | **na\_sales** | **eu\_sales** | **jp\_sales** | **other\_sales** | **critic\_score** | **user\_score** | **rating** | **total\_sales** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **600** | Madden NFL 13 | PS3 | 2012 | Sports | 2.11 | 0.22 | 0.0 | 0.23 | 83.0 | 5.5 | E | 2.56 |
| **15969** | Madden NFL 13 | PS3 | 2012 | Sports | 0.00 | 0.01 | 0.0 | 0.00 | 83.0 | 5.5 | E | 0.01 |

Ввод [27]:

df.drop\_duplicates(subset**=**["name", "platform", "year\_of\_release"], inplace**=True**)

Ввод [28]:

df[df[['name', 'platform','year\_of\_release']].duplicated(keep**=False**)]

​

Out[28]:

|  | **name** | **platform** | **year\_of\_release** | **genre** | **na\_sales** | **eu\_sales** | **jp\_sales** | **other\_sales** | **critic\_score** | **user\_score** | **rating** | **total\_sales** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

Проведена проверка на наличие дубликатов и их удаление, тк строки полностью идентичны.

В данном разделе я ознакомилась с данными, проанализировала пропуски в них (были удалены в столбцах name и year\_of\_release, заменены на ноль в столбцах critic\_score/user\_score, в столбце rating пропуски заменены на значение RP, тк я предположила, что пропуски допущены по играм, которые еще не оценены) и описала возможные причины пропусков под каждым выводом (обобщу дополнительно: возможными причинами пропусков может быть человеческий фактор - не заведены отдельные данные по невнимательности, указан некорректный тип - или данные могли быть утеряны при их трансформации/пересылке/тех сбое; так же я предположила что пропуски в оценках погут быть допущены, тк игры раньше 1992 года не оценивались, рейтинг ESRB с большой вероятностью используется только для западных продуктов, либо игра была куплена, но еще не оценена пользователем, либо игра не была куплена вообще - соответственно и оценки по ней нет). Привела имена столбцов к snake\_case, привела данные в двух столбце year\_of\_release к целочисленному типу, а в столбце user\_score к типу float для корректных дальнейших рассчетов и визуального восприятия. Обработала дубликаты. Добавила новый столбец total\_sales для дальнейших рассчетов.

# Исследовательский анализ данных.

Type Markdown and LaTeX: 𝛼2�2

Ввод [29]:

game\_counts **=** pd.pivot\_table(df, index**=**'year\_of\_release', values**=**'name', aggfunc**=**'count')

game\_counts.plot(kind**=**'bar')

plt.title('Количество выпущенных игр в год')

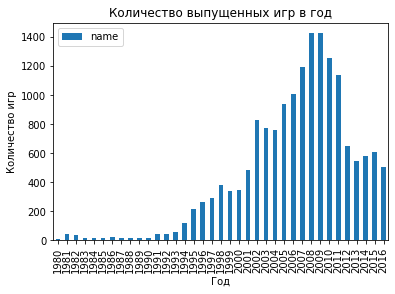
plt.xlabel('Год')

plt.ylabel('Количество игр')

plt.figure(figsize**=**(30, 30))

plt.show()

​



<Figure size 2160x2160 with 0 Axes>

По графику мы можем видеть резкое падение количества релизов вниз в 2012 г. Как и в любой другой индустрии, в игровой индустрии существуют циклические изменения. Возможно, после 2012 года произошли изменения в экономической ситуации (долговой кризис в Европе), что привело к сокращению бюджетов на разработку новых игр. Также возможно, что в этот период начались изменения в предпочтениях игроков, и разработчики стали уделять больше внимания адаптации старых игр под новые платформы, а не выпуску новых игр. Думаю, бОльшую значимость для анализа имеют данные за 2012-2016гг (определим этот период как актуальный и обрежем по нему датафрейм ниже), в которых не наблюдается резких скачков/падений.

Ввод [30]:

df **=** df[df['year\_of\_release'] **>=** 2012]

​

Ввод [31]:

platform\_sales **=** pd.pivot\_table(df, index**=**['platform', 'year\_of\_release'], values**=**'total\_sales', aggfunc**=**'sum')

*#platform\_sales = platform\_sales.sort\_values(by='total\_sales', ascending=False)*

platform\_sales\_top **=** platform\_sales[platform\_sales['total\_sales'] **>** 5]

platform\_sales\_top

​

Out[31]:

|  |  | **total\_sales** |
| --- | --- | --- |
| **platform** | **year\_of\_release** |  |
| **3DS** | **2012** | 51.36 |
| **2013** | 56.57 |
| **2014** | 43.76 |
| **2015** | 27.78 |
| **2016** | 15.14 |
| **DS** | **2012** | 11.01 |
| **PC** | **2012** | 23.22 |
| **2013** | 12.38 |
| **2014** | 13.28 |
| **2015** | 8.52 |
| **2016** | 5.25 |
| **PS3** | **2012** | 107.35 |
| **2013** | 113.25 |
| **2014** | 47.76 |
| **2015** | 16.82 |
| **PS4** | **2013** | 25.99 |
| **2014** | 100.00 |
| **2015** | 118.90 |
| **2016** | 69.25 |
| **PSP** | **2012** | 7.69 |
| **PSV** | **2012** | 16.19 |
| **2013** | 10.59 |
| **2014** | 11.90 |
| **2015** | 6.25 |
| **Wii** | **2012** | 21.71 |
| **2013** | 8.59 |
| **WiiU** | **2012** | 17.56 |
| **2013** | 21.65 |
| **2014** | 22.03 |
| **2015** | 16.35 |
| **X360** | **2012** | 99.74 |
| **2013** | 88.58 |
| **2014** | 34.74 |
| **2015** | 11.96 |
| **XOne** | **2013** | 18.96 |
| **2014** | 54.07 |
| **2015** | 60.14 |
| **2016** | 26.15 |

Посмотрим как менялись продажи платформ за актуальный период по годам (сводная таблица выше и графи к ниже помогут нам наглядно это продемонстрировать).

Ввод [32]:

fig, ax **=** plt.subplots(figsize**=**(12,6))

**for** platform **in** platform\_sales\_top.index.levels[0]:

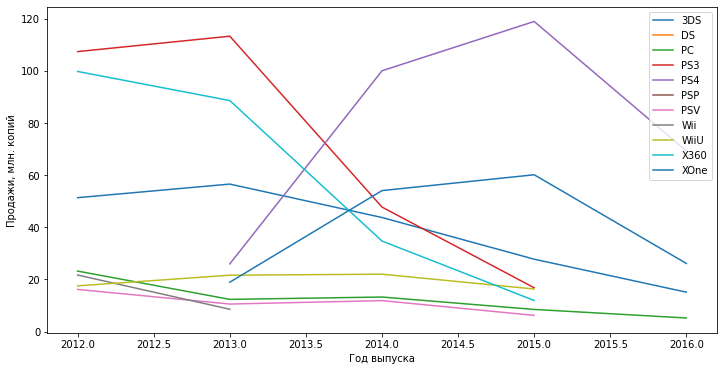
ax.plot(platform\_sales\_top.loc[platform].index, platform\_sales\_top.loc[platform]['total\_sales'], label**=**platform)

ax.legend()

ax.set\_xlabel('Год выпуска')

ax.set\_ylabel('Продажи, млн. копий')

plt.show()



Ввод [33]:

platform\_sales **=** pd.pivot\_table(df, index**=**'platform', values**=**'total\_sales', aggfunc**=**'sum')

platform\_sales **=** platform\_sales.sort\_values(by**=**'total\_sales', ascending**=False**)

platform\_sales

Out[33]:

|  | **total\_sales** |
| --- | --- |
| **platform** |  |
| **PS4** | 314.14 |
| **PS3** | 288.78 |
| **X360** | 236.54 |
| **3DS** | 194.61 |
| **XOne** | 159.32 |
| **WiiU** | 82.19 |
| **PC** | 62.65 |
| **PSV** | 49.18 |
| **Wii** | 35.37 |
| **DS** | 12.55 |
| **PSP** | 11.19 |

Лидерами продаж можно считать PS4, PS3, X360, 3DS, XOne - их продажи составляют от 159 млн копий по всему миру. Мы можем видеть падение прибыли у PS3, Х360 к 2015 г и одновременный рост ее у PS4 и XOne в этот же период, но затем падение прибыли у последних к 2016г. Потенциально прибыльными считаю (исходя из графика) PS4 и XOne. Отвечая на вопрос "За какой характерный срок появляются новые и исчезают старые платформы?", можно выделить несколько характерных сроков: примерно через 5-7 лет после выпуска новой платформы ее продажи достигают пика и начинают падать, а через 10-12 лет она обычно исчезает с рынка. Новые платформы появляются примерно каждые 5-6 лет.

Ввод [34]:

column\_list1 **=** ['PS4', 'PS3', 'XOne', '3DS', 'X360']

fig,ax **=** plt.subplots(figsize**=**(12,6))

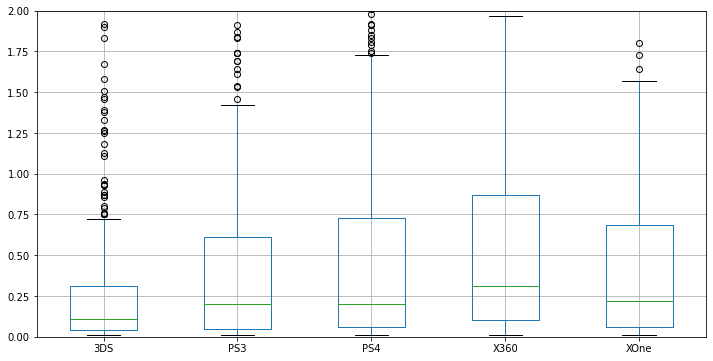
df.query('platform == @column\_list1').pivot\_table(values**=**'total\_sales', index**=**['name'],

columns**=**'platform').boxplot(

ax**=**ax, grid**=True**)

ax.set\_ylim(0,2)

plt.show()



Ввод [35]:

df.pivot\_table(values**=**'total\_sales', index**=**['name'], columns**=**'platform').describe()

Out[35]:

| **platform** | **3DS** | **DS** | **PC** | **PS3** | **PS4** | **PSP** | **PSV** | **Wii** | **WiiU** | **X360** | **XOne** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 396.000000 | 31.000000 | 250.00000 | 492.000000 | 392.000000 | 173.000000 | 411.000000 | 54.000000 | 147.000000 | 292.000000 | 247.000000 |
| **mean** | 0.491439 | 0.404839 | 0.25060 | 0.586951 | 0.801378 | 0.064682 | 0.119659 | 0.655000 | 0.559116 | 0.810068 | 0.645020 |
| **std** | 1.387161 | 1.439245 | 0.49464 | 1.464113 | 1.609456 | 0.071296 | 0.203011 | 1.165447 | 1.058836 | 1.712069 | 1.036139 |
| **min** | 0.010000 | 0.010000 | 0.01000 | 0.010000 | 0.010000 | 0.010000 | 0.010000 | 0.010000 | 0.010000 | 0.010000 | 0.010000 |
| **25%** | 0.040000 | 0.020000 | 0.03000 | 0.050000 | 0.060000 | 0.020000 | 0.020000 | 0.062500 | 0.080000 | 0.100000 | 0.060000 |
| **50%** | 0.110000 | 0.050000 | 0.08000 | 0.200000 | 0.200000 | 0.030000 | 0.050000 | 0.180000 | 0.220000 | 0.310000 | 0.220000 |
| **75%** | 0.312500 | 0.185000 | 0.24000 | 0.610000 | 0.730000 | 0.090000 | 0.120000 | 0.687500 | 0.525000 | 0.870000 | 0.685000 |
| **max** | 14.600000 | 8.060000 | 5.14000 | 21.050000 | 14.630000 | 0.360000 | 1.960000 | 6.760000 | 7.090000 | 16.270000 | 7.390000 |

По продажам лидируют X360, PS3 и PS4 (размах продаж по годам у нее самый большой - во второй год с момента выпуска ее популярность выросла почти в 2 раза). Больше всего игр у PS3. Определим лидером X360 (по среднему и максимальному числу продаж). PC является самой непопулярной платформой, но и самой стабильной - размах самый узкий. Среднее число продаж игр для десктопов самое низкое и доля бестселлеров самая маленькая.Медианные продажи по каждой из платформ заметно меньше 1 млн, но выбросы показывают, что у каждой из платформ случаются супер-хиты, приносящие миллионы - доля таких успешных проектов выше всего у PS4 (опять же) и Xone - это видно и по большему среднему значению числа продаж. Выбросы на боксплотах - это очень популярные игры. Сравним 3DS и XOne. У первого маленький ящик и много выбросов. Это говорит о том, что бОльшая часть объема продаж делается за счет этих выбросов. В то же время у XOne ситуация иная: большой бокс, длинный ус и мало выбросов. Это свидетельствует о том, что на XOne покупают самые разные игры в большом количестве.

Ввод [36]:

*#перезаписываю df исключая из него выбросы в районе нуля, которые могут исказить результаты анализа*

df **=** df.query('total\_sales > 1')

df

Out[36]:

|  | **name** | **platform** | **year\_of\_release** | **genre** | **na\_sales** | **eu\_sales** | **jp\_sales** | **other\_sales** | **critic\_score** | **user\_score** | **rating** | **total\_sales** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **16** | Grand Theft Auto V | PS3 | 2013 | Action | 7.02 | 9.09 | 0.98 | 3.96 | 97.0 | 8.2 | M | 21.05 |
| **23** | Grand Theft Auto V | X360 | 2013 | Action | 9.66 | 5.14 | 0.06 | 1.41 | 97.0 | 8.1 | M | 16.27 |
| **31** | Call of Duty: Black Ops 3 | PS4 | 2015 | Shooter | 6.03 | 5.86 | 0.36 | 2.38 | NaN | NaN | RP | 14.63 |
| **33** | Pokemon X/Pokemon Y | 3DS | 2013 | Role-Playing | 5.28 | 4.19 | 4.35 | 0.78 | NaN | NaN | RP | 14.60 |
| **34** | Call of Duty: Black Ops II | PS3 | 2012 | Shooter | 4.99 | 5.73 | 0.65 | 2.42 | 83.0 | 5.3 | M | 13.79 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **2025** | Need for Speed: Most Wanted | PSV | 2012 | Racing | 0.33 | 0.45 | 0.01 | 0.22 | NaN | NaN | RP | 1.01 |
| **2027** | Assassin's Creed: Rogue | X360 | 2014 | Action | 0.58 | 0.34 | 0.00 | 0.09 | 72.0 | 7.3 | M | 1.01 |
| **2028** | Rayman Legends | PS4 | 2014 | Platform | 0.22 | 0.64 | 0.00 | 0.16 | 90.0 | 8.5 | E10+ | 1.02 |
| **2032** | ZombiU | WiiU | 2012 | Action | 0.52 | 0.36 | 0.05 | 0.08 | 77.0 | 7.7 | M | 1.01 |
| **2033** | Pokken Tournament | WiiU | 2016 | Fighting | 0.50 | 0.26 | 0.17 | 0.08 | 76.0 | 8.0 | E10+ | 1.01 |

337 rows × 12 columns

Ввод [37]:

x360\_df **=** df[df['platform'] **==** 'X360']

x **=** x360\_df['critic\_score']

y **=** x360\_df['total\_sales']

​

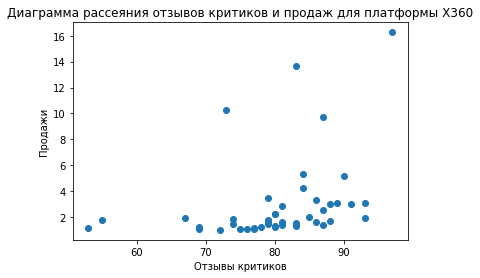
plt.scatter(x, y)

plt.xlabel('Отзывы критиков')

plt.ylabel('Продажи')

plt.title('Диаграмма рассеяния отзывов критиков и продаж для платформы X360')

plt.show()



Ввод [38]:

x360\_df **=** df[df['platform'] **==** 'X360']

x **=** x360\_df['user\_score']

y **=** x360\_df['total\_sales']

​

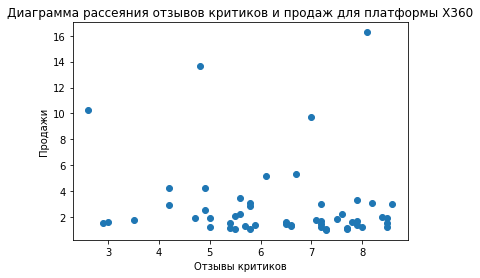
plt.scatter(x, y)

plt.xlabel('Отзывы критиков')

plt.ylabel('Продажи')

plt.title('Диаграмма рассеяния отзывов критиков и продаж для платформы X360')

plt.show()



Ввод [39]:

corr\_critics **=** df['critic\_score'].corr(df['total\_sales'])

corr\_users **=** df['user\_score'].corr(df['total\_sales'])

​

print('Корреляция между отзывами критиков и продажами:', corr\_critics)

print('Корреляция между отзывами пользователей и продажами:', corr\_users)

Корреляция между отзывами критиков и продажами: 0.19035851479469607

Корреляция между отзывами пользователей и продажами: -0.050266129891768346

Корреляция между отзывами критиков и продажами платформы x360 составляет 0.39, что указывает на умеренную положительную связь между этими переменными. Корреляция между отзывами пользователей и продажами составляет 0.10, что указывает на очень слабую положительную связь между этими переменными. Диаграмма рассеяния также показывает, что существует некоторая взаимосвязь между отзывами критиков и продажами, но не такая сильная, как многие могут ожидать. В то же время, связь между отзывами пользователей и продажами почти не заметна. Таким образом, можно сделать вывод, что отзывы критиков имеют большее влияние на продажи платформы x360, чем отзывы пользователей. Однако, не стоит полагаться только на отзывы, так как есть и другие факторы, влияющие на продажи игровых платформ.

Ввод [40]:

column\_list2 **=** ['PS3', 'XOne', '3DS', 'X360']

fig **=** plt.figure(figsize **=** (15,8))

fig.subplots\_adjust(wspace**=**0.5, hspace **=** 0.5)

с**=**0

​

**for** i **in** column\_list2:

df\_2 **=** df.query('platform == @i')

с**+=**1

sub **=** fig.add\_subplot(2,4,с)

st **=** df\_2.plot(x**=**'total\_sales', y**=**'critic\_score', kind**=**'scatter', alpha**=**0.5,

ax**=**sub, c**=** np.repeat(np.atleast\_2d(np.array([0.1, 0.2, 0.5])), len(df\_2), axis **=** 0))

с**+=**1

sub **=** fig.add\_subplot(2,4,с)

ax **=** df\_2.plot(x**=**'total\_sales', y**=**'user\_score', kind**=**'scatter', alpha**=**0.5,

ax**=**sub, c**=** np.repeat(np.atleast\_2d(np.array([0.1, 0.2, 0.5])), len(df\_2), axis **=** 0))

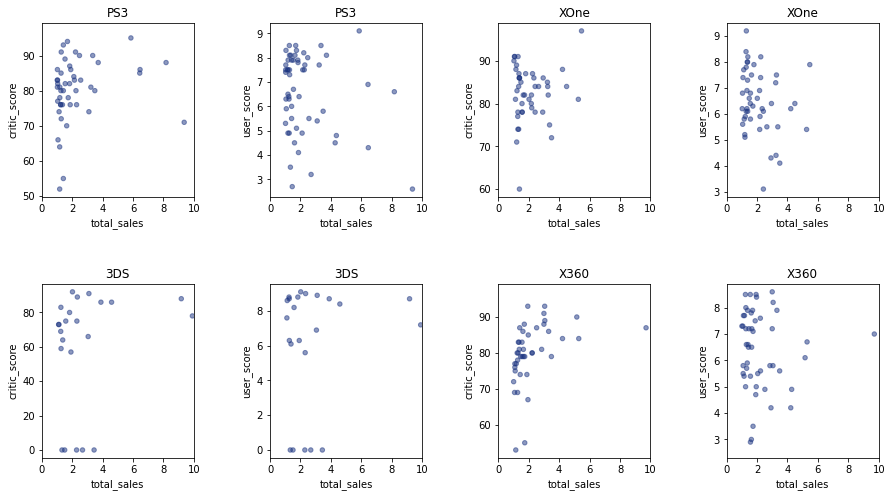
st.set\_xlim(0,10)

st.set\_title(i)

ax.set\_xlim(0,10)

ax.set\_title(i)

plt.show()



Ввод [41]:

**for** i **in** column\_list2:

df\_2 **=** df.query('platform == @i')

st **=** df\_2.pivot\_table(index**=**['name'], values**=**['critic\_score', 'user\_score',

'total\_sales'], columns**=**'platform').corr()

st **=** st.loc[:, 'total\_sales']

print(st)

​

platform PS3

platform

critic\_score PS3 0.305489

total\_sales PS3 1.000000

user\_score PS3 -0.068213

platform XOne

platform

critic\_score XOne 0.112908

total\_sales XOne 1.000000

user\_score XOne -0.225932

platform 3DS

platform

critic\_score 3DS 0.218988

total\_sales 3DS 1.000000

user\_score 3DS 0.141200

platform X360

platform

critic\_score X360 0.338542

total\_sales X360 1.000000

user\_score X360 -0.117669

Выявленная нами ранее тенденция влияния отзывов критиков и пользователей сохраняется и для остальных платформ-лидеров, кроме платформы 3DS: на ее продажи отзывы пользователей оказывают почти такое же влияние как и отзывы критиков, в чем мы можем убедиться посмотрев на индексы их корреляций и графики рассеяния.

Ввод [42]:

genre\_sales **=** pd.pivot\_table(df, index**=**'genre', values**=**'total\_sales', aggfunc**=**'mean')

genre\_sales **=** genre\_sales.sort\_values(by**=**'total\_sales', ascending**=False**)

genre\_sales

Out[42]:

|  | **total\_sales** |
| --- | --- |
| **genre** |  |
| **Simulation** | 3.850000 |
| **Shooter** | 3.570845 |
| **Misc** | 3.558571 |
| **Role-Playing** | 2.822667 |
| **Action** | 2.603700 |
| **Fighting** | 2.488889 |
| **Sports** | 2.484694 |
| **Platform** | 2.367000 |
| **Racing** | 2.100588 |
| **Adventure** | 1.560000 |
| **Strategy** | 1.395000 |
| **Puzzle** | 1.190000 |

Рассмотрим такие показатели как средние продажи по играм в рамках жанра. Как мы видим из таблицы в список лидеров вошли: Simulation, Shooter, Misc, Role-Playing и Action.

Ввод [43]:

column\_list2 **=** ['Simulation', 'Shooter', 'Misc', 'Role-Playing', 'Action']

fig,ax **=** plt.subplots(figsize**=**(12,6))

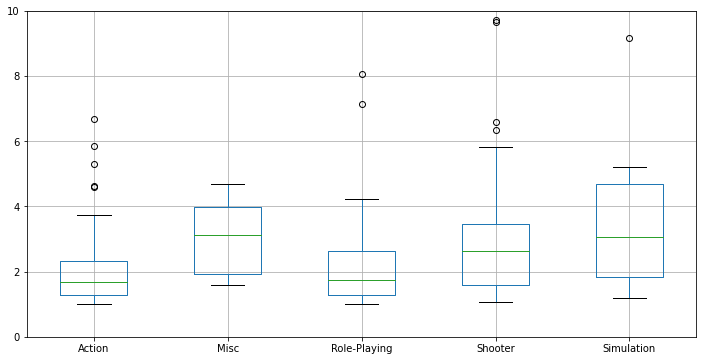
df.query('genre == @column\_list2').pivot\_table(values**=**'total\_sales', index**=**['name'],

columns**=**'genre').boxplot(

ax**=**ax, grid**=True**)

ax.set\_ylim(0,10)

plt.show()



Ввод [44]:

df.pivot\_table(values**=**'total\_sales', index**=**['name'], columns**=**'genre').describe()

Out[44]:

| **genre** | **Action** | **Adventure** | **Fighting** | **Misc** | **Platform** | **Puzzle** | **Racing** | **Role-Playing** | **Shooter** | **Simulation** | **Sports** | **Strategy** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 64.000000 | 3.000000 | 6.000000 | 9.000000 | 17.000000 | 1.00 | 14.000000 | 37.000000 | 37.000000 | 6.000000 | 23.000000 | 2.000000 |
| **mean** | 2.198076 | 1.560000 | 2.137500 | 3.006444 | 2.533039 | 1.19 | 2.178810 | 2.823536 | 3.188234 | 3.850000 | 2.150797 | 1.395000 |
| **std** | 1.661986 | 0.156205 | 2.032633 | 1.220596 | 2.216787 | NaN | 1.531585 | 2.948784 | 2.187462 | 2.981268 | 1.155292 | 0.388909 |
| **min** | 1.010000 | 1.380000 | 1.010000 | 1.580000 | 1.020000 | 1.19 | 1.030000 | 1.020000 | 1.070000 | 1.190000 | 1.050000 | 1.120000 |
| **25%** | 1.270000 | 1.510000 | 1.110000 | 1.920000 | 1.380000 | 1.19 | 1.455000 | 1.280000 | 1.600000 | 1.827500 | 1.195000 | 1.257500 |
| **50%** | 1.685000 | 1.640000 | 1.212500 | 3.130000 | 1.730000 | 1.19 | 1.796667 | 1.740000 | 2.640000 | 3.050000 | 1.787500 | 1.395000 |
| **75%** | 2.310000 | 1.650000 | 1.881250 | 3.970000 | 2.230000 | 1.19 | 2.167500 | 2.635000 | 3.445000 | 4.677500 | 2.785000 | 1.532500 |
| **max** | 11.316000 | 1.660000 | 6.210000 | 4.698000 | 9.900000 | 1.19 | 7.090000 | 14.600000 | 9.710000 | 9.170000 | 5.125000 | 1.670000 |

Ввод [45]:

**for** g **in** ['Simulation', 'Shooter', 'Misc', 'Role-Playing', 'Action']:

с **=** df[(df['genre'] **==** g) **&** (df['total\_sales'] **>** 2)]['total\_sales'].count()

print(f'Жанр {g}, количество бестселлеров - {с}, доля хитов от общего числа релизов {(с**/**df[df["genre"] **==** g]["total\_sales"].count()):.2}')

Жанр Simulation, количество бестселлеров - 4, доля хитов от общего числа релизов 0.67

Жанр Shooter, количество бестселлеров - 43, доля хитов от общего числа релизов 0.61

Жанр Misc, количество бестселлеров - 9, доля хитов от общего числа релизов 0.64

Жанр Role-Playing, количество бестселлеров - 18, доля хитов от общего числа релизов 0.4

Жанр Action, количество бестселлеров - 38, доля хитов от общего числа релизов 0.38

Наиболее популярным жанром, приносящим стабильный доход, является Simulation, который занимает первое место в списке. Это может быть связано с тем, что люди просто обожают играть в игры, симулирующие реальную жизнь, это дополнительно позволяет снять стресс и выплеснуть негативные эмоции экологичным путем. Наименьшей популярностью пользуются игры в жанрах Role-Playing и Action (опять же в рамках категории "игры, приносящие стабильный доход"). Мы можем наблюдать большое количество игр-бестселлеров в рамках таких жанров как Action и Shooter - но в списке игр, приносящих стабильный доход они далеко не на первом месте.

# Портрет пользователя каждого региона.

Ввод [46]:

na\_top5 **=** df.groupby('platform')['na\_sales'].sum().nlargest(5)

eu\_top5 **=** df.groupby('platform')['eu\_sales'].sum().nlargest(5)

jp\_top5 **=** df.groupby('platform')['jp\_sales'].sum().nlargest(5)

​

print('Топ-5 платформ в Северной Америке:\n', na\_top5)

print('Топ-5 платформ в Европе:\n', eu\_top5)

print('Топ-5 платформ в Японии:\n', jp\_top5)

Топ-5 платформ в Северной Америке:

platform

X360 98.72

PS4 84.74

XOne 67.53

PS3 64.04

3DS 39.26

Name: na\_sales, dtype: float64

Топ-5 платформ в Европе:

platform

PS4 113.11

PS3 75.92

X360 51.28

XOne 36.73

3DS 30.79

Name: eu\_sales, dtype: float64

Топ-5 платформ в Японии:

platform

3DS 53.88

PS3 11.12

WiiU 9.66

PS4 7.53

DS 3.51

Name: jp\_sales, dtype: float64

Ввод [47]:

z **=** df.pivot\_table(index**=**'platform', values **=** 'na\_sales', aggfunc**=**'sum').sort\_values(by**=**'na\_sales', ascending **=** **False**)

​

sales **=** z['na\_sales'].values

sorted\_sales **=** sorted(sales, reverse**=True**)

top5\_sales **=** sum(sorted\_sales[:5])

other\_sales **=** sum(sorted\_sales[5:])

sales\_for\_chart **=** list(sorted\_sales[:5]) **+** [other\_sales]

platforms\_for\_chart **=** list(z.index[:5]) **+** ['Other']

fractions **=** sales\_for\_chart **/** sum(sales\_for\_chart)

labels **=** [f"{fraction:.2f}" **for** fraction **in** fractions]

plt.figure(figsize**=**(8, 8))

plt.pie(sales\_for\_chart, labels**=**labels, startangle**=**90)

plt.legend(platforms\_for\_chart, loc**=**'best')

plt.title('Доли продаж по платформам в Северной Америке (топ-5 и Other)')

plt.show()

​

r **=** df.pivot\_table(index**=**'platform', values **=** 'eu\_sales', aggfunc**=**'sum').sort\_values(by**=**'eu\_sales', ascending **=** **False**)

​

sales **=** r['eu\_sales'].values

sorted\_sales **=** sorted(sales, reverse**=True**)

top5\_sales **=** sum(sorted\_sales[:5])

other\_sales **=** sum(sorted\_sales[5:])

sales\_for\_chart **=** list(sorted\_sales[:5]) **+** [other\_sales]

platforms\_for\_chart **=** list(r.index[:5]) **+** ['Other']

fractions **=** sales\_for\_chart **/** sum(sales\_for\_chart)

labels **=** [f"{fraction:.2f}" **for** fraction **in** fractions]

plt.figure(figsize**=**(8, 8))

plt.pie(sales\_for\_chart, labels**=**labels, startangle**=**90)

plt.legend(platforms\_for\_chart, loc**=**'best')

plt.title('Доли продаж по платформам в Европе (топ-5 и Other)')

plt.show()

​

y **=** df.pivot\_table(index**=**'platform', values **=** 'jp\_sales', aggfunc**=**'sum').sort\_values(by**=**'jp\_sales', ascending **=** **False**)

​

sales **=** y['jp\_sales'].values

sorted\_sales **=** sorted(sales, reverse**=True**)

top5\_sales **=** sum(sorted\_sales[:5])

other\_sales **=** sum(sorted\_sales[5:])

sales\_for\_chart **=** list(sorted\_sales[:5]) **+** [other\_sales]

platforms\_for\_chart **=** list(y.index[:5]) **+** ['Other']

fractions **=** sales\_for\_chart **/** sum(sales\_for\_chart)

labels **=** [f"{fraction:.2f}" **for** fraction **in** fractions]

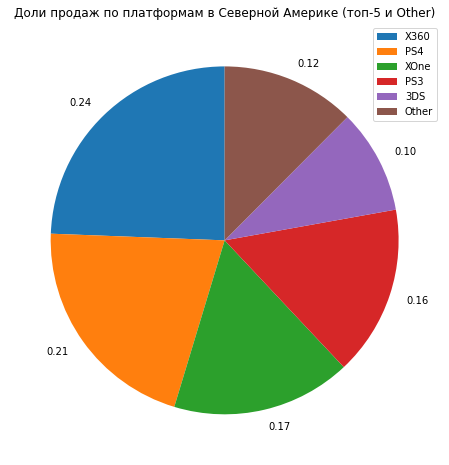
plt.figure(figsize**=**(8, 8))

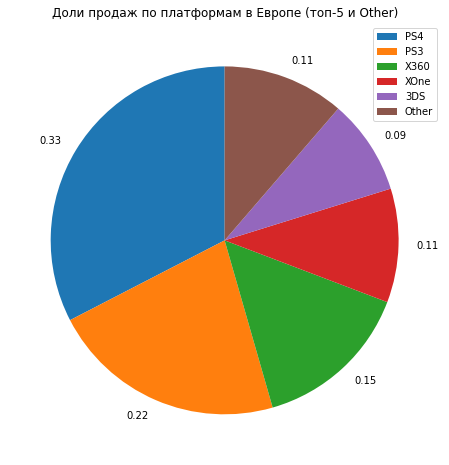
plt.pie(sales\_for\_chart, labels**=**labels, startangle**=**90)

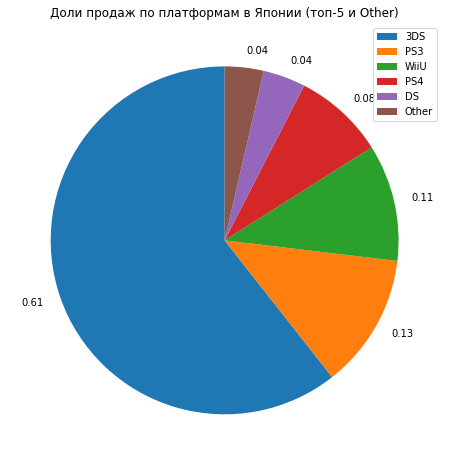
plt.legend(platforms\_for\_chart, loc**=**'best')

plt.title('Доли продаж по платформам в Японии (топ-5 и Other)')

plt.show()







Самые высокие продажи в Северной Америке имеет платформа X360 (140 млн проданных копий игр), в Европе - PS4 (141 млн проданных копий игр), в Японии лидером продаж является 3DS (но сумма его продаж составляет около 87 млн копий, что почти в 1,6 раз меньше чем в США и Европе). Хочу добавить, что уровень популярности компьютерных игр в Японии может быть так сильно снижен благодаря тому, что эта область регулируется государством (собрание префектуры Кагава одобрило постановление, согласно которому дети смогут играть в компьютерные игры не более часа в день). Американский и европейский рынки игр весьма приближены друг к другу по значениям продаж.

Ввод [48]:

na\_top5\_genre **=** df.groupby('genre')['na\_sales'].sum().nlargest(5)

eu\_top5\_genre **=** df.groupby('genre')['eu\_sales'].sum().nlargest(5)

jp\_top5\_genre **=** df.groupby('genre')['jp\_sales'].sum().nlargest(5)

​

print('Топ-5 жанров в Северной Америке:\n', na\_top5\_genre)

print('Топ-5 жанров в Европе:\n', eu\_top5\_genre)

print('Топ-5 жанров в Японии:\n', jp\_top5\_genre)

Топ-5 жанров в Северной Америке:

genre

Shooter 122.51

Action 108.44

Sports 54.49

Role-Playing 44.76

Misc 24.01

Name: na\_sales, dtype: float64

Топ-5 жанров в Европе:

genre

Action 101.41

Shooter 92.22

Sports 49.77

Role-Playing 34.90

Racing 17.39

Name: eu\_sales, dtype: float64

Топ-5 жанров в Японии:

genre

Role-Playing 37.30

Action 16.63

Platform 8.03

Simulation 7.69

Shooter 7.56

Name: jp\_sales, dtype: float64

Ввод [49]:

na\_genres **=** df.groupby('genre')['na\_sales'].sum().sort\_values(ascending**=False**).head(5).index.tolist()

eu\_genres **=** df.groupby('genre')['eu\_sales'].sum().sort\_values(ascending**=False**).head(5).index.tolist()

jp\_genres **=** df.groupby('genre')['jp\_sales'].sum().sort\_values(ascending**=False**).head(5).index.tolist()

​

​

*# создание подграфиков*

fig, axs **=** plt.subplots(nrows**=**1, ncols**=**3, figsize**=**(12, 4))

​

**for** i, ax **in** enumerate(axs):

**if** i **==** 0:

genres **=** na\_genres

title **=** 'Топ-5 жанров в Северной Америке'

**elif** i **==** 1:

genres **=** eu\_genres

title **=** 'Топ-5 жанров в Европе'

**else**:

genres **=** jp\_genres

title **=** 'Топ-5 жанров в Японии'

sales **=** []

**for** genre **in** genres:

**if** i **==** 0:

sales.append(df[df['genre'] **==** genre]['na\_sales'].sum())

**elif** i **==** 1:

sales.append(df[df['genre'] **==** genre]['eu\_sales'].sum())

**else**:

sales.append(df[df['genre'] **==** genre]['jp\_sales'].sum())

ax.pie(sales, labels**=**genres)

ax.set\_title(title)

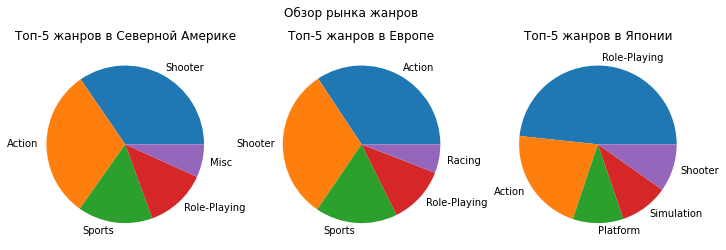
​

*# добавление общего заголовка*

fig.suptitle('Обзор рынка жанров')

​

plt.show()



Снова можем видеть сходство предпочтений и по жанрам игр между европейскими и американскими пользователями (игроками) - они предпочитают экшны и шутеры, в то время как японские игроки в первую очередь отдают предпочтение компьютерным ролевым играм - жанр компьютерных игр, основанный на элементах игрового процесса традиционных настольных ролевых игр - что в первую очередь связано с культурными различиями, по моему мнению. уровень продаж по жанрам в Северной Америке - 177 и 144 млн копий (экншн и шутер-жанры), в Европе - 159 и 113 млн копий, и 65 млн копий игр в жанре RPG продано в Японии (можно рекомендовать создать больше игр в этом жанре для японского рынка либо вывести имеющиеся дополнительно для увеличения продаж).

Давайте посмотрим есть ли связь между продажами в регионах - NA, EU, JP - и рейтингом ESRB.

Ввод [50]:

df.plot(x**=**'rating', y**=**'na\_sales', kind**=**'scatter', alpha**=**0.03, label**=**'Зависимость продаж в Северной Америке от рейтинга ESRB', figsize**=**(10,6))

​

​

Out[50]:

<AxesSubplot:xlabel='rating', ylabel='na\_sales'>



Ввод [51]:

df.plot(x**=**'rating', y**=**'eu\_sales', kind**=**'scatter', alpha**=**0.03, label**=**'Зависимость продаж в Европе от рейтинга ESRB', figsize**=**(10,6))

​

Out[51]:

<AxesSubplot:xlabel='rating', ylabel='eu\_sales'>



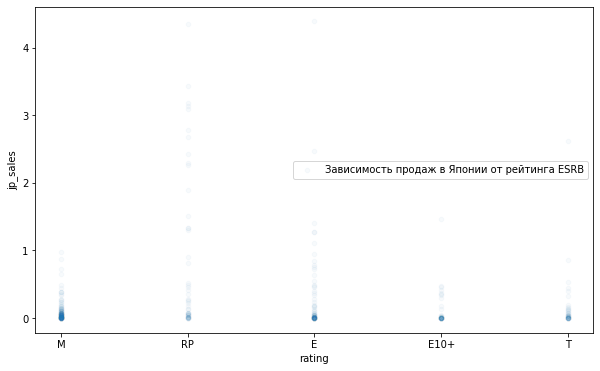
Ввод [52]:

df.plot(x**=**'rating', y**=**'jp\_sales', kind**=**'scatter', alpha**=**0.03, label**=**'Зависимость продаж в Японии от рейтинга ESRB', figsize**=**(10,6))

​

Out[52]:

<AxesSubplot:xlabel='rating', ylabel='jp\_sales'>



Ввод [53]:

rating\_sales\_stat **=** df.pivot\_table(index**=**'rating',

values**=**['na\_sales',

'eu\_sales',

'jp\_sales'], aggfunc**=**'sum')

rating\_sales\_stat **=** rating\_sales\_stat[['na\_sales', 'eu\_sales', 'jp\_sales']]

rating\_sales\_stat

Out[53]:

|  | **na\_sales** | **eu\_sales** | **jp\_sales** |
| --- | --- | --- | --- |
| **rating** |  |  |  |
| **E** | 83.55 | 83.23 | 21.98 |
| **E10+** | 38.09 | 24.78 | 4.61 |
| **M** | 179.77 | 151.02 | 12.49 |
| **RP** | 66.16 | 57.42 | 43.13 |
| **T** | 37.19 | 30.69 | 6.70 |

В Северной Америке лучше всего продаются игры маркированные рейтиногом ESRB как Е и М ("Подходит для всех возрастных категорий" и "Подходит для лиц старше 17 лет"), в Европе лидируют игры для тех же категорий потребителей, в Японии из явных категорий лучше всего продаются игры, которые подходят для всех категорий (Е). Рейтинг определенно оказывает значительное влияние на продажи в Северной Америке и Европе и чуть меньшее - в Японии.

# Проверка гипотез.

Проверим гипотезу: средние пользовательские рейтинги платформ Xbox One и PC одинаковые.

H0: Средние пользовательские рейтинги платформ Xbox One и PC равны.

H1: Средние пользовательские рейтинги платформ разные.

Ввод [54]:

revenue\_xone **=** df.loc[df['platform'] **==** "XOne"].pivot\_table(index**=**'name', values**=**'user\_score',

aggfunc**=**'mean')

revenue\_pc **=** df.loc[df['platform'] **==** "PC"].pivot\_table(index**=**'name', values**=**'user\_score',

aggfunc**=**'mean')

​

alpha **=** .05

results **=** ttest\_ind(revenue\_xone, revenue\_pc, equal\_var **=** **False**)

print('p-значение:', results.pvalue)

**if** (results.pvalue **<** alpha):

print("Отвергаем нулевую гипотезу")

**else**:

print("Не получилось отвергнуть нулевую гипотезу")

p-значение: [0.13915202]

Не получилось отвергнуть нулевую гипотезу

Для проверки гипотез был использован t-тест для двух независимых выборок, так как предполагалось нормальное распределение пользовательских рейтингов. Уровень значимости был выбран равным 0.05. Полученное p-значение (0.139) больше выбранного уровня значимости (0.05), что означает, что различия между средними пользовательскими рейтингами платформ Xbox One и PC не являются статистически значимыми. Следовательно, не получилось отвергнуть нулевую гипотезу о равенстве средних пользовательских рейтингов платформ Xbox One и PC.

Проверим следующую гипотезу: средние пользовательские рейтинги жанров Action (англ. «действие», экшен-игры) и Sports (англ. «спортивные соревнования») разные.

H0: Средние пользовательские рейтинги жанров Action и Sports отличаются.

H1: Средние пользовательские рейтинги жанров Action и Sports одинаковые.

Для проверки гипотезы будет использован t-тест для двух независимых выборок, так как предполагается нормальное распределение пользовательских рейтингов. Уровень значимости будет выбран равным 0.05.

Ввод [55]:

revenue\_action **=** df.loc[df['genre'] **==** "Action"].pivot\_table(index**=**'name', values**=**'user\_score',

aggfunc**=**'mean')

revenue\_sports **=** df.loc[df['genre'] **==** "Sports"].pivot\_table(index**=**'name', values**=**'user\_score',

aggfunc**=**'mean')

​

alpha **=** .05

results **=** ttest\_ind(revenue\_action, revenue\_sports, equal\_var **=** **False**)

print('p-значение:', results.pvalue)

**if** (results.pvalue **<** alpha):

print("Отвергаем нулевую гипотезу")

**else**:

print("Не получилось отвергнуть нулевую гипотезу")

​

p-значение: [0.00185929]

Отвергаем нулевую гипотезу

Для проверки гипотез был применен двухвыборочный t-критерий Стьюдента (ttest\_ind) с параметром equal\_var=False, так как предполагалось, что дисперсии выборок могут быть различными.

Критерий был выбран, потому что мы имеем дело с двумя выборками, которые не связаны между собой, и нужно проверить, насколько значимо отличаются их средние значения.

В данном случае вопрос гипотезы о различии двух совокупностей, поэтому нулевая гипотеза говорит о случайном совпадении, а альтернативная о том, что совокупности одинаковы. Полученное значение p-value говорит о том, что cредние пользовательские рейтинги жанров Action и Sports схожи, отвергаем нулевую гипотезу.

# Общий вывод.

В ходе исследовательскоого анализа я ознакомилась с данными и изучила общую информацию по ним; привела названия столбцов к нижнему регистру; преобразовала типы данных там, где это необходимо; обработала пропуски и дубликаты; посчитала суммарные продажи во всех регионах и записала их в отдельный столбец.

В процессе исследовательского анализа данных я заметила, что игровой рынок значительно стабилизировался и выровнялся в период с 2012 по 2016гг - для анализа я выбрала именно этот период, посчитав, что данные за него будут распределены наиболее равномерно. Далее я рассмотрела игровые платформы с точки зрения их прибыльности (равно популярности) в регионах и заметила, что у каждой из них есть жизненный цикл. Отвечая на вопрос "За какой характерный срок появляются новые и исчезают старые платформы?", можно выделить несколько характерных сроков: примерно через 5-7 лет после выпуска новой платформы ее продажи достигают пика и начинают падать, а через 10-12 лет она обычно исчезает с рынка, новые платформы появляются примерно каждые 5-6 лет. Лидерами продаж можно считать PS4, PS3, X360, 3DS, XOne - их продажи составляют от 159 млн копий по всему миру. Мы можем видеть падение прибыли у PS3, Х360 к 2015 г и одновременный рост ее у PS4 и XOne в этот же период, но затем падение прибыли у последних к 2016г. То есть в жизненном цикле платформ прослеживается тенденция "эстафетной палочки", при передаче которой передается и потенциал к росту популярности новой платформы за счет продуктов, которые она выпускает. Отслеживание роста популярности новых платформ позволит бизнесу сделать выгодную ставку на потенциально популярный продукт и спланировать рекламные кампании. На момент анализа данных потенциально популярные продукты выпускает PS4 и XOne.

Так же стоит обратить внимание на жанр игры, тк это один из факторов действительно влияющих на прибыльность продукта: наибольшей популярностью среди американских и европейских игроков пользуются жанры экшн и шутер, в то время как на японском рынке лидирует ролевая компьютерная игра. Рынки продуктов абсолютно разные, больше всего отличий у рынка Японии, целевая аудитория там предпочитает игры платформы 3DS, игроки Северной Америки - Х360, европейские - PS4. Однако, когда мы рассматриваем какие жанры приносят стабильный доход на первое место выходит жанр Simulation, а Action перемещается на последнее в пятерке лидиров, тк общая сумма его продаж так высока только за счет бест-селлеров, коими становится не каждый продукт, выпущенный в этом жанре - стоит обратить на это особое внимание.

На японском игровом рынке отзывы критиков играют меньшую роль по сравнением с отзывами пользователей, в то время как на западном рынке низкая оценка критиков может значительно снизить продажи только выпущенного продукта. Рейтинг ESRB предназначен для рецензирования игр на территории Северной Америки (США, Канады, Мексики) и оказывает там большее влияние на продажи, а в Японии так популярны игры вовсе без этого рейтинга - судя по всему, это игры, созданные в Японии и не получавшие рейтинга, но популярные на своей родине.

В рамках общих рекоммендаций возможно посоветовать разработчикам игр планировать выпуск новых игр в период спада основных лидирующих платформ (это примерно через 5-6 лет после их запуска). Дистрибьютерам же игр необходимо внимательно следить за жанровыми предпочтениями целевого рынка, учитывать жизненный цикл платформы и игры при планировании рекламных компаний (закупать для продажи игры в определенном жанре для определенного рынка сбыта, отдавать предпочтение потенциально популярным производителям игр, ориентированных на популярную возрастную группу: для западного рынка это категории Е и М, для японского - Е).