

```

1  --1.数据清洗
2  --1.1 查看发行年份的组成数据值，并保留发行年份在2000年至2017年之间的数据
3
4  --查看数据总行数
5  SELECT COUNT(*) FROM video_games;
6  /* 结果：16719*/
7
8  --查看发行年份列数据组成成分
9  SELECT Year_of_Release, COUNT(*) FROM video_games
10 GROUP BY Year_of_Release;
11 /*结果：
12 2006      1006
13 1985      14
14 2008      1427
15 2009      1426
16 1996      263
17 1989      17
18 1984      14
19 2005      939
20 1999      338
21 2007      1197
22 2010      1255
23 2013      544
24 2004      762
25 1990      16
26 1988      15
27 2002      829
28 2001      482
29 2011      1136
30 1998      379
31 2015      606
32 2012      653
33 2014      581
34 1992      43
35 1997      289
36 1993      62
37 1994      121
38 1982      36
39 2016      502
40 2003      775
41 1986      21
42 2000      350
43 N/A 269
44 1995      219
45 1991      41
46 1981      46
47 1987      16
48 1980      9
49 1983      17
50 2020      1
51 2017      3
52
53 可以看到有一个N/A的异常值，等等一并删除 */
54
55 --查询1999年游戏总数（后续有用）
56 SELECT COUNT(*) FROM video_games
57 WHERE Year_of_Release = 1999 AND Name IS NOT NULL;
58 /*结果：338*/
59
60 --删除异常值
61 DELETE FROM video_games WHERE Year_of_Release = 'N/A';
62 DELETE FROM video_games WHERE Year_of_Release < 2000 OR Year_of_Release > 2016;
63
64 --查看剩余数据行数
65 SELECT COUNT(*) FROM video_games;
66 /* 结果：14470*/
67
68 --1.2 清洗姓名列（Name）
69
70 --查看姓名列是否有NULL值或空值
71 SELECT COUNT(*) FROM video_games
72 WHERE Name IS NULL;
73 /* 结果：0，无空值*/

```

```

74
75 --查看姓名列是否有重复值
76 SELECT * FROM video_games
77 WHERE Name IN (SELECT Name FROM video_games
78 GROUP BY Name
79 HAVING COUNT(*) > 1)
80 ORDER BY Name;
81 /*结果：（由于结果太长，故截取部分）
82
83 Frozen: Olaf's Quest 3DS 2013 Platform Disney Interactive Studios 0.27
84 0.27 0 0.05 0.6
85 Frozen: Olaf's Quest DS 2013 Platform Disney Interactive Studios 0.21
86 0.26 0 0.04 0.52
87 [Prototype 2] PC 2012 Action Activision 0.07 0.03 0 0.01 0.11
88 76 12 6.4 389 Radical Entertainment M
89 [Prototype 2] X360 2012 Action Activision 0.48 0.24 0 0.07
90 0.79 74 69 7 173 Radical Entertainment M
91 [Prototype 2] PS3 2012 Action Activision 0.36 0.28 0 0.1 0.74 79
92 39 6.8 179 Radical Entertainment M
93 [Prototype] X360 2009 Action Activision 0.84 0.35 0 0.12 1.31
94 78 83 7.8 356 Radical Entertainment M
95 [Prototype] PS3 2009 Action Activision 0.65 0.4 0 0.19 1.24 79 53
96 7.7 308 Radical Entertainment M
97 007: Quantum of Solace PC 2008 Action Activision 0.01 0.01 0 0
98 0.03 70 18 6.3 55 Treyarch T
99 007: Quantum of Solace Wii 2008 Action Activision 0.29 0.28 0.01
100 0.07 0.65 54 11 7.5 26 Treyarch T
101 007: Quantum of Solace DS 2008 Action Activision 0.11 0.01 0 0.01
102 0.13 65 10 tbd Vicarious Visions T
103 007: Quantum of Solace PS2 2008 Action Activision 0.17 0 0 0.26
104 0.43
105 007: Quantum of Solace X360 2008 Action Activision 0.82 0.51 0.01
106 0.14 1.48 65 69 7.1 71 Treyarch T
107 007: Quantum of Solace PS3 2008 Action Activision 0.43 0.51 0.02
108 0.19 1.14 65 42 6.6 47 Treyarch T
109 007: The World is not Enough N64 2000 Action Electronic Arts 1.13 0.38
110 0.02 0.03 1.55
111 007: The World is not Enough PS 2000 Action Electronic Arts 0.51 0.35
112 0 0.06 0.92 61 11 6.7 44 Black Ops Entertainment T
113
114 可以看出是因为同一个游戏发行在不同的平台造成的，故我们不做删除处理，并将不同平台的同一
115 游戏视作不同的游戏*/
116
117 --1.3 清洗平台列（Platform）
118
119 --查看平台列是否有NULL值或空值
120 SELECT COUNT(*) FROM video_games
121 WHERE Platform IS NULL;
122 /* 结果： 0，无空值*/
123
124 --查看平台列组成成分
125 SELECT DISTINCT(Platform) FROM video_games;
126 /*结果：
127 Wii
128 DS
129 X360
130 PS3
131 PS2
132 GBA
133 PS4
134 3DS
135 XB
136 PC
137 PSP
138 XOne
139 WiiU
140 GC
141 GB
142 PS
143 N64
144 PSV
145 DC
146 WS

```

```

131 无异常值*/
132
133 --1.4 清洗游戏类型列 (Genre)
134
135 --查看游戏类型列是否有NULL值或空值
136 SELECT COUNT(*) FROM video_games
137     WHERE Genre IS NULL;
138 /* 结果: 0, 无空值*/
139
140 --查看游戏类型列组成成分
141 SELECT DISTINCT(Genre) FROM video_games;
142 /*结果:
143 Sports
144 Racing
145 Platform
146 Misc
147 Simulation
148 Action
149 Role-Playing
150 Puzzle
151 Shooter
152 Fighting
153 Adventure
154 Strategy
155 分为运动类、竞速类、平台跳跃类、杂项、模拟类、动作类、角色扮演类、益智类、射击类、格斗
    类、冒险类、策略类游戏*/
156
157 --1.5 清洗发行商列 (Publisher)
158
159 --查看发行商列是否有NULL值或空值
160 SELECT COUNT(*) FROM video_games
161     WHERE Publisher IS NULL;
162 /* 结果: 0, 无空值*/
163
164 --查看游戏类型列组成成分
165 SELECT DISTINCT(Publisher) FROM video_games;
166 /*结果过多, 不一一展示, 但是可以看出有些公司命名不规范或经过收购, 一家公司出现两种说法
    , 如Activision和Activision Value统一改为Activision Value;
    Ascaron Entertainment和Ascaron Entertainment GmbH统一改为后者等等。*/
167
168 --修改不规范名称
169
170 UPDATE video_games SET Publisher = 'Activision Value'
171     WHERE Publisher = 'Activision';
172 UPDATE video_games SET Publisher = 'Ascaron Entertainment GmbH'
173     WHERE Publisher = 'Ascaron Entertainment';
174 UPDATE video_games SET Publisher = 'ASCII Entertainment'
175     WHERE Publisher = 'ASCII Media Works';
176 UPDATE video_games SET Publisher = 'Avanquest Software'
177     WHERE Publisher = 'Avanquest';
178 UPDATE video_games SET Publisher = 'Big Ben Interactive'
179     WHERE Publisher = 'Bigben Interactive';
180 UPDATE video_games SET Publisher = 'Codemasters'
181     WHERE Publisher = 'Codemasters Online';
182 UPDATE video_games SET Publisher = 'Daedalic Entertainment'
183     WHERE Publisher = 'Daedalic';
184 UPDATE video_games SET Publisher = 'FuRyu Corporation'
185     WHERE Publisher = 'FuRyu';
186 UPDATE video_games SET Publisher = 'Hudson Soft'
187     WHERE Publisher = 'Hudson Entertainment';
188 UPDATE video_games SET Publisher = 'Idea Factory'
189     WHERE Publisher = 'Idea Factory International';
190 UPDATE video_games SET Publisher = 'Kadokawa Games'
191     WHERE Publisher = 'Kadokawa Shoten';
192 UPDATE video_games SET Publisher = 'Marvelous Entertainment'
193     WHERE Publisher = 'Marvelous Games' OR Publisher = 'Marvelous Interactive';
194 UPDATE video_games SET Publisher = 'Milestone'
195     WHERE Publisher = 'Milestone S.r.l' OR Publisher = 'Milestone S.r.l.';
196 UPDATE video_games SET Publisher = 'Nippon'
197     WHERE Publisher = 'Nippon Amuse' OR Publisher = 'Nippon Columbia' OR Publisher =
        'Nippon Ichi Software';
198 UPDATE video_games SET Publisher = 'Paon Corporation'
199     WHERE Publisher = 'Paon';
200 UPDATE video_games SET Publisher = 'Paradox'

```

```

201     WHERE Publisher = 'Paradox Development' OR Publisher = 'Paradox Interactive';
202 UPDATE video_games SET Publisher = 'Rebellion'
203     WHERE Publisher = 'Rebellion Developments';
204 UPDATE video_games SET Publisher = 'Revolution Software'
205     WHERE Publisher = 'Revolution (Japan)';
206 UPDATE video_games SET Publisher = 'SNK'
207     WHERE Publisher = 'SNK Playmore';
208 UPDATE video_games SET Publisher = 'Sony Computer Entertainment'
209     WHERE Publisher = 'Sony Computer Entertainment America' OR Publisher = 'Sony
    Computer Entertainment Europe'
210     OR Publisher = 'Sony Music Entertainment' OR Publisher = 'Sony Online
    Entertainment';
211 UPDATE video_games SET Publisher = 'Square'
212     WHERE Publisher = 'Square Enix' OR Publisher = 'Square Enix ' OR Publisher =
    'SquareSoft';
213 UPDATE video_games SET Publisher = 'System 3'
214     WHERE Publisher = 'System 3 Arcade Software';
215 UPDATE video_games SET Publisher = 'Takara'
216     WHERE Publisher = 'Takara Tomy';
217 UPDATE video_games SET Publisher = 'TDK'
218     WHERE Publisher = 'TDK Core' OR Publisher = 'TDK Mediactive';
219 UPDATE video_games SET Publisher = 'Ubisoft'
220     WHERE Publisher = 'Ubisoft Annecy';
221 UPDATE video_games SET Publisher = 'Valve Software'
222     WHERE Publisher = 'Valve';
223
224 --1.6 清洗销量列 (NA_Sales, EU_Sales, JP_Sales, Other_Sales, Global_Sales)
225
226 --查看销量列是否有NULL值或空值
227 SELECT COUNT(*) FROM video_games
228     WHERE NA_Sales IS NULL;
229 SELECT COUNT(*) FROM video_games
230     WHERE EU_Sales IS NULL;
231 SELECT COUNT(*) FROM video_games
232     WHERE JP_Sales IS NULL;
233 SELECT COUNT(*) FROM video_games
234     WHERE Other_Sales IS NULL;
235 SELECT COUNT(*) FROM video_games
236     WHERE Global_Sales IS NULL;
237 /* 结果: 全部为0, 无空值*/
238
239 --1.7 清洗媒体评分和媒体总数列 (Critic_Score, Critic_Count)
240
241 --查看是否有NULL值或空值
242 SELECT COUNT(*) FROM video_games
243     WHERE Critic_Score IS NULL;
244 SELECT COUNT(*) FROM video_games
245     WHERE Critic_Count IS NULL;
246 /* 结果: 均为6586, 因数量过多, 故删除这两列*/
247 ALTER TABLE video_games DROP COLUMN Critic_Score, DROP COLUMN Critic_Count;
248
249 --1.8 删除开发商列 (Developer) 此项目不研究此项
250 ALTER TABLE video_games DROP COLUMN Developer;
251
252 ----1.9 清洗用户评分和用户总数列 (User_Score, User_Count)
253
254 --通过观察发现评分列不仅有空值, 还有tbd值
255 --查看空值和tbd值数量
256 SELECT COUNT(*) FROM video_games
257     WHERE User_Score IS NULL OR User_Score = 'tbd';
258 SELECT COUNT(*) FROM video_games
259     WHERE User_Count IS NULL;
260 /* 结果: 均为7102, 因数量过多, 故删除这两列*/
261 ALTER TABLE video_games DROP COLUMN User_Score, DROP COLUMN User_Count;
262
263 --1.10 清洗评级列 (Rating)
264
265 --查看是否有NULL值或空值
266 SELECT COUNT(*) FROM video_games
267     WHERE Rating IS NULL;
268 /* 结果: 4810, 根据美国评级系统, 填入RP*/
269 UPDATE video_games SET Rating = 'RP'
270     WHERE Rating IS NULL;

```

```

271
272 --2.数据分析 (MySQL+Tableau)
273 --2.1 游戏数量-增长率-年份关系
274 SELECT Year_of_Release AS '发行年份', COUNT(*) AS '游戏总数',
275        CASE a.Year_of_Release
276        WHEN 2000 THEN ((COUNT(*)/338)-1)
277        ELSE ((COUNT(*)/(SELECT COUNT(*) FROM video_games AS b
278                        WHERE b.Year_of_Release = a.Year_of_Release-1
279                        GROUP BY b.Year_of_Release))-1)
280        END AS '增长率'
281 FROM video_games AS a
282 GROUP BY Year_of_Release
283 ORDER BY Year_of_Release;
284 --2.2 各平台发行游戏数-平均销量-年份关系
285 SELECT Year_of_Release AS '发行年份', Platform AS '发行平台', COUNT(*) AS
286 '游戏总数', SUM(Global_Sales)/COUNT(*) AS '平均销量'
287 FROM video_games
288 GROUP BY Year_of_Release, Platform;
289 --2.3 游戏类型发行数-年份关系
290 SELECT Year_of_Release AS '发行年份', Genre AS '游戏类型', COUNT(*) AS '发行数量'
291 FROM video_games
292 GROUP BY Year_of_Release, Genre;
293 --2.4 各游戏类型销量-年份关系
294 SELECT Year_of_Release AS '发行年份', Genre AS '游戏类型',
295 SUM(Global_Sales)/COUNT(*) AS '平均销量'
296 FROM video_games
297 GROUP BY Year_of_Release, Genre;
298 --2.5 2000-2016年间各发行商总发行游戏数
299 SELECT ROW_NUMBER(), Publisher AS '发行商', COUNT(*) AS '发行总数'
300 FROM video_games
301 GROUP BY Publisher;
302 --2.6 发行总数前十的发行商及总发行游戏数
303 SELECT Publisher AS '发行商', COUNT(*) AS '总发行游戏数'
304 FROM video_games
305 GROUP BY Publisher
306 ORDER BY COUNT(*) DESC LIMIT 10;
307 --2.7 发行总数前十的发行商的平均销量-总游戏数关系
308 SELECT Publisher AS '发行商', COUNT(*) AS '总发行游戏数', SUM(Global_Sales)/COUNT(*)
309 AS '平均销量'
310 FROM video_games
311 GROUP BY Publisher
312 ORDER BY COUNT(*) DESC LIMIT 10;
313 --2.8 发行总数前十的发行商游戏发行数占比/平均销量占比-年份关系
314 CREATE VIEW TOP10 AS
315 (SELECT Publisher FROM video_games
316  GROUP BY Publisher
317  ORDER BY COUNT(*) DESC LIMIT 10) a);
318 CREATE VIEW EveryYear AS
319 (SELECT Year_of_Release, COUNT(*) AS C, SUM(Global_Sales) AS S
320  FROM video_games
321  GROUP BY Year_of_Release);
322 SELECT P_C.Year_of_Release AS '发行年份', P_C.Publisher AS '发行商',
323 P_C.C1/EveryYear.C AS '发行数量占比', P_C.S1/everyyear.S AS '销量占比'
324 FROM
325 (SELECT Publisher, Year_of_Release, COUNT(*) AS C1, SUM(Global_Sales) AS S1
326  FROM video_games
327  WHERE Publisher IN (SELECT Publisher FROM TOP10)
328  GROUP BY Year_of_Release, Publisher) P_C
329 LEFT JOIN EveryYear
330 ON P_C.Year_of_Release = EveryYear.Year_of_Release;
331 --2.9 各地区销量占比-年份关系
332 CREATE VIEW AreaSales AS
333 (SELECT Year_of_Release,
334        SUM(NA_Sales) a,
335        SUM(EU_Sales) b,
336        SUM(JP_Sales) c,
337        SUM(Other_Sales) d,
338        SUM(Global_Sales) e
339  FROM video_games
340  GROUP BY Year_of_Release);
341 SELECT Year_of_Release AS '发行年份',
342        a/e AS '北美',
343        b/e AS '欧洲',

```

```

341         c/e AS '日本',
342         d/e AS '其他地区'
343     FROM AreaSales;
344 --2.11 各地区喜爱游戏类型-年份关系
345 SELECT m.Year_of_Release AS '发行年份', m.Genre AS '游戏类型',
346        m.na/AreaSales.a AS '北美', m.eu/AreaSales.b AS '欧洲',
347        m.jp/AreaSales.c AS '日本', m.other/AreaSales.d AS '其他地区'
348 FROM
349     (SELECT Year_of_Release, Genre,
350            SUM(NA_Sales) na,
351            SUM(EU_Sales) eu,
352            SUM(JP_Sales) jp,
353            SUM(Other_Sales) other
354     FROM video_games
355     GROUP BY Year_of_Release, Genre) AS m
356 LEFT JOIN AreaSales
357 ON m.Year_of_Release = AreaSales.Year_of_Release;
358

```