

```
1  --creating the user table
2  CREATE TABLE topshot_user(
3      --create cols
4      topshot_user_id int identity
5      , topshot_username varchar(50) NOT NULL
6      , user_join_date datetime NOT NULL
7      , showcases_count int
8      , moments_count int
9      , user_team varchar(50)
10     --place constraints
11     , CONSTRAINT PK_topshot_user PRIMARY KEY (topshot_user_id)
12     , CONSTRAINT U1_topshot_user UNIQUE(topshot_username)
13 )
14 -- End Creating the User table
15
16 --Creating Moments table
17 CREATE TABLE moments(
18     --create cols
19     moment_unique_id int identity
20     , serial_number int NOT NULL
21     , player_name varchar(200) NOT NULL
22     , game_date datetime NOT NULL
23     , pack_name varchar(100) NOT NULL
24     , highlight_type varchar(100) NOT NULL
25     , moment_type varchar(100) NOT NULL
26     , circulation_count int NOT NULL
27     , circulation_status varchar(50) NOT NULL
28     , highlight_season varchar(50) NOT NULL
29     -- place constraints
30     , CONSTRAINT PK_moments PRIMARY KEY(moment_unique_id)
31 )
32
33 --Creating table with User's moments
34 CREATE TABLE user_moments_collection(
35     --creating cols
36     collection_id int identity
37     , topshot_user_id int NOT NULL
38     , moment_unique_id int NOT NULL
39     , purchase_date datetime NOT NULL
40     --place constraints
41     , CONSTRAINT PK_user_moments_collection PRIMARY KEY(collection_id)
42     , CONSTRAINT FK1_user_moments_collection FOREIGN KEY(moment_unique_id) REFERENCES moments(moment_unique_id)
43     , CONSTRAINT FK2_user_moments_collection FOREIGN KEY(topshot_user_id) REFERENCES topshot_user(topshot_user_id)
44     , CONSTRAINT U1_user_moments_collection UNIQUE(moment_unique_id)
45 )
46
47 DROP TABLE user_moments_collection
```

```
48
49 --Creating purchase order table for Moments
50 CREATE TABLE moments_purchase_order(
51     moment_po_transaction_id int identity
52     , topshot_user_id int NOT NULL
53     , moment_unique_id int NOT NULL
54     , moment_purchase_price int NOT NULL
55     , payment_type varchar(50) NOT NULL
56     --place constraints
57     , CONSTRAINT PK_moments_purchase_order PRIMARY KEY           ↗
58       (moment_po_transaction_id)
59     , CONSTRAINT FK1_moments_purchase_order FOREIGN KEY(topshot_user_id)   ↗
60       REFERENCES topshot_user(topshot_user_id)
61     , CONSTRAINT FK2_moments_purchase_order FOREIGN KEY(moment_unique_id)   ↗
62       REFERENCES moments(moment_unique_id)
63 )
64
65 --Creating table for Packs
66 CREATE TABLE packs(
67     pack_id int identity
68     , pack_type varchar(50) NOT NULL
69     , moments_count int NOT NULL
70     , drop_number int NOT NULL
71     , pack_circulation_status varchar(50)
72     , pack_name varchar(50)
73     -- place constraints
74     CONSTRAINT PK_packs PRIMARY KEY(pack_id)
75 )
76
77 -- Creating table for Packs Purchase Order
78 CREATE TABLE pack_purchase_order(
79     pack_po_transaction_id int identity
80     , topshot_user_id int NOT NULL
81     , pack_id int NOT NULL
82     , pack_purchase_price int NOT NULL
83     , payment_type varchar(50) NOT NULL
84     --place constraints
85     , CONSTRAINT PK_pack_purchase_order PRIMARY KEY (pack_po_transaction_id)
86     , CONSTRAINT FK1_pack_purchase_order FOREIGN KEY(topshot_user_id)   ↗
87       REFERENCES topshot_user(topshot_user_id)
88     , CONSTRAINT FK2_pack_purchase_order FOREIGN KEY(pack_id) REFERENCES packs ↗
89       (pack_id)
90 )
91
92 --Creating table for Flow blockchain that validates all transactions
93 CREATE TABLE flow_blockchain(
94     ledger_transaction_id int identity
95     , order_timestamp datetime NOT NULL
96     , pack_id int
```

```

92     , pack_po_transaction_id int
93     , moment_po_transaction_id int
94     , moment_unique_id int
95     --place constraints
96     , CONSTRAINT PK_flow_blockchain PRIMARY KEY(ledger_transaction_id)
97     , CONSTRAINT FK1_flow_blockchain FOREIGN KEY(pack_id) REFERENCES packs
100     (pack_id)
101     , CONSTRAINT FK2_flow_blockchain FOREIGN KEY(pack_po_transaction_id)
102     REFERENCES pack_purchase_order(pack_po_transaction_id)
103     , CONSTRAINT FK3_flow_blockchain FOREIGN KEY(moment_po_transaction_id)
104     REFERENCES moments_purchase_order(moment_po_transaction_id)
105     , CONSTRAINT FK4_flow_blockchain FOREIGN KEY(moment_unique_id) REFERENCES
106     moments(moment_unique_id)
107 )
108
109 -- There is a bug in our payment system that need to be resolved,
110 -- it's only affecting the base set packs so we have to change these from
111 available to unavailable.
112
113 SELECT * FROM packs
114
115 -- We do not have the unavailable pack circulation status so we have to insert
116 that into our table first
117
118 --adding unavailable status, so we know which packs are not sold out, and
119 available for purchase once we fix the bug
120 UPDATE packs SET pack_circulation_status = 'unavailable' WHERE
121 pack_circulation_status = 'available'
122
123 SELECT * FROM packs
124
125 --cool, now our ecommerce platform has marked our available base set cards to
126 unavailable,
127 --that way once the bug is fixed we can switch the unavails back to avail
128
129 -- insert our users into the topshot_user table
130 insert into topshot_user (user_join_date, topshot_username, showcases_count,
131 moments_count, user_team)
132 values
133 ('2020-06-15 02:06:31', 'rbrownsmith0', 6, 1184, null)
134 ,('2020-02-06 02:32:01', 'anesbeth1', 6, 54, 'Phoenix Suns')
135 ,('2019-08-26 20:40:08', 'tcustance2', null, 1261, 'Milwaukee Bucks')
136 , ('2020-11-17 20:10:35', 'glamey3', 10, 903, 'San Antonio Spurs')
137 , ('2020-04-04 16:43:18', 'braffels4', null, 293, 'Indiana Pacers')
138 , ('2020-08-26 07:04:01', 'lwessing5', 9, 1030, 'Washington Wizards')
139 , ('2019-09-05 14:18:15', 'anazer6', 2, 974, null)
140 , ('2019-08-03 20:25:05', 'llawey7', 4, 552, 'Golden State Warriors')

```

```

131 , ('2019-08-26 02:46:36', 'mbosward8', 6, 1234, null)
132 , ('2019-08-27 09:19:55', 'hhasted9', null, 354, 'Phoenix Suns')
133 , ('2020-01-27 01:50:29', 'gaugustea', null, 678, 'Oklahoma City Thunder')
134 , ('2020-10-03 07:46:36', 'cponcefordb', null, 605, 'Portland Trail Blazers')
135 , ('2020-01-15 14:50:08', 'tsperac', null, 635, 'San Antonio Spurs')
136 , ('2020-07-16 21:52:48', 'hnandd', 10, 460, 'Phoenix Suns')
137 , ('2019-07-15 17:54:12', 'tmullingere', 10, 893, 'Los Angeles Lakers')
138 , ('2020-10-21 04:18:11', 'dhearnef', null, 1093, 'New York Knicks')
139 , ('2019-10-21 00:41:34', 'murlingg', null, 1006, null)
140 , ('2020-09-08 10:57:29', 'tskinh', 1, 212, 'Utah Jazz')
141 , ('2020-08-22 06:19:17', 'ldoughtyi', 3, 789, 'Atlanta Hawks')
142 , ('2020-09-03 16:17:18', 'cberthomierj', null, 893, 'Utah Jazz')
143 , ('2020-03-19 12:46:33', 'zvelak', 4, 866, 'Cleveland Cavaliers')
144 , ('2020-10-15 20:29:09', 'leynaldl', null, 216, 'Los Angeles Clippers')
145 , ('2019-10-12 19:57:04', 'pbarnwallm', null, 240, 'Dallas Mavericks')
146
147 --Let's check our data!
148
149 SELECT * FROM topshot_user
150
151 --We have users now! Lot's of them have not created showcases, some have not
    selected their favorite team, and one has not purchases any moments!
152
153 -- Insert data officially licensed nba highlights into the moments table
154 insert into moments (serial_number, player_name , game_date , pack_name ,
    highlight_type , moment_type , circulation_count, circulation_status ,
    highlight_season)
155 values (39, 'Steven Adams', '07/29/2020', 'base set', 'layup', 'rare', 3326,
    'cc', '2019-2020')
156 ,(94, 'Paul Millsap', '08/20/2020', 'metallic hold le', 'block', 'common',
    3326, 'cc', '2018-2019')
157 ,(171, 'OG Anunoby', '10/02/2020', 'metallic hold le', 'steal', 'common', 299,
    'le', '2019-2020')
158 ,(8, 'Royce O'Neale', '08/31/2020', 'so fresh', 'steal', 'common', 299, 'cc',
    '2019-2020')
159 ,(90, 'Joe Harris', '10/07/2020', 'so fresh', 'handles', 'rare', 299, 'cc',
    '2019-2020')
160 ,(180, 'Nemanja Bjelica', '10/16/2020', 'metallic hold le', 'handles',
    'common', 275, 'le', '2019-2020')
161 ,(74, 'Miles Bridges', '08/08/2020', 'premium pack', 'handles', 'common', 2343,
    'le', '2019-2020')
162 ,(55, 'Caris LeVert', '07/28/2020', 'metallic hold le', 'handles', 'common',
    3326, 'cc', '2019-2020')
163 ,(242, 'Jonathan Isaac', '08/09/2020', 'base set', 'dunk', 'common', 345, 'cc',
    '2019-2020')
164 ,(6, 'Tyler Herro', '10/16/2020', 'metallic hold le', 'steal', 'common', 3134,
    'le', '2019-2020')
165 ,(138, 'Patty Mills', '07/27/2020', 'premium pack', 'layup', 'common', 3134,
    'cc', '2019-2020')

```

```
166 ,(123, 'OG Anunoby', '08/08/2020', 'lace em' up', 'handles', 'common', 1232,  ↗
    'cc', '2019-2020')
167 ,(79, 'Derrick White', '08/07/2020', 'so fresh', 'steal', 'common', 150, 'cc',  ↗
    '2018-2019')
168 ,(12, 'Willie Cauley-Stein', '09/28/2020', 'premium pack', 'assist', 'common',  ↗
    3134, 'le', '2019-2020')
169 ,(20, 'Seth Curry', '07/30/2020', 'so fresh', 'dunk', 'common', 3134, 'cc',  ↗
    '2019-2020')
170 ,(173, 'Doug McDermott', '10/10/2020', 'lace em' up', 'block', 'common', 3326,  ↗
    'cc', '2018-2019')
171 ,(30, 'Brandon Clarke', '10/02/2020', 'premium pack', 'dunk', 'common', 299,  ↗
    'le', '2019-2020')
172 ,(240, 'John Henson', '09/20/2020', 'base set', 'steal', 'common', 3134, 'le',  ↗
    '2018-2019')
173 ,(295, 'Bogdan Bogdanovic', '08/11/2020', 'premium pack', 'dunk', 'common',  ↗
    3134, 'cc', '2019-2020')
174 ,(255, 'Patty Mills', '08/21/2020', 'base set', 'steal', 'common', 2343, 'le',  ↗
    '2019-2020')
175 ,(238, 'Bojan Bogdanovic', '07/23/2020', 'base set', 'handles', 'rare', 150,  ↗
    'cc', '2019-2020')
176 ,(114, 'Donovan Mitchell', '07/22/2020', 'lace em' up', 'assist', 'rare', 299,  ↗
    'cc', '2019-2020')
177 ,(137, 'Pascal Siakam', '10/09/2020', 'metallic hold le', 'jump shot',  ↗
    'common', 3134, 'cc', '2019-2020')
178 ,(46, 'Jae Crowder', '08/07/2020', 'metallic hold le', 'block', 'common', 150,  ↗
    'cc', '2019-2020')
179 ,(36, 'Ish Smith', '07/28/2020', 'so fresh', 'dunk', 'common', 1232, 'cc',  ↗
    '2018-2019')
180 ,(280, 'P.J. Tucker', '10/12/2020', 'lace em' up', 'dunk', 'rare', 345, 'le',  ↗
    '2019-2020')
181 ,(138, 'RJ Barrett', '10/12/2020', 'metallic hold le', 'layup', 'common', 2343,  ↗
    'le', '2019-2020')
182 ,(281, 'Pascal Siakam', '09/15/2020', 'metallic hold le', 'block', 'common',  ↗
    3134, 'cc', '2019-2020')
183 ,(249, 'Doug McDermott', '09/24/2020', 'metallic hold le', 'block', 'common',  ↗
    2432, 'cc', '2019-2020')
184 ,(121, 'Aron Baynes', '09/03/2020', 'so fresh', 'handles', 'common', 2343,  ↗
    'cc', '2019-2020')
185 ,(79, 'Kenneth Faried', '07/29/2020', 'so fresh', 'jump shot', 'common', 1232,  ↗
    'cc', '2019-2020')
186 ,(138, 'Kosta Koufos', '09/27/2020', 'so fresh', 'layup', 'common', 345, 'cc',  ↗
    '2019-2020')
187 ,(275, 'Pascal Siakam', '09/09/2020', 'lace em' up', 'steal', 'rare', 275,  ↗
    'cc', '2019-2020')
188 ,(223, 'Dejounte Murray', '09/11/2020', 'premium pack', 'dunk', 'common', 2432,  ↗
    'le', '2019-2020')
189 ,(292, 'Tyler Zeller', '08/26/2020', 'base set', 'assist', 'rare', 2432, 'cc',  ↗
    '2019-2020')
190 ,(65, 'Denzel Valentine', '10/08/2020', 'base set', 'assist', 'common', 345,  ↗
```

```
'cc', '2018-2019')
191
192 --Lots of rows, but that's bc we want a large inventory of highlights for users to purchase!
193
194 --Now let's insert moments to user's collections that were already purchased from July - November 2020
195
196 insert into user_moments_collection (topshot_user_id, moment_unique_id, purchase_date)
197 values(25, 2953, '8/8/2020')
198 , (18, 2007, '11/19/2020')
199 , (4, 2534, '10/30/2020')
200 , (10, 2505, '7/21/2020')
201 , (1, 2216, '7/13/2020')
202 , (1, 2783, '11/24/2020')
203 , (3, 2385, '11/18/2020')
204 , (14, 2524, '8/7/2020')
205 , (6, 2373, '11/29/2020')
206 , (8, 2155, '11/1/2020')
207 , (1, 2744, '8/20/2020')
208 , (6, 2168, '11/5/2020')
209 , (12, 2204, '7/25/2020')
210 , (10, 2419, '8/27/2020')
211 , (12, 2876, '7/18/2020')
212 , (12, 2737, '8/15/2020')
213 , (4, 2082, '8/8/2020')
214 , (3, 2332, '10/3/2020')
215 , (7, 2063, '8/15/2020')
216 , (18, 2925, '9/5/2020')
217 , (18, 2962, '9/14/2020')
218 , (5, 2681, '7/24/2020')
219 , (16, 2420, '7/15/2020')
220 , (1, 2916, '8/5/2020')
221 , (25, 2150, '7/23/2020')
222 , (18, 2794, '11/19/2020')
223 , (18, 2578, '7/20/2020')
224 , (4, 2675, '8/5/2020')
225 , (8, 2046, '9/28/2020')
226 , (14, 2278, '10/9/2020')
227 , (6, 2686, '10/2/2020')
228 , (14, 2712, '7/13/2020')
229 , (12, 2410, '9/29/2020')
230 , (17, 2395, '7/4/2020')
231 , (10, 2615, '11/23/2020')
232 , (15, 2110, '11/13/2020')
233 , (19, 2283, '11/14/2020')
234 , (3, 2209, '7/6/2020')
235 , (17, 2493, '10/27/2020')
```

```
236 , (11, 2132, '10/15/2020' )
237 , (21, 2343, '8/6/2020' )
238 , (12, 2428, '11/8/2020' )
239 , (4, 2279, '8/25/2020' )
240 , (23, 2223, '11/26/2020' )
241 , (3, 2695, '11/24/2020' )
242 , (7, 2259, '9/15/2020' )
243 , (9, 2598, '9/25/2020' )
244 , (6, 2701, '10/22/2020' )
245 , (18, 2420, '11/11/2020' )
246 , (12, 2681, '11/26/2020' )
247 , (6, 2449, '7/12/2020' )
248 , (10, 2885, '9/30/2020' )
249 , (23, 2299, '11/11/2020' )
250 , (7, 2277, '8/1/2020' )
251 , (24, 2942, '9/30/2020' )
252 , (19, 2353, '8/10/2020' )
253 , (3, 2395, '9/21/2020' )
254 , (12, 2187, '11/28/2020' )
255 , (25, 2383, '8/31/2020' )
256 , (2, 2064, '9/27/2020' )
257 , (11, 2278, '10/31/2020' )
258 , (5, 2130, '9/23/2020' )
259 , (20, 2395, '9/2/2020' )
260 , (23, 2980, '9/4/2020' )
261 , (12, 2207, '8/14/2020' )
262 , (15, 2065, '9/16/2020' )
263 , (5, 2763, '11/2/2020' )
264 , (21, 2085, '11/7/2020' )
265 , (12, 2058, '9/15/2020' )
266 , (23, 2459, '10/25/2020' )
267 , (12, 2360, '11/24/2020' )
268 , (3, 2519, '8/25/2020' )
269 , (6, 2297, '7/18/2020' )
270 , (1, 2269, '10/4/2020' )
271 , (14, 2467, '9/10/2020' )
272 , (13, 2274, '11/27/2020' )
273 , (8, 2211, '10/28/2020' )
274 , (15, 2346, '8/23/2020' )
275 , (9, 2661, '10/2/2020' )
276 , (25, 2846, '9/8/2020' )
277 , (10, 2257, '10/19/2020' )
278 , (14, 2359, '7/6/2020' )
279 , (17, 2568, '9/18/2020' )
280 , (11, 2667, '11/22/2020' )
281 , (2, 2164, '7/18/2020' )
282 , (8, 2298, '10/21/2020' )
283 , (1, 2798, '9/6/2020' )
284 , (13, 2688, '7/30/2020' )
```

```
285 , (20, 2537, '9/8/2020')
286 , (20, 2779, '9/7/2020')
287 , (19, 2010, '9/23/2020')
288 , (5, 2707, '10/3/2020')
289 , (11, 2382, '10/23/2020')
290 , (10, 2021, '8/1/2020')
291 , (5, 2690, '10/14/2020')
292 , (17, 2984, '8/20/2020')
293 , (8, 2054, '8/3/2020')
294 , (15, 2314, '10/3/2020')
295 , (7, 2372, '7/3/2020')
296 , (21, 2254, '9/24/2020')
297
298 SELECT * FROM user_moments_collection
299
300 --Now let's insert information about packs available for purchase where users can buy their moments
301 insert into packs (pack_type, pack_purchase_price , moments_count , drop_number , pack_circulation_status , pack_name)
302 values ('common', '20', '8', 3, 'available', 'base set')
303 , ('rare', '99', '10', 1, 'sold out', 'lace em up')
304 , ('common', '20', '8', 1, 'available', 'base set')
305 , ('rare', '99', '10', 3, 'sold out', 'lace em up')
306 , ('common', '20', '8', 2, 'available', 'base set')
307 , ('rare', '99', '10', 3, 'sold out', 'lace em up')
308 , ('common', '20', '8', 1, 'available', 'base set')
309 , ('common', '20', '8', 1, 'available', 'base set')
310 , ('common', '20', '8', 1, 'available', 'base set')
311 , ('rare', '99', '10', 1, 'sold out', 'lace em up')
312 , ('common', '20', '8', 2, 'available', 'base set')
313 , ('common', '20', '8', 1, 'available', 'base set')
314 , ('common', '20', '8', 3, 'available', 'base set')
315 , ('common', '20', '8', 1, 'available', 'base set')
316 , ('common', '20', '8', 3, 'available', 'base set')
317 , ('common', '20', '8', 2, 'available', 'base set')
318 , ('common', '20', '8', 3, 'available', 'base set')
319 , ('common', '20', '8', 3, 'available', 'base set')
320 , ('common', '20', '8', 3, 'available', 'base set')
321 , ('common', '20', '8', 2, 'available', 'base set')
322
323 --Now that we have our packs for sale, let's add most recent Pack transactions
324 insert into pack_purchase_order (topshot_user_id, pack_id, payment_type)
325 values (13, 16, 'debit card')
326 , (1, 16, 'paypal')
327 , (17, 15, 'ethereum')
328 , (4, 10, 'ethereum')
329 , (5, 9, 'paypal')
330 , (13, 16, 'credit card')
331 , (17, 14, 'paypal')
```



```
332 , (1, 7, 'ethereum')
333
334 --Now let's add moments that were most recently purchased
335 insert into moments_purchase_order (topshot_user_id, moment_unique_id,
    moment_purchase_price , payment_type)
336 values (14, 2591, 1457, 'ethereum')
337 , (10, 2694, 943, 'debit card')
338 , (23, 2277, 1987, 'paypal')
339 , (23, 2865, 2145, 'debit card')
340 , (23, 2660, 1488, 'ethereum')
341 , (15, 2899, 2166, 'paypal')
342 , (11, 2684, 1943, 'credit card')
343 , (19, 2436, 1972, 'credit card')
344
345 -- finally, let's add data to the flow blockchain which validates all our
    transactions
346 insert into flow_blockchain (order_timestamp , pack_id,
    pack_po_transaction_id, moment_po_transaction_id, moment_unique_id)
347 values ('2020-06-09 08:04:02', 9, 7, null, null)
348 , ('2020-03-12 07:15:13', 18, 4, 101, 2079)
349 , ('2020-06-26 11:01:18', 20, 2, 102, 2748)
350 , ('2020-07-24 01:03:28', 4, 3, 105, 2179)
351 , ('2019-12-05 20:20:28', 15, 1, null, null)
352 , ('2020-01-22 08:41:14', 2, 6, 107, 2160)
353 , ('2020-06-05 15:30:05', 19, 3, 104, 2497)
354 , ('2020-07-18 02:33:49', 2, 1, 107, 2576)
355 , ('2020-07-14 11:47:55', 16, 8, 104, 2149)
356 , ('2019-12-15 22:47:53', 6, 4, 104, 2764)
357 , ('2020-07-09 22:57:31', 10, 2, 107, 2554)
358 , ('2020-07-29 11:35:46', 18, 6, 102, 2581)
359 , ('2020-11-21 15:19:31', 13, 8, 103, 2202)
360 , ('2020-08-31 04:04:07', 4, 7, 104, 2236)
361 , ('2020-06-12 06:11:02', 3, 7, 102, 2430)
362 , ('2019-12-18 16:00:18', 6, 4, 101, 2133)
363 , ('2020-01-23 09:10:47', 17, 7, 105, 2825)
364 , ('2020-08-16 14:53:31', 6, 4, 108, 2092)
365
366
367 --Create view of User and their moments
368 CREATE VIEW TopShotUserMomentsLibrary
369 AS
370 SELECT
371     topshot_user.topshot_username AS UserName
372     , moments.moment_unique_id as MOMENT
373 FROM user_moments_collection
374 INNER JOIN user_moments_collection ON user_moments_collection.topshot_user_id =
    topshot_user.topshot_user_id
375 GO
376
```

```
377 -- Data Question #1: From our early adopters cohort, who owns the most moments and how many?
378
379 SELECT topshot_user.user_join_date, topshot_user.topshot_username,
        topshot_user.user_team, topshot_user.moments_count
380 FROM topshot_user
381 WHERE moments_count > 1000
382 ORDER BY topshot_user.moments_count DESC;
383
384 -- Data Question #2: From the most recent pack purchase order transactions,
        what pack was the best selling?
385
386 SELECT packs.pack_name, packs.pack_purchase_price,
        topshot_user.topshot_username
387 FROM topshot_user
388 INNER JOIN (packs INNER JOIN pack_purchase_order
389 ON packs.pack_id = pack_purchase_order.pack_id)
390 ON topshot_user.topshot_user_id = pack_purchase_order.topshot_user_id;
391
392
393 -- Data Question #3: From most recent moment order transactions, who spent the
        most money and how much?
394 SELECT topshot_user.topshot_username, Count
        (moments_purchase_order.payment_type) AS CountOfpayment_type
395 , Sum(moments_purchase_order.moment_purchase_price) AS
        SumOfmoment_purchase_price
396 FROM topshot_user INNER JOIN moments_purchase_order
397 ON topshot_user.topshot_user_id = moments_purchase_order.topshot_user_id
398 GROUP BY topshot_user.topshot_username
399 ORDER BY Count(moments_purchase_order.payment_type);
400
401 -- Data Question #4: From our most recent moment purchases, what was the
        average transaction per moment type by each user?
402 SELECT topshot_user.topshot_username,
403 Avg(moments_purchase_order.moment_purchase_price) AS
        AvgOfmoment_purchase_price, moments.moment_type
404 FROM moments INNER JOIN (topshot_user INNER JOIN moments_purchase_order
405 ON topshot_user.topshot_user_id = moments_purchase_order.topshot_user_id)
406 ON moments.moment_unique_id = moments_purchase_order.moment_unique_id
407 GROUP BY topshot_user.topshot_username, moments.moment_type
408 ORDER BY topshot_user.topshot_username;
409
410 --Data Question #5: What was the most popular payment type from the most recent
        transactions?
411 SELECT Count(moments_purchase_order.moment_po_transaction_id)
412 AS CountOfmoment_po_transaction_id, moments_purchase_order.payment_type
413 FROM moments_purchase_order
414 GROUP BY moments_purchase_order.payment_type;
415
```