

STORED PROCEDURE

```
1 CREATE DEFINER='root'@'%' PROCEDURE `Result`(IN Keyword VARCHAR(255), IN Category_id INT, IN Region VARCHAR(255))
2 BEGIN
3     DECLARE done int default 0;
4     DECLARE B_Power int default 0;
5     DECLARE B_Rating VARCHAR(255) default ' ';
6     DECLARE CurrTag VARCHAR(255);
7     DECLARE CurrLikes INT;
8     DECLARE CurrViews INT;
9     DECLARE CurrVideos INT;
10    -- DECLARE CurrTopVideo VARCHAR(255);
11
12    DECLARE cur CURSOR FOR (select t.tag_name, t.total_likes, t.total_views, t.total_videos
13        from Tags t join UsedBy s on(t.tag_name=s.tag_name) join TrendingIn ti on (s.video_id=ti.video_id)
14        where ti.region_abbrev=Region AND s.video_id = ANY(select sub.video_id from UsedBy sub where sub.tag_name like CONCAT('%', Keyword, '%'))
15        group by t.tag_name
16        order by t.total_views DESC
17        limit 20);
18
19    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
20
21    #table stuff
22    DROP TABLE IF EXISTS ReturnTable;
23
24    CREATE TABLE ReturnTable (
25        TagName VARCHAR(255),
26        -- Top_video VARCHAR(255),
27        Total_views INT,
28        Total_likes INT,
29        Total_videos INT,
30        Rating VARCHAR(255)
31    );
32
33    DROP TEMPORARY TABLE IF EXISTS TopCategories;
34
35    CREATE TEMPORARY TABLE TopCategories AS
36    SELECT category_id
37    FROM Trendify.Category u join Trendify.Video v on (u.category_id=v.category_id)
38    Group by category_id
39    Order by sum(view_count) DESC, sum(likes) DESC, sum(dislikes)
40    limit 5;
41
42    #cursor stuff
43    open cur;
44
45    REPEAT
46    Fetch cur into CurrTag, CurrLikes, CurrViews, CurrVideos;
47
48    SELECT Count(distinct v.category_id)
49    INTO B_Power
50    FROM Video v
51    WHERE v.tags like CONCAT('%', CurrTag, '%') and v.category_id IN (Select c.category_id from TopCategories c);
52
53    -- SELECT video_id
54    -- INTO CurrTopVideo
55    -- From Video v
56    -- WHERE v.tags like CONCAT('%', CurrTag, '%')
```

```

57 |         -- Order By v.view_count
58 |         -- limit 1;
59 |
60 |         IF CurrViews >= 3000000 OR B_Power >= 3 THEN
61 |             SET B_Rating = "A";
62 |         ELSEIF CurrViews >= 1500000 OR B_Power >= 2 THEN
63 |             SET B_Rating = "B";
64 |         ELSEIF CurrViews >= 500000 OR B_Power >= 1 THEN
65 |             SET B_Rating = "C";
66 |         ELSEIF CurrViews < 500000 THEN
67 |             SET B_Rating = "D";
68 |         END IF;
69 |
70 |         INSERT INTO ReturnTable(TagName, Total_views, Total_likes, Total_videos, Rating)
71 |         VALUES(CurrTag, CurrViews, CurrLikes, CurrVideos, B_Rating);
72 |
73 |         -- INSERT INTO ReturnTable(TagName, Top_video, Total_views, Total_likes, Total_videos, Rating)
74 |         -- VALUES(CurrTag, CurrTopVideo, CurrViews, CurrLikes, CurrVideos, B_Rating);
75 |
76 |     UNTIL done
77 |     END REPEAT;
78 |
79 |     # selecting table results
80 |     SELECT distinct *
81 |     FROM ReturnTable
82 |     ORDER BY Rating, Total_views DESC, Total_likes DESC, Total_videos DESC
83 |     LIMIT 20;
84 |
85 |
86 |     CLOSE cur;
87 |
88 | END

```

TRIGGER

```

1 | DELIMITER //
2 | CREATE TRIGGER Fav
3 |     BEFORE UPDATE ON Favorite
4 |     FOR EACH ROW
5 |     BEGIN
6 |
7 |         DECLARE currRating INT default 0;
8 |
9 |         SELECT rating
10 |        INTO currRating
11 |        FROM Favorite
12 |        WHERE old.tag_name = new.tag_name;
13 |
14 |        IF currRating >= 10 and currRating <= 50 THEN
15 |            SET new.rating = old.rating + 2;
16 |        ELSEIF currRating > 50 THEN
17 |            SET new.rating = old.rating + 4;
18 |        ELSEIF currRating < 10 THEN
19 |            SET new.rating = old.rating + 1;
20 |        END IF;
21 |    END
22 | //
23 | DELIMITER ;
24 |

```