METHODOLOGY

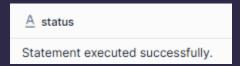
SNOWFLAKE

BRIGHT_TV | Ksetlhatsoe@gmail.com



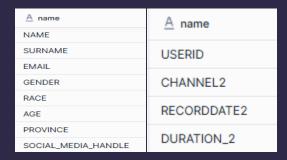
1. Setting the working database and schema

USE DATABASE Bright; USE SCHEMA BRIGHT_TV;



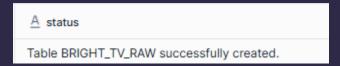
2. Inspecting column names to confirm Social Media Handles and Duration 2

DESC TABLE USER_PROFILES; DESC TABLE VIEWERSHIP;



3. Creating the joined base table and This combines user profile info with viewership sessions

CREATE OR REPLACE TABLE BRIGHT_TV_RAW AS **SELECT** up.UserID, up.Name, up.Surname, up.Email, up.Gender, up.Race, up.Age, up.Province, up.SOCIAL_MEDIA_HANDLE, vw.Channel2, vw.RecordDate2. vw.DURATION 2 FROM USER_PROFILES up FULL OUTER JOIN VIEWERSHIP vw ON up.UserID = vw.UserID;



4. Adding date and time columns for transformation

ALTER TABLE BRIGHT_TV_RAW ADD COLUMN record_time TIME; ALTER TABLE BRIGHT_TV_RAW ADD COLUMN record_date DATE; ALTER TABLE BRIGHT_TV_RAW ADD COLUMN record_date_text VARCHAR; ALTER TABLE BRIGHT_TV_RAW ADD COLUMN record_time_text VARCHAR;

(return a list of all current columns) DESC TABLE BRIGHT_TV_RAW;

RECORD_DATE_TEXT	VARCHAR(1677721	COLUMN	Υ	null
RECORD_TIME_TEXT	VARCHAR(1677721	COLUMN	Υ	null

5. Extracting and converting date

UPDATE BRIGHT_TV_RAW
SET record_date_text = REPLACE(LEFT(RecordDate2, 10), '/', '-');

UPDATE BRIGHT_TV_RAW
SET record_date = TO_DATE(record_date_text, 'YYYY-MM-DD');

	# number of rows updated	# number of multi-joined rows updated
1	10989	0

6. Extracting and converting time

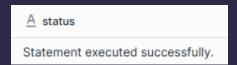
UPDATE BRIGHT_TV_RAW
SET record_time_text = RIGHT(RecordDate2, 5);

UPDATE BRIGHT_TV_RAW
SET record_time = TO_TIME(record_time_text);

# number of rows updated :	# number of multi-joined rows updated
10989	0

7. Cleaning up temporary columns

ALTER TABLE BRIGHT_TV_RAW DROP COLUMN record_date_text; ALTER TABLE BRIGHT_TV_RAW DROP COLUMN record_time_text; ALTER TABLE BRIGHT_TV_RAW DROP COLUMN RecordDate2;



8. Creating enriched table with buckets & calculations

CREATE OR REPLACE TABLE BRIGHTLIGHT_TV AS SELECT *,



Table BRIGHTLIGHT_TV successfully created.

-- TIME BUCKET

CASE

WHEN record_time BETWEEN '00:00:01' AND '04:00:00' THEN '12AM-4AM' WHEN record_time BETWEEN '04:00:01' AND '08:00:00' THEN '4AM-8AM' WHEN record_time BETWEEN '08:00:01' AND '12:00:00' THEN '8AM-12PM' WHEN record_time BETWEEN '12:00:01' AND '16:00:00' THEN '12PM-4PM' WHEN record_time BETWEEN '16:00:01' AND '20:00:00' THEN '4PM-8PM' ELSE '8PM-12AM'

END AS time_bucket FROM BRIGHT_TV_RAW;

A DURATION_2	© RECORD_TIME	© RECORD_DATE	A TIME_BUCKET
00:02:00	18:09:00	2016-01-04	4PM-8PM
00:10:06	09:04:00	2016-03-30	8AM-12PM
00:00:42	17:01:00	2016-03-30	4PM-8PM
00:00:10	15:23:00	2016-03-25	12PM-4PM
00:01:33	17:18:00	2016-02-02	4PM-8PM
00:00:10	16:56:00	2016-03-18	4PM-8PM
00:17:15	04:25:00	2016-03-20	4AM-8AM
00:01:13	06:43:00	2016-02-03	4AM-8AM

-- AGE GROUP

SELECT *,

CASE

WHEN Age BETWEEN 0 AND 12 THEN 'Child'

WHEN Age BETWEEN 13 AND 19 THEN 'Teen'

WHEN Age BETWEEN 20 AND 24 THEN 'Young Adult'

WHEN Age BETWEEN 25 AND 34 THEN 'Emerging Adult'

WHEN Age BETWEEN 35 AND 49 THEN 'Adult'

WHEN Age BETWEEN 50 AND 59 THEN 'Advanced Middle Age'

WHEN Age BETWEEN 60 AND 74 THEN 'Elderly'

ELSE 'Advanced Elderly'

END AS age_group

FROM BRIGHT_TV_RAW;

A DURATION_2	© RECORD_TIME	© RECORD_DATE	A AGE_GROUP
00:02:00	18:09:00	2016-01-04	Emerging Adult
00:10:06	09:04:00	2016-03-30	Child
00:00:42	17:01:00	2016-03-30	Emerging Adult
00:00:10	15:23:00	2016-03-25	Advanced Middle Aç
00:01:33	17:18:00	2016-02-02	Child
00:00:10	16:56:00	2016-03-18	Child
00:17:15	04:25:00	2016-03-20	Child
00:01:13	06:43:00	2016-02-03	Child

-- MONTH

SELECT *,

CASE

WHEN record_date BETWEEN '2016-01-01' AND '2016-01-31' THEN 'JAN' WHEN record_date BETWEEN '2016-02-01' AND '2016-02-29' THEN 'FEB' WHEN record_date BETWEEN '2016-03-01' AND '2016-03-31' THEN 'MAR' ELSE 'APR'

END AS month

FROM BRIGHT_TV_RAW;

A DURATION_2	L RECORD_TIME	© RECORD_DATE	A MONTH_
00:02:00	18:09:00	2016-01-04	JAN
00:10:06	09:04:00	2016-03-30	MAR
00:00:42	17:01:00	2016-03-30	MAR
00:00:10	15:23:00	2016-03-25	MAR
00:01:33	17:18:00	2016-02-02	FEB
00:00:10	16:56:00	2016-03-18	MAR
00:17:15	04:25:00	2016-03-20	MAR
00:01:13	06:43:00	2016-02-03	FEB

-- DAY OF WEEK

SELECT*,

DAYNAME(record_date) AS day_of_week, FROM BRIGHT_TV_RAW;

A DURATION_2	© RECORD_TIME	© RECORD_DATE	A DAY_OF_WEEK
00:02:00	18:09:00	2016-01-04	Mon
00:10:06	09:04:00	2016-03-30	Wed
00:00:42	17:01:00	2016-03-30	Wed
00:00:10	15:23:00	2016-03-25	Fri
00:01:33	17:18:00	2016-02-02	Tue
00:00:10	16:56:00	2016-03-18	Fri
00:17:15	04:25:00	2016-03-20	Sun
00:01:13	06:43:00	2016-02-03	Wed

--TRY_TO_TIME INSTEAD OF TO_TIME

SELECT*,

DATE_PART('HOUR', TRY_TO_TIME(DURATION_2)) * 3600 +
DATE_PART('MINUTE', TRY_TO_TIME(DURATION_2)) * 60 +
DATE_PART('SECOND', TRY_TO_TIME(DURATION_2)) AS duration_seconds
FROM BRIGHT_TV_RAW;

A DURATION_2	© RECORD_TIME	© RECORD_DATE	# DURATION_SECONDS
00:02:00	18:09:00	2016-01-04	120
00:10:06	09:04:00	2016-03-30	606
00:00:42	17:01:00	2016-03-30	42
00:00:10	15:23:00	2016-03-25	10
00:01:33	17:18:00	2016-02-02	93
00:00:10	16:56:00	2016-03-18	10
00:17:15	04:25:00	2016-03-20	1035
00:01:13	06:43:00	2016-02-03	73

9. Sampling queries for Power BI insights

--VIEWERSHIP BY PROVINCE

SELECT Province, COUNT(UserID) AS number_of_sessions FROM BRIGHTLIGHT_TV WHERE DURATION_2 > '00:00:00' GROUP BY Province ORDER BY number_of_sessions DESC;

	A PROVINCE	# NUMBER_OF_SESSIONS
1	Gauteng	3352
2	Western Cape	1709
3	Kwazulu Natal	898
4	Mpumalanga	854
5	Limpopo	687
6	Eastern Cape	626
7	North West	308
8	Free State	267
9	None	227

--VIEWERSHIP BY RACE AND GENDER

SELECT Race, Gender, COUNT(UserID) AS number_of_sessions FROM BRIGHTLIGHT_TV GROUP BY Race, Gender;

	A RACE	A GENDER	# NUMBER_OF_SESSIONS
1	indian_asian	male	1494
2	black	female	532
3	white	male	1203
4	null	female	4
5	null	male	11
6	coloured	male	1520
7	None	None	788
8	coloured	female	139
9	other	male	100

--DISTINCT VIEWERS KPI

SELECT COUNT(DISTINCT UserID) AS total_unique_viewers FROM BRIGHTLIGHT_TV;

	# TOTAL_UNIQUE_VIEWERS
1	5375

--VIEWERSHIP BY AGE GROUP

SELECT age_group, SUM(duration_seconds) AS total_watch_time FROM BRIGHTLIGHT_TV GROUP BY age_group ORDER BY total_watch_time DESC;

	A AGE_GROUP	# TOTAL_WATCH_TIME
1	Advanced Elderly	2803
2	Child	204426
3	Young Adult	565208
4	Advanced Middle Age	205125
5	Adult	1852034
6	Elderly	38603
7	Emerging Adult	2366486
8	Teen	158920

--DAILY VIEWERSHIP TREND

SELECT record_date, COUNT(UserID) AS sessions FROM BRIGHTLIGHT_TV GROUP BY record_date ORDER BY record_date;

	© RECORD_DATE	# SESSIONS
1	2016-01-01	70
2	2016-01-02	73
3	2016-01-03	72
4	2016-01-04	60
5	2016-01-05	85
6	2016-01-06	85
7	2016-01-07	67
8	2016-01-08	84
9	2016-01-09	79

--VIEWERSHIP BY TIME BUCKET

SELECT time_bucket, COUNT(UserID) AS sessions FROM BRIGHTLIGHT_TV GROUP BY time_bucket ORDER BY sessions DESC;

	A TIME_BUCKET	# SESSIONS
1	12PM-4PM	2536
2	4PM-8PM	2482
3	8AM-12PM	2279
4	8PM-12AM	2126
5	4AM-8AM	1286
6	12AM-4AM	280

--TOP 10 MOST WATCHED CONTENT

SELECT Channel2, SUM(duration_seconds) AS total_watch_time FROM BRIGHTLIGHT_TV GROUP BY Channel2 ORDER BY total_watch_time DESC LIMIT 10;

	A CHANNEL2	# TOTAL_WATCH_TIME
1	null	null
2	ICC Cricket World Cup 2011	1483652
3	Supersport Live Events	1127516
4	Channel O	693921
5	Trace TV	690927
6	SuperSport Blitz	326446
7	Boomerang	263733
8	Cartoon Network	252917
9	CNN	243999

-- DAY OF WEEK VIEWERSHIP

SELECT day_of_week, COUNT(UserID) AS sessions FROM BRIGHTLIGHT_TV GROUP BY day_of_week ORDER BY sessions DESC;

	A DAY_OF_WEEK	# SESSIONS
1	Sat	1655
2	Fri	1642
3	Wed	1539
4	Thu	1471
5	Sun	1398
6	Tue	1321
7	null	989
8	Mon	974

-- NEW VS RETURNING USERS

```
WITH first_sessions AS (
 SELECT UserID, MIN(record_date) AS first_seen
 FROM BRIGHTLIGHT_TV
 GROUP BY UserID
user_activity AS (
 SELECT b.UserID, COUNT(*) AS total_sessions, f.first_seen
 FROM BRIGHTLIGHT_TV b
 JOIN first_sessions f ON b.UserID = f.UserID
 GROUP BY b.UserID, f.first_seen
SELECT
 CASE
   WHEN total_sessions = 1 THEN 'New'
   ELSE 'Returning'
 END AS user_type,
 COUNT(*) AS user_count
FROM user_activity
GROUP BY user_type;
```

A USER_TYPE	# USER_COUNT
Returning	2139
New	3236