

CNIT 372 Final Report

Group 4 chose to explore the social media platform, Youtube from the perspective of a content creator. In doing so, we wanted to understand how content creators utilize user data and statistics to improve their viewership and popularity. The questions range from those about measuring user attention span through video duration to predicting when other artists are releasing content to find optimal release dates. Each question paints a unique outlook on how creators use consumer analytics to curate quality content. We answered these questions using the knowledge gathered during our time in this course, CNIT 372. Calling upon our knowledge of concepts like cursors, functions, Packages, and triggers we were able to unravel how content creators employ user search history, genre popularity, and other user analytics to keep up with trends and maintain. In this report, we included the actual script within the function, however in the GitHub repository, you can see the entire package and functions all together.

Our database consists of 4 tables. These tables are users, videos, search history and watch history. The User table consists of two rows, userid and username. Userid is the primary key of this table. The Videos table consists of six rows, video_id, title, genre, release_date, duration_seconds, artist_youtuber. Video_id is the primary key of the videos table. The Search History table consists of six rows, search_id, userid, search_term, search_date, video_id, genre. This table has two foreign keys, userid and video_id. Our final table, the Watch History table, consists of four rows, history_id, user_id, video_id, and watch_date. The table utilizes two foreign keys, user_id and video_id.

From the option chosen, we generated 10 questions that can be used to gain meaningful insights that can be utilized by content creators on the platform. The first question we asked was "What are the peak hours for viewership on YouTube?". We were able to see the hour of the day

that gets the most views from users on the platform. This can be helpful because youtubers can decide to release their content at hours that have the most activity and they can get more engagement on their channels. From our data, we could see that 10PM was the most popular time for watching videos on youtube, so content creators can plan to upload videos from 7PM onwards so that it is already out when the most people are on the platform. The code can be seen below and it is getting the number of hours a day and the amount of views that occurred during that hour from all users who's watch history we generated.

Question 1

```
SELECT
    TO_CHAR(watch_date, 'HH24') AS watching_times,
    COUNT(*) AS viewership_count
FROM
    watch_history
GROUP BY
    TO_CHAR(watch_date, 'HH24')
ORDER BY
    viewership_count DESC;
```

The results were:

WA	VIEWERSHIP_COUNT
--	-----
22	55
14	53
12	45
17	43
19	39
16	38
18	35
09	33
11	29
20	27
10	21

WA	VIEWERSHIP_COUNT
--	-----
08	19
21	19
13	12
15	10
07	8
23	4

17 rows selected.

The second question we asked was “What are the most watched videos on YouTube and what time are they watched?”. From this, we were able to see popular user favorites. This is insightful because it can help content creators see what is something that is a big hit amongst users and having the most recent watch time can let them know if that video is still trending and getting views or if it has already passed. As seen from our data, the most popular video is Culinary Adventure: Global Cuisine which tells content creators that cooking adventure types of videos are trending in popularity amongst users and they can try to release a related video to garner attention.

The code can be seen below and it counts the number of times that a video has been watched by counting views and then finds when the most recent time the video was watched.

```
SELECT *
FROM (
    SELECT
        v.title AS most_watched_video,
        COUNT(*) AS watch_count,
        MAX(h.watch_date) AS most_recent_watch_time
    FROM
        watch_history h
    JOIN
        videos v ON h.video_id = v.video_id
    GROUP BY
        v.title
    ORDER BY
        watch_count DESC
)
WHERE ROWNUM = 1;
```

The results were:

```
MOST_WATCHED_VIDEO
WATCH_COUNT MOST_RECENT_WATCH_TIME
-----
-----
Culinary Adventure: Global Cuisine
2 25-SEP-23 10.15.00.000000000 AM
```

The third question asked was “Based on the top 5 most popular genres searched, how would you say the users utilize youtube as a resource ?” This can be insightful in providing content creators insight on how users are using the platform and what it is that they are looking for. This way, they can make sure that their content hits those markings and is something that people want to see.

The code can be seen below and is counting the amount of times that a genre appeared in the user search history table. It displayed the top 5 genres. For our data, you can see that the top 5 genres are cooking, comedy, music, technology, and science.

```
SELECT
    genre,
    COUNT(*) AS search_count
FROM
    search_history
GROUP BY
    genre
ORDER BY
    search_count DESC
FETCH FIRST 5 ROWS ONLY;
```

The results are:

GENRE	SEARCH_COUNT
Cooking	65
Comedy	64
Music	55
Technology	35
Science	20

The fourth question we asked was “Which category of videos is most popular and which time of year is it watched?” This is insightful to users because it allows for content creators to see not only the categories that are popular during which months, but also provides the count of views that the specific category got during the month and year. As seen from our data below, the most popular genre was DIY during April of 2023 with 12 views in that category. This is an example using our dummy data, which has over 60 genres.

The code can be seen below and it counts the number of times each video was watched, then takes the genre from them. The VideoPopularity function calculates watch count for the

combination of video category and month and with the outer select, it is choosing the one with the highest watch count per month.

```
WITH VideoPopularity AS (  
    SELECT  
        v.genre AS video_category,  
        TO_CHAR(w.watch_date, 'YYYY-MM') AS watch_month,  
        COUNT(*) AS watch_count  
    FROM  
        watch_history w  
    JOIN  
        videos v ON w.video_id = v.video_id  
    GROUP BY  
        v.genre, TO_CHAR(w.watch_date, 'YYYY-MM')  
)  
SELECT  
    video_category,  
    watch_month,  
    watch_count  
FROM (  
    SELECT  
        video_category,  
        watch_month,  
        watch_count,  
        ROW_NUMBER() OVER (PARTITION BY watch_month ORDER BY  
watch_count DESC) AS pop  
    FROM  
        VideoPopularity  
)  
WHERE  
    pop = 1  
ORDER BY  
    watch_count DESC, watch_month, video_category;
```

The results were:

```
VIDEO_CATEGORY  
WATCH_M WATCH_COUNT
```

```
-----  
-----  
DIY  
2023-04          12  
Gaming  
2023-01          11  
Cooking  
2023-02          10  
Technology  
2023-03          10  
Gaming  
2018-12           2  
Cooking  
2023-09           2  
Sports  
2018-03           1
```

Travel	
2018-04	1
Music	
2018-05	1
Education	
2018-06	1
Funk	
2018-07	1

VIDEO_CATEGORY
WATCH_M WATCH_COUNT

Sports	
2018-08	1
Sports	
2018-10	1
Punk	
2019-01	1
RndB	
2019-02	1
Entertainment	
2019-04	1
Education	
2019-05	1
Entertainment	
2019-07	1
Food nd Cooking	
2019-08	1
Sports	
2019-09	1
Sports	
2019-10	1
J-Pop	
2019-11	1

VIDEO_CATEGORY
WATCH_M WATCH_COUNT

Gaming	
2020-02	1
Philosophy	
2020-03	1
Entertainment	
2020-04	1
Fashion	
2020-05	1
Bluegrass	
2020-07	1
Documentary	
2020-08	1
Gaming	
2020-09	1
Sports	
2020-11	1

Thriller	
2020-12	1
Sports	
2021-01	1
Sports	
2021-02	1

VIDEO_CATEGORY
WATCH_M WATCH_COUNT

Alternative	
2021-03	1
Entertainment	
2021-04	1
Sports	
2021-06	1
Horror	
2021-07	1
Gaming	
2021-10	1
Entertainment	
2021-11	1
History	
2021-12	1
Gaming	
2022-05	1
Entertainment	
2022-06	1
Education	
2022-07	1
Nature	
2022-08	1

VIDEO_CATEGORY
WATCH_M WATCH_COUNT

Fashion	
2022-10	1
K-Pop	
2023-05	1
Fitness	
2023-06	1
Travel	
2023-07	1
Gaming	
2023-08	1
Art	
2023-10	1
Art	
2023-11	1
Education	
2023-12	1

52 rows selected.

Question five. The query identifying upcoming content releases by popular artists or YouTubers is helpful to a content creator for strategic planning. It provides insights into future releases, allowing creators to synchronize their content schedule, explore collaboration opportunities, and engage with their audience effectively. we got no rows are selected and decided to leave this as an answer to show that there might be a few things concerning a question and why it cant be answered, this may be due to release dates being in the past, lack of popularity criteria, data quality issues, or misalignment with the defined criteria. It underscores the importance of accurate data and thoughtful planning to ensure the query aligns with the content creator's objectives and audience expectations.

```
CREATE OR REPLACE TRIGGER upcoming_releases_trigger
AFTER INSERT ON videos
FOR EACH ROW
BEGIN
    IF :new.release_date >= SYSDATE THEN
        INSERT INTO watch_history (user_id, video_id, watch_date)
        VALUES (:new.user_id, :new.video_id, :new.release_date);
    END IF;
END;
/

SELECT
    user_id,
    video_id,
    watch_date AS release_date
FROM
    watch_history
WHERE
    watch_date >= SYSDATE;
```

The results are:

no rows selected.

Question six was “How does the genre of videos change during different times of the year?” This provides a different type of insight as it focuses both on the popularity of genres and the rotational content that occurs each season, holiday, etc.. By getting information on how video genres change throughout the year, content creators have the opportunity to recognize seasonal trends and can create content that fits. This information would also help system managers determine which content will be more popular and when, helping them prepare for high page traffic on certain pages and within certain genres. The data could be used to monetize and advertise for different businesses, as well.

The code can be seen below and it counts the number of videos per genre and groups them by the month they were released.

```
SELECT
    EXTRACT(MONTH FROM release_date) AS month,
    genre,
    COUNT(*) AS genre_count
FROM
    videos
GROUP BY
    EXTRACT(MONTH FROM release_date),
    genre
ORDER BY
    EXTRACT(MONTH FROM release_date),
    genre;
```

The results are:

	MONTH	GENRE	GENRE_COUNT
	1	Acoustic	1
	1	Art	1
	2	Comedy	1
	2	Cooking	1
	3	DIY	1
	5	Education	1
	3	Entertainment	1
	3	Fitness	1
	1	Food and Cooking	1
	1	Gaming	1
	3	Indie	1
	1	Literature	1

3	1 Music
1	1 Nature
1	1 Photography
1	1 Reggae
2	1 Science
4	1 Sports
1	1 Technology
7	1 Travel
1	2 Art
2	2 Comedy

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

3	2 Cooking
1	2 Documentary
4	2 Education
3	2 Entertainment
1	2 Fitness
1	2 Funk
8	2 Gaming
1	2 History
2	2 Literature
7	2 Music
1	2 Science nd Technology

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

3	2 Sports
---	----------

1	2 Street
2	2 Technology
1	2 Trap
1	2 Travel
1	2 Wellness
1	3 Americana
1	3 Art
1	3 Books nd Literature
1	3 Comedy
1	3 Cooking

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

5	3 DIY
1	3 Drama
2	3 Education
2	3 Entertainment
1	3 Film
2	3 Fitness
4	3 Gaming
3	3 Literature
2	3 Nature
1	3 RndB
1	3 Science Fiction

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

2	3 Sports
---	----------

3	3 Technology
3	3 Travel
1	4 Art
3	4 Comedy
4	4 Cooking
5	4 DIY
2	4 Drama
1	4 Education
2	4 Entertainment
2	4 Fitness
2	

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

4	4 Gaming
1	4 Gospel
1	4 History
1	4 Jazz Funk
1	4 Literature
1	4 Music
1	4 Philosophy
2	4 Photography
1	4 Science
2	4 Sports
3	4 Technology

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

1	4 Wellness
---	------------

2	5 Adventure
6	5 Art
3	5 Cooking
3	5 DIY
2	5 Education
2	5 Entertainment
2	5 Fitness
3	5 Gaming
3	5 Literature
1	5 Psychedelic

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

1	5 Punk
1	5 Romance
2	5 Science
1	5 Science nd Technology
3	5 Sports
2	5 Theatre
3	5 Travel
1	5 Wellness
1	6 Art
2	6 Comedy
1	6 Cooking

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

2	6 Education
---	-------------

2	6 Entertainment
1	6 Fantasy
1	6 Film
2	6 Fitness
6	6 Gaming
1	6 Hardcore
1	6 History
2	6 Music
1	6 News nd Politics
1	6 Photography

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

5	6 Science
1	6 Science nd Technology
1	6 Sports
1	6 Talk Show
2	6 Technology
1	6 World
1	7 Alternative
1	7 Art
1	7 Comedy
3	7 Cooking
1	7 Country

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

4	7 DIY
---	-------

1	7 Documentary
2	7 Education
3	7 Entertainment
1	7 Fashion
1	7 Film
2	7 Fitness
3	7 Gaming
1	7 Health nd Fitness
2	7 Literature
4	7 Music

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

2	7 Nature
2	7 Sports
1	7 Thriller
4	7 Travel
1	7 Wellness
2	8 Art
3	8 Comedy
2	8 Cooking
6	8 DIY
2	8 Education
2	8 Entertainment

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

1	8 Film nd Animation
---	---------------------

4	8 Fitness
5	8 Gaming
1	8 History
1	8 Horror
1	8 J-Pop
1	8 Literature
1	8 Movie Reviews
4	8 Music
1	8 Photography
1	8 Science

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

2	8 Sports
---	----------

1	8 Technology
---	--------------

1	9 Art
---	-------

1	9 Bluegrass
---	-------------

1	9 Comedy
---	----------

2	9 Cooking
---	-----------

5	9 DIY
---	-------

3	9 Education
---	-------------

3	9 Entertainment
---	-----------------

1	9 Fashion
---	-----------

3	9 Fitness
---	-----------

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

4	9 Gaming
---	----------

6	9 Literature
1	9 Movie Reviews
4	9 Music
1	9 Pop
1	9 Rock
1	9 Science Fiction
3	9 Sports
1	9 Technology
1	9 Theatre
3	9 Travel

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

2	10 Art
1	10 Beauty and Fashion
3	10 Comedy
1	10 Cooking
5	10 DIY
1	10 Dance
1	10 Disco
2	10 Education
2	10 Entertainment
1	10 Fantasy
1	10 Fitness

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

5	10 Gaming
---	-----------

1	10 Hip-Hop
5	10 Literature
1	10 Movie Reviews
3	10 Music
1	10 Science
2	10 Sports
3	10 Technology
1	10 Wellness
1	11 Action
2	11 Art

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

1	11 Blues
1	11 Classical
3	11 Comedy
2	11 Cooking
3	11 DIY
1	11 Drama
1	11 Drum and Bass
5	11 Education
3	11 Entertainment
2	11 Fashion
2	11 Film

MONTH	GENRE	GENRE_COUNT
-------	-------	-------------

3	11 Fitness
---	------------

4	11 Gaming
3	11 Literature
1	11 Movie Reviews
2	11 Music
2	11 Nature
3	11 Sports
1	11 Technology
2	11 Travel
1	11 Wellness
2	12 Art

MONTH	GENRE	GENRE_COUNT

1	12 Books nd Literature
4	12 Comedy
2	12 Cooking
1	12 Documentary
3	12 Education
1	12 Electronic
3	12 Entertainment
1	12 Film nd Animation
2	12 Fitness
4	12 Gaming
2	12 History

MONTH	GENRE	GENRE_COUNT

1	12 Horror
---	-----------

1	12 K-Pop
1	12 Literature
1	12 Metal
1	12 Movie Reviews
1	12 Music
3	12 Sports
3	12 Technology
2	12 Travel
1	

240 rows selected.

Question 7 is "What are the top 11 trending genres watched?" This data could help content creators come up with names and tags that are correlated with trending keywords. This would make their content appear within these trending genres and searches, improving their numbers and viewerships.

The code seen below selects the watched genres and counts the videos watched to find the top 11 trending genres watched.

```
SELECT
    v.genre AS most_popular,
    count(h.User_id) AS total_views
FROM
    watch_history h
JOIN
    videos v ON h.video_ID = v.video_ID
GROUP BY
    genre
ORDER BY
    total_views DESC;
```

```
-MOST_POPULAR
TOTAL_VIEWS
```

```
-----
-----
Gaming
53
DIY
37
Music
32
Education
31
Sports
30
```

```

Entertainment
30
Cooking
29
Literature
28
Comedy
26
Art
24
Fitness
24
*/

```

The next one is question 8 and is “What type of video duration content are users mostly consuming?” This is insightful to content creators because it gives them knowledge of the attention span of the primary users on the platform. By seeing if reels or more longer style videos are popular, they are able to make sure their videos fit that sweet spot.

The code can be seen below and works by gathering the average watch duration of users from their watch history in seconds.

```

SELECT
    u.user_id,
    u.username,
    AVG(v.duration_seconds) AS average_duration
FROM
    users u
JOIN
    watch_history wh ON u.user_id = wh.user_id
JOIN
    videos v ON wh.video_id = v.video_id
GROUP BY
    u.user_id, u.username
ORDER BY
    average_duration DESC;

```

The result is:

```

USER_ID USERNAME
AVERAGE_DURATION
-----
-----
-----
298 dancey_dani
960

```

960	5	sam_jackson
960	258	liam_lagoon
960	23	alex_cooper
900	292	sophia_sculpture
900	241	isabella_illuminate
900	200	EliEpicExploration
900	211	IslaIntrigueInsider
900	30	aaliyah_martin
900	137	ThumbsUpTwist
900	252	william_whimsy

	USER_ID	USERNAME	AVERAGE_DURATION
--	---------	----------	------------------

900	12	linda_carter
900	350	peter_parker
900	341	grace_kelly
900	35	jordan_perez
840	272	maya_moonbeam
840	19	kevin_turner
840	44	zoey_brown
840	261	mia_mirage
840	247	mia_melody

	334	quentin_quotable
810		
	224	SkylarSculptureStory
810		

USER_ID	USERNAME	AVERAGE_DURATION

	216	OwenOdysseyOfficial
810		
	222	SebastianSeekerShow
810		
	42	grace_walker
780		
	28	mia_hill
780		
	233	emma_echo
780		
	376	penny_penguin
765		
	112	RetroReverieReviews
750		
	330	winona_wondrous
750		
	336	bob_marley
750		
	402	parker_peach
740		
	366	freddie_froggy
720		

USER_ID	USERNAME	AVERAGE_DURATION

	250	elijah_enigma
720		
	277	oliver_orbit
720		
	178	TravelerTreasureTrove
720		

720	37	gavin_cook
720	314	lola_labyrinth
720	270	hannah_horizon
720	285	leo_lullaby
720	21	patrick_morgan
720	219	LaylaLifestyleLogs
720	338	diana_prince
720	126	ComedyCanvasCraze

	USER_ID	USERNAME	AVERAGE_DURATION
--	---------	----------	------------------

	245	sophia_serendipity	
720	263	brooke_blossom	
720	290	ella_essence	
720	296	lanie_kozlowski	
720	10	olivia_brown	
720	32	zoey_clark	
720	239	mia_muse	
710	213	MiaMelodicMoments	
700	265	sophie_solar	
693	232	AlexAdventureAwaits	
690	127	UrbanUmbrella360	

USER_ID USERNAME
AVERAGE_DURATION

690	319	nolan_nebula
690	328	olivia_outstanding
690	348	neil_armstrong
690	237	ava_aesthetic
690	295	kenny_daniels
690	322	xena_xerox
690	325	quincy_quintessence
690	4	alice_walker
690	208	VictorVoyageVision
690	332	xavier_xclusive
690	324	ursula_umbra

USER_ID USERNAME
AVERAGE_DURATION

690	289	liam_labyrinth
690	308	quentin_quasar
680	279	max_meadow
675	287	noah_nectar
672	203	GracefulGlobeVibes
660	16	natalie_green

```
        301 daisy_delta
660
        283 owen_odyssey
660
        310 oliver_onyx
660
        305 sasha_sphinx
660
        243 harper_harmony
660
```

```
        USER_ID USERNAME
AVERAGE_DURATION
```

```
-----
-----
-----
        220 NoahNomadNarrative
660
        49 william_nelson
660
        339 elvis_presley
660
        254 ethan_eclipse
660
        226 LeoLearningLegacy
660
        327 daniel_dapper
660
        268 emma_eden
660
        54 da_boys
660
        8 susan_miller
660
        229 MiaMelodicMoments
660
        34 amy_morris
660
```

```
        USER_ID USERNAME
AVERAGE_DURATION
```

```
-----
-----
-----
```

398	logan_lavender
660	
26	hailey_taylor
660	
235	olivia_orbit
660	
39	mason_davis
660	
281	aiden_aurora
660	
171	CodeCraftsmanChronicles
660	
294	eric_cartman
660	
273	nathan_nestle
660	
190	ArtisticAscentAdventures
660	
372	lucas_lyric
645	
187	TechTrendTrailblazing
645	

USER_ID	USERNAME
AVERAGE_DURATION	

378	riley_rainbow
640	
58	TechExplorer365
630	
343	irene_adler
630	
207	LilyLandscapeLife
630	
214	CarterCraftsChronicle
630	
346	leonardo_daVinci
630	
354	tina_turner
630	
230	EvanExplorationEncounter
630	

```
227 LaylaLifestyleLogs
630
347 marilyn_monroe
624
421 isabel_iris
620
```

```
USER_ID USERNAME
AVERAGE_DURATION
```

```
-----
```

```
-----
```

```
-----
```

```
79 StarryEyedExplorer
615
41 nathan_smith
600
323 rhysand_thesecond
600
304 zane_zephyr
600
249 amelia_artistry
600
321 floyd_fantasia
600
198 AdrianArtistryAvenue
600
154 EnchantedEvermore
600
175 ElementalEpicExplorer
600
275 ethan_echo
600
320 willow_wonder
600
```

```
USER_ID USERNAME
AVERAGE_DURATION
```

```
-----
```

```
-----
```

```
-----
```

```
329 nicholas_noteworthy
600
260 nolan_novel
600
```

600	3	bob_jones
600	1	john_doe
600	267	jaxon_journey
600	14	emma_hall
600	256	owen_observatory
600	349	olivia_wilde
600	300	nolan_chris
600	297	tashi_poo
593.333333	169	EnchantedEra
585		

	USER_ID	USERNAME	AVERAGE_DURATION
--	---------	----------	------------------

	471	azriel_amethyst	
585	253	avery_anchor	
585	223	CarterCraftyChronicle	
580	259	zara_zephyr	
576	307	iris_iridescence	
570	210	EvanExplorationEncounter	
570	209	AvaArtisanAdventures	
570	312	nina_nyquist	
570	313	quincy_quantum	
570	316	emma_enigma	
570			

231 IslaIntrigueInsider
570

USER_ID USERNAME
AVERAGE_DURATION

228 VictorVoyageVision
570
72 NatureEnthusiastTV
570
205 PenelopePioneerPage
570
311 isla_illusion
570
417 emma_evergreen
570
201 BellaBeyondBorders
570
206 ChaseCraftyChronicle
570
302 xavier_zeppelin
570
303 amber_azure
570
202 XavierExplorerTV
570
215 SkylarSculptureStory
570

USER_ID USERNAME
AVERAGE_DURATION

345 katherine_jones
570
444 samwise_stardust
560
174 GourmetGazelleGazette
560
487 starfall_stardust
555

	317	dexter_dazzle
555		
	91	VogueVoyagerHQ
555		
	113	DIYDazzlerStudio
548.571429		
	236	noah_narrative
540		
	291	jacob_journey
540		
	6	emily_white
540		
	344	jack_johnson
540		

	USER_ID	USERNAME	AVERAGE_DURATION
--	---------	----------	------------------

	152	FearlessDreamer
540		
	286	emma_enigma
540		
	234	liam_lyric
540		
	439	nancy_drew
540		
	315	jason_jovial
540		
	257	audrey_azure
540		
	246	jackson_journey
540		
	38	hannah_johnson
540		
	45	adam_martinez
540		
	217	RubyRhythmicRambles
540		
	17	brian_morris
540		

	USER_ID	USERNAME	AVERAGE_DURATION
--	---------	----------	------------------

```

-----
-----
-----
276 lily_lagoon
540
264 evan_echelon
540
251 chloe_canvas
540
24 grace_kelly
540
31 peter_robinson
540
340 frank_sinatra
540
498 amren_abyss
525
422 jason_juniper
517.5
335 alice_wonderland
510
306 leo_lagoon
510
218 LeoLearningLegacy
510

```

```

      USER_ID USERNAME
AVERAGE_DURATION
-----
-----
-----
2 jane_smith
510
138 SkinnyDipping
510
86 BookishBardChannel
495
33 eric_wright
480
271 logan_lustrous
480
293 nolan_nimbus
480
25 brandon_fisher
480

```



```
318 olivia_obsidian
480
248 oliver_ode
480
262 isaac_illusion
480
282 mia_mirage
480
```

```
USER_ID USERNAME
AVERAGE_DURATION
```

```
-----
```

```
-----
```

```
278 grace_galaxy
480
299 nat_chamb
480
326 emily_esteemed
480
412 zachary_zephyr
480
15 jason_king
480
9 chris_evans
480
36 lily_rivera
480
288 isla_illuminate
480
132 AlmostLoveAffair
480
242 lucas_luminous
480
331 felicity_fantastic
480
```

```
USER_ID USERNAME
AVERAGE_DURATION
```

```
-----
```

```
-----
```

```
64 TravelTreasureHunter
480
```

212	AlexAdventureAwaits
480	
333	uma_unique
480	
445	tony_tesseract
465	
225	RubyRhythmicRambles
450	
342	harry_potter
450	
337	charlie_chaplin
450	
238	logan_lyric
450	
309	marley_mirage
450	
204	IanInsightfulInnings
450	
148	RunAndHideSeek
440	

USER_ID	USERNAME	AVERAGE_DURATION
---------	----------	------------------

40	ella_jones
420	
255	lily_luminary
420	
244	aiden_artisan
420	
269	caleb_crescent
420	
266	mark_2
420	
13	ryan_clark
420	
351	quincy_jones
420	
221	StellaSerenityStudio
420	
27	daniel_nelson
420	

```

        20 ava_stewart
420
        240 ethan_epic
420

USER_ID USERNAME
AVERAGE_DURATION
-----
-----
-----
274 ava_avalanche
420
284 ruby_rainforest
420
18 sophia_martinez
360
29 logan_wright
360
43 owen_williams
360
280 zoey_zephyr
360
491 feyre_archerons
360
7 david_jones
360
22 victoria_wood
300
11 michael_adams
300

241 rows selected.

```

Question 9 is "What are the most popular genres among users based on their watch history?" can be highly beneficial for content creators in several ways, the popularity of genres among users offers valuable insights that content creators can leverage to optimize their content strategy, enhance audience engagement, and make informed decisions about collaborations and monetization. This gives another option of how to find popular genres that give these creators more options.

```

DECLARE
    CURSOR genre_cursor IS
        SELECT DISTINCT genre FROM videos;

    genre_name videos.genre%TYPE;
    user_count INT;

```

```

BEGIN
    -- Open the cursor
    OPEN genre_cursor;

    -- Loop through genres
    LOOP
        -- Fetch the next genre
        FETCH genre_cursor INTO genre_name;

        -- Exit the loop if no more genres
        EXIT WHEN genre_cursor%NOTFOUND;

        -- Count the number of unique users for the genre
        SELECT COUNT(DISTINCT wh.user_id)
        INTO user_count
        FROM videos v
        JOIN watch_history wh ON v.video_id = wh.video_id
        WHERE v.genre = genre_name;

        -- Display the result
        DBMS_OUTPUT.PUT_LINE('Genre: ' || genre_name || ', Unique Users: ' ||
user_count);
    END LOOP;

    -- Close the cursor
    CLOSE genre_cursor;
END;
/

```

```

Genre: Gaming, Unique Users: 24
Genre: Adventure, Unique Users: 1
Genre: Philosophy, Unique Users: 1
Genre: Wellness, Unique Users: 7
Genre: Sports, Unique Users: 2
Genre: DIY, Unique Users: 35
Genre: Education, Unique Users: 2
Genre: Theatre, Unique Users: 2
Genre: Travel, Unique Users: 18
Genre: History, Unique Users: 4
Genre: Film, Unique Users: 4
Genre: Fitness, Unique Users: 23
Genre: Photography, Unique Users: 4
Genre: Comedy, Unique Users: 23
Genre: Cooking, Unique Users: 27
Genre: Art, Unique Users: 23
Genre: Literature, Unique Users: 27
Genre: Drama, Unique Users: 1
Genre: Science, Unique Users: 11
Genre: Music, Unique Users: 28
Genre: Technology, Unique Users: 19
Genre: Science Fiction, Unique Users: 1
Genre: Fashion, Unique Users: 1
Genre: Nature, Unique Users: 5

```

Next is question 10, and we asked the question “Is the content the user consuming diverse, or are they all from primarily a specific genre?” This provides insights into the preferences and behavior of their audience. Understanding the diversity of content consumption can impact content creation strategies and decision-making in several ways. Understanding the diversity of content consumption is crucial for content creators to make informed decisions about content creation, audience engagement, and overall strategy. By leveraging this information, creators can build a more robust and sustainable relationship with their audience.

```
SELECT
    w.user_id,
    v.genre,
    COUNT(*) AS watch_count
FROM
    watch_history w
JOIN
    videos v ON w.video_id = v.video_id
GROUP BY
    w.user_id, v.genre;
```

	USER_ID	GENRE	WATCH_COUNT
	3	Technology	1
	2	Technology	1
	16	DIY	1
	18	Comedy	1
	20	Gaming	1
	23	Science	1
	32	Science	2
	44	Cooking	1
	204	DIY	1
	227	Comedy	1
	229	Cooking	1

2	240 Literature
1	201 Gaming
1	224 Science
1	232 Comedy
1	254 Travel
1	260 Art
1	263 Gaming
1	267 Comedy
1	272 Literature
1	275 Wellness
1	297 DIY

USER_ID	GENRE	WATCH_COUNT
---------	-------	-------------

1	301 Technology
1	304 DIY
1	307 Comedy
1	316 Nature
1	340 DIY
1	343 Comedy
1	305 Film
1	307 Cooking
1	308 Art
1	313 DIY
1	325 Travel

USER_ID	GENRE	WATCH_COUNT
---------	-------	-------------

1	340	Technology
1	344	Music
1	346	Literature
1	349	Music
1	1	Comedy
1	2	Gaming
1	6	Fitness
1	11	Technology
1	37	Cooking
1	38	Art
1	217	Cooking

USER_ID	GENRE	WATCH_COUNT
---------	-------	-------------

1	218	Art
1	226	Literature
1	231	DIY
1	213	Wellness
1	223	Art
1	248	Art
1	256	Fitness
1	270	Music
1	284	Travel
1	285	Comedy
1	287	Cooking

USER_ID	GENRE	WATCH_COUNT
---------	-------	-------------

1	296	Fitness
1	310	Music
1	335	Cooking
1	344	Fitness
1	346	Art
1	303	Music
1	311	Literature
1	324	Music
1	327	Fitness
1	331	Literature
1	334	Music

	USER_ID	GENRE
WATCH_COUNT		

1	347	Gaming
1	223	Adventure
1	412	Music
1	4	Science
1	1	Fitness
1	3	Literature
1	9	Comedy
1	17	Literature
1	21	Cooking
1	28	Travel
2	214	DIY

	USER_ID	GENRE
WATCH_COUNT		

1	216	Gaming
1	223	Science
1	228	Sports
2	237	Art
2	238	Nature
1	216	Cooking
1	218	Education
1	219	DIY
1	221	Gaming
1	225	Fitness
1	227	Music

	USER_ID	GENRE
	WATCH_COUNT	

1	231	Literature
1	244	Travel
1	246	Fitness
1	253	Music
1	258	Literature
1	286	Fitness
1	306	Travel
1	324	Fitness
1	333	Comedy
1	339	Gaming
1	347	Technology

	USER_ID	GENRE
	WATCH_COUNT	

1	348	Literature
1	302	DIY
1	289	Drama
1	1	Travel
2	4	Comedy
1	4	Art
1	10	Travel
1	24	Fitness
1	25	DIY
1	31	Art
1	34	DIY
1		

	USER_ID	GENRE
WATCH_COUNT		

1	45	Photography
1	132	Technology
1	178	Cooking
1	205	Literature
1	220	DIY
1	222	Cooking
2	225	DIY
1	206	Fitness
1	243	Music
1	257	DIY
1	274	Film
1		

	USER_ID	GENRE
WATCH_COUNT		

1	280 Literature
1	282 DIY
1	302 Literature
1	319 Gaming
1	322 Travel
1	301 Gaming
1	310 Technology
1	342 Gaming
1	343 DIY
1	345 Technology
1	350 Travel

USER_ID	GENRE
WATCH_COUNT	

1	126 Science Fiction
1	54 Philosophy
1	187 Music
1	26 Literature
1	35 Nature
1	39 Science
1	42 Literature
1	215 Theatre
1	232 Music
1	207 DIY
1	211 Photography

USER_ID	GENRE
WATCH_COUNT	

1	212	Gaming
1	226	DIY
1	241	Gaming
1	242	DIY
1	247	Cooking
1	249	Technology
1	261	Technology
1	268	Fitness
1	269	DIY
1	273	History
1	277	Art

USER_ID	GENRE
WATCH_COUNT	

1	279	Technology
1	288	Art
1	290	Literature
1	291	Gaming
1	293	Music
1	300	Art
1	308	Fitness
1	313	Wellness
1	314	Cooking
1	317	Technology
1	318	Literature

USER_ID	GENRE
WATCH_COUNT	

1	328 Literature
1	342 Travel
1	345 Cooking
1	304 History
1	312 Gaming
1	314 Music
1	315 Travel
1	317 Fitness
1	321 Literature
1	322 Gaming
1	326 Comedy

USER_ID	GENRE
WATCH_COUNT	

1	328 Cooking
1	336 Comedy
1	8 Literature
1	12 Cooking
1	14 Science
1	213 Fitness
2	233 History
2	235 Fitness
1	209 Comedy
1	210 Cooking
1	230 DIY

USER_ID	GENRE
WATCH_COUNT	

1	250 Literature
1	252 DIY
1	259 Gaming
1	265 Music
1	271 Science
1	283 Music
1	294 Travel
1	299 Gaming
1	315 Art
1	326 Art
1	330 DIY

USER_ID	GENRE
WATCH_COUNT	

1	331 Music
1	334 Fitness
1	337 Technology
1	341 Music
1	320 Technology
1	329 Art
1	332 Gaming
1	341 Literature
1	348 DIY
1	351 Comedy
1	2 Music

USER_ID	GENRE
WATCH_COUNT	

1	3	Art
1	3	Cooking
1	19	Travel
1	22	Art
1	27	Comedy
1	29	Gaming
1	43	Comedy
1	201	Cooking
1	212	Science
1	221	Music
1	224	Wellness
1		

	USER_ID	GENRE
WATCH_COUNT		

1	230	Photography
1	234	Film
2	202	DIY
1	203	Music
1	204	Travel
1	205	Comedy
1	208	Literature
1	215	Music
1	251	Gaming
1	255	Comedy
1	264	DIY
1		

	USER_ID	GENRE
WATCH_COUNT		

1	278	Nature
1	298	Literature
1	305	Music
1	312	Film
1	320	DIY
1	327	Technology
1	329	Gaming
1	336	Art
1	318	Cooking
1	319	Art
1	339	Art

USER_ID	GENRE
WATCH_COUNT	

1	439	Music
1	4	Cooking
1	2	DIY
1	5	Science
1	7	DIY
1	13	Art
1	15	Fitness
1	30	Cooking
1	33	Fitness
1	36	Gaming
1	40	Fitness

USER_ID	GENRE
WATCH_COUNT	

1	41	DIY
1	200	Music
1	202	Science
1	203	Wellness
1	206	Comedy
1	207	Sports
1	208	Cooking
1	209	Photography
1	210	DIY
1	211	Music
1	219	Education

USER_ID	GENRE
WATCH_COUNT	

2	236	Cooking
2	239	Technology
1	137	Wellness
1	217	Art
1	220	Theatre
1	229	Fitness
1	245	Comedy
1	262	Literature
1	276	Cooking
1	289	Technology
1	292	DIY

USER_ID	GENRE
WATCH_COUNT	

```

1      295 Comedy
1      303 Gaming
1      309 DIY
1      311 History
1      321 Music
1      325 Cooking
1      332 Travel
1      338 Literature
1      306 Wellness
1      309 Nature
1      316 Comedy
1
USER_ID  GENRE
WATCH_COUNT
-----
-----
1      330 Technology
1      333 DIY
1      335 Travel
1      337 Fitness
1      338 Cooking
1      376 Travel
1      113 Fashion
1
293 rows selected.

```

The team has 3 members. Sarah Abrams is a Junior in CIT and worked on generating the dummy data and populating the Watch History table, she created the questions and queries for 5 and 7, wrote the background and database description, and contributed to updating the questions in the report. Victoria Nnaji is a junior in CIT and worked on

generating dummy data and populating the Video table, she created the questions and queries to answer 6, 8, 9 and 10, she contributed to adding the questions to the report and updating the repository. Daniella is a senior in CIT and worked on generating the dummy data, populating tables Search history and users with a script, and created the queries to answer questions 1, 2, 3, and 4, also working on adding the questions to the report and the readme file.