Methodology:

Checking duplicates for user_profile table:

```sql
--Checking duplicates from user profiles table
SELECT userid, COUNT(*) AS record_count
FROM user_profiles
GROUP BY userid
HAVING COUNT(*) > 1;
```

Results | Chart

USERID	RECORD_COUNT

Query produced no results

User_profile table has no duplicates.

Checking duplicates for viewership table:

```sql
--Checking duplicates from viewership table
SELECT userid, channel2, recorddate2, COUNT(*) AS record_count
FROM viewership
GROUP BY userid, channel2, recorddate2
HAVING COUNT(*) > 1;
```

Results | Chart

	USERID	CHANNEL2	RECORDDATE2	RECORD_COUNT
1	809015	Channel O	2016/02/19 04:30	2
2	809330	Africa Magic	2016/01/11 17:56	2
3	809422	Africa Magic	2016/02/21 16:46	2
4	2204356	DStv Events 1	2016/02/03 11:22	2
5	931362	ICC Cricket World Cup 2011	2016/03/30 13:49	2
6	786910	Channel O	2016/01/08 20:47	2
7	810044	Supersport Live Events	2016/02/18 07:50	2

Checking and fixing duplicates in your dataset is essential to guaranteeing data quality, accuracy, and reliable analysis as they can provide skewed results, misleading insights and wasted resources.

Removing duplicates from viewership table and creating cleaned temp table:

```sql
--Removing viewership duplicates (keeping 1 record for eah duplicate)
CREATE OR REPLACE TEMP TABLE viewership_deduped AS
SELECT *
FROM (
  SELECT *,
         ROW_NUMBER() OVER (PARTITION BY userid, channel2, recorddate2 ORDER BY duration_2 DESC) AS rn
  FROM viewership
) AS sub
WHERE rn = 1;
```

Results | Chart

	status
1	Table VIEWERSHIP_DEDUPED successfully created.

```sql
DELETE FROM viewership
WHERE (userid, channel2, recorddate2, duration_2) NOT IN (
  SELECT userid, channel2, recorddate2, duration_2
  FROM (
    SELECT *,
           ROW_NUMBER() OVER (PARTITION BY userid, channel2, recorddate2 ORDER BY duration_2 DESC) AS rn
    FROM viewership
  WHERE rn = 1
);
SELECT*
```

Results | Chart

number of rows deleted
2

Cleaning data and joining the user profile and viewership table:

```
--CLEANING DATA and joining the tables
create or replace Temporary table viewership_temp_tbl AS (
SELECT
A.Userid,
A.Name,
A.Surname,
A.Email,
CASE
WHEN A.Gender IS NULL OR A.Gender = 'None'THEN 'other'
ELSE A.Gender
END AS gender,
CASE
WHEN A.Race IS NULL OR A.Race = 'None' THEN 'other'
ELSE A.Race
END AS Race,
A.Age,
CASE
WHEN A.Province IS NULL OR A.Province = 'None' THEN 'other'
ELSE A.Province
END AS Province,
B.Channel2 AS Channel,
B.Recorddate2 AS Date,
B.Duration_2 AS Duration
FROM
user_profiles AS A
INNER JOIN VIEWERSHIP AS B
ON A.userid=B.userid
)
--
```

Results | Chart

	status
1	Table VIEWERSHIP_TEMP_TBL successfully created.

Converting UTC to SA time:

```
-- Converting UTC to SA time
UPDATE VIEWERSHIP_TEMP_TBL
SET date= CONVERT_TIMEZONE('UTC', 'Africa/Johannesburg', TO_TIMESTAMP(date, 'YYYY/MM/DD HH24:MI'));
```

Results | Chart

number of rows updated	number of multi-joined rows updated
10000	0

Adding time column and timestamp:

```
--
UPDATE VIEWERSHIP_TEMP_TBL
SET date = CONVERT_TIMEZONE('UTC', 'Africa/Johannesburg', TO_TIMESTAMP(date, 'YYYY-MM-DD HH24:MI:SS.FF3'));

--Adding timestamp to time
UPDATE VIEWERSHIP_TEMP_TBL
SET Time= TO_CHAR(TO_TIMESTAMP(date, 'YYYY-MM-DD HH24:MI:SS.FF3'), 'HH24:MI');
```

Results | Chart

	number of rows updated	number of multi-joined rows updated
1	10000	0

Adding date column and timestamp:

```
69  --Adding date column
70  ALTER TABLE VIEWERSHIP_TEMP_TBL
71  ADD COLUMN Date2 DATE;
72  --adding timestamp to date
73  UPDATE VIEWERSHIP_TEMP_TBL
74  SET Date2= TO_CHAR(TO_TIMESTAMP(date, 'YYYY-MM-DD HH24:MI:SS.FF3'), 'YYYY-MM-DD');
```

Results | Chart

number of rows updated	number of multi-joined rows updated
10000	0

QUESTION 1:

Age buckets

```
85  --QUESTION 1
86   --create age buckets
87   SELECT
88   CASE
89     WHEN age = 0 THEN 'Not specified'
90     WHEN age BETWEEN 1 AND 12 THEN 'Kids'
91     WHEN age BETWEEN 13 AND 19 THEN 'Teenagers'
92     WHEN age BETWEEN 20 AND 24 THEN 'Young Adults'
93     WHEN age BETWEEN 25 AND 34 THEN 'Early Adulthood'
94     WHEN age BETWEEN 35 AND 44 THEN 'Mid Adulthood'
95     WHEN age BETWEEN 45 AND 54 THEN 'Mature Adults'
96     ELSE 'Seniors'
97   END AS age_group,
98  COUNT (*) AS viewer_count
99  FROM VIEWERSHIP_TEMP_TBL
100 GROUP BY age_group
101 ORDER BY viewer_count DESC;
102
```

Results | Chart

	AGE_GROUP	VIEWER_COUNT
1	Early Adulthood	4148
2	Mid Adulthood	2635
3	Young Adults	1228
4	Mature Adults	927

Time buckets

```
102 --Create time of day buckets
103 SELECT
104   CASE
105     WHEN TO_TIME(Time) BETWEEN '00:00:00' AND '05:59:59' THEN 'Late Night'
106     WHEN TO_TIME(Time) BETWEEN '06:00:00' AND '11:59:59' THEN 'Morning'
107     WHEN TO_TIME(Time) BETWEEN '12:00:00' AND '17:59:59' THEN 'Afternoon'
108     WHEN TO_TIME(Time) BETWEEN '18:00:00' AND '23:59:59' THEN 'Evening'
109     ELSE 'Unknown'
110   END AS time_bucket,
111   COUNT(*) AS total_views,
112 FROM VIEWERSHIP_TEMP_TBL
113 GROUP BY time_bucket
114 ORDER BY total_views DESC;
```

Results | Chart

	TIME_BUCKET	TOTAL_VIEWS
1	Evening	3807
2	Afternoon	3085
3	Late Night	2352
4	Morning	756

Group by Province:

```sql
--Group by province
SELECT
  province,
  COUNT(*) AS total_views,
FROM VIEWERSHIP_TEMP_TBL
GROUP BY province
ORDER BY total_views DESC;
```

Results | Chart

	PROVINCE	TOTAL_VIEWS
1	Gauteng	3654
2	Western Cape	1845
3	Kwazulu Natal	1001
4	Mpumalanga	918

Most viewed channels

```sql
--Most viewed channels
SELECT
  channel,
  COUNT(*) AS total_views,
FROM VIEWERSHIP_TEMP_TBL
GROUP BY channel
ORDER BY total_views DESC;
```

Results | Chart

	CHANNEL	TOTAL_VIEWS
1	Supersport Live Events	1638
2	ICC Cricket World Cup 2011	1465
3	Channel O	1050
4	Trace TV	952

Views per day:

```sql
--views per day
SELECT
  date2,
  COUNT(*) AS views,
FROM VIEWERSHIP_TEMP_TBL
GROUP BY date2
ORDER BY date2;
```

Results | Chart

	DATE2	VIEWS
1	2016-01-01	51
2	2016-01-02	79
3	2016-01-03	63
4	2016-01-04	70

Grouping by gender and age

```sql
--Grouping gender and age
SELECT
  gender,
  CASE
    WHEN age = 0 THEN 'Not specified'
    WHEN age BETWEEN 1 AND 12 THEN 'Kids'
    WHEN age BETWEEN 13 AND 19 THEN 'Teenagers'
    WHEN age BETWEEN 20 AND 24 THEN 'Young Adults'
    WHEN age BETWEEN 25 AND 34 THEN 'Early Adulthood'
    WHEN age BETWEEN 35 AND 44 THEN 'Mid Adulthood'
    WHEN age BETWEEN 45 AND 54 THEN 'Mature Adults'
    ELSE 'Seniors'
  END AS age_group,
  COUNT(*) AS viewer_count,
FROM VIEWERSHIP_TEMP_TBL
GROUP BY gender, age_group
ORDER BY age_group, gender;
```

Results | Chart

GENDER	AGE_GROUP	VIEWER_COUNT
female	Early Adulthood	417
male	Early Adulthood	3731
female	Kids	26
male	Kids	73

Top viewers by watch time:

```sql
--top users by watch time
SELECT
 userid,
 SUM(DATE_PART('minute', Duration) * 60 + DATE_PART('second', Duration)) / 60 AS total_minutes_watched
FROM VIEWERSHIP_TEMP_TBL
GROUP BY userid
ORDER BY total_minutes_watched DESC
LIMIT 10;
```

Results | Chart

USERID	TOTAL_MINUTES_WATCHED
2415121	337.066667
2000272	226.033333
810145	197.250000
789220	196.700000