```
CREATE TABLE AppleStore_description as

SELECT * from appleStore_description1
UNION ALL
SELECT * from appleStore_description2
UNION ALL
SELECT * from appleStore_description3
UNION ALL
SELECT * from appleStore_description4;
```

****EXPLORATORY DATA ANALYSIS****

```
-- check the number of unique apps in both tables

SELECT
	count(DISTINCT c1) as UniqueAppleIDs
FROM AppleStore;

SELECT
	count(DISTINCT c1) as UniqueAppleIDs
FROM AppleStore_Description;

SELECT
	count(*) as Missing_Values
FROM AppleStore
WHERE
	track_name is null
	or
	user_rating is null
	or
	prime_genre is NULL;
```

```
SELECT
        count(*) as Missing_Values
FROM AppleStore_description
WHERE app_desc is null

-- Find the number of apps per genre

SELECT
        prime_genre,
        count(*) as NumApps
FROM AppleStore
GROUP BY prime_genre
ORDER BY NumApps DESC

-- Get an overview of apps' ratings

SELECT
        min(user_rating) as MinRating,
        max(user_rating) as MaxRating,
        avg(user_rating) as AvgRating
FROM AppleStore
```

****DATA ANALYSIS****

```
-- Determine whether paid apps have higher average than free apps

SELECT
        CASE
                WHEN price>0 then "Paid"
                ELSE "Free"
          END as App_Type,
          avg(user_rating) as AvgRating
FROM AppleStore
GROUP BY App_Type;
```

```
-- Check if apps with more supported languages have higher ratings

SELECT
   CASE
      WHEN lang_num < 10 THEN "<10 Languages"
      WHEN lang_num BETWEEN 10 AND 30 THEN "10-30 Languages"
      ELSE ">30 Languages"
   END AS language_bucket,
   AVG(user_rating) AS Avg_Rating
FROM AppleStore
GROUP BY language_bucket
ORDER BY Avg_Rating DESC;


-- Check genres with low rating

SELECT prime_genre,
            avg(user_rating) as Avg_Rating
FROM AppleStore
GROUP BY prime_genre
ORDER BY Avg_Rating ASC
LIMIT 1;

-- Check if there is a correlation between the length of the app
description and the user rating.

SELECT
      CASE
            WHEN length(b.app_desc)<500 THEN 'short'
            WHEN length(b.app_desc) BETWEEN 500 and 1000
            THEN'medium'
            ELSE 'long'
      END AS description_length_bucket,
      avg(user_rating) as Avg_Rating
FROM
```

```
        AppleStore as a
JOIN
  AppleStore_Description as b
ON
        a.id=b.id
GROUP BY description_length_bucket
ORDER BY Avg_Rating DESC

--Check the top-rated apps for each genreAppleStore

SELECT
        prime_genre,
        track_name,
        user_rating
        FROM(
                SELECT
                        prime_genre,
                        track_name,
                        user_rating,
                RANK() OVER(PARTITION by prime_genre order by
user_rating desc, rating_count_tot DESC) as rank
       FROM
       AppleStore
     ) as a
WHERE
a.rank=1;
```

****RECOMMENDATIONS****

- Paid Apps generally get slightly higher ratings than free counterparts; it could be because users that pay perceive the app to be of higher value leading to better ratings. If client believes the product is valuable, they should consider charging for it.

- Found that Apps supporting between 10 and 30 languages had the highest average rating; it is not about the quantity of languages the app supports but about focusing on the right languages.

- Finance and Book Apps have low user ratings; it could be because user needs are not being met. It could present a market opportunity if client can meet these needs.

- Length of app descriptions has a positive correlation with user ratings; Users appreciate having detailed explainations of apps they to download.

- A new App should aim for an average rating above 3.5.

- Games and Entertainment genres are saturated; entering this space could be challenging but there is high user demand in these sectors.