|  |
| --- |
| 1. Why should your database be built? 2. My database is going to essentially contain data on the ever-evolving world of streaming platforms over the internet such as twitch, YouTube, Facebook and much more. Don’t get this confused with streaming services like Netflix, Amazon Prime, Hulu and so on. When I talk about streaming for this database I have in mind, I’m talking about individuals that stream various things online to their audiences of thousands of people. For example, popular streamers mainly stream themselves playing video games and it can be more than just that, it can be art, music making, coding, just chatting and more. **Why should this database be built?** Streaming has been growing in popularity over the years. It has started many careers, helped people make a ton of money in a new way and as the years of the digital era go by streaming just becomes more and more popular. Twitch, for example, is a streaming platform and in 2020 alone they mad 2.3 billion dollars in revenue and had about 18.6 billion hours of content consumed. This is all thanks to the individual streamers on their platform that continue to make content that advertisers can make use of as well. This will be one part of the digital future and I feel like it can be important showcasing just how much it has grown. 3. What tasks, operations or processes the system supports, and what user group or groups are served? 4. The database will support the entry, deletion, and changes of any streamer contract updates, revenue changes, analytic updates and so on. Relationships will be made between tables within the database to connect streamers to their platforms. 5. The group that this will serve to will be myself and since this type of database is going to be made with the use of public analytics it can serve the general public 6. Limitations of the system? 7. The main limitation of this system will be the fact that unless there is a way to automatically update the numbers (I’m sure there is but it would probably be too complicated at this level of programming) it will be almost impossible to accurately stay up to date with specific streamers averages on viewer numbers and subscription counts since this number can change daily, weekly, monthly, and yearly. Streaming statistics is always evolving. 8. List of the DML, DDL and TCL statements used? 9. DML – SELECT, INSERT, UPDATE, DELETE 10. DDL – CREATE, ALTER, DROP, RENAME 11. TCL – COMMIT, ROLLBACK, SAVEPOINT\*   2. Schema and ERD |
| 3. Create 5 Tables |
| 4. Add at least 2 different defaults |
| 5. Add a NOT NULL constraint    6. Add at least 1 check constraint using ENUM or check constraint |
| 7. Each table should have a primary key and 4 of the tables should have foreign keys |
| 8.  a. ) Create a table from another table – structure and data      b. ) Drop the created table |
| 9. Insert atleast 10 records into 3 of the tables  Use the following at least ONCE  VARCHAR, INT, DATE, FLOAT/DECIMAL/DOUBLE |
| 10. Update at least 3 records  a.)        b.) |
| c.) |
| 11.) Group by functions,  Query with: Sum/Average, Count, Min/Max  a.)      b.)    c.) |
| 12.) Queries  a.) Range criteria    b.) Using IN operator      c.) Using LIKE operator with wildcards |
| d.) Another Range Criteria    e.) Use column alias    f.) Perform basic arithmetic operations    g.) Concatenate when reporting data        h.) Use Distinct keyword    i.) Order the output      j.) No Null values in database    k.) Extract the month from the date        l.) Use SUBSTR to pull out something    m.) Create a column that provides the length of one of your columns      n.) Round output from one of your columns to 1 digit      13.) Use an if/case statement |
| 14.) Joins  a.) Query that joins 3 tables and limits output      b.) Query with an outer join      d.) |