|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Features  Rank 1-5 | MSSQL  Rate 0-5 | Oracle  Rate 0-5 | SQLite  Rate 0-5 | MySQL(or MariaDB)  Rate 0-5 | PostgreSQL  Rate 0-5 | Microsoft Access  Rate 0-5 | LibreOffice Base  Rate 0-5 |
| Simple and fast (5) | 3 (15) | 1 (5) | 5 (25) | 5 (25) | 4 (20) | 3 (15) | 3 (15) |
| Minimal setup (no special hardware/software requirements) (5) | 1 (5) | 1 (5) | 5 (25) | 3 (15) | 3 (15) | 3 (15) | 3 (15) |
| Cross-platform compatibility (5) | 1 (5) | 1 (5) | 5 (25) | 5 (25) | 5 (25) | 2 (10) | 5 (25) |
| Concurrency (1) | 4 (4) | 4 (4) | 1 (1) | 4 (4) | 5 (5) | 3 (3) | 3 (3) |
| Networkable (1) | 4 (4) | 4 (4) | 1 (1) | 5 (5) | 5 (5) | 4 (4) | 4 (4) |
| Replication (3) | 3 (9) | 3 (9) | 5 (15) | 2 (6) | 4 (12) | 3 (9) | 3 (9) |
| Low Cost (5) | 1 (5) | 1 (5) | 5 (25) | 5 (25) | 5 (25) | 4 (20) | 5 (25) |
| Compatible with Python (4) | 3 (12) | 3 (12) | 5 (20) | 5 (20) | 5 (20) | 4 (16) | 3 (12) |
| Total | 59 | 49 | 137 | 125 | 127 | 92 | 108 |

The table above shows the ranking and rating of the listed database options according to the criteria provided by my boss based on the situation provided for Topic Challenge – Module 6D.

By multiplying my ratings to the ranks of the features listed, the database option with the highest score is ***SQLite***.

SQLite works out-of-the-box with little to no setup required, is low-cost, and compatible with Python. This database is perfect for the ‘feasibility test’ scenario given.