Report on Data Build Task

Mahir Hashimov
12695120
Unit 7
Deciphering Big Data
University of Essex Online
21 June 2024

Contents

Report on Data Build Task 3
Introduction 3
Steps Taken 3
Challenges Faced 4
Critical Analysis 4
Conclusion 5
Output 6
References 7

Report on Data Build Task

Introduction

The purpose of this report is to document the process of creating and testing a relational database system based on the provided student data table. The tasks involved include designing the database schema, implementing it using SQL, populating it with data, and ensuring referential integrity. This report also discusses the challenges encountered, the steps taken, and a critical analysis of the process and results.

Steps Taken

1. Designing the Database Schema

Tables and Relationships: The provided data was analyzed to identify the necessary tables and their relationships. The primary tables identified were Students, Courses, ExamBoards, Teachers, and three junction tables: StudentCourses, StudentExamBoards, and StudentTeachers to manage many-to-many relationships.

2. Creating the Database
    - SQL Script for Schema: A comprehensive SQL script was written to create the tables with the appropriate primary and foreign keys. The script included the creation of junction tables to ensure referential integrity between the main tables.
3. Populating the Database
    - Data Insertion: SQL INSERT statements were used to populate the tables with the provided data. This involved adding records for students, courses, exam boards, and teachers, as well as establishing the relationships between them in the junction tables.
4. Testing the Database
    - Querying Relationships: SQL SELECT queries were used to join tables and verify the relationships between students, their courses, exam boards, and teachers. This step ensured that the data was correctly inserted, and the referential integrity was maintained.

Challenges Faced

1. Schema Design Complexity
   - Designing a normalized database schema that accurately represents the relationships in the provided data was challenging. Ensuring that all many-to-many relationships were appropriately handled through junction tables required careful planning.
2. Handling Referential Integrity
   - Ensuring that all foreign key constraints were correctly implemented and maintained throughout the database operations was critical. Any omission or error in defining these constraints could lead to data integrity issues.
3. SQL Execution Environment
   - Initially, the SQL script was attempted to be run in a Jupyter Notebook environment, which resulted in syntax errors. This highlighted the importance of choosing the correct execution environment for SQL scripts.
4. Data Duplication
   - Careful attention was needed to avoid data duplication, especially when populating junction tables. Each record had to be accurately mapped to ensure correct relationships without redundancy.

Critical Analysis

1. Methodology and Tools
   - The use of SQLite for this task was appropriate due to its simplicity and ease of use for small to medium-sized databases (Gaffney et al., 2022). The methodology of breaking down the problem into schema design, data insertion, and testing proved effective.
2. Normalization and Integrity
   - The database schema was designed to be fully normalized, which helped in avoiding data redundancy and ensuring data integrity. The use of foreign key constraints further reinforced the integrity of the relationships between tables (Amato, xxxx).
3. Efficiency of the Process
   - The process was efficient in terms of design and implementation, but initial attempts at running the SQL script in an inappropriate environment highlighted the importance of selecting the right tools.

The transition to a standalone Python script with SQLite resolved these issues effectively.

4. Error Handling and Debugging
    - Encountering and resolving the OperationalError due to existing tables underscored the importance of error handling and debugging. Adding scripts to drop existing tables before creation ensured a clean slate for the database operations.
    - Scalability and Performance
    - While the current database system is adequate for the provided data, scalability could become an issue with larger datasets. Future considerations could include migrating to more robust database systems like PostgreSQL or MySQL for handling larger volumes of data and more complex queries.

Conclusion

The task of creating and testing a relational database system was successfully accomplished by following a structured approach to database design, implementation, and testing. Despite the challenges faced, particularly in ensuring referential integrity and handling the execution environment, the process proved effective. The resulting database system is normalized, maintains data integrity, and accurately represents the relationships in the provided data.

Future work could involve enhancing the database system to handle larger datasets, optimizing performance, and potentially integrating more advanced features like stored procedures and triggers for improved data management and automation. Overall, the task provided valuable insights into the complexities and critical considerations in database design and implementation.

Output

```
Students and their Courses:

('Bob Baker', 'Maths')
('Bob Baker', 'Physics')
('Sally Davies', 'Maths')
('Sally Davies', 'Biology')
('Sally Davies', 'Music')
('Mark Hanmill', 'Computer Science')
('Mark Hanmill', 'Maths')
('Mark Hanmill', 'Physics')
('Anas Ali', 'Maths')
('Anas Ali', 'Physics')
('Anas Ali', 'Biology')
('Cheuk Yin', 'Computer Science')
('Cheuk Yin', 'Maths')
('Cheuk Yin', 'Music')

Students and their Exam Boards:
('Bob Baker', 'BCS')
('Bob Baker', 'EdExcel')
('Bob Baker', 'OCR')
('Sally Davies', 'AQA')
('Sally Davies', 'WJEC')
('Sally Davies', 'AQA')
('Mark Hanmill', 'BCS')
('Mark Hanmill', 'EdExcel')
('Mark Hanmill', 'OCR')
('Anas Ali', 'AQA')
('Anas Ali', 'OCR')
('Anas Ali', 'WJEC')
('Cheuk Yin', 'BCS')
('Cheuk Yin', 'EdExcel')
('Cheuk Yin', 'AQA')

Students and their Teachers:
('Bob Baker', 'Mr Jones')
('Bob Baker', 'Ms Parker')
('Bob Baker', 'Mr Peters')
('Sally Davies', 'Ms Parker')
('Sally Davies', 'Mrs Patel')
('Sally Davies', 'Ms Daniels')
('Mark Hanmill', 'Mr Jones')
('Mark Hanmill', 'Ms Parker')
('Mark Hanmill', 'Mr Peters')
('Anas Ali', 'Ms Parker')
('Anas Ali', 'Mr Peters')
('Anas Ali', 'Mrs Patel')
('Cheuk Yin', 'Mr Jones')
('Cheuk Yin', 'Ms Parker')
('Cheuk Yin', 'Ms Daniels')
```

References

Gaffney, K.P., Prammer, M., Brasfield, L., Hipp, D.R., Kennedy, D., and Patel, J.M., 2022. SQLite: past, present, and future. Proceedings of the VLDB Endowment, 15(12), pp.3535-3547. Available at: https://doi.org/10.14778/3554821.3554842.

Amato, N., Mastering database normalization: A comprehensive exploration of normal forms. Available at: https://www.researchgate.net/profile/Nicola-Antonio-Roberto-Amato/publication/374509386_Mastering_database_normalization_A_comprehensive_exploration_of_normal_forms/links/652106cffc5c2a0c3bbe361d/Mastering-database-normalization-A-comprehensive-exploration-of-normal-forms.pdf.