Steps:

1. **Setting up the Environment:**
   - Installed the required Python library (`mysql-connector-python`) for connecting to and interacting with a MySQL database.
   - Defined the necessary database connection details (host, username, password, database name).
2. **Establishing Database Connection:**
   - Used the `mysql.connector` library to establish a connection to the MySQL database.
   - Created a cursor object to execute SQL statements.
3. **Executing SQL Statements:**
   - Converted each SQL INSERT statement into a corresponding Python statement using the `cursor.execute()` method.
   - Handled foreign key constraints by inserting data into parent tables first and then using the generated primary key values in child table insertions.
   - Addressed errors related to column names, foreign key constraints, and data integrity violations by modifying the queries or executing additional SQL statements (e.g., `ALTER TABLE` to drop or modify foreign key constraints).
4. **Retrieving and Displaying Data:**
   - Executed SELECT statements to retrieve data from the populated tables.
   - Printed the retrieved data to the console for verification.
5. **Testing Foreign Key Constraints:**
   - Attempted to delete a record from the parent table (`users`) to test the foreign key constraints.
   - Verified that related records in child tables (`polls` , `votes`) were appropriately handled (deleted or updated) based on the defined constraints.
6. **Committing Changes and Closing Connections:**
   - Committed the changes to the database using `db.commit()` .
   - Closed the database connection and cursor objects.
7. **Deleting Data:**
   - Provided an example of how to delete data from the tables, considering the order of deletion due to foreign key constraints (child tables first, then parent tables).
8. **Documentation:**
   - Summarized the conversion process, including the steps taken, libraries used, and any specific considerations or challenges encountered during the conversion.

The resulting Python script can be executed to populate the database with the provided sample data, retrieve and display the data, test foreign key constraints, and optionally delete data from the tables. This conversion process demonstrates the integration of SQL statements into a Python application, enabling programmatic interaction with a MySQL database.