PROG8850 – Assignment 3

1. Student Information

Name: Jay Nileshkumar Patel
Student ID: 9062044
Date: 16th October 2025

2. Overview and Key Features

Flyway

- **Overview:**
  - Flyway is an open-source database migration tool that uses simple SQL scripts for versioned migrations. It is widely adopted for its simplicity and reliability.
- **Key Features:**
  - Supports SQL and Java-based migrations
  - Works with most relational databases
  - Easy CLI and API integration
  - Tracks schema version in a dedicated table
  - Supports undo and repeatable migrations (Teams edition)
  - Integrates with CI/CD tools

Liquibase

- **Overview:**
  - Liquibase is a powerful open-source database schema change management tool. It uses XML, YAML, JSON, or SQL for defining migrations (called changelogs).
- **Key Features:**
  - Supports multiple file formats for migrations
  - Extensive rollback and change tracking
  - Works with most relational databases
  - Rich CLI, API, and UI integrations
  - Advanced diff and audit capabilities
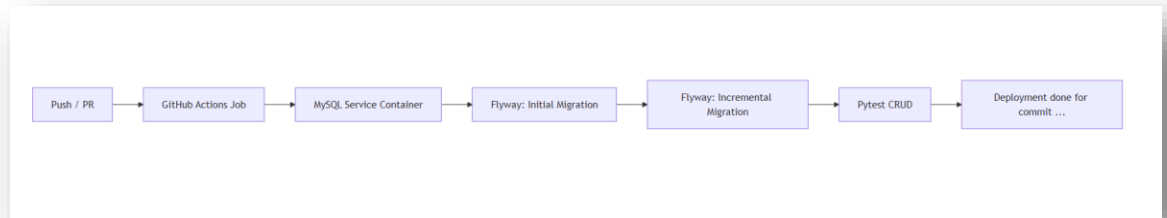 - Integrates with CI/CD tools

3. Comparison Table

Feature	Flyway	Liquibase
Ease of Use	Very simple, SQL-based	Flexible, but more complex
CI/CD Integration	Native support, easy setup	Native support, more options
Supported DBs	Most RDBMS	Most RDBMS + NoSQL (some)
Migration Format	SQL, Java	SQL, XML, YAML, JSON
Rollback Support	Limited (Teams)	Extensive
Change Tracking	Basic (schema version table)	Advanced (audit, diff)

4. Integration Strategy for CI/CD

To integrate Flyway and Liquibase into a CI/CD pipeline:

1. **Repository Structure:**
    –   Organize migration scripts in dedicated folders (e.g., `/flyway_migrations`, `/liquibase_changelogs`).
2. **Pipeline Steps:**
    –   On code push or PR:
        •   Spin up a test database instance
        •   Run Flyway migrations (for simple, versioned changes)
        •   Run Liquibase changelogs (for complex, auditable changes)
        •   Run automated tests
        •   Rollback or clean up as needed
3. **Automation:**
    –   Use GitHub Actions, Jenkins, or similar tools to automate migration and testing steps.
    –   Print deployment status to the console/logs.
4. **Best Practices:**
    –   Ensure each migration is idempotent and reversible
    –   Use environment variables for DB credentials
    –   Store migration history and audit logs

Push / PR → GitHub Actions Job → MySQL Service Container → Flyway: Initial Migration → Flyway: Incremental Migration → Pytest CRUD → Deployment done for commit …

5. References

[Flyway Documentation](#)

[Liquibase Documentation](#)