**Assignment 4**

**Student Name: Puneet Mishra**

**Student ID: 8910233**

**Program Name: Cloud Development and Operations**

**College Name: Conestoga College**

**Course Code: PROG8850 - Winter 2025 - Section 1**

**Course Name: Database Automation**

**Assignment Due Date: 31st March 2025**

**Question 1: Analysis and Integration of Database Automation Tools (8 Points) 1.1) Select any two database automation tools from the following list: Jenkins, GitHub Actions, Azure DevOps, GitLab CI/CD, Liquibase, Flyway, Ansible. For each tool: - Provide a brief overview and key features.**

**Selected Tools: Flyway and Ansible.  
1. Flyway – Overview and Key Features**

**Overview:**

Flyway is a database migration tool that has version-controlled database changes through SQL or Java-based migration scripts. It easily integrates with CI/CD pipelines and is primarily for managing schema change.

**Key Features:**

Version-controlled SQL scripts supported (e.g., V1\_\_init.sql, V2\_\_add\_column.sql)

idempotent migrations supported (one execution of every migration)

Has an easy integration with CI/CD tools like Jenkins, GitHub Actions, etc.

Schema validation and repair commands supported

Works with multiple database systems (MySQL, PostgreSQL, Oracle, SQL Server, etc.)

**2. Ansible – Overview and Key Features**

Ansible is open-source IT automation software for provisioning infrastructure, deploying applications, and configuring them. Ansible was utilized in this exercise to manage the lifecycle of a MySQL Docker container and trigger Flyway migrations.

**Major Features:**

YAML-playbooks for simple readable automation flows

Agentless architecture (client software doesn't need to be installed)

Supports Docker, Kubernetes, Jenkins, and a whole lot more

Best suited for provisioning databases, infrastructures, and application environments

Can manage multi-step deployment sequences (e.g., start container → apply migration)

**Comparison Table: Flyway vs Ansible**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Flyway** | **Ansible** |
| **Ease of Use** | **Easy for DBAs and developers familiar with SQL** | **Requires some learning curve, YAML syntax needed** |
| **Integration with CI/CD** | **Native support for Jenkins, GitHub Actions, etc.** | **Integrates well with CI/CD tools using playbooks** |
| **Supported Databases** | **MySQL, PostgreSQL, Oracle, SQL Server, H2, etc.** | **Works with any database (indirectly) via provisioning and scripting** |
| **Main Focus** | **Database migrations and version control** | **Infrastructure automation (including databases)** |
| **Idempotency** | **Ensures migrations run only once per version** | **Built-in idempotency in playbooks** |
| **Strength** | |  | | --- | |  |  |  | | --- | | **Structured, reliable schema migration tool** | | **Powerful orchestration across services and environments** |

**1.2) Integration Strategy: Propose a strategy to integrate the two selected tools into a CI/CD pipeline for a software project.**To build an automated and streamlined database deployment pipeline, Flyway and Ansible can be integrated into a CI/CD pipeline. The two are complementary: Ansible takes care of infrastructure operations like the startup or shutdown of a MySQL container, and Flyway maintains database schema changes versioned and deployed uniformly across environments.

The integration begins by developers writing and committing their Flyway SQL database migration scripts into a version control system like GitHub. The scripts are the ones that alter the database schema, such as defining tables or inserting columns. After committing, the changes trigger a CI/CD pipeline through mechanisms like GitHub Actions, Jenkins, or Azure DevOps.

Along the pipeline run, Ansible is used first to provision the database environment. It begins a MySQL container running in a Docker container and waits for it to become fully up and running. When the database's running, Flyway is invoked to execute the versioned migration scripts automatically to the database. This does a good job ensuring the database structure is in the latest state required by the application.

After migrations, the database schema can be validated by running a Python script to verify that the database schema is still in place, with key columns like id, email, and subscription\_date existing. This validation provides an additional layer of protection prior to application building and deployment.

Finally, after deployment or testing has been achieved, Ansible is then used once more to elegantly shut down the MySQL container. Developers can also utilize this step if need be to dump the existing data out of the database using mysqldump so that in the future the database can be created the same way again.

Teams also have a reproducible, versioned, and fully automated process of dealing with database changes using Flyway and Ansible through the CI/CD pipeline, reducing inconsistencies and human errors with the deployment environments. **Question 2: Hands-on Exercise Using Ansible (12 Points)  
Installed docker and Git.**  
A screenshot of a computer

AI-generated content may be incorrect.

**Chocolatey (Package manager) is already present.**

A black screen with many small lights

AI-generated content may be incorrect.

**Flyway successfully installed but not Ansible**A screen shot of a computer program

AI-generated content may be incorrect.

**Installed WSL(Ubuntu Linux) , so I could successfully install ansible.**

**A screen shot of a computer

AI-generated content may be incorrect.**

**Ansible and python successfully installed   
A screen shot of a computer

AI-generated content may be incorrect.**

**A screen shot of a computer

AI-generated content may be incorrect.**

**Cloned Repository**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Folder Structure Setup. Created two files up.yaml,down.yaml and a folder flyway and inside it created sql folder.  
A screenshot of a computer

AI-generated content may be incorrect.**

**Initial Flyway MigrationA screenshot of a computer

AI-generated content may be incorrect.**

**Started a MySQL Docker container**

**A screen shot of a computer screen

AI-generated content may be incorrect.**

**Ran ansible-playbook up.yaml**

**A black background with many lines

AI-generated content may be incorrect.**

**Installed pip3**

**A screen shot of a computer

AI-generated content may be incorrect.**

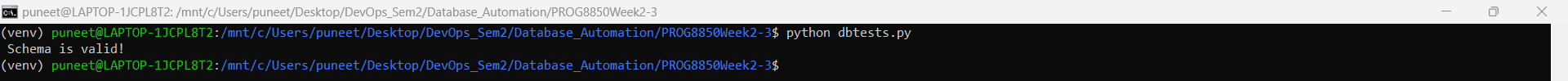
**Installed the missing packageA screen shot of a computer

AI-generated content may be incorrect.**

**Created Python Schema Validation (dbtests.py)  
A screenshot of a computer

AI-generated content may be incorrect.**

**Validated the Initial Schema using python dbtests.py**

****

**Flyway Migration Check**

| **Item** | **Result** |
| --- | --- |
| **Database Created-** | **subscribersdb → Yes** |
| **Migration Table Created-** | **flyway\_schema\_history → Yes** |
| **First Table Migrated-** | **subscriber → Yes** |

**A black screen with white text

AI-generated content may be incorrect.**

**Flyway is connected to my MySQL database running on port 3307.**

**It found that the version 1 migration had already applied.**

**It then executed the new migration: V2\_\_Add\_subscription\_date.sql.**

**The schema is now at version 2, meaning the new subscription\_date column was added!  
A black screen with green and blue lights

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A computer screen with text and numbers

AI-generated content may be incorrect.**

**Seed Data Migration  
This adds test data to the subscriber table.  
A computer screen shot of a black screen

AI-generated content may be incorrect.**

**Export/Seed the data using mysqldump  
A computer screen with text on it

AI-generated content may be incorrect.**

**Ran down.yaml to stop MySQL**

**A screenshot of a computer

AI-generated content may be incorrect.  
Ran ansible-playbook up.yaml again to check Check the Inserted Test Data in MySQL.**

**A screen shot of a computer

AI-generated content may be incorrect.**