DAY 31 — Daily Assessment

Theme: DevOps Foundation + Git + Build Tools (Maven & Gradle)

Time: 120 minutes
Marks: 50(self evaluation)
Assessment Type: User Story Driven + Hands-on Tasks

Build & Version MetroRide Java API

MetroRide, a small transport information service, has a Java-based Spring Boot API that gives real-time metro timings. The engineering team wants to version their code, manage builds with Maven and Gradle, and streamline basic DevOps workflows.

You are assigned as a DevOps Engineer to prepare their repository, version control workflows, and build automation.

User Stories & Tasks

User Story 1 — Git Repository Setup

As a DevOps engineer, I want a version-controlled repository so that developers can collaborate without conflict.

Task 1 (8 Marks)

- Create a Git repository locally
- Add a sample Java Spring Boot application (simple REST controller acceptable)
- Add .gitignore for Java + Maven

- Create meaningful commit history (minimum 3 commits)
- Show branch creation:
    - main
    - feature/build-setup
- Provide screenshots of commit logs and branches

User Story 2 — Maven Build Pipeline

As a build engineer, I want a reliable Maven build so that dependencies and packaging are consistent.

Task 2 (10 Marks)

- Create a Maven pom.xml
- Add minimum dependencies:
    - spring-boot-starter-web
- Run mvn clean package
- Generate JAR file
- Provide screenshots of build success
- Add final JAR in submission ZIP

User Story 3 — Gradle Build Setup

As a DevOps engineer, I want an alternative Gradle build for teams that prefer Gradle.

Task 3 (10 Marks)

- Add build.gradle

- Add same dependencies
- Run gradle build
- Provide screenshots
- Export build folder

User Story 4 — Basic Git Workflow Automation

As a team lead, I want a standardized Git workflow so all developers follow the same process.

Task 4 (7 Marks)

Create a CONTRIBUTING.md explaining:

- Branch naming rules
- Commit message format
- Merge request guidelines

User Story 5 — Versioning Strategy

As a release engineer, I want semantic versioning applied to keep release history clean.

Task 5 (5 Marks)

- Add a VERSION.txt file
- Add version: 1.0.0
- Explain the next version after introducing a new feature

User Story 6 — Documentation

As a new joiner, I want to understand how to build and run the application.

Task 6 (5 Marks)

Create README.md covering:

- How to run using Maven
- How to run using Gradle
- Sample API endpoint

Self Evaluating Rubrics (50 Marks)

Section	Criteria	Marks
Git Setup	Repo + Commit History + Branching	8
Maven Build	pom.xml + Build Success	10
Gradle Build	build.gradle + Build Success	10
Git Workflow Document	CONtributing.md	7
Versioning	VERSION.txt + Explanation	5
README	Clarity & Completeness	5
Structure & Quality	Clean Submission	5
Total		**50**
