Exercise 5.1: CI/CD for Talend ETL Job using GitLab + Jenkins (with Webhook Trigger)

Input File: `customers.csv`

```
id,name,email
1,John Doe,john@example.com
2,Jane Smith,jane@example.com
3,Alex Ray,alex@example.com
```

Tools Required

- **Talend Open Studio** (Free Version)
- **GitLab** (with repository created)
- **Jenkins** (with plugins installed: GitLab, Git)
- **Java JDK** (e.g., 17)

PART 1: Talend Job Design in TOS

◆ Step-by-step Job (TOS):

Components:

1. `tFileInputDelimited` → 2. `tLogRow`

Configuration:

1. **tFileInputDelimited**
    - File name: `customers.csv`
    - Field Separator: `,`
    - Schema:
        - `id` → Integer
        - `name` → String
        - `email` → String
2. **tLogRow**
    - Mode: **Table**

Final Job Name: `CustomerLoggerJob`

Build the Job:

- Right-click on Job → Build Job → Export as ZIP or JAR
- Choose `Standalone Job` option
- Output Directory: Save to local folder like `C:/talend_projects/CustomerLoggerJob`

PART 2: Push Project to GitLab

◆ Steps:

1. On **GitLab**, create a **new repository**: `talend-ci-example`
2. On your system:

```
git init
git remote add origin https://gitlab.com/your-username/talend-ci-example.git
git add .
git commit -m "Initial Talend Job"
git push -u origin master
```

*Make sure the **`CustomerLoggerJob.zip` or full build folder** is committed.

PART 3: Jenkins Configuration

◆ Step-by-step Jenkins Setup:

1. Install Jenkins Plugins:

- GitLab Plugin
- Git Plugin
- Pipeline Plugin

2. Create Jenkins Job:

- Go to Jenkins → New Item → Choose **Pipeline**
- Name: `TalendCIJob`
- Enable **GitLab webhook trigger** under Build Triggers:
    - ✓ "Build when a change is pushed to GitLab"

3. Pipeline Script (Jenkinsfile):

You can store this in your GitLab repo as `Jenkinsfile`, or inline in Jenkins.

Groovy Code:

```
pipeline {
    agent any

    environment {
        JAVA_HOME = "C:\\Program Files\\Java\\jdk-21"
        TALEND_CLI = "C:\\Talend\\Talend-Studio-win-x86_64"
        JOB_NAME = "CI_CD_DEMO_PROJECT"
    }

    parameters {
        string(name: 'CSV_FILE_PATH', defaultValue: 'C:/TalendStudio/customers.csv')
    }
    stages {
        stage('Checkout') {
            steps {
                git branch: 'main',
                    url: 'https://github.com/nagireddykmit/talendtest.git'
            }
        }
        stage('Unzip Job') {
            steps {
                bat 'powershell -Command "Expand-Archive -Force 
CI_CD_DEMO_PROJECT_0.1.zip .\\job2"'
            }
        }
        stage('Run Talend Job') {
            steps {
                bat '''
                cd job2\\CI_CD_DEMO_PROJECT
                call CI_CD_DEMO_PROJECT_run.bat --context=Default ^
                --context_param CSV_FILE_PATH=%CSV_FILE_PATH% ^
                '''
            }
        }
    }
}
```

PART 4: GitLab Webhook to Jenkins

Steps:

1. Go to GitLab → Repository → Settings → Webhooks
2. URL: `http://<your_jenkins_ip>:8080/project/TalendCIJob`
3. Secret token: optional
4. ✅ Push events
5. Click “Test webhook”

✅ Expected Output (on Jenkins Console)

```
[Pipeline] echo
Unzipping and simulating Talend execution...
[Pipeline] sh
Unzipping...
Running CustomerLoggerJob_run.sh...

[statistics] connecting to socket on port...
+ id=1 name=John Doe email=john@example.com
+ id=2 name=Jane Smith email=jane@example.com
+ id=3 name=Alex Ray email=alex@example.com
...
```

Summary:

Task	Tool
ETL Job with CSV	Talend Open Studio
Version Control	Git + GitLab
CI/CD Execution	Jenkins
Build Trigger	GitLab Webhook
Logging Output	Jenkins Console via `tLogRow`