# Jenkinsfile

Video 4.2

# What you will learn in this video

**Theory**:

- What is a Jenkinsfile
- Declarative and scripted pipelines

**Practice**:

- Create and run simple pipeline jobs

# Jenkinsfile

- A **script containing the definition and the steps to execute** in a Jenkins pipeline

# Jenkinsfile

- A **script containing the definition and the steps to execute** in a Jenkins pipeline
- Typically **stored under version control at the root level** of the repository

# Jenkinsfile

- A **script containing the definition and the steps to execute** in a Jenkins pipeline
- Typically **stored under version control at the root level** of the repository
- Written using a **DSL based on Groovy**

# Jenkinsfile

- A **script containing the definition and the steps to execute** in a Jenkins pipeline
- Typically **stored under version control at the root level** of the repository
- Written using a **DSL based on Groovy**
- Can use **declarative or scripted syntax**

# Scripted vs Declarative Pipelines

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script
- Could become messy if unfamiliar with Groovy

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script
- Could become messy if unfamiliar with Groovy

## Declarative Pipeline

- Declarative programming model

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script
- Could become messy if unfamiliar with Groovy

## Declarative Pipeline

- Declarative programming model
- Suited for regular requirements

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script
- Could become messy if unfamiliar with Groovy

## Declarative Pipeline

- Declarative programming model
- Suited for regular requirements
- Very readable, opinionated syntax, stricter and pre-defined structure

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script
- Could become messy if unfamiliar with Groovy

## Declarative Pipeline

- Declarative programming model
- Suited for regular requirements
- Very readable, opinionated syntax, stricter and pre-defined structure
- Can integrate scripted syntax

# Scripted vs Declarative Pipelines

## Scripted Pipeline

- Imperative programming model
- Suited for complex requirements
- Flexible and extensible
- Can do anything that Groovy can script
- Could become messy if unfamiliar with Groovy

## Declarative Pipeline

- Declarative programming model
- Suited for regular requirements
- Very readable, opinionated syntax, stricter and pre-defined structure
- Can integrate scripted syntax
- Learning curve if unfamiliar with declarative programming

# Next Video

Video 4.3 CD Pipeline Overview and Intro to Docker