Red Hat
Training and
Certification

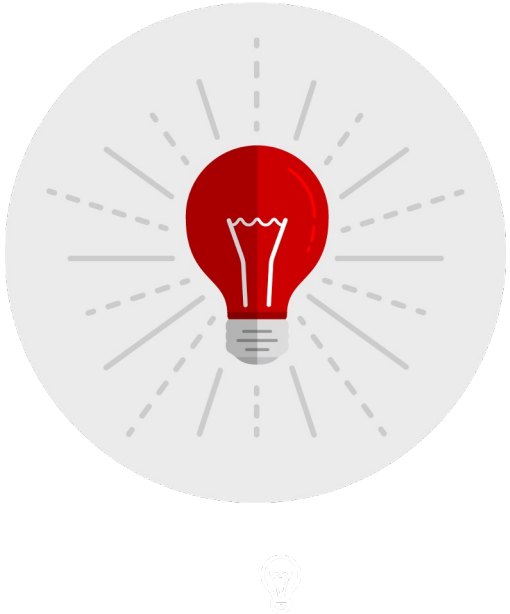
RED HAT
OPEN INNOVATION LABS

DevOps Culture and Practice Enablement

Exercise 1 - Introduction

Red Hat

Topics



Lab Setup and Access

How to setup your account and access CodeReady Workspaces and the OpenShift Environment

Codeready Workspaces

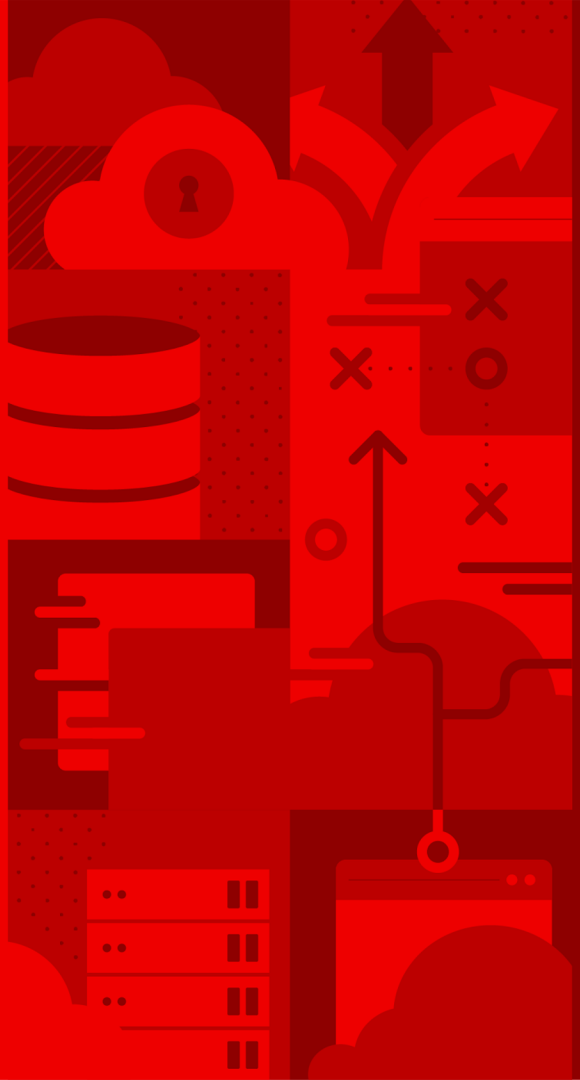
Accessing and understanding CodeReady Workspaces

Technical Exercise Introduction

Introduction to the technical exercise and what is to be accomplished

Technical Hands-On Exercises

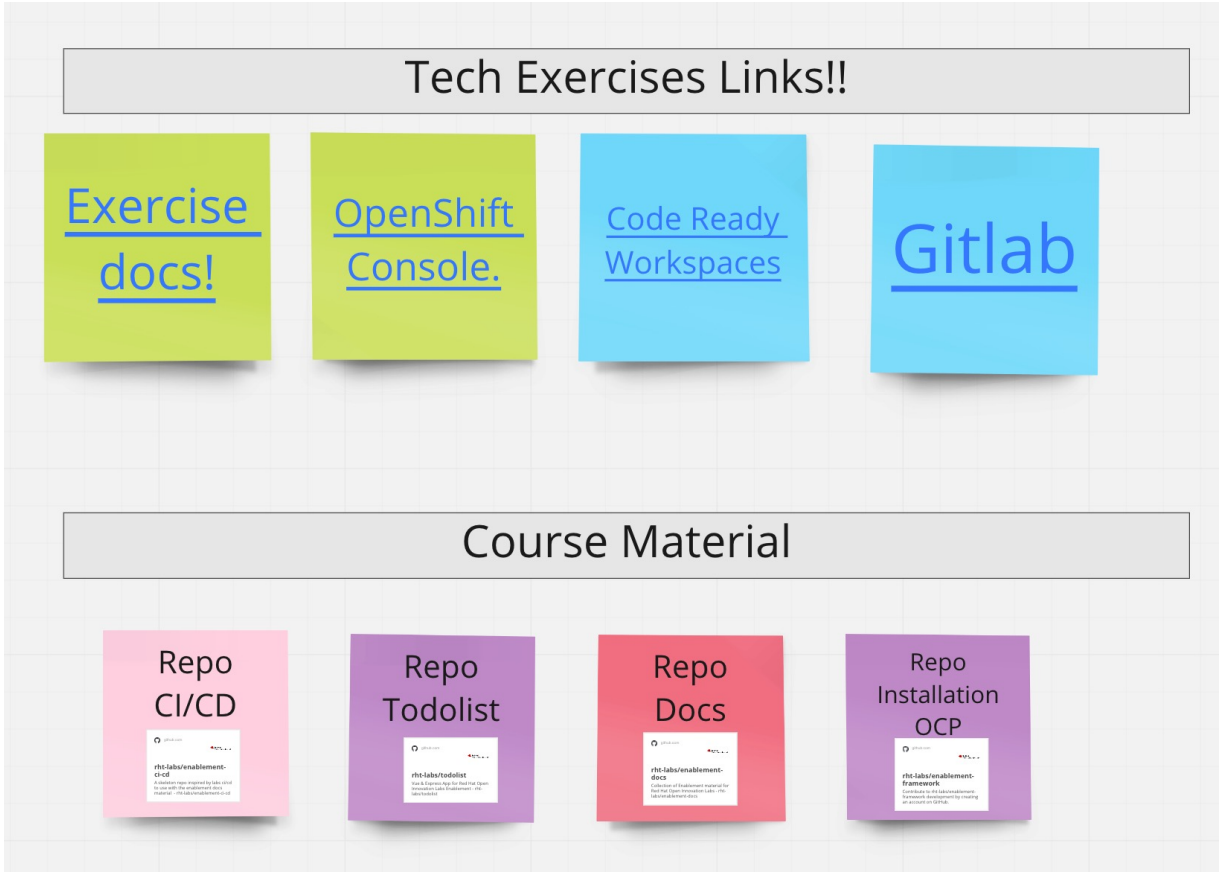
- Jenkins
- Nexus
- Gitlab
- ArgoCD
- OpenShift / HELM
- Eclipse Che (CodeReady Workspace)



Lab Setup & Access

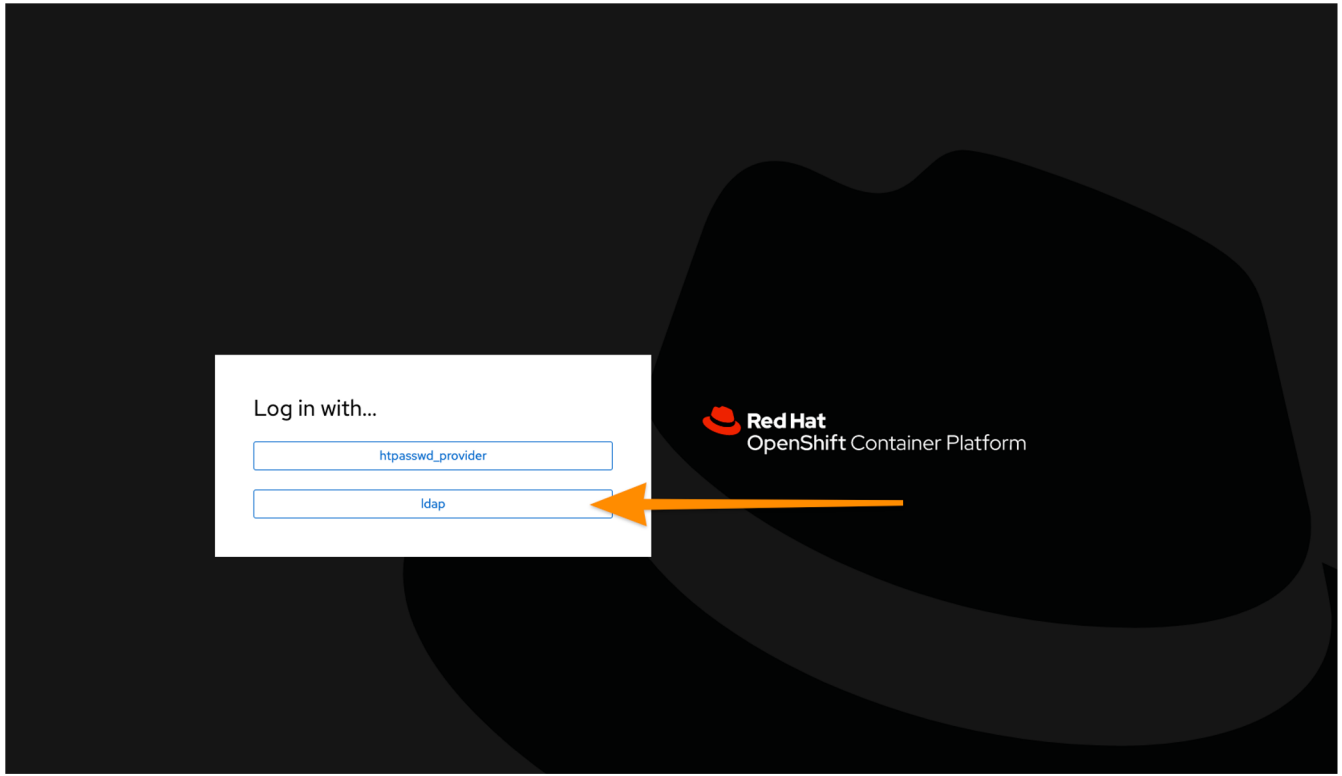
Red Hat

CodeReady Workspace Setup



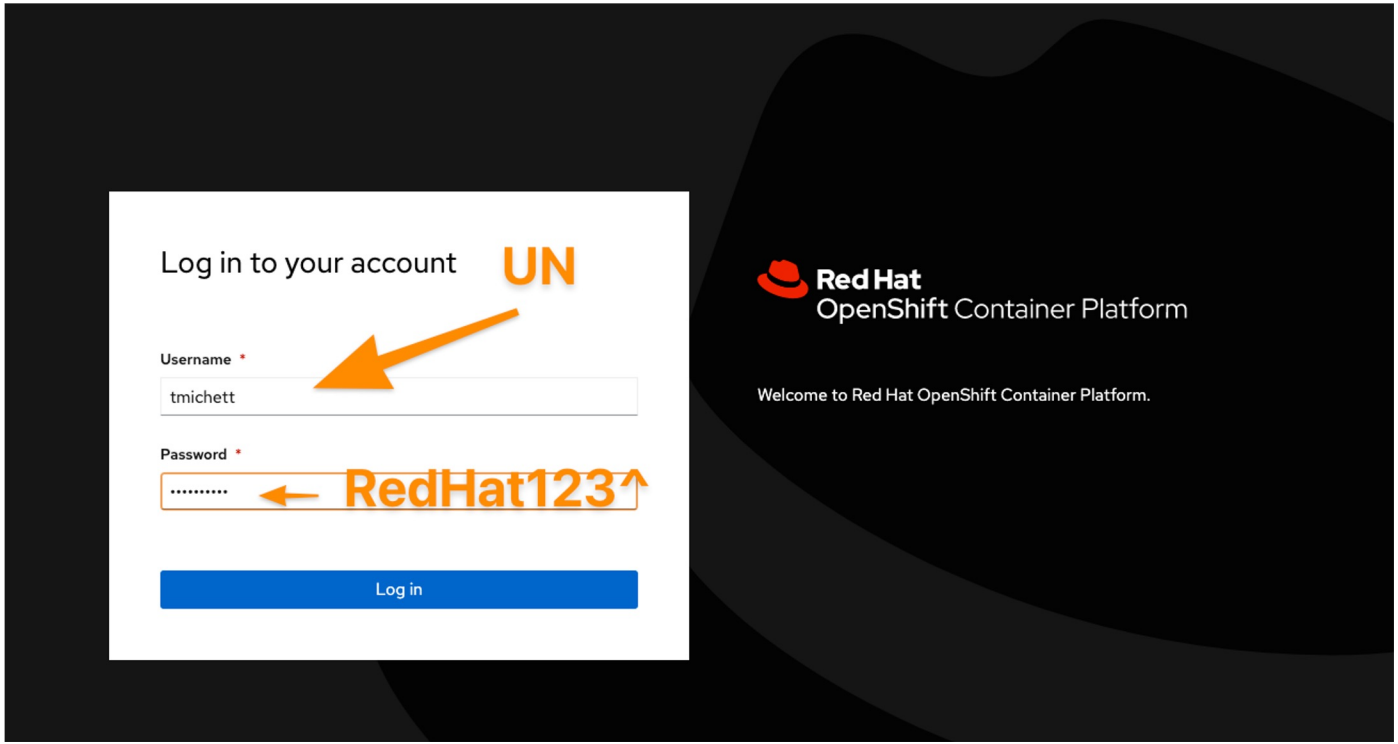
Red Hat

CodeReady Workspace Setup



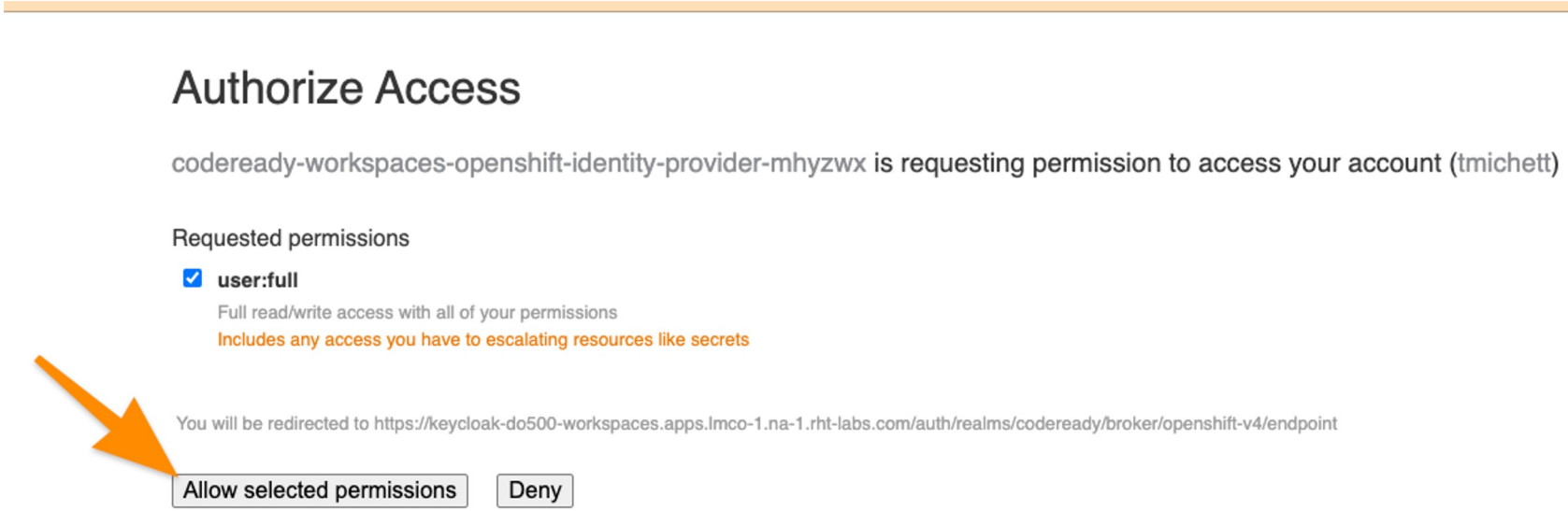
Red Hat

CodeReady Workspace Setup Cont.



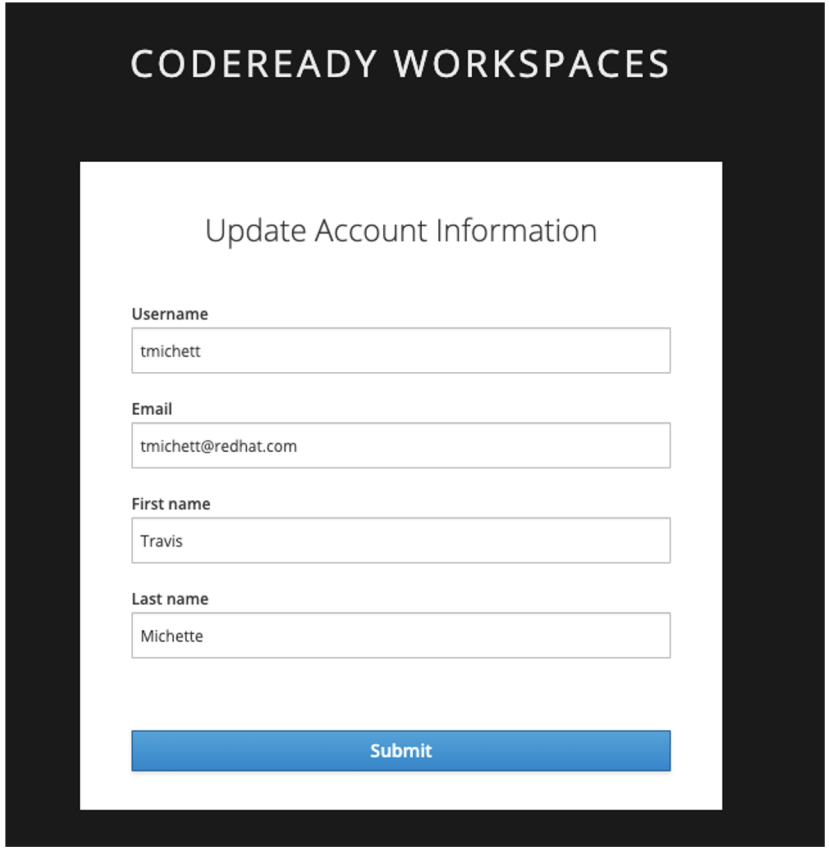
Red Hat

CodeReady Workspace Setup Cont.



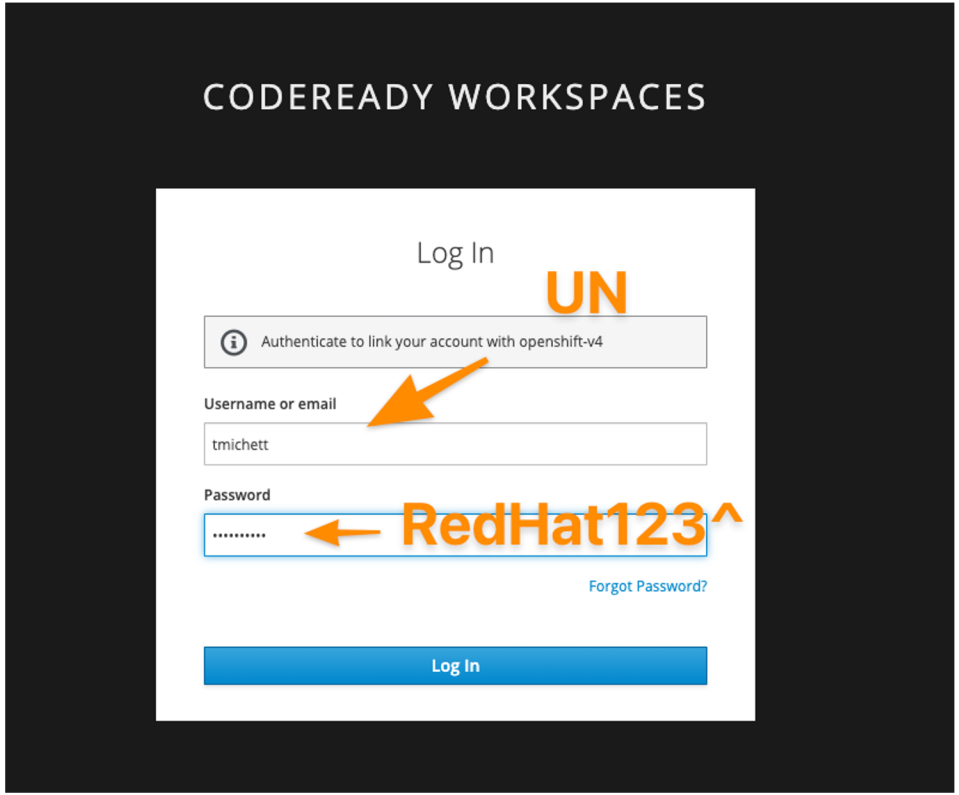
Red Hat

CodeReady Workspace Setup Cont.



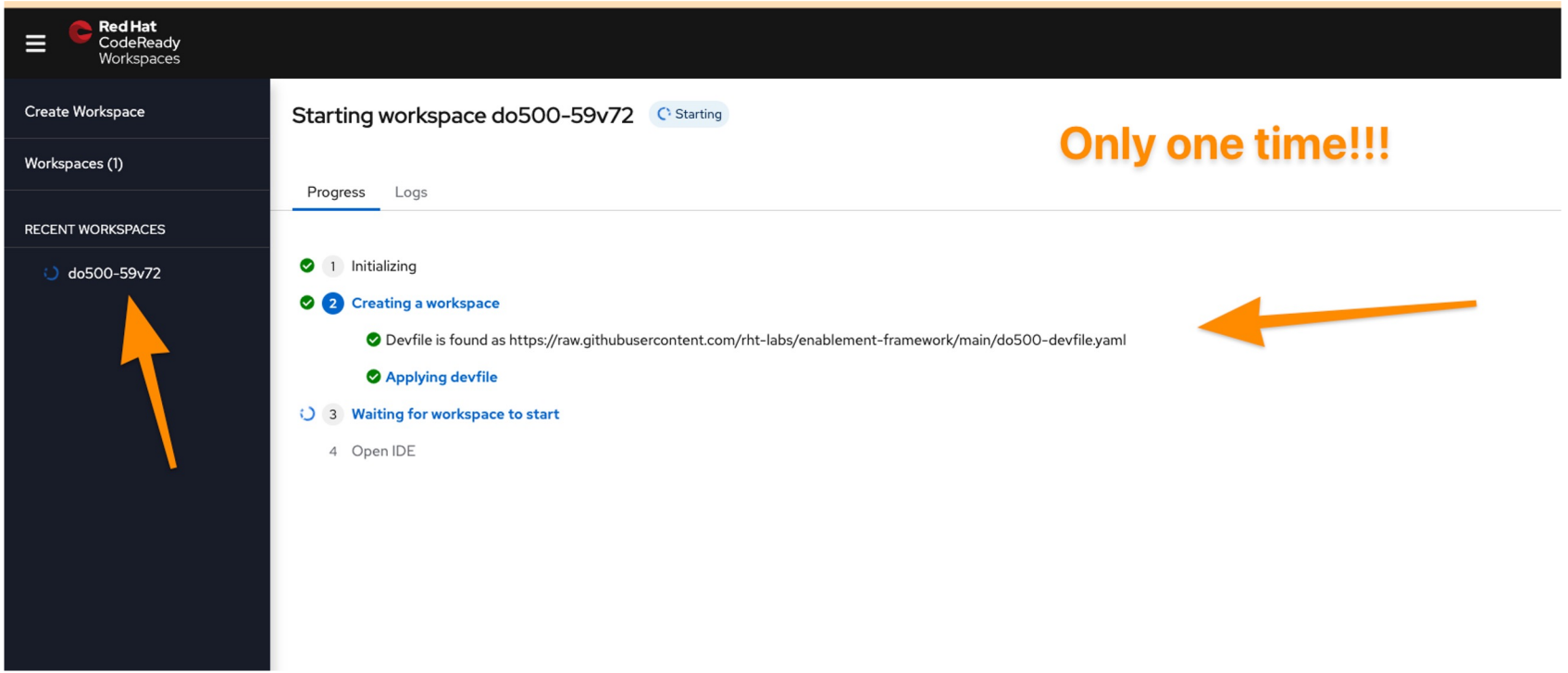
Red Hat

CodeReady Workspace Setup Cont.



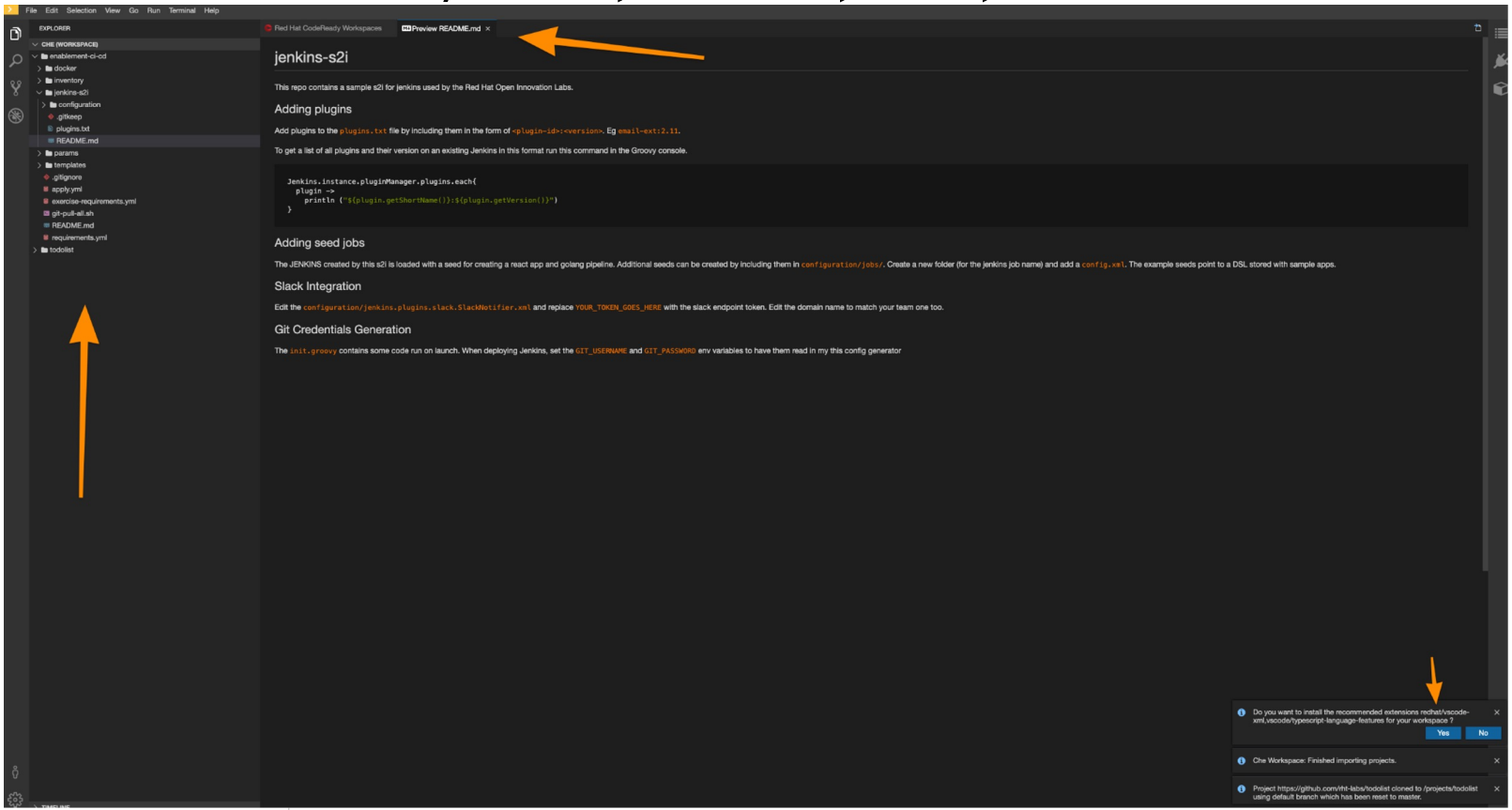
Red Hat

CodeReady Workspace Setup Cont.

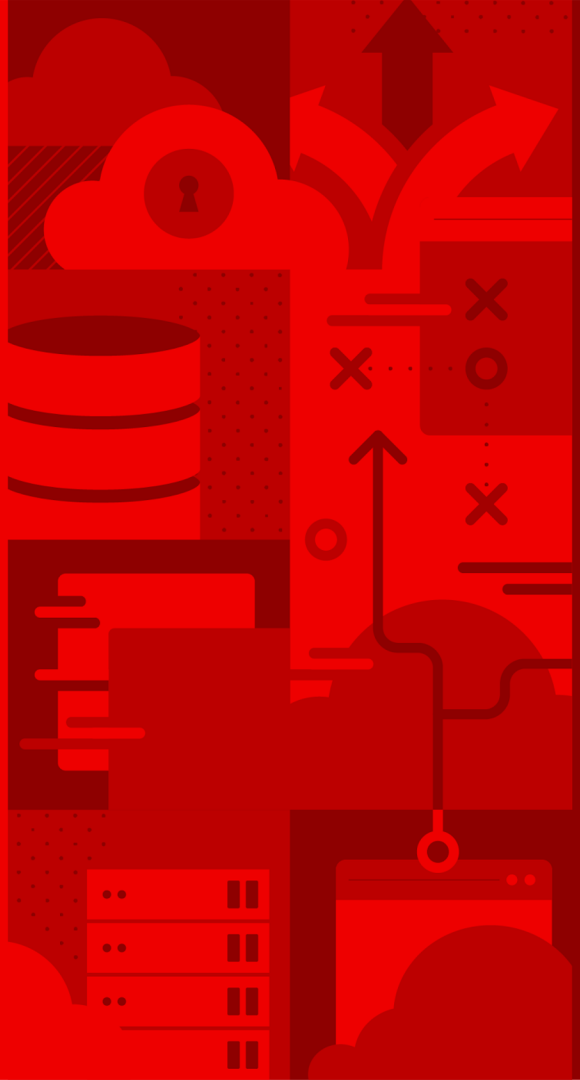


Red Hat

CodeReady Workspace Setup Completed



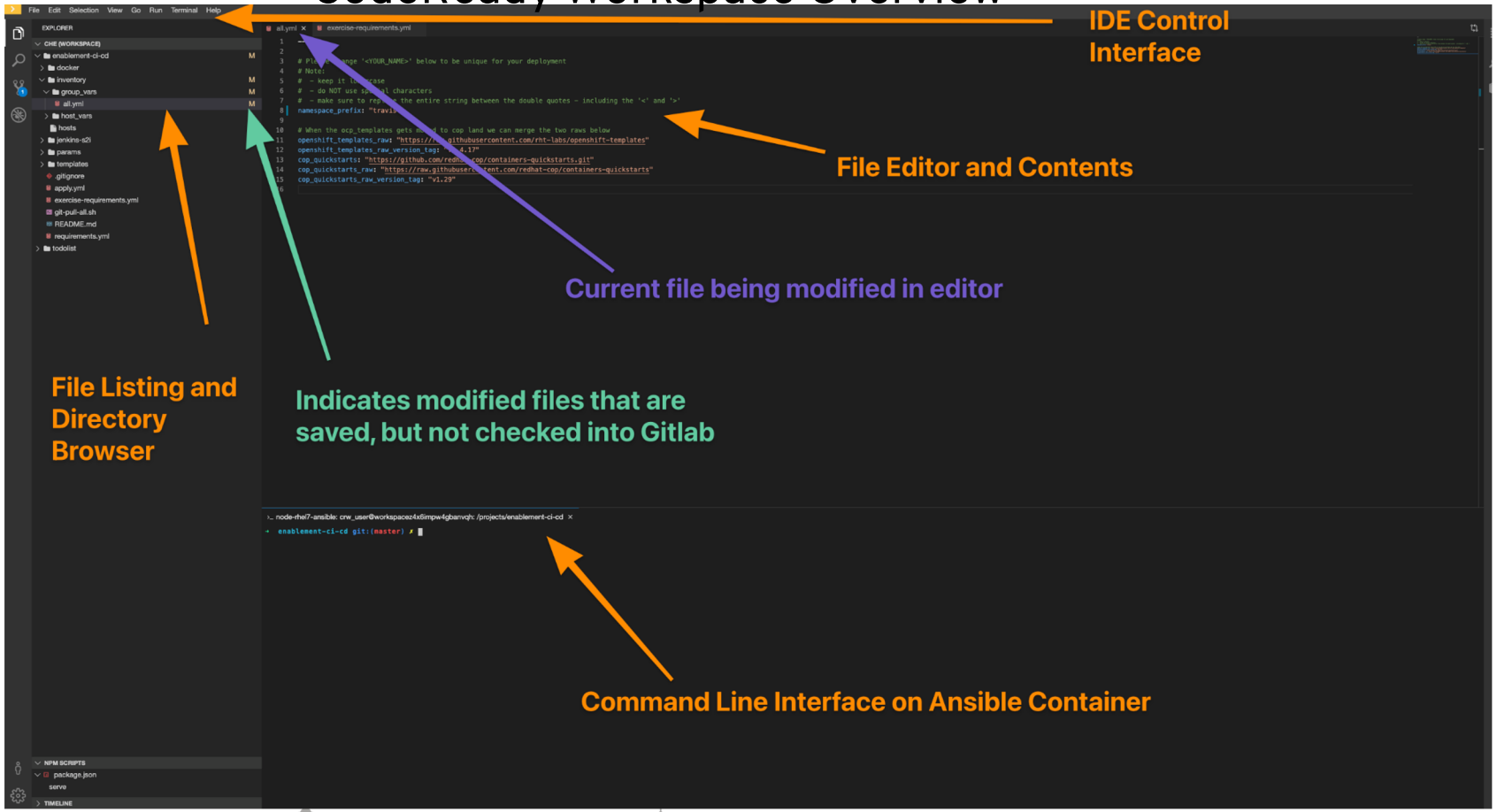
Red Hat



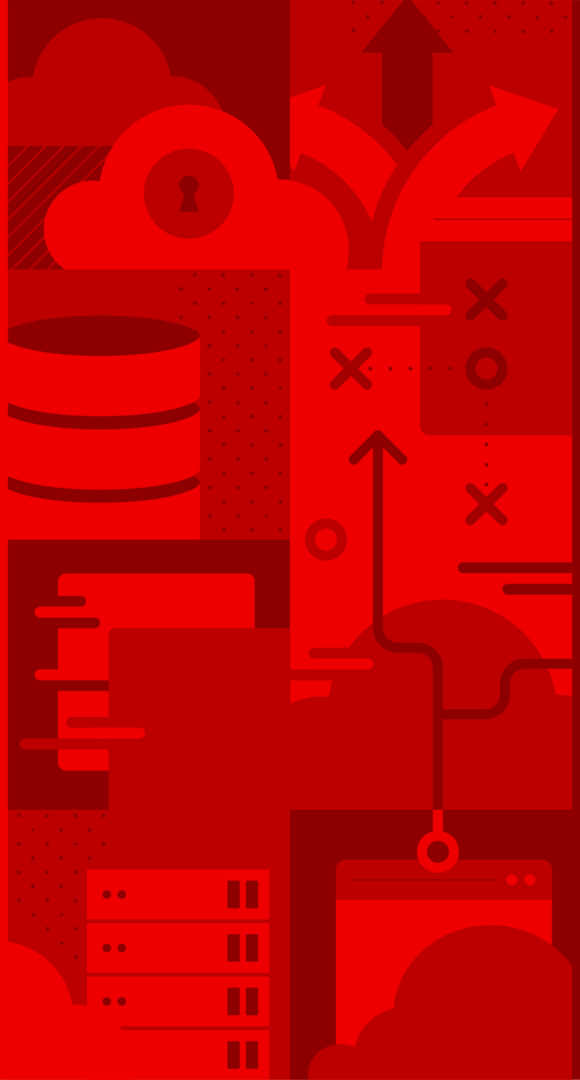
CodeReady Workspace Introduction

Red Hat

CodeReady Workspace Overview



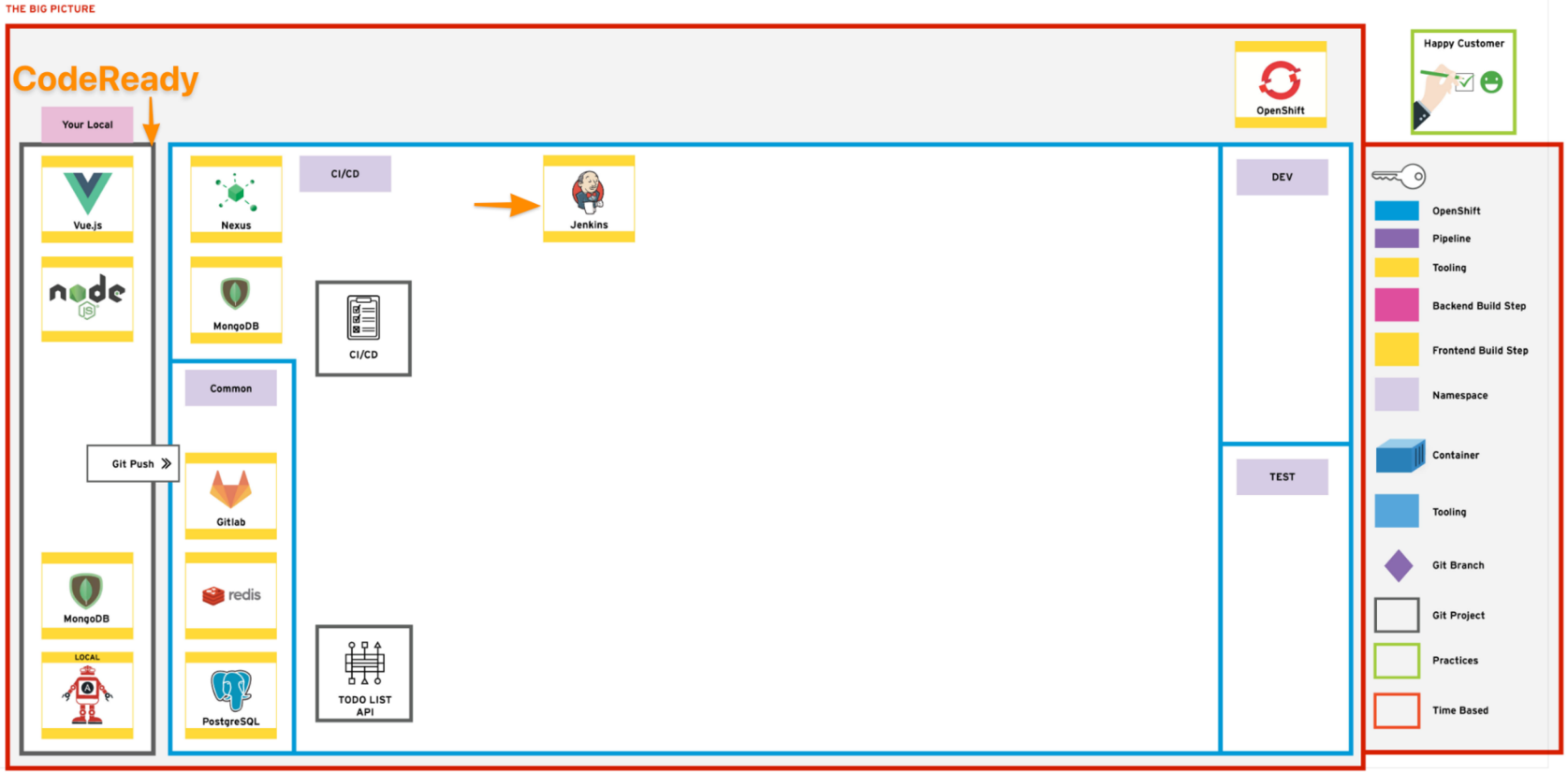
Red Hat



Technical Exercise Introduction

Red Hat

The Big Picture



Red Hat

Exercise Objectives

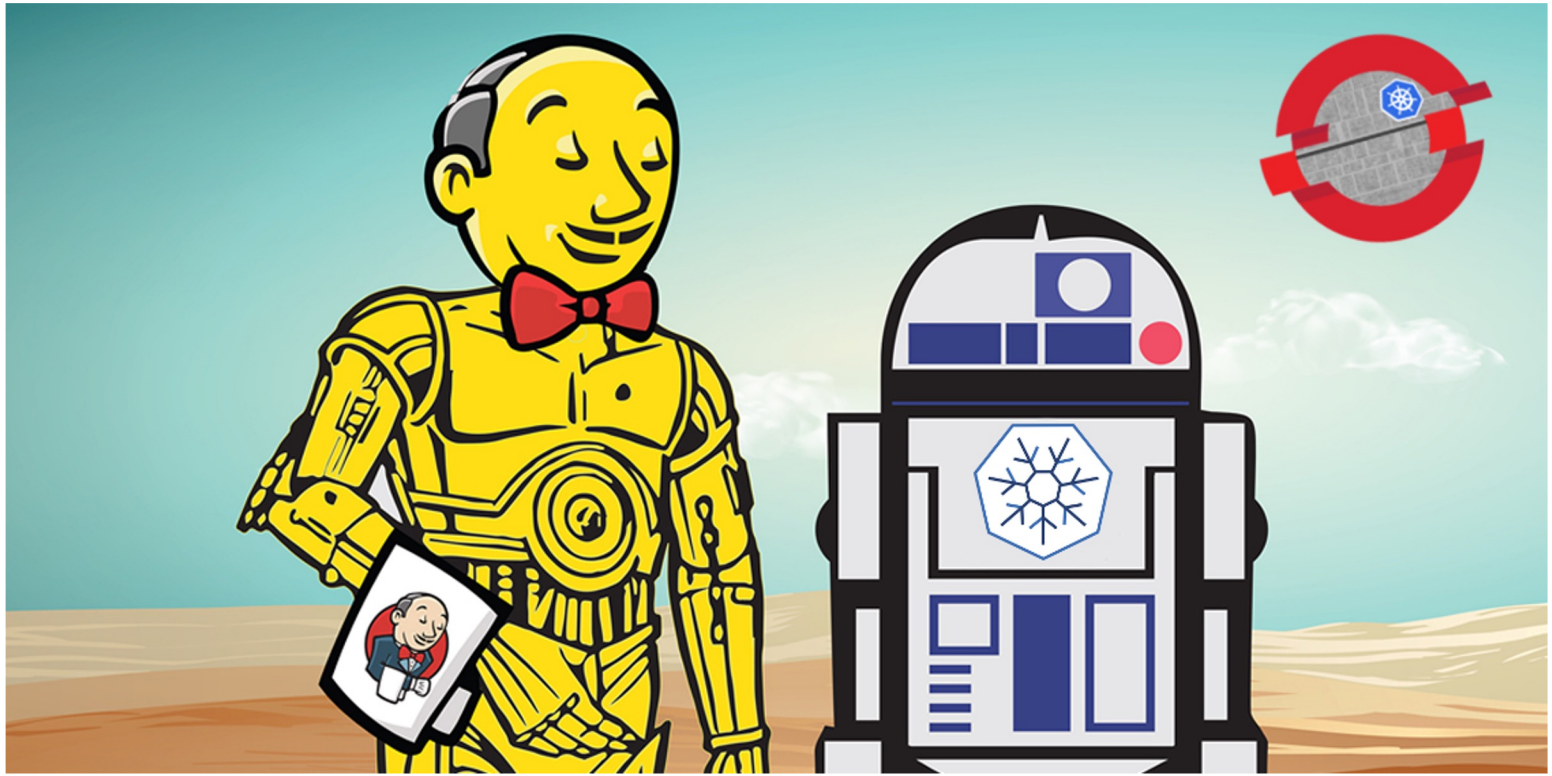
1. Setup CodeReady Workspace and OpenShift environment including user accounts
2. Access CodeReady/Che Workspace for hands-on technical exercises
3. Create initial automation environment with Configuration-as-Code
4. Building initial foundation tooling to facilitate deployment and lifecycle management of a custom application which can be automated
   - ➢ Create Project Namespaces for:
     - CI/CD Tooling
     - Development Environment
     - Test Environment
5. Create an Infrastructure-as-Code environment for a sample DevOps deployment

Tools and Frameworks Used

- Openshift Container Platform - Red Hat's private, on-premise cloud application deployment and hosting platform
- GitLab - Community driven Git server now with integrated DevOps Toolchain.
- Nexus - Repository manager for storing lots of application types. Can also host npm and Docker registries.
- Jenkins - OpenSource Build automation server. Highly customisable with plugins.
- Helm - Helps us to define, install, and upgrade Kubernetes application.
- ArgoCD - A controller which continuously monitors application and compare the current state against the desired.

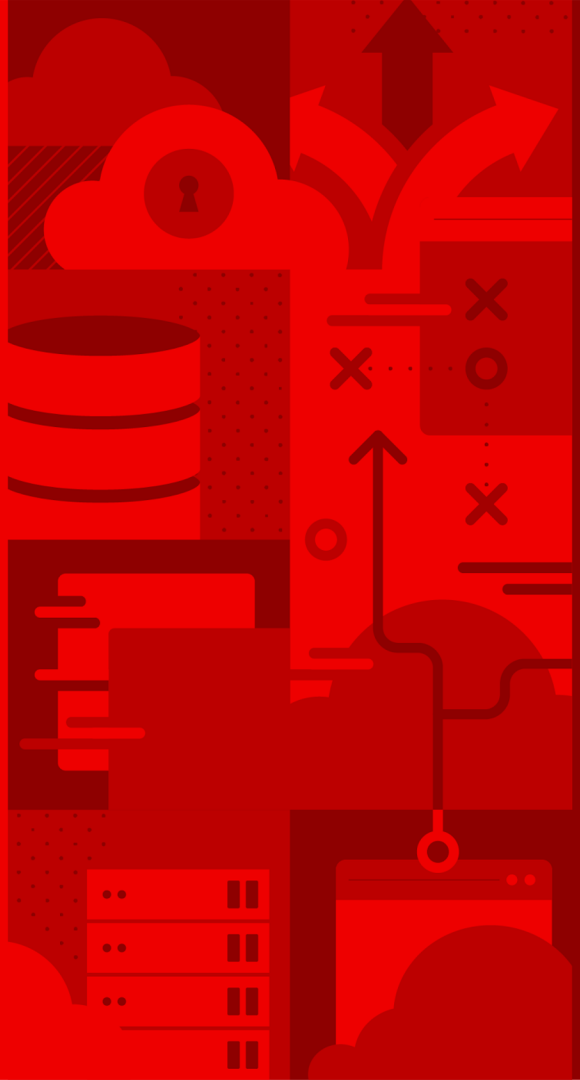
Exercise Steps (10,000' View)

1. Clone the repo `https://github.com/rht-labs/enablement-ci-cd` which contains the scaffold of the project. *(Already completed for you as part of CodeReady Workspace)*
2. Create `<your-name>-ci-cd`, `<your-name>-dev` and `<your-name>-test` project namespaces using the inventory and run them with the OpenShift Applier to populate the cluster.
3. Use the templates provided to create build and deployment configs in `<your-name>-ci-cd` project to be leveraged throughout the rest of the guided exercises.
    a. Nexus
    b. GitLab
    c. Jenkins (using an s2i to pre-configure jenkins)
4. Commit your `enablement-ci-cd` repository to the GitLab Instance you've created.
5. Burn it all down and re-apply your inventory proving config-as-code works.



Exercise 1 - Introduction Crawl

Red Hat



Technical Hands-on Exercises

What is this exercise about?

- This exercise is all about automation and **configuration-as-code**.
- This exercise is aimed at the creation of the supporting tooling that will be used to support the rest of the exercises **using code.**
- The high level goal is to create a collection of project namespaces and populate them with Git, Jenkins & Nexus **using code.**

- As a learner, you will be able to:

    1. Understand the benefits gained from GitOps approach
    2. Deploy helm charts manually
    3. Drive tool installations through GitOps

Red Hat

Why is automation important and how does it help?

- Assurance - Prevents unwanted config changes from people making arbitrary changes to environments. No more Snowflake servers!
- Traceability - Commiting **configuration-as-code** means a user has approved and changes can be tracked.
- Phoenix Server - Burn it all to the ground and bring it back; exactly the way it was!

Red Hat

Technical Hands-On Exercises

Hands-On Activities

- Infrastructure-as-Code (Gitlab)
- Pipelines (Jenkins)
- Containers and Orchestration (OpenShift)
- Container Registries (Nexus)
- Automation (Ansible)
- Local development environment (CodeReady Workspace)
- … and more

Red Hat

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

https://www.facebook.com/RedHatServices

https://twitter.com/RedHat_Services

E-Mail: tmichett@redhat.com

Red Hat