

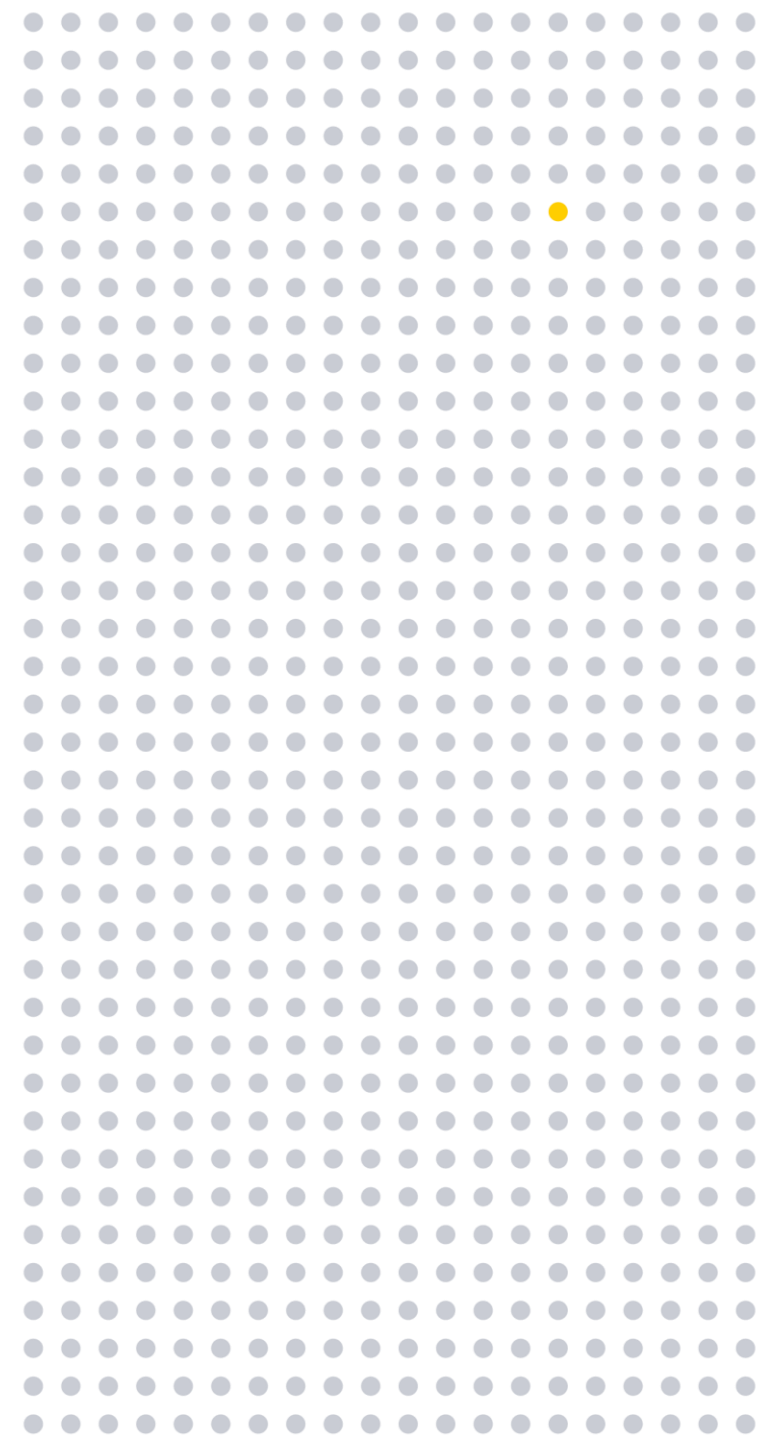


# LAB 2

# PURPOSE OF LAB

- Get familiar with Pipeline jobs
- Understanding the config of a pipeline job
- Learn how to troubleshoot a pipeline job
- Learn to trigger jobs on pull request
- Learn to start jobs from other jobs

NOTE: All exercises assume that you have gone through the functional presentation.



# CREATE PIPELINE JOB

- Create a Pipeline job called “Lab 2\_1 <your name>”
- The pipeline should have 3 stages:
  - Setup
    - Clone the repository <https://github.com/dev4242/project-x>
  - Build
    - Source the script “ci/setup.sh”
    - Run the script “apps/app00/test/test.sh”
  - Cleanup
    - Remove the cloned repository

# CREATE A PIPELINE JOB WITH CUSTOM CONFIG

- Copy the job from the last exercise and name it Lab2\_2 <Your name>:
  - Add setup path and script path as parameters
  - Add an option to not run the cleanup

# FIX THE FAILING PIPELINE

- Create a new job as a copy of the “Lab2 Error Pipeline” job and name it “Lab2\_3 <Your name>”
- There are errors in the pipeline
  - Check the logs to identify the errors
  - Fix the errors

# TRIGGER PIPELINE ON PULL REQUEST

- Create a job that is a copy of “Lab2\_1 <your name>”, name it “Lab2\_4 <your name>”
- Modify the job so that it triggers when a pull request is made in repo <https://github.com/dev4242/project-x>
- Print the commit message of the triggering commit in the pipeline job
- Trigger a pull request by running [Rebuild Pull Request 1](#)

# TRIGGER PIPELINE ON PULL REQUEST CONT.

- Create a new job called Lab 2\_5 <your name>”
  - At build-stage it should build the job “Lab 2\_2 <your name>” with appropriate parameters.