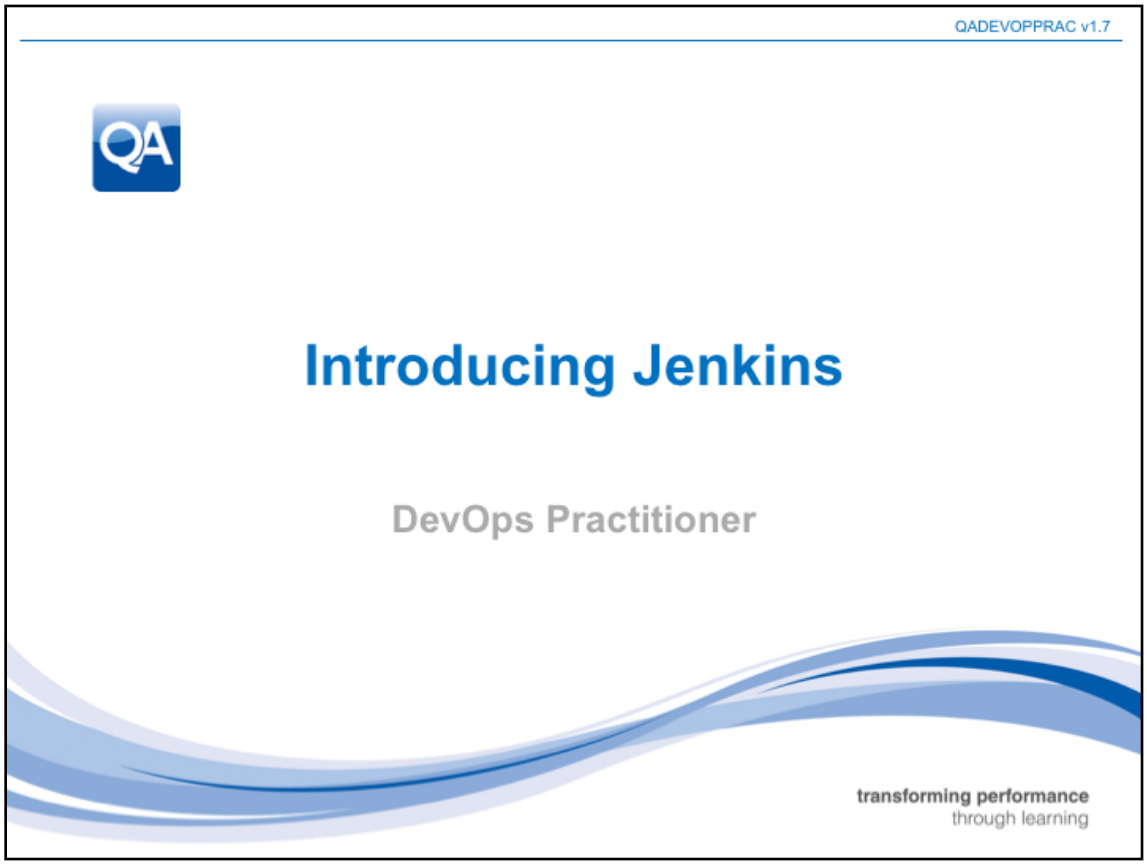
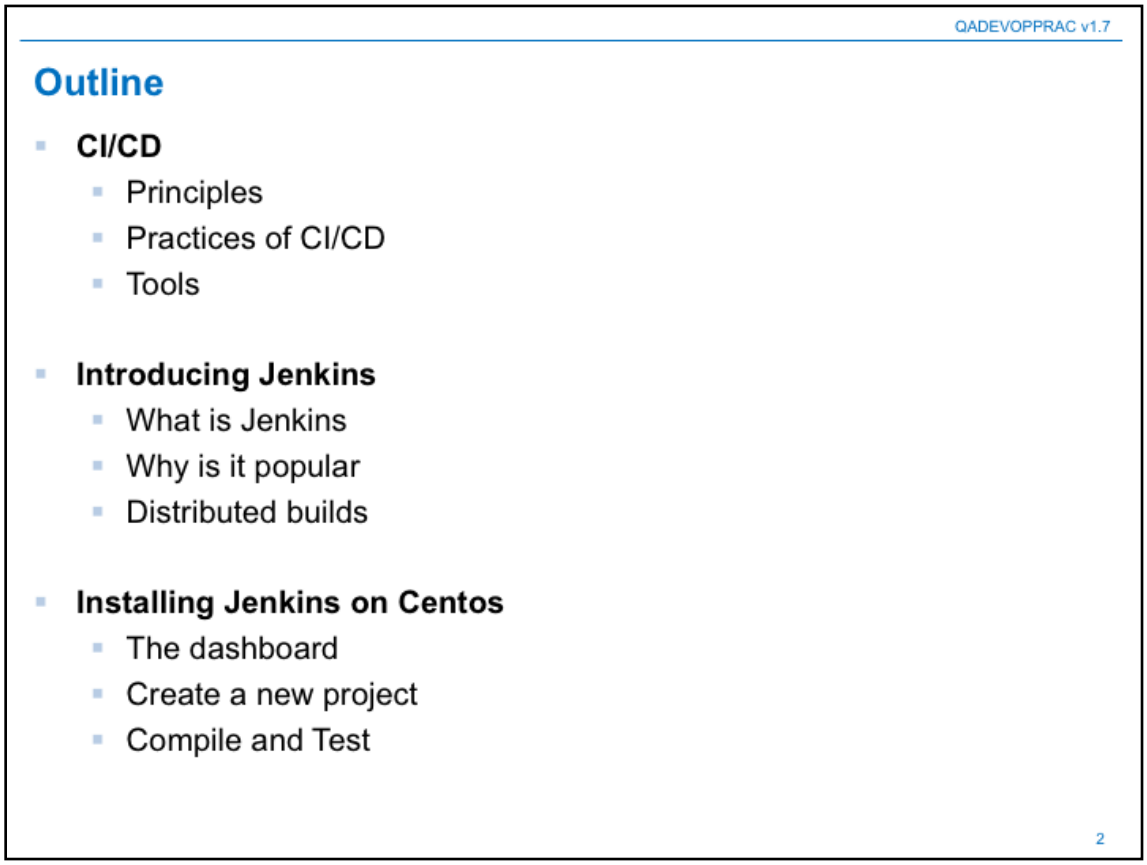
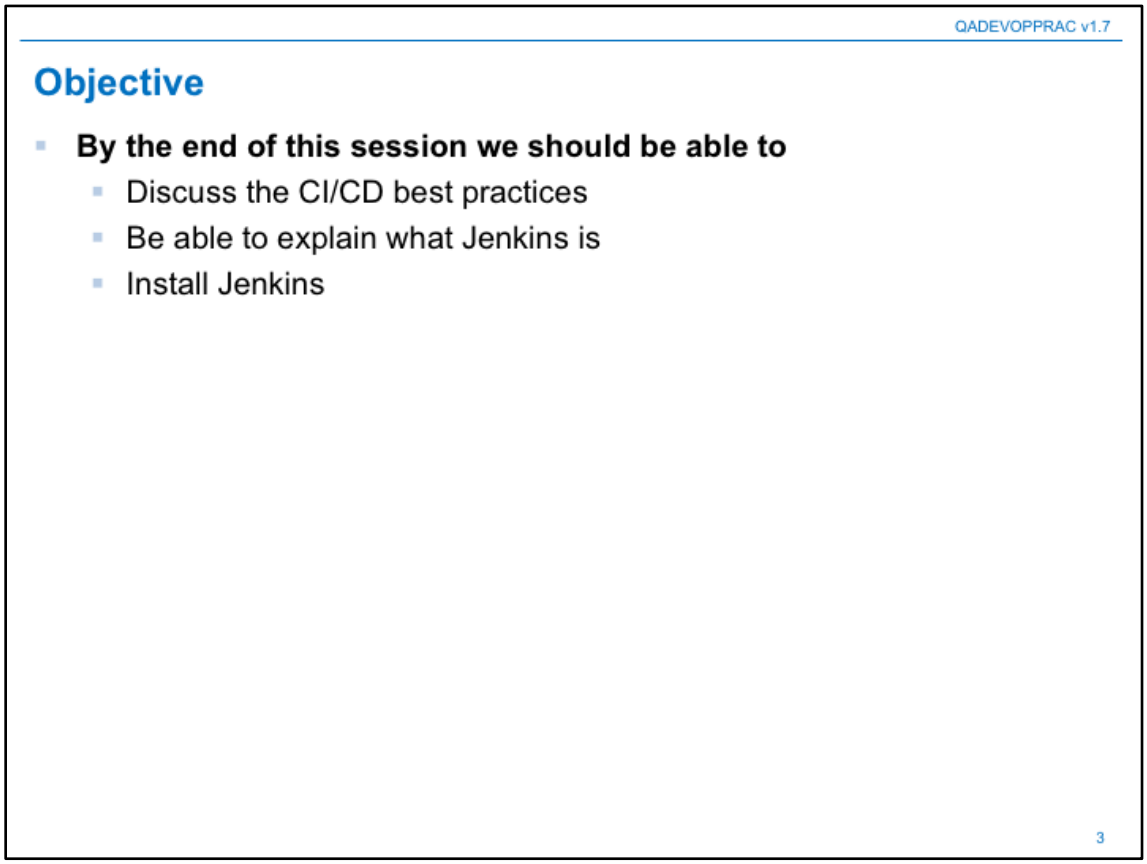
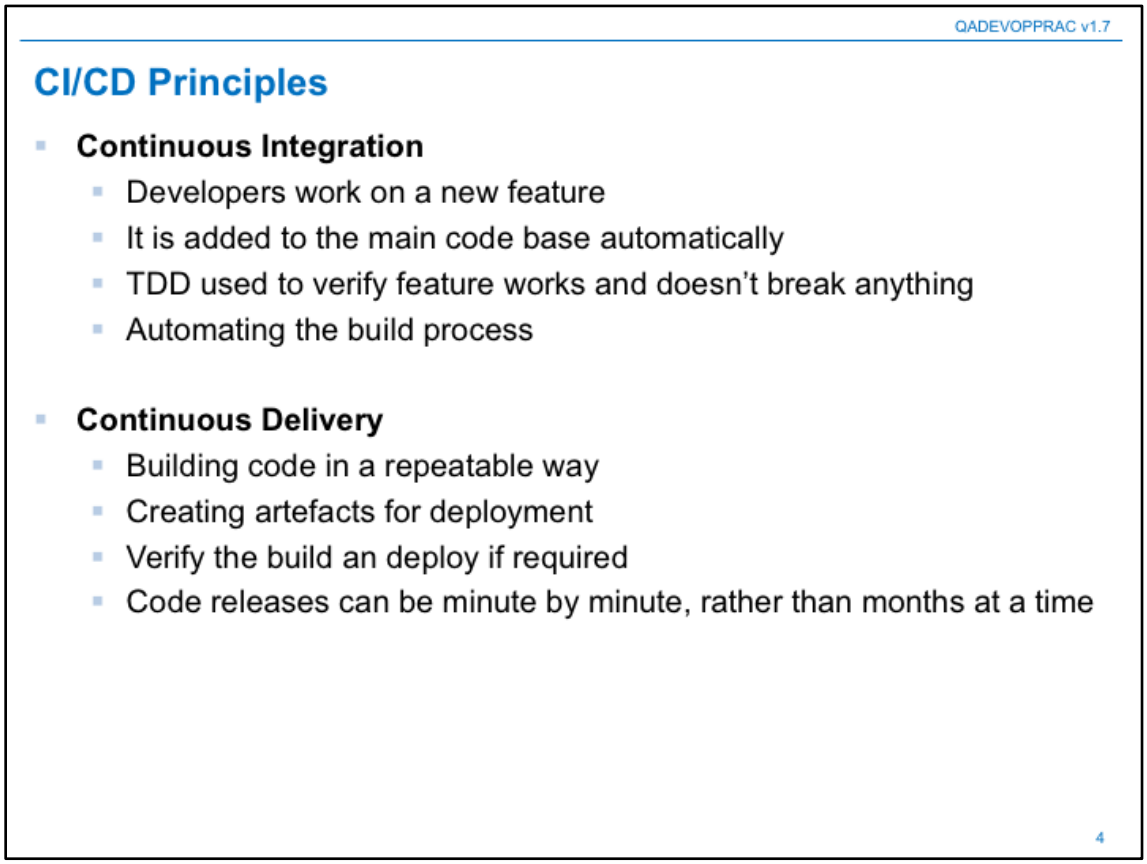

QADEVOPPRAC v1.7
Introducing Jenkins
DevOps Practitioner
transforming performance
through learning


QADEVOPPRAC v1.7
Outline
CI/CD
Principles
Practices of CI/CD
Tools
Introducing Jenkins
What is Jenkins
Why is it popular
Distributed builds
Installing Jenkins on Centos
The dashboard
Create a new project
Compile and Test
2


QADEVOPPRAC v1.7
Objective
By the end of this session we should be able to
Discuss the CI/CD best practices
Be able to explain what Jenkins is
Install Jenkins
3



Continuous Integration – source control (specifically using git)

Developers work on a new feature – on the same time.

Code can be submitted when is done to repo. Repo monitors merging conflicts of a code from two people.

Testing – following TDD through out.

Automating the build process – when code is submitted, its auto tested and if it does not have conflicts – built.

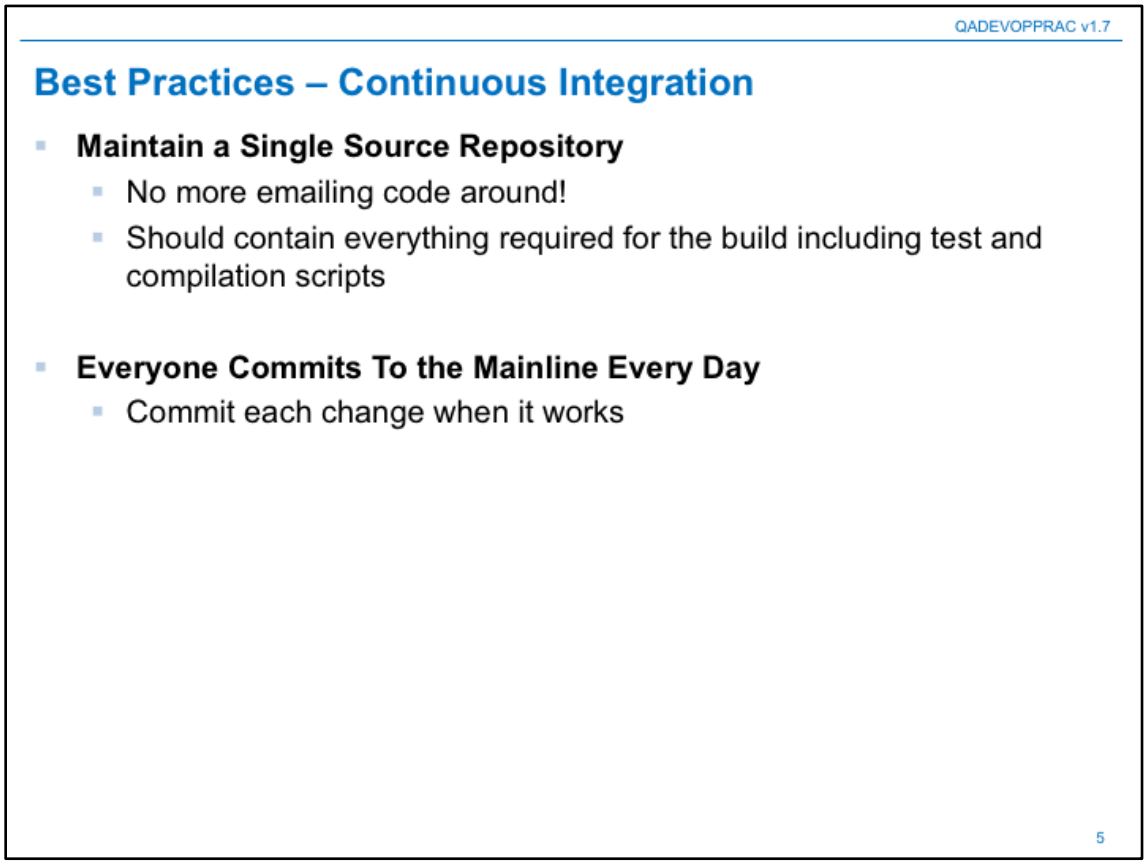
Continuous Delivery – process to make sure, any time the commit happens to source control, it then builds a code.

Building code in a repeatable way –

Creating artefacts for deployment – always have deployment artefacts that is ready to go to deployment

Verify the build an deploy if required - when the code is build, it goes to deployment stage and goes through other specific tests: to make sure, that is safe to go into production

Code releases can be minute by minute, rather than months at a time

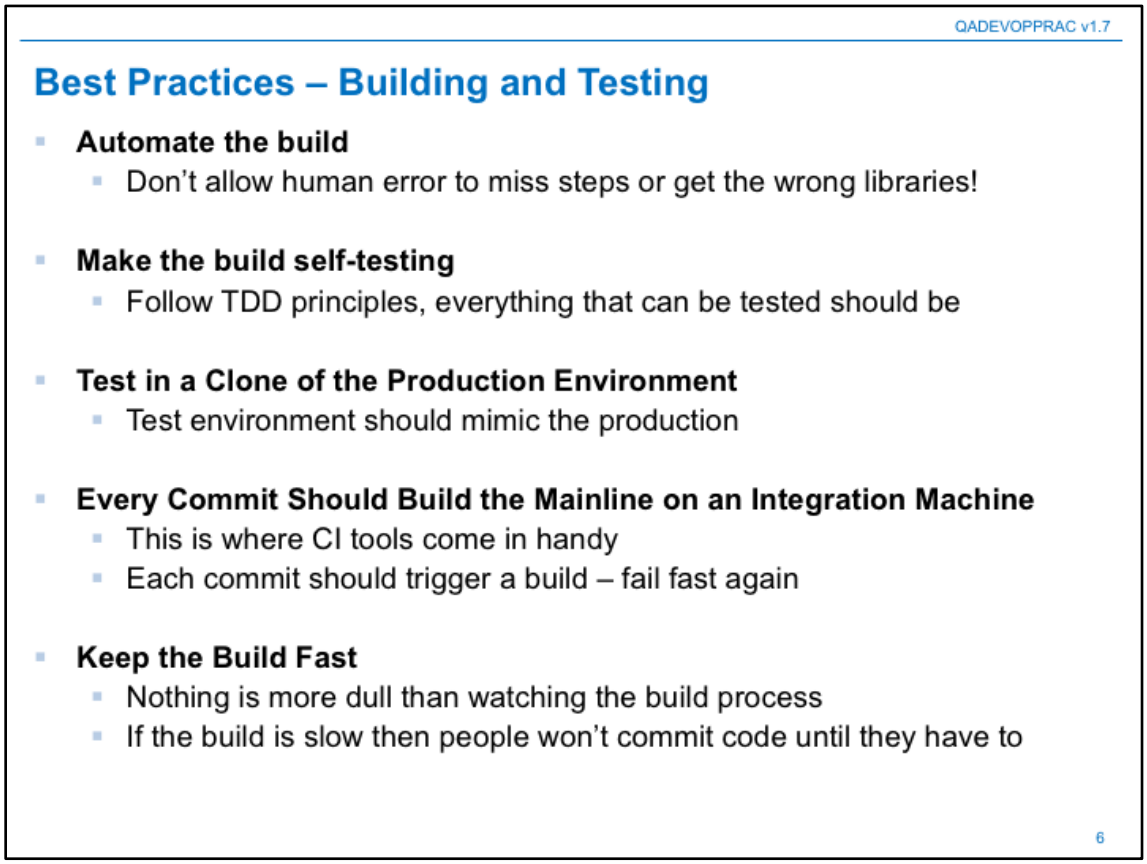


Adapted From: http://martinfowler.com/articles/continuousIntegration.html

Maintain a Single Source Repository

One source repo for the project (not one repo for one person.

After you complete the work (and it work without errors), you push it to repo.



We want to automate every single step in the build process (locally with Maven, server based solution with Jenkins).

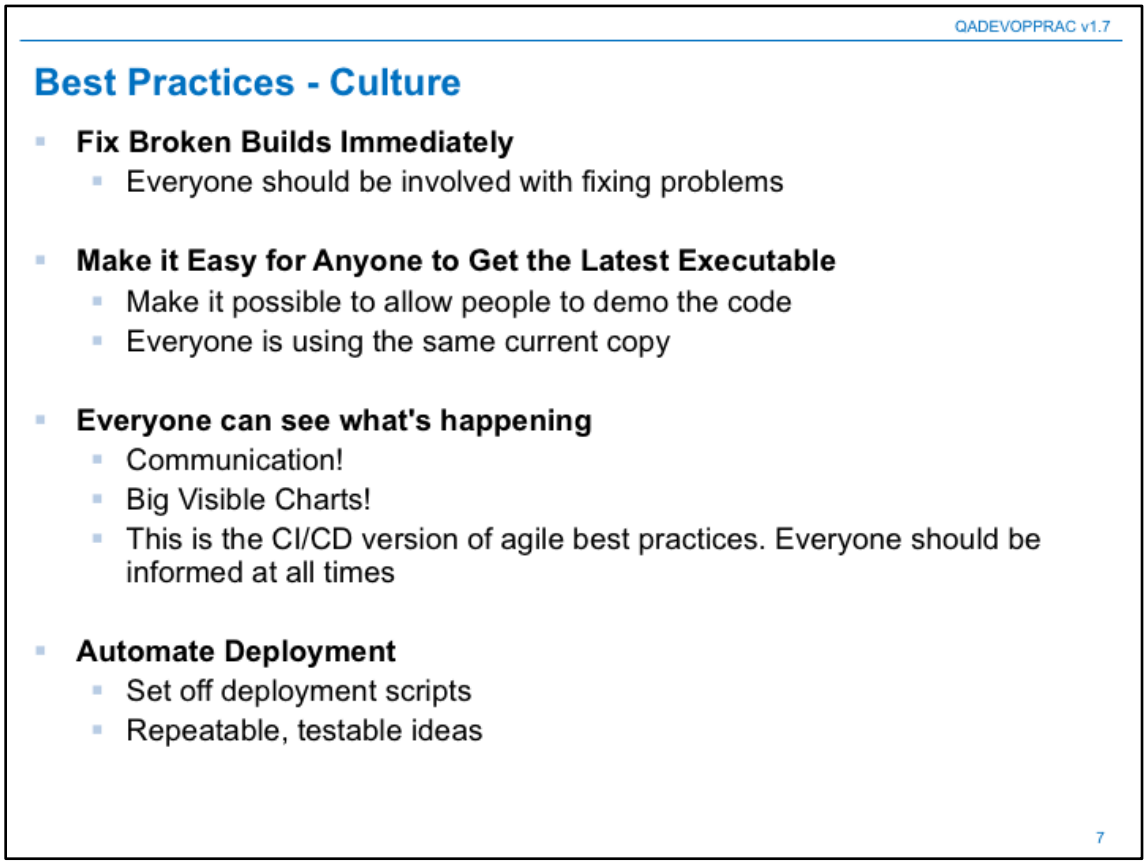
Make the build self-testing

Test environment should mimic the production – test it on the different machines

Every commit should build that branch in the server machine.

Every new code is submitted – should be build automatically

Build fast!!!

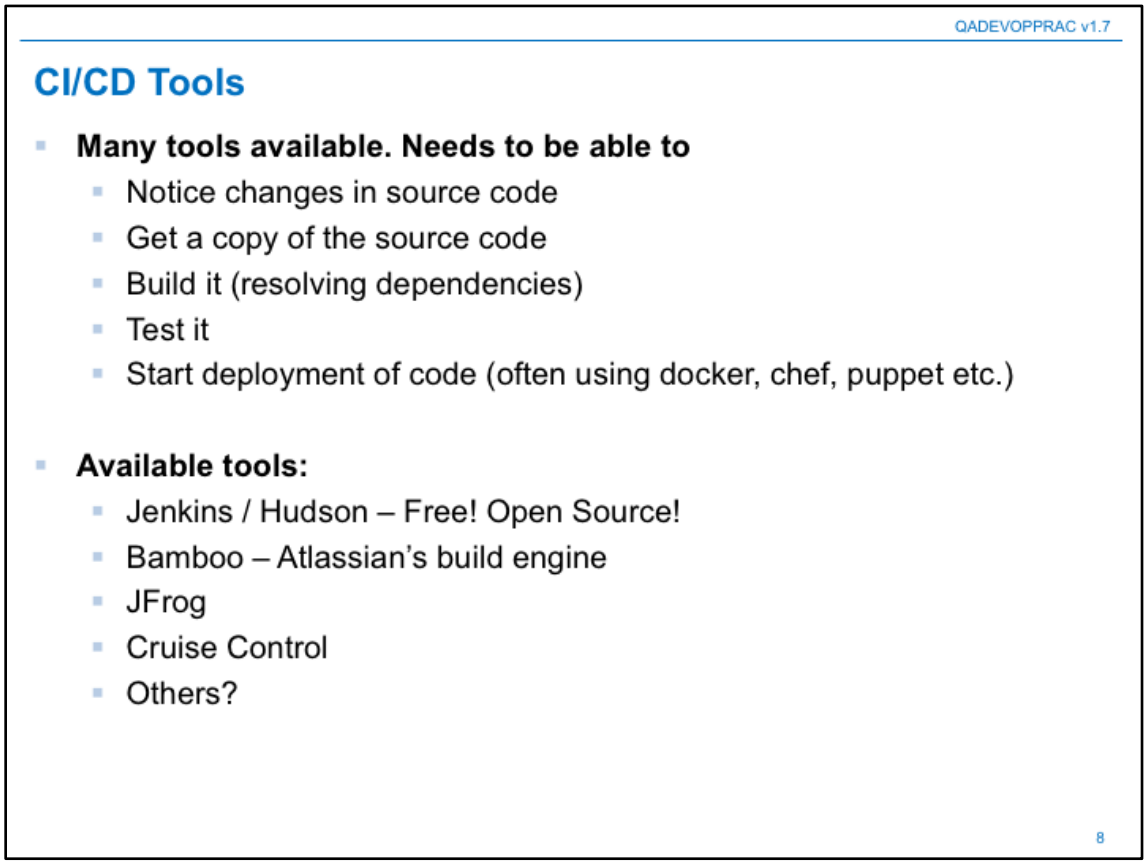


Fix Broken Builds Immediately – don't build a new code on a broken code

Don't hide the latest executable code from your team or boss (make sure that is in the single place and people knows how to get it)

Everyone needs to be informed what is going with the project on exact time (charts, board in the room etc).

Automate as much as humanly possible deployment process



CI / CD tools are a collective group covering the source control, the package managers and the build tool itself. We have been using git and maven for the first two. We will be investigating Jenkins for the build process.

Introducing Jenkins

- **First released in 2005 as the Hudson project**
  - Created by Sun Microsystems
  - Oracle bought the company and so had control of the Hudson name
  - Name changed to Jenkins for the open source project
  - Both the Hudson and Jenkins project are still active but development has diverged
- **Created as a CI / CD tool**
  - Aim was to be easy to install and use
  - Plugins to connect to source control
  - Builds projects using the project setup – so can work with any language
  - Public / company facing dashboard lets everyone know the status of the current build

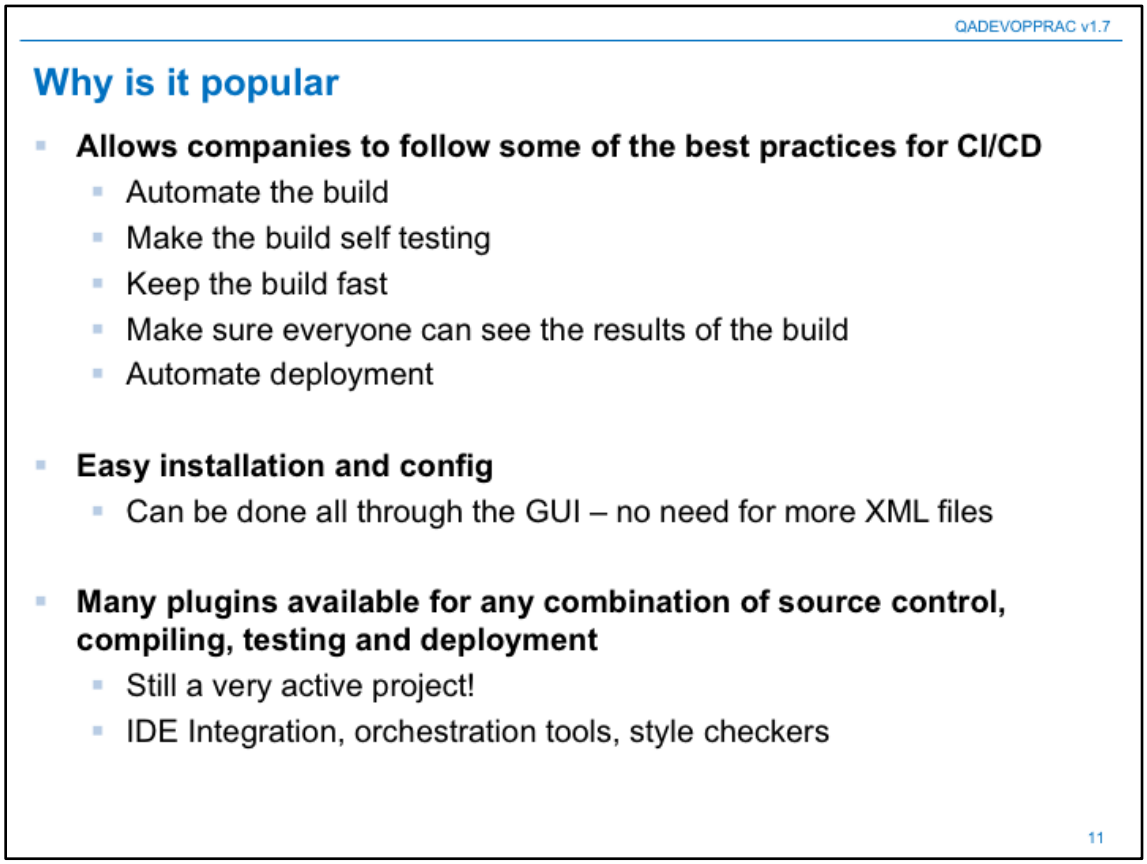
Open source version of Hudson was called Jenkins.

Jenkins project is more active then Hudson. There are a lot of references from Jenkins to Hudson

It gives a public dashboards – everyone can see what is going on with the project

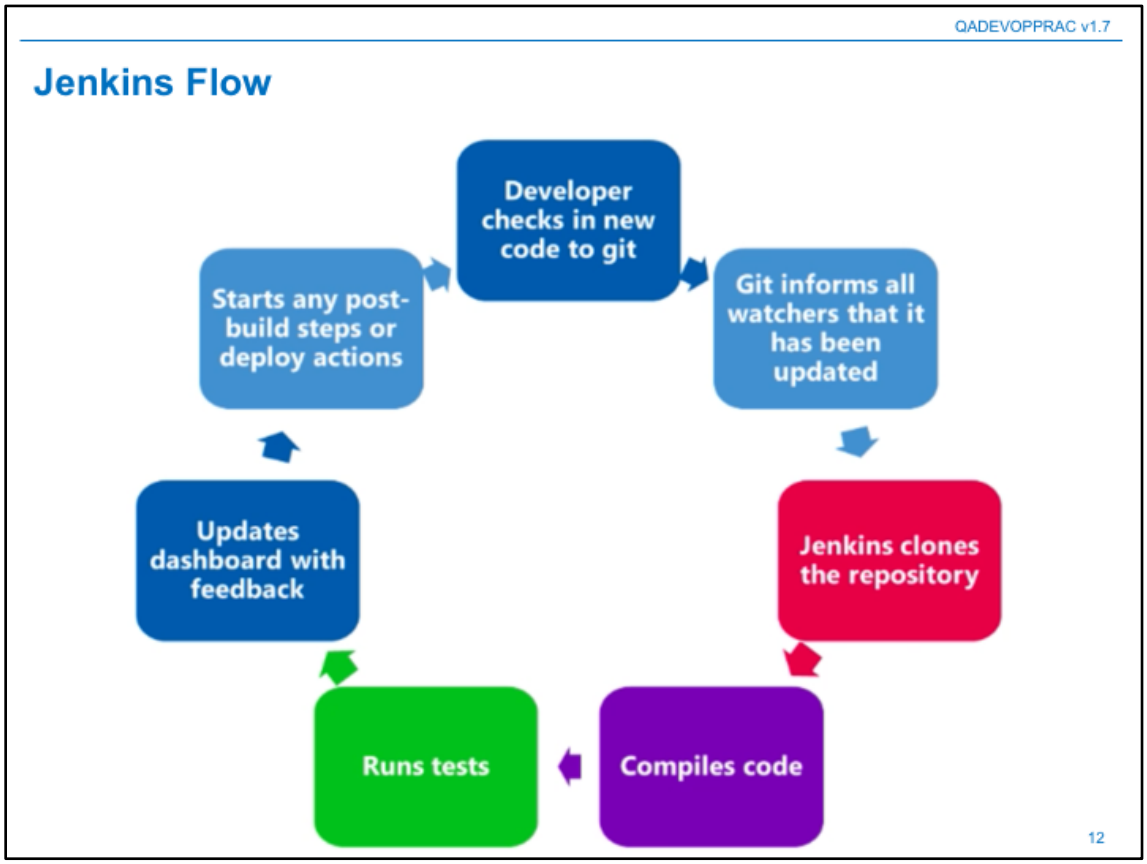
What does Jenkins do

- **Jenkins is a one stop shop for testing and building code**
  - It can even deploy code onto servers if they already exist
- **Usual workflow:**
  - Developer checks some new code into git
  - Git informs all watchers that it has updated
  - The Jenkins git plugin starts the build process
  - Clones the project
  - Compiles, runs tests
  - Updates the dashboard with any feedback
  - Start the post-build steps or actions to deploy



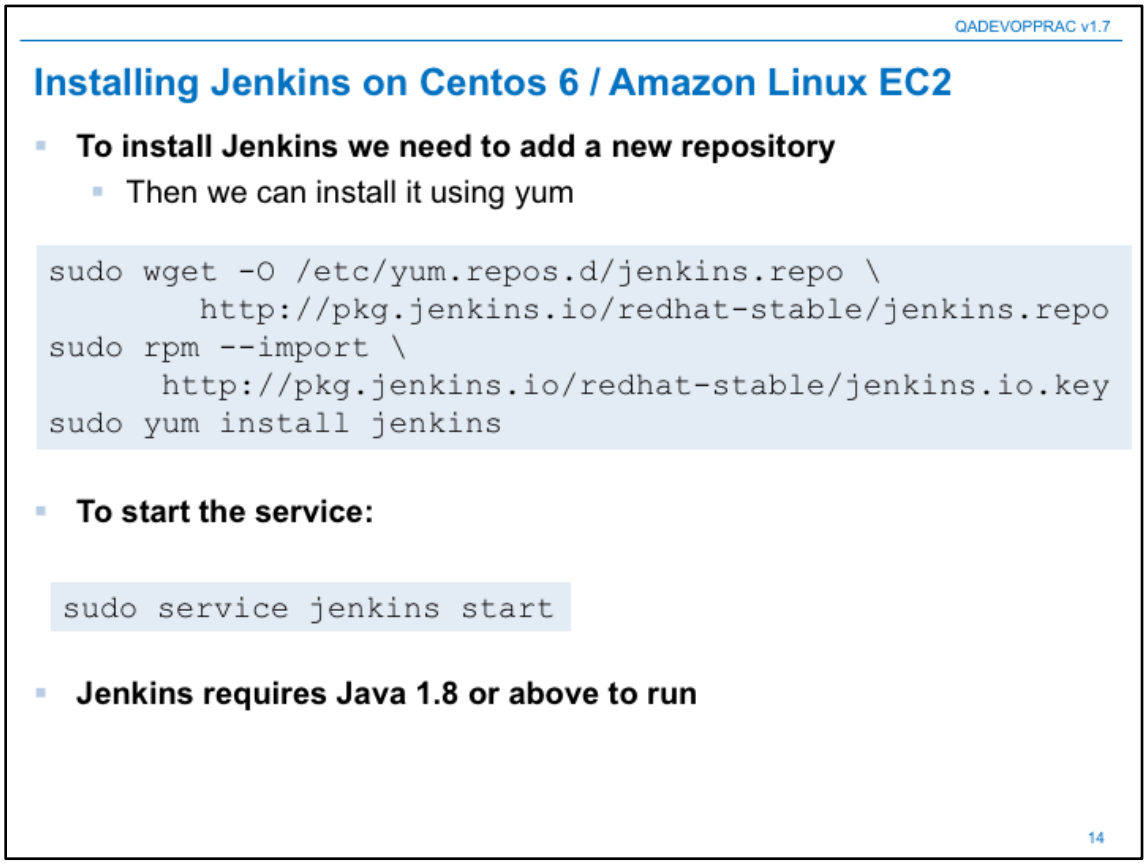
Want to avoid integration hell: http://c2.com/cgi/wiki?IntegrationHell

Publically visible dashboards: https://www.google.co.uk/search?ie=UTF-8&q=%22Dashboard+%5BJenkins%5D%22&gws_rd=cr,ssl&ei=XLB2VZ6KEfCU7QbB_4PwBQ


QADEVOPPRAC v1.7
Jenkins Flow
Developer checks in new code to git
Git informs all watchers that it has been updated
Jenkins clones the repository
Compiles code
Runs tests
Updates dashboard with feedback
Starts any post-build steps or deploy actions
12

Distributed Jenkins Servers

- **Jenkins uses agents to scale systems**
  - By default there is one agent which will build using one core
  - May need to build for different architectures
    - Windows / Mac / Linux
    - Android / iOS / Windows Phone
  - Master / Slave servers
    - Can exist anywhere
- **Single report can be generated over all the builds**



You may need to install wget:

```
sudo yum install -y wget
```

Installing Jenkins on other systems

- **Ubuntu**

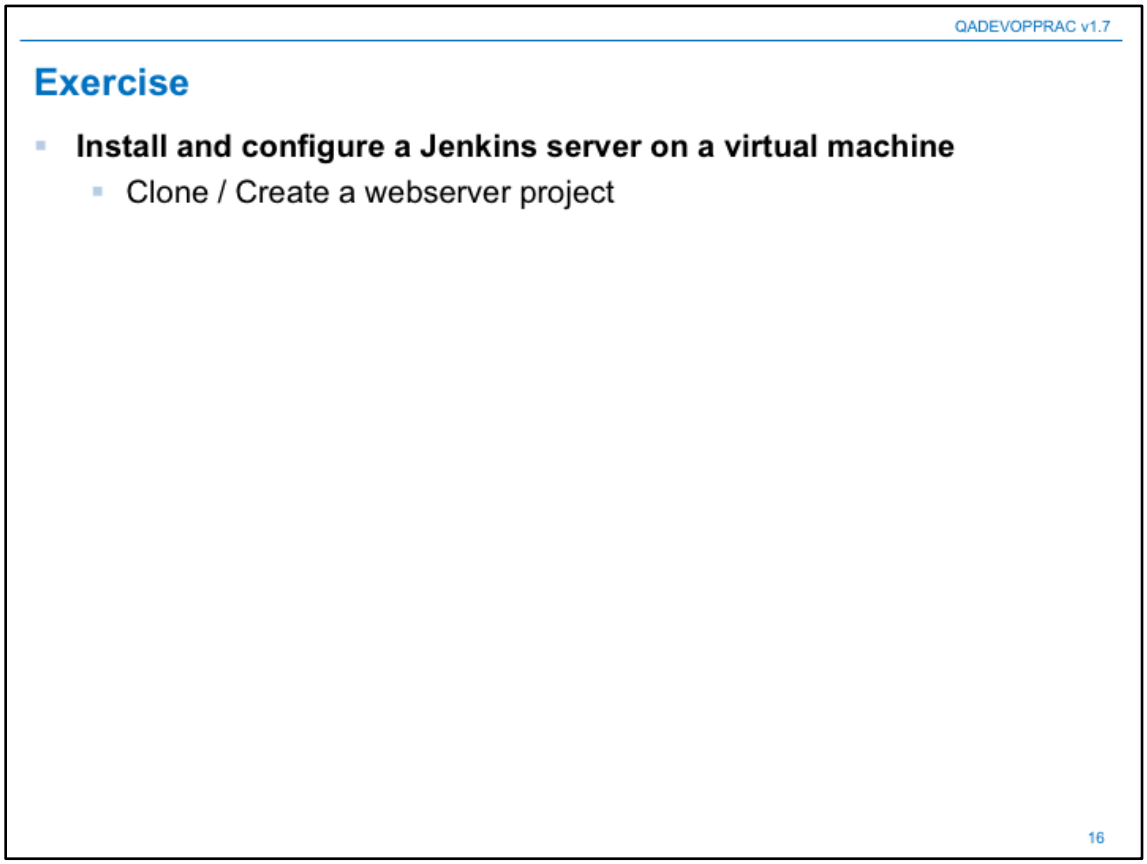
```
$ sudo apt-get install jenkins
```

- **Using Docker:**

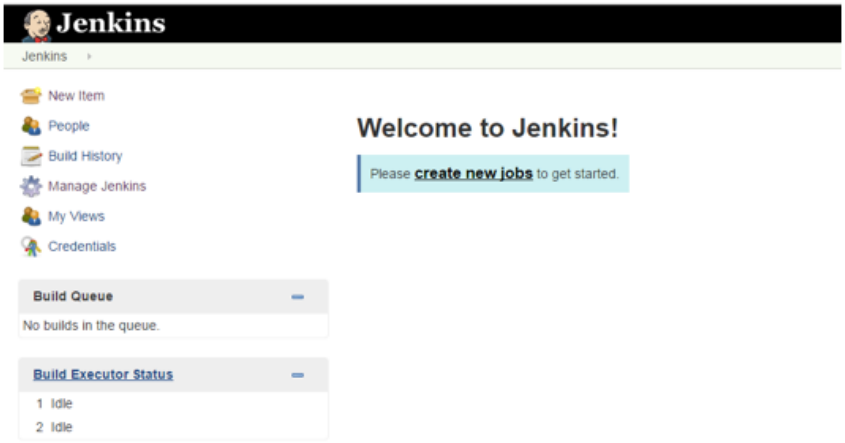
```
$ docker run -d -p 8080:8080 jenkins
```

- **Windows:**
  - Uses the installer

```
https://wiki.jenkins-ci.org/display/
JENKINS/Installing+Jenkins+as+a+Windows+service
```


QADEVOPPRAC v1.7
Exercise
Install and configure a Jenkins server on a virtual machine
Clone / Create a webserver project
16

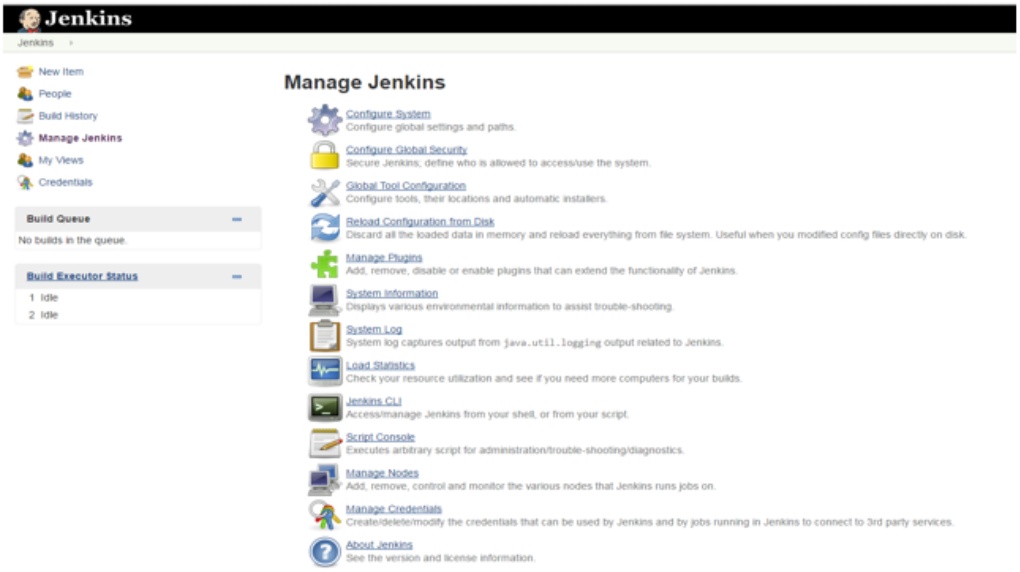
The Dashboard



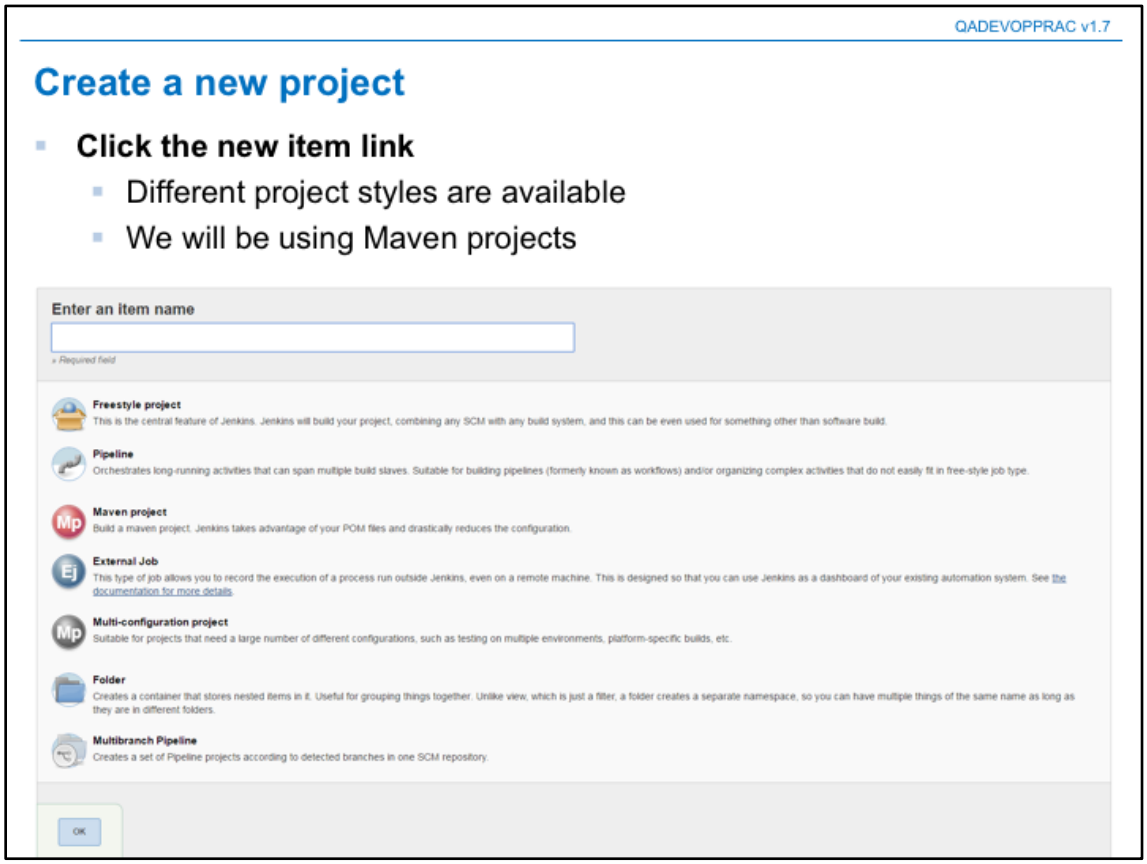
- Main control centre for Jenkins
- Configure projects from here

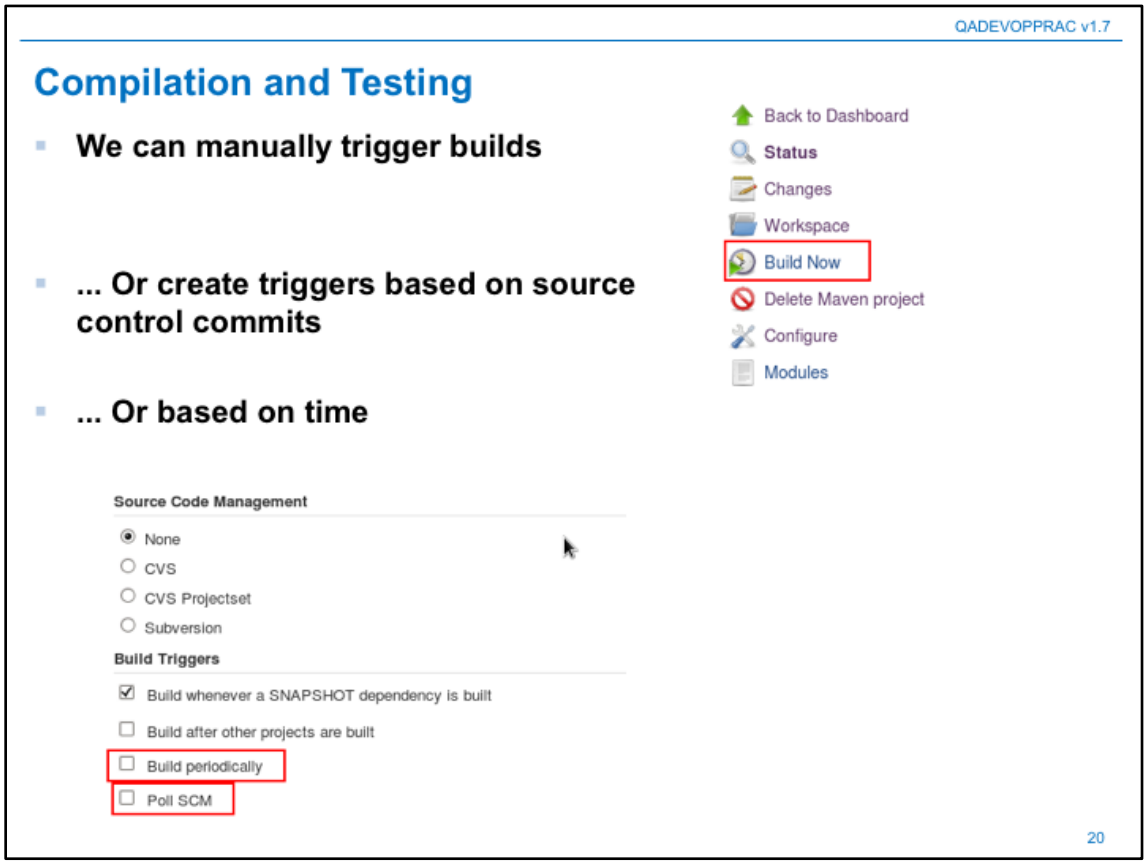
Managing Jenkins

- **The Manage Jenkins link on the home page contains all the admin side of Jenkins**



18


QADEVOPPRAC v1.7
Create a new project
Click the new item link
Different project styles are available
We will be using Maven projects
Enter an item name
» Required field
Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
Pipeline
Orchestrates long-running activities that can span multiple build slaves. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
External Job
This type of job allows you to record the execution of a process run outside Jenkins, even on a remote machine. This is designed so that you can use Jenkins as a dashboard of your existing automation system. See the documentation for more details.
Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.
OK


QADEVOPPRAC v1.7
Compilation and Testing
We can manually trigger builds
... Or create triggers based on source control commits
... Or based on time
Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Maven project
Configure
Modules
Source Code Management
None
CVS
CVS Projectset
Subversion
Build Triggers
Build whenever a SNAPSHOT dependency is built
Build after other projects are built
Build periodically
Poll SCM
20

To read more about Jenkins

Books:

- Learning Continuous Integration with Jenkins - Second Edition (2017. link: https://www.packtpub.com/virtualization-and-cloud/learning-continuous-integration-jenkins-second-edition)
- Jenkins 2.x Continuous Integration Cookbook (2017. link: https://www.safaribooksonline.com/library/view/jenkins-2x-continuous/9781788297943/)
- Continuous Integration, Delivery, and Deployment (2017. link: https://www.safaribooksonline.com/library/view/continuous-integration-delivery/9781787286610/)

Tutorials:

- https://www.tutorialspoint.com/jenkins/
- https://uk.ctl.io/knowledge-base/cloud-application-manager/tutorials/jenkins-ci-cd-tutorial/
- http://www.vogella.com/tutorials/Jenkins/article.html

Summary

- **CI/CD**
  - Principles
  - Practices of CI/CD
  - Tools
- **Introducing Jenkins**
  - What is Jenkins
  - Why is it popular
  - Distributed builds
- **Installing Jenkins on Centos**
  - The dashboard
  - Create a new project
  - Compile and Test