

Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
Maven
Testing (Foundation)
Java Intermediate
HTML
CSS
Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)
Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
Security
Cloud Fundamentals
AWS Foundations
AWS Intermediate
Linux
DevOps

# Jobs

## Contents

- [Overview](#)
- [Create a Job](#)
- [Workspaces](#)
- [Help!](#)
- [Job Configuration](#)
  - [General Settings](#)
- [Build History & Console Output](#)
- [Tutorial](#)
- [Exercises](#)

## Overview

A Jenkins project (job) is a repeatable build job, which contains steps and post-build actions.

A job can really do anything, it just depends what you configure it to do.

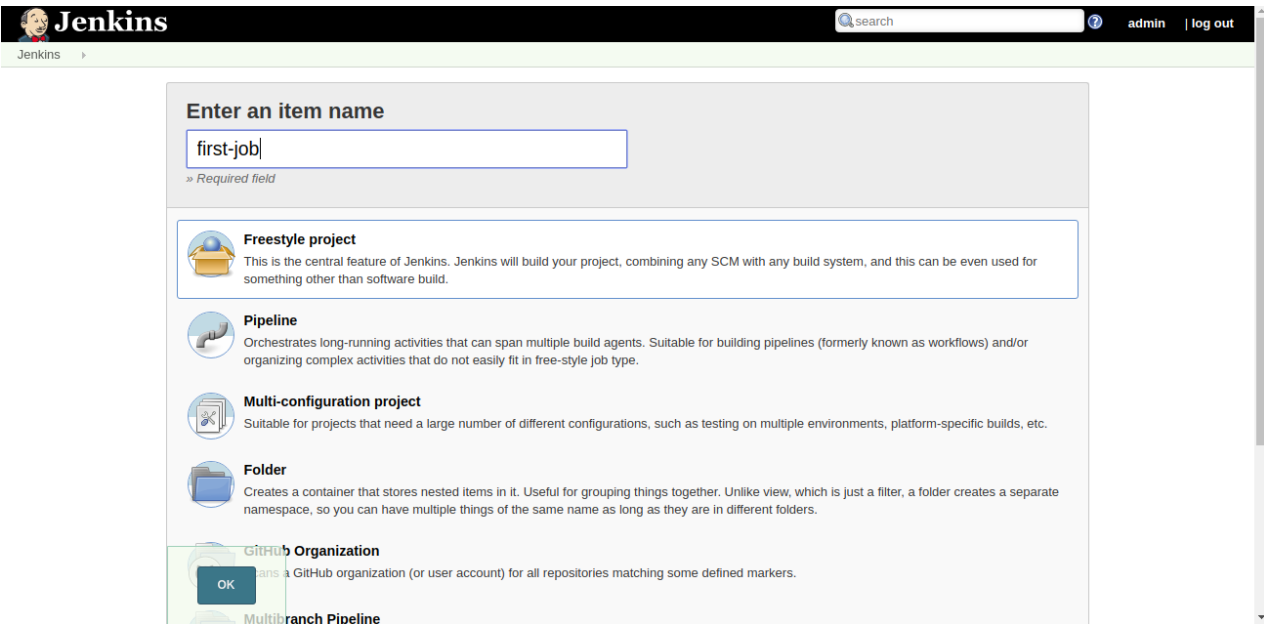
An example of what a job can be used to do is - automatically build a project and deploy it on a server, to be accessed over the internet.

## Create a Job

To create a new job, you can navigate to the **New Item** link on the Jenkins dashboard.

This will then present you with some options for what type of job to create.

Go ahead and name your job **first-job**, select **Freestyle Project** and then select **OK**.



## Workspaces

A workspace in Jenkins is basically just a folder that is on the host machine.

Jenkins will run every single job in its own workspace.

This is to keep jobs seperate from each other, which avoids conflicts with files and configurations that the jobs may be using.

## Help!

There are many configurations to choose from when setting up a Jenkins job; however, there is usually a **?** symbol on the right side of the page, which gives a comprehensive explanation of what the option does.

## Job Configuration

<div>Jenkins Introduction</div> <div><div><div></div></div><div>Installation Wizard</div></div> <div><div><div></div></div><div>Jobs</div></div> <div><div><div></div></div><div>Freestyle Project</div></div> <div><div><div></div></div><div>Builds</div></div> <div><div><div></div></div><div>Plugins</div></div>
---

This is where you can set up what your job is going to do. There are many options, but this module will just focus on the General Settings.

## General Settings

The settings and options throughout all job configurations will depend on what plugins are installed; this module aims to explain the ones that are suggested when you go through the Jenkins setup.

Setting Name	Description
Description	This is just an open text field to fill in information about the job. You'll likely want to put some instructions about your job here, or some information about its current state; for example, if it's a work in progress.
Discard Old Builds	Disk space doesn't grow on trees. If your job uses up quite a bit of space, then, after you have built it many times, considerable disk space might be used by old builds that no longer serve any purpose.
GitHub Project	Jenkins jobs are commonly associated with some kind of source code repository, like GitHub. You can add a link to that repository here, so it's easier to navigate to from the job's page.
Parameters	To make your job more generic and able to accept different configurations, you can pass it parameters.
Disable Project	Prevention is better than cure. If there are issues with the job's current configuration and you would like to make sure that it doesn't get executed, then you can check this option.
Concurrent Builds	Use this option carefully. There are many cases where your job will not be able to run at the same time as another instance of it, which is why this is disabled by default.

## Build History & Console Output

The build history is simply a list of all the individual builds that have run. These are especially useful for checking the *Console Output*, to acces this you can either:

- Click the blue or red 'bulb' on the job to go straight to the Console Output
- Click the number and then once you have been sent to the build page, select the *Console Output* link

## Tutorial

Have a go at entering some simple configurations into your first job:

- Choose any project from [GitHub](#)
- Enter some information about the project into the **Project url** field, for the **GitHub project** option
- Set a **String Parameter** for the job:
  - Set the **NAME** field to be **VERSION**
  - Set the **Default Value** field to be **0.1.0**
  - Set the **Description** field to be **Version of the application to build**
- Select the **Save** button (you will be redirected to the job's page)
- Select **Build with Parameters**. You should then see the parameter that you configured
- Select **Delete Project** and confirm the deletion

## Exercises

There are no exercises for this module.