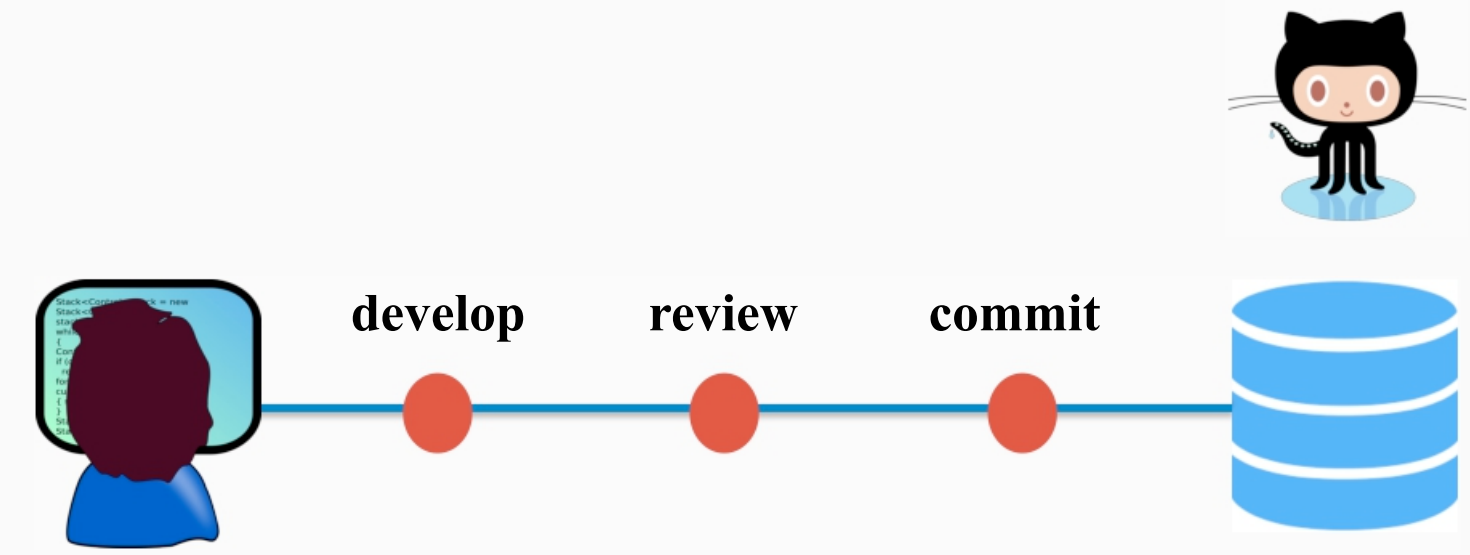
CYDEO

CI / CD

Contents


develop
review
commit
Developers
Version control

sonarqube

JUnit

build

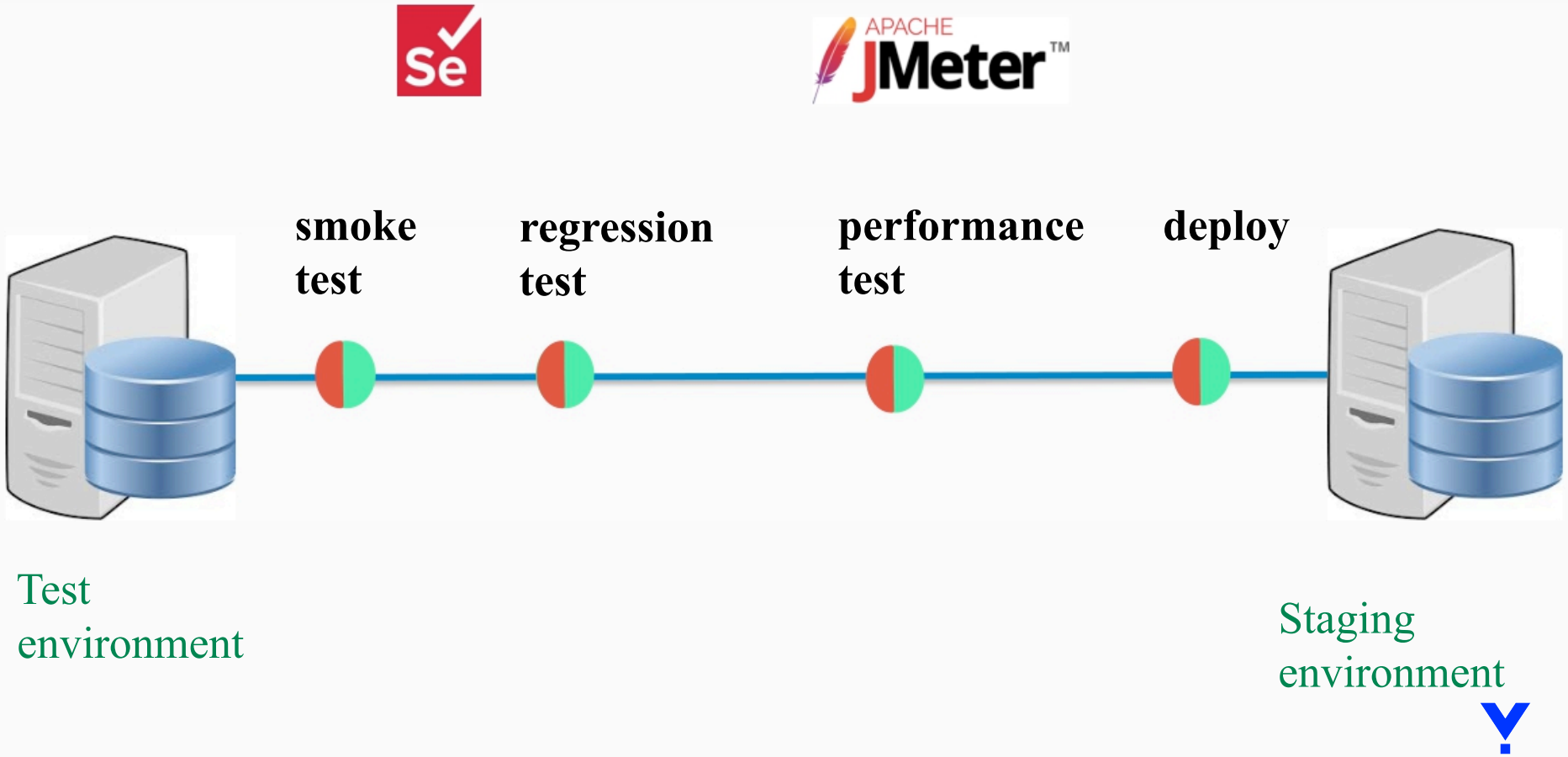
codeanalysis

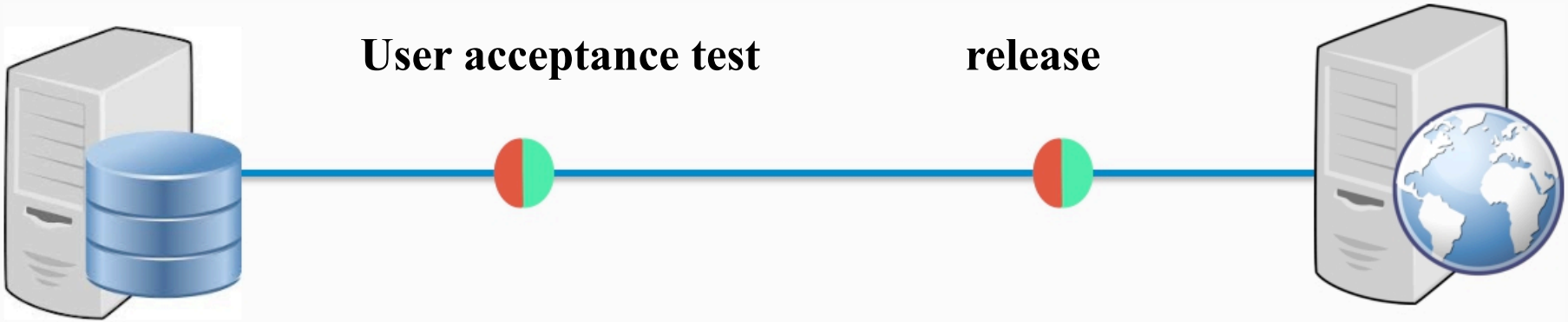
unit testing

integration test

VCS (github)

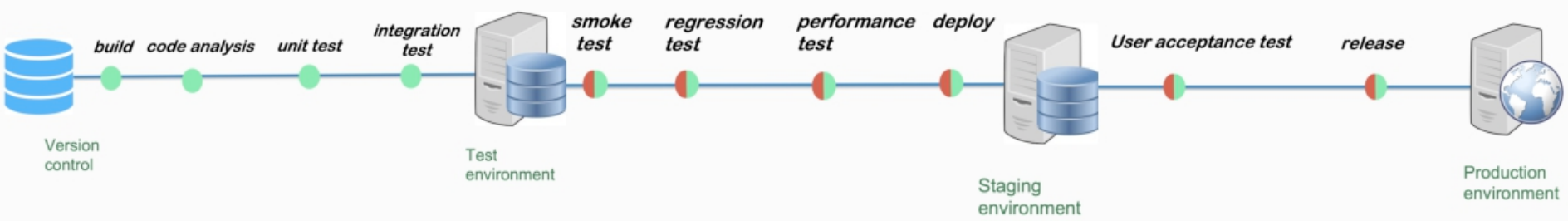
Test environment


smoke test
regression test
performance test
deploy
Test environment
Staging environment



Staging environment

Production environment



Production Pipeline

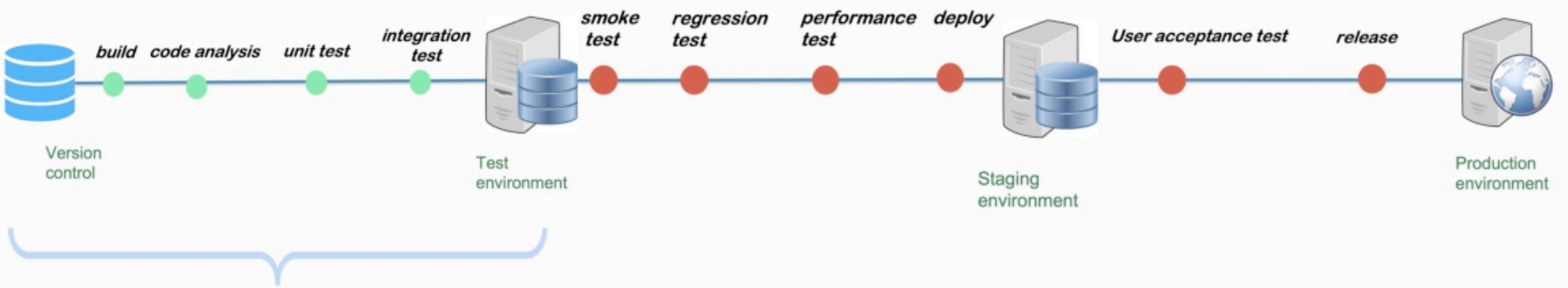
Pipeline

Pipeline

Pipeline is a set of processes that take the code from version control and compile, build, test and deploy to production in automated fashion.

The pipeline breaks down the software delivery process into stages. Each stage is made of different tasks which can be carried out in parallel. When all tasks in a stage passes, next stage is triggered.

Continuous Integration



Continuous Integration

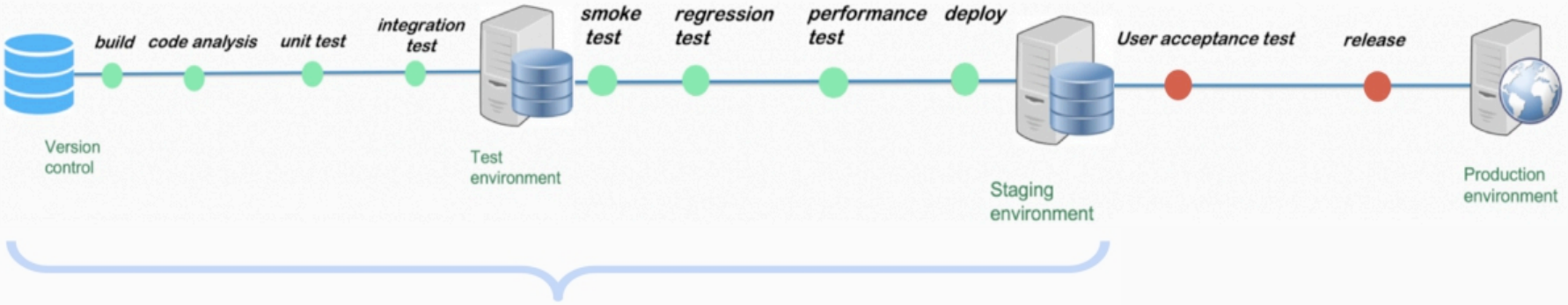
Continuous Integration

Continuous Integration is a process of automated build, unit testing, integration testing and code analysis.

The Continuous Integration process is comprised of automatic tools that assert the new code's correctness before integration. It reduces integration problems allowing to deliver software more rapidly by providing quick feedback every time new code is added to the source control. Usually Continuous Integration does not involve testing the functionality of the application.

In a simple words: the practice of merging all developers' working copies to a shared mainline several times a day.

Continuous Delivery



Continuous Delivery

Continuous Delivery is an automated build and execution of at unit and integration tests, execution of code analysis, functional tests and also deploying to any supported platform any time. Each time a build or a set of code passes the tests, it's automatically deployed out to a staging environment.

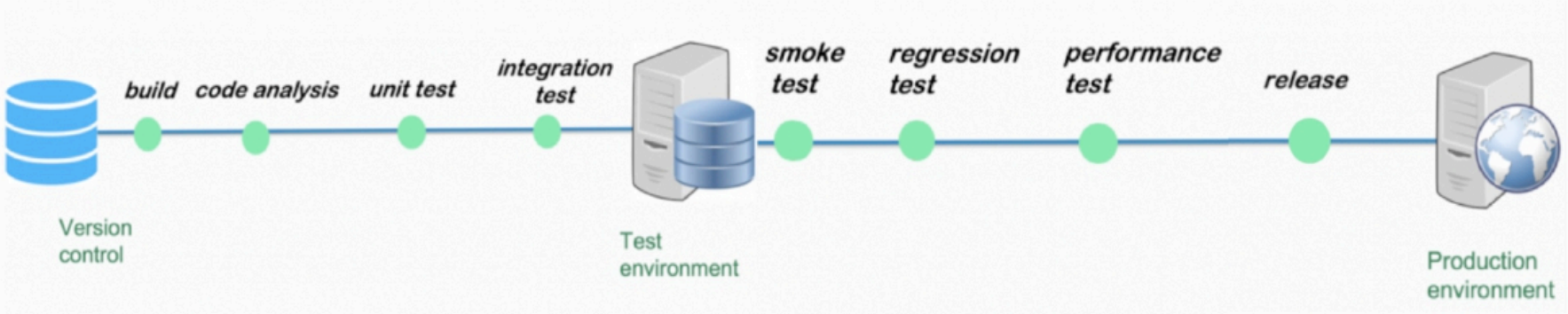
In Continuous Delivery releasing to end users is a manual process. Continuous delivery involves human decision- making when it comes to deciding when to release the software to the customers.

Continuous Delivery

In a simple words: the practice in which teams produce and release value in short cycles.
8 Principles of Continuous Delivery:

1. Repeatable Reliable Process
2. Automate Everything
3. Version Control Everything
4. Bring the Pain Forward
5. Build-in Quality
6. "Done" Means Released
7. Everyone is Responsible
8. Continuous Improvement

Continuous Deployment



Continuous Deployment

Continuous Deployment

Continuous deployment means that every change that you make, goes through the pipeline, and if it passes all the tests, it automatically gets deployed into production.

When a developer checks in code, the automated processes take the code and move it through the entire lifecycle and if it passes each gate, it gets deployed directly to production. The delivery speeds are notably faster due to elimination of manual steps.

In a simple words: the practice in which the value is delivered frequently through automated deployments.

DevOps

Methodology (like agile, waterfall)

DevOps merges developer, tester and operations roles together. whoever writes the software is also responsible and deploying and maintaining it.

DevOps organizations break down the barriers between Operations and Engineering by cross - training each team in the other's skills.

DevOps focuses on culture highlighting roles that emphasize responsiveness and breaking down barriers between developers and operations teams.

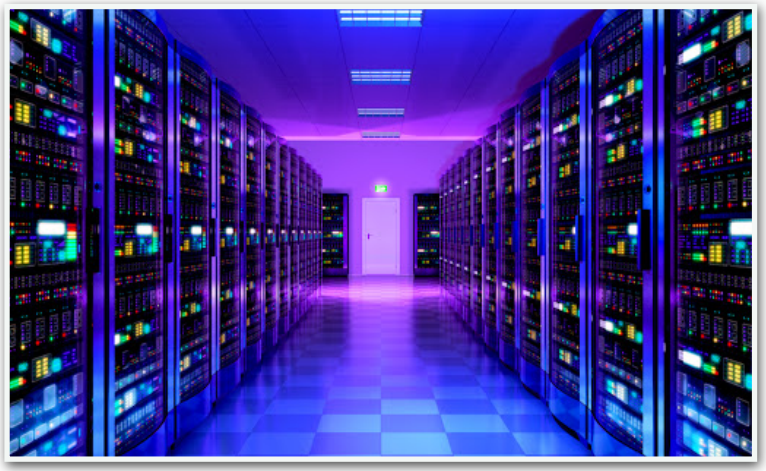
What is Server ?

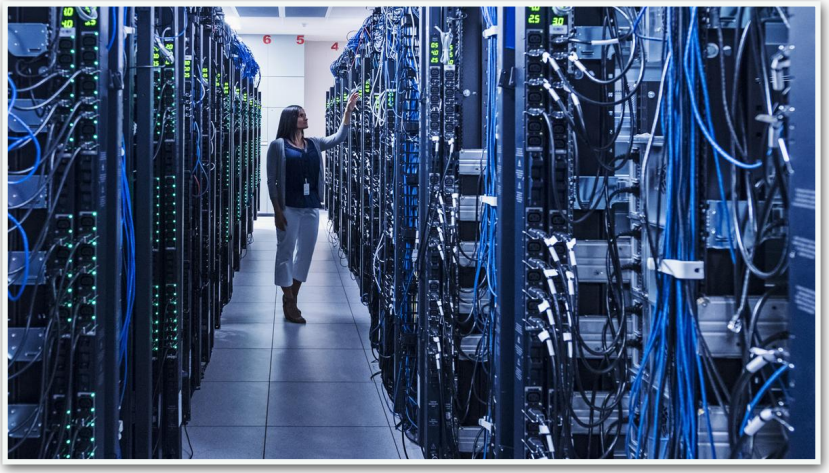
What is a server?

A server is a computer program or device that provides a service to another computer program and its user, also known as the client. In a data center, the physical computer that a server program runs on is also frequently referred to as a server. That machine might be a dedicated server or it might be used for other purposes.

In the client/server programming model, a server program awaits and fulfills requests from client programs, which might be running in the same or other computers. A given application in a computer might function as a client with requests for services from other programs and as a server of requests from other programs.

What is Server ?





Physical and Virtual Servers

1. A physical server is simply a computer that is used to run server software.

2. A virtual server is a virtual representation of a physical server. Like a physical server, a virtual server includes its own operating system and applications. These are kept separate from any other virtual servers that might be running on the physical server.

The process of creating virtual machines involves installing a lightweight software component called a hypervisor onto a physical server. The hypervisor's job is to enable the physical server to function as a virtualization host. The virtualization host makes the physical server's hardware resources - such as CPU time, memory, storage and network bandwidth - available to one or more virtual machines.

Servers in the Cloud

Servers in the cloud have revolutionized the IT industry.

- Scale capacity up and down based on demands.
- Storage, more memory, and computing power can be added as needed.
- Obtain servers in minutes.
- No need for onsite hardware or capital expenses.

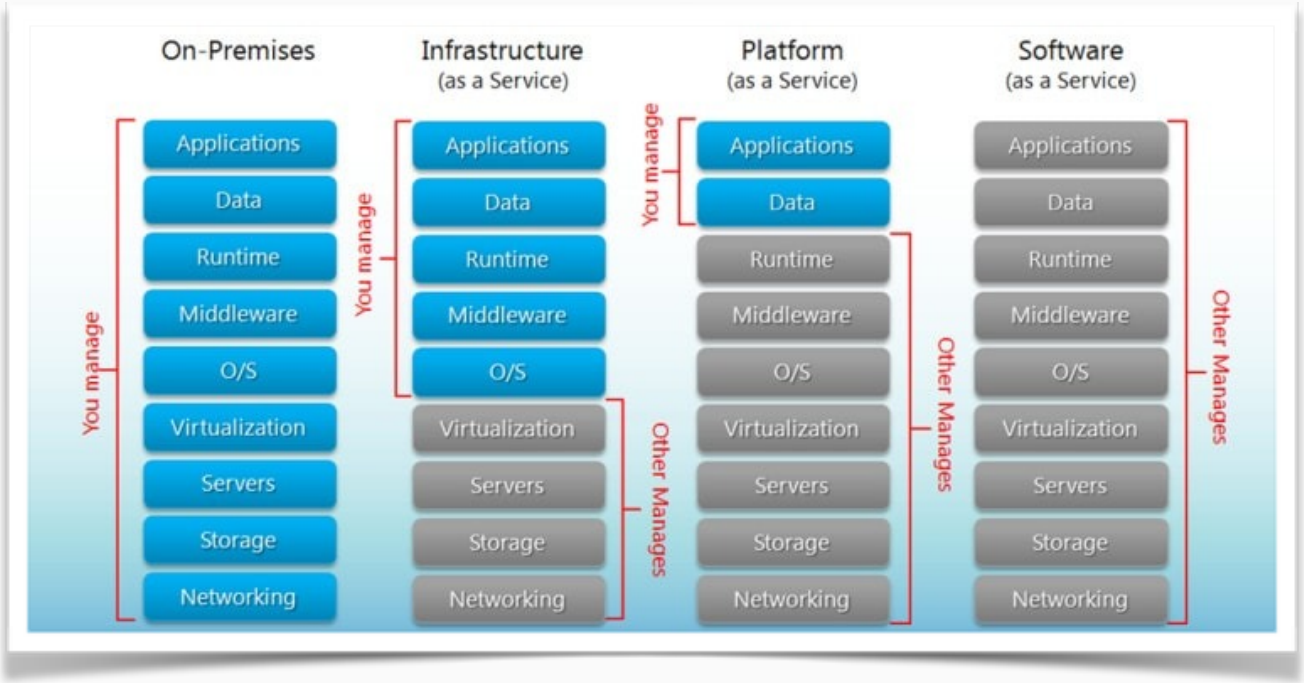
Main Types of Cloud Computing

SaaS, PaaS, and IaaS are simply three ways to describe how you can use the cloud.

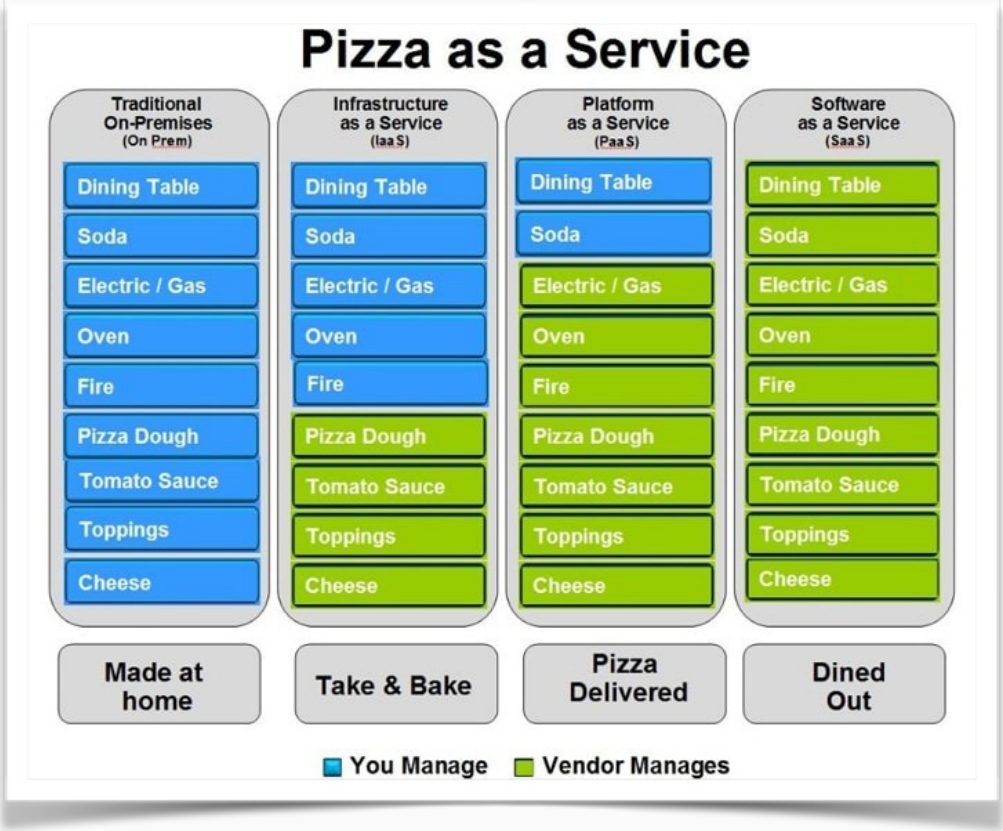
- IaaS: cloud-based services, pay-as-you-go for services such as storage, networking, and virtualization.
- PaaS: hardware and software tools available over the internet.
- SaaS: software that's available via a third-party over the internet.
- On-premise: software that's installed in the same building as your business.

*IaaS - Infrastructure As A Service
*PaaS - Platform As A Service
*SaaS - Software As A Service

Main Types of Cloud Computing



Main Types of Cloud Computing



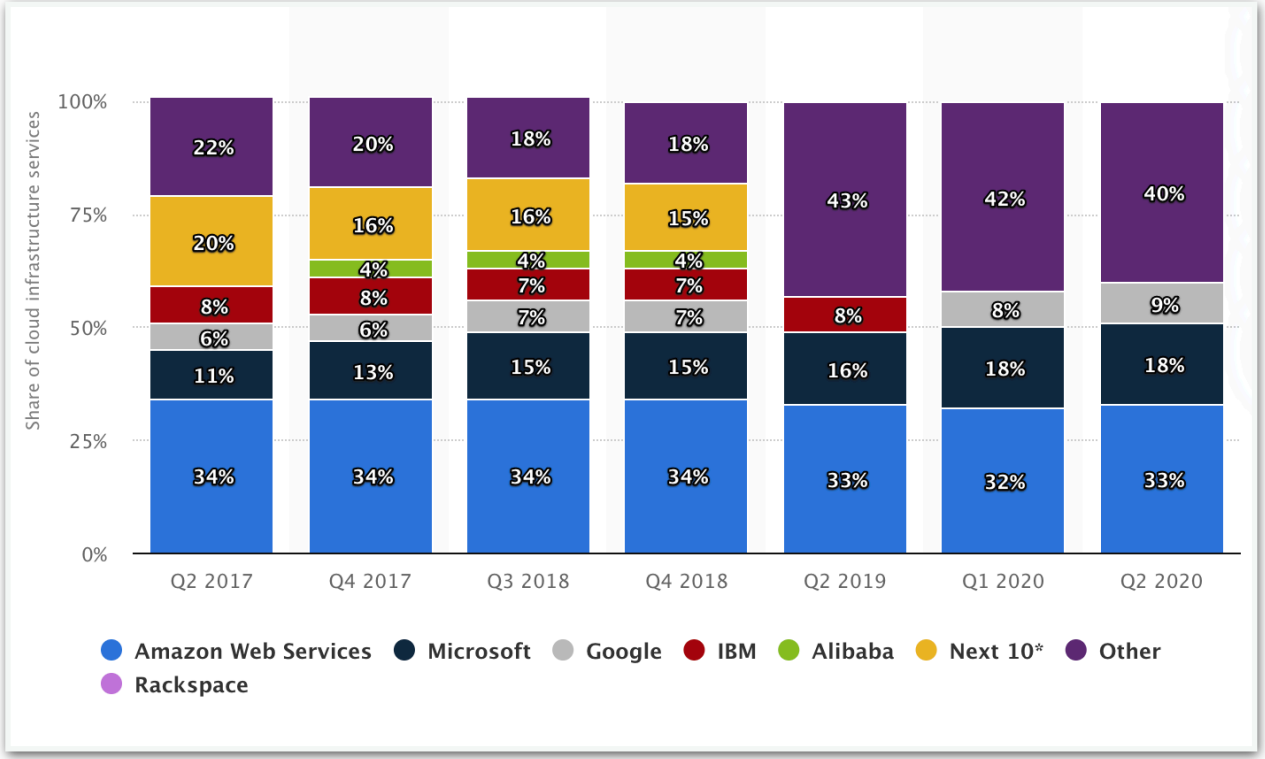
Main Types of Cloud Computing

SaaS: BigCommerce, Google Apps, Salesforce, Dropbox, MailChimp, ZenDesk, DocuSign, Slack, Hubspot.

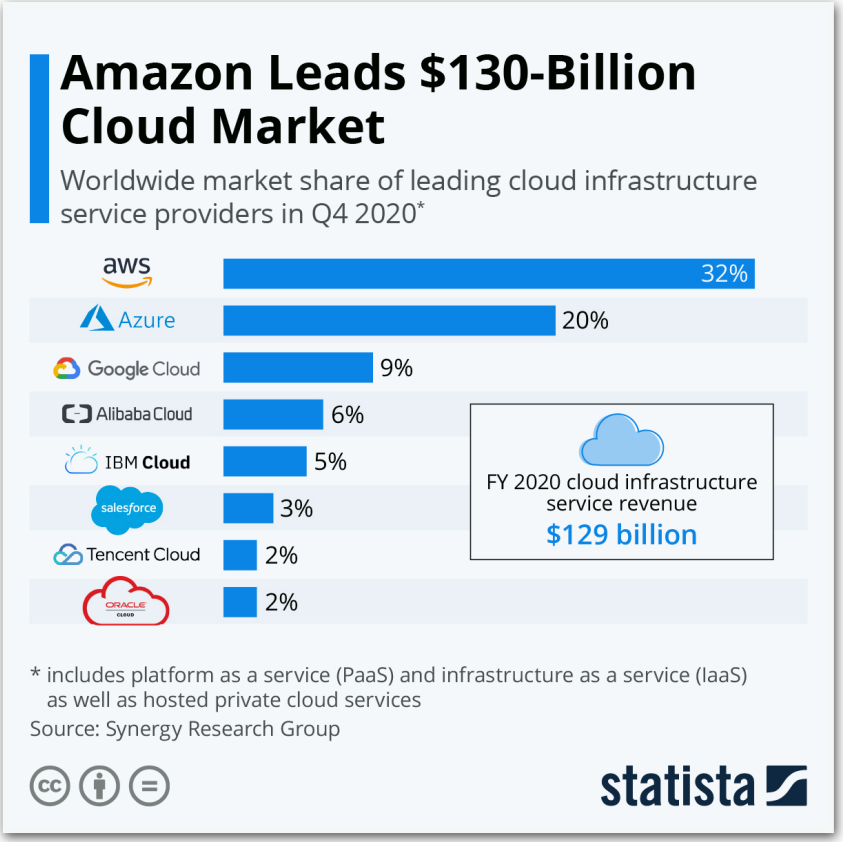
PaaS: AWS Elastic Beanstalk, Heroku, Windows Azure (mostly used as PaaS), Force.com, OpenShift, Apache Stratos, Magento Commerce Cloud.

IaaS examples: AWS EC2, Rackspace, Google Compute Engine (GCE), Digital Ocean, Magento 1 Enterprise Edition*.

Cloud Computing Market



Cloud Computing Market



Cloud Computing with AWS

Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud platform, offering over 200 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies—are using AWS to lower costs, become more agile, and innovate faster.

Cloud computing is the on-demand delivery of compute power, database storage, applications and other IT resources through a cloud services platform via the Internet with pay-as-you go pricing

AWS is made up of so many different cloud computing products and services. The highly profitable Amazon division provides servers, storage, networking, remote computing, email, mobile development, and security. AWS can be broken into three main products: EC2, Amazon's virtual machine service, Glacier, a low-cost cloud storage service, and S3, Amazon's storage system. AWS is so large and present in the computing world that it's far outpaced its competitors.

Cloud Computing with AWS

AWS has 81 availability zones in which its servers are located. These serviced regions are divided in order to allow users to set geographical limits on their services (if they so choose), but also to provide security by diversifying the physical locations in which data is held. Overall, AWS spans 245 countries and territories.

AWS EC2

Elastic Cloud Compute or EC2 is a foundational piece of AWS' cloud computing platform and is a service that provides servers for rent in the cloud.

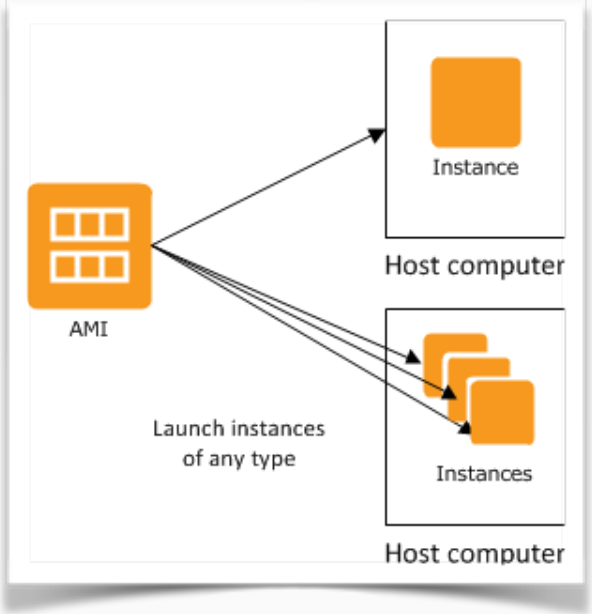
Pricing Options
There are several pricing options for EC2.

- On Demand - Pay as you go, no contract.
- Dedicated Hosts - You have your own dedicated hardware and don't share it with others.
- Spot - You place a bid on an instance price. If there is extra capacity that falls below your bid, an EC2 instance is provisioned. If the price goes above your bid while the instance is running, the instance is terminated.
- Reserved Instances - You earn huge discounts if you pay up front and sign a 1-year or 3-year contract.

AMI

An *Amazon Machine Image (AMI)* is a template that contains a software configuration (for example, an operating system, an application server, and applications).

From an AMI, you launch an *instance*, which is a copy of the AMI running as a virtual server in the cloud. You can launch multiple instances of an AMI, as shown in the following figure.



JENKINS

1. CI/CD tool
2. Open source
3. Automates building, testing, packaging, staging, deploying the application
4. Integrates with different tools using plugins

Alternatives: Travis, TeamCity, Bamboo , CircleCI, GitHub Actions

Jenkins Job

In Jenkins everything is done by creating a job

1. a task that Jenkins performs based its schedule
2. be made of several steps
3. can have a schedule or a trigger which determines when it runs
4. reports the results of the run automatically

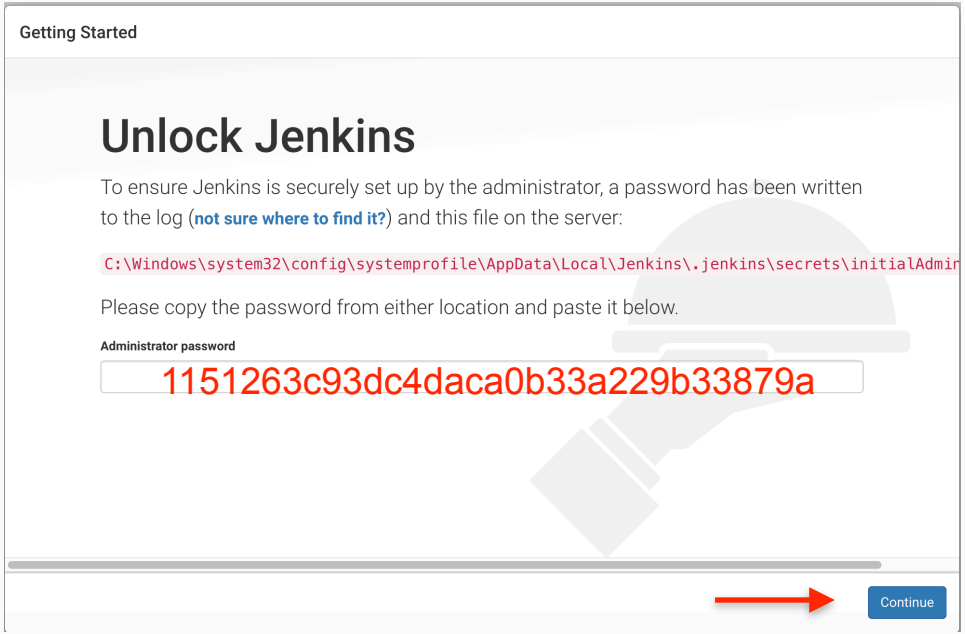
CYDEO

Jenkins Setup

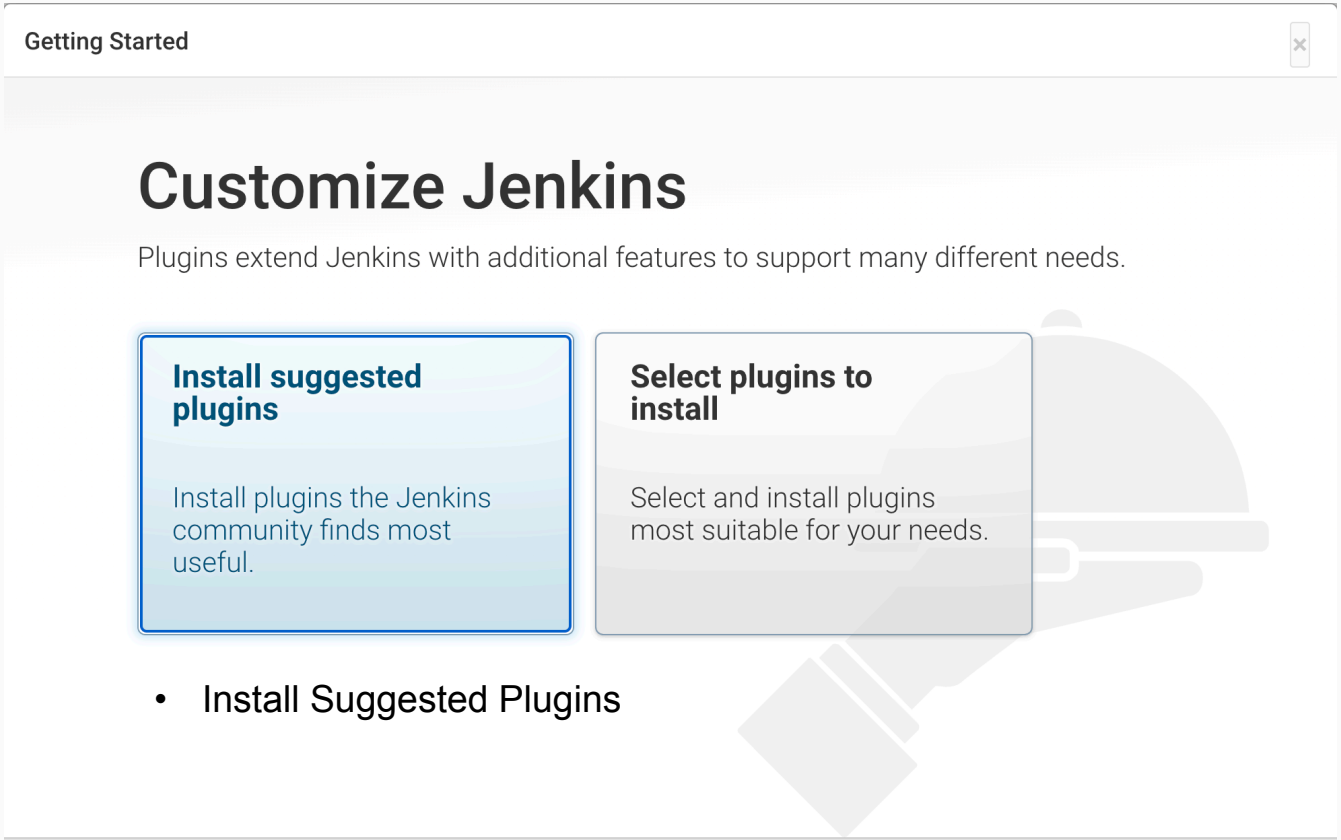
JENKINS

1. Open any browser
2. Type here YOUR IP from EC2 instance → **Example** : 54.165.159.235:8081
3. Use this password to unlock Jenkins
4. Click Continue

This password same for ALL USERS



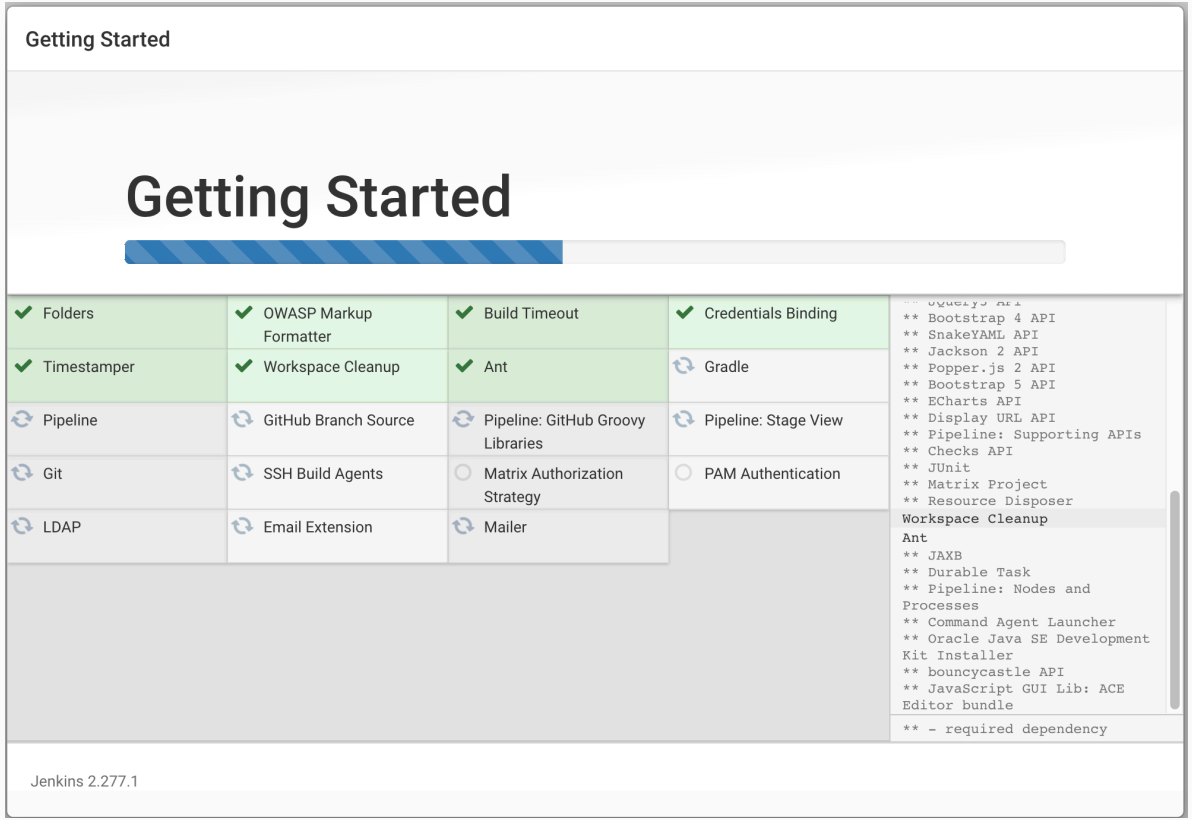
JENKINS



- Install Suggested Plugins

JENKINS

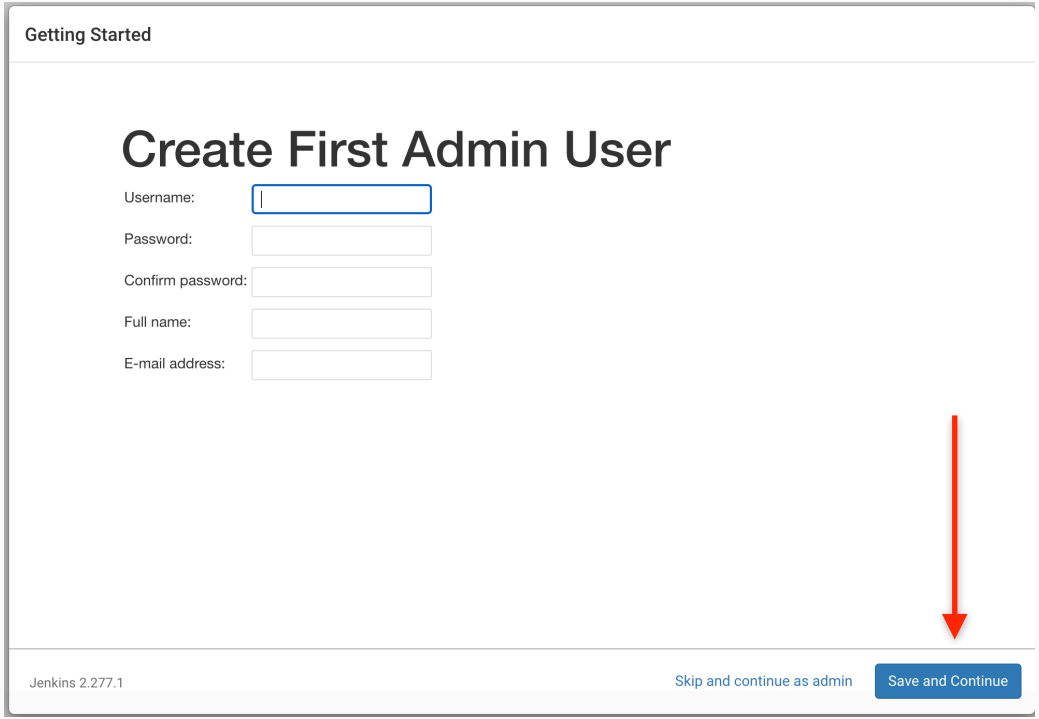
- Wait for all installations



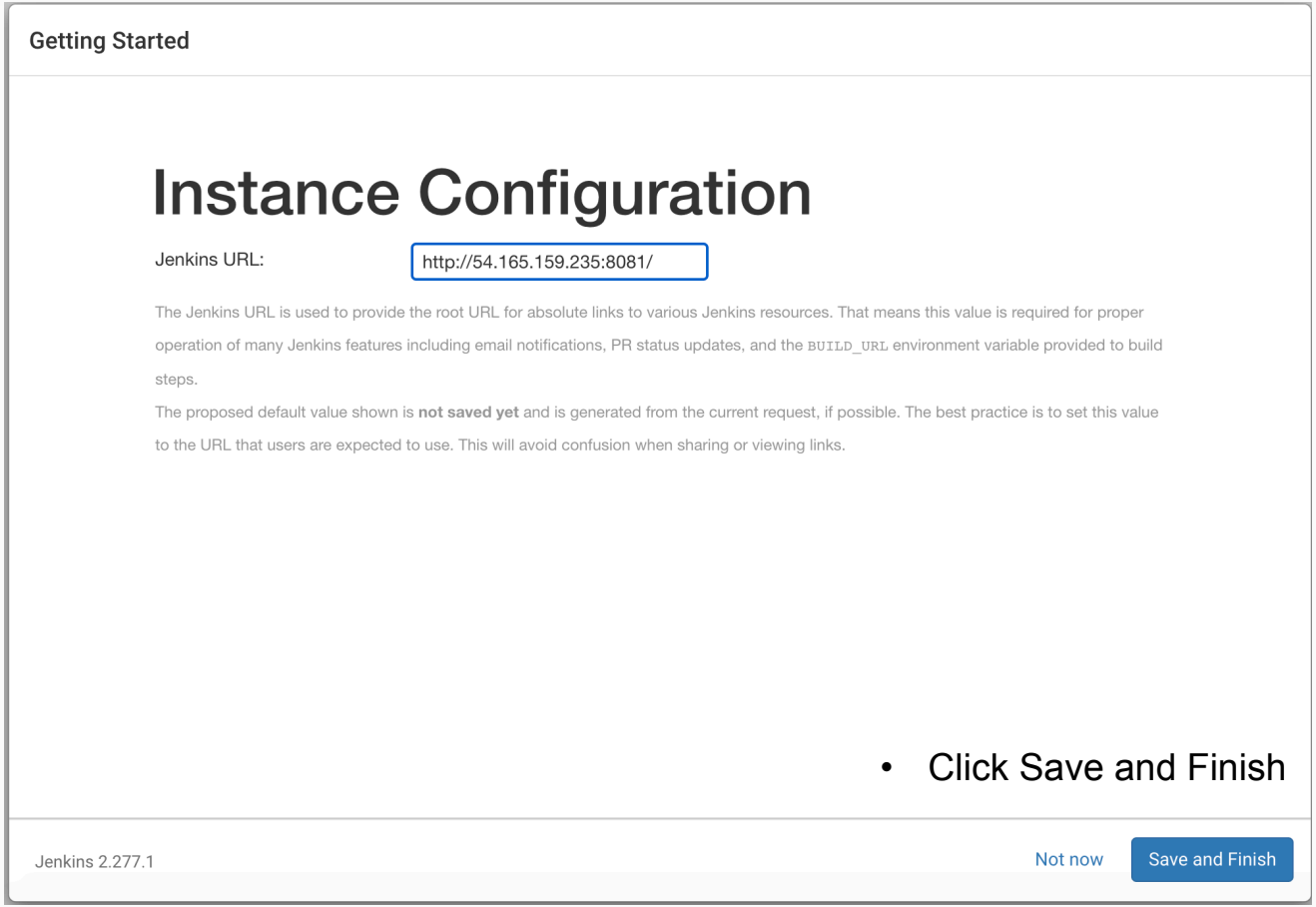
JENKINS

1. Create your user in JENKINS
2. These will be your information
3. After fill the form click
   Save and Continue

☑ DON'T FORGET PASSWORD

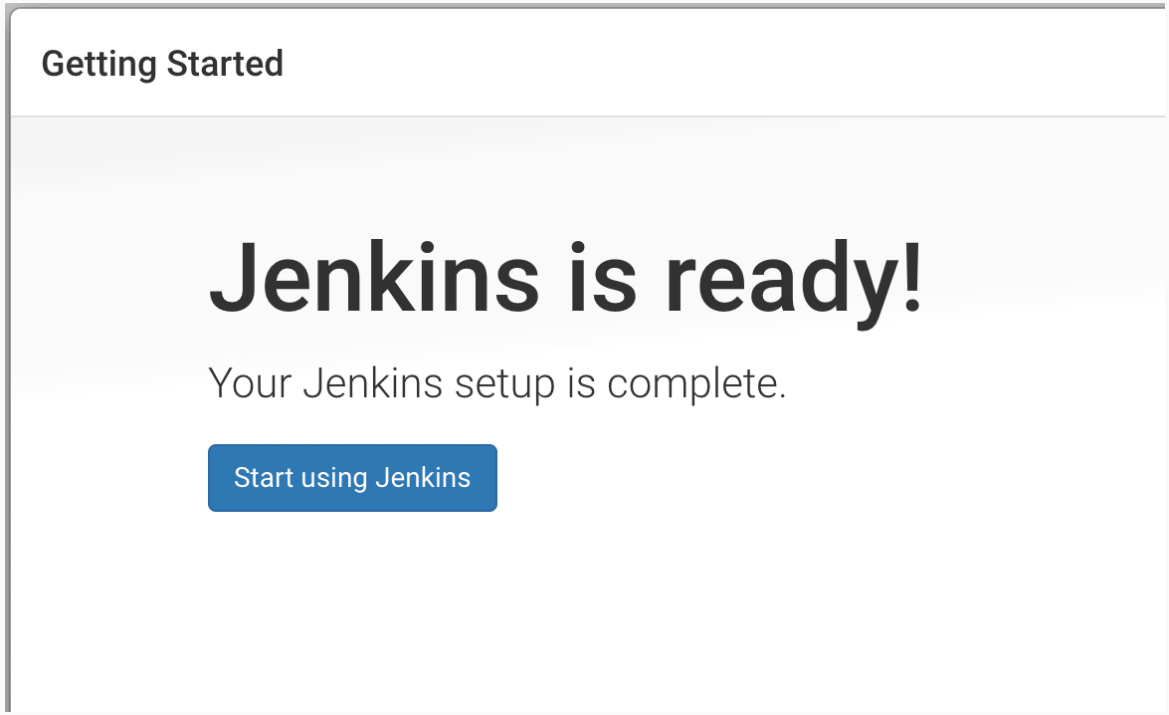


JENKINS



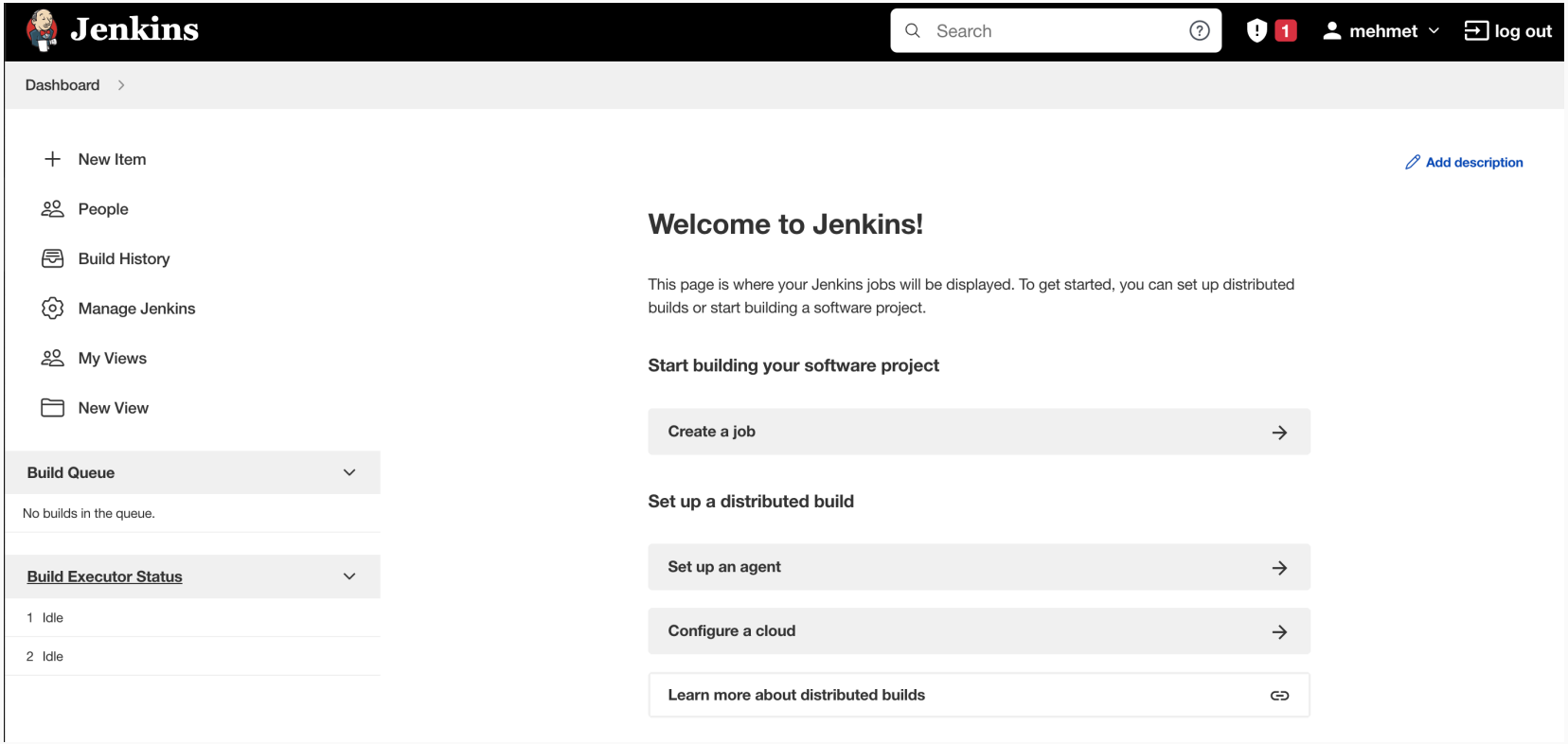
- Click Save and Finish

JENKINS



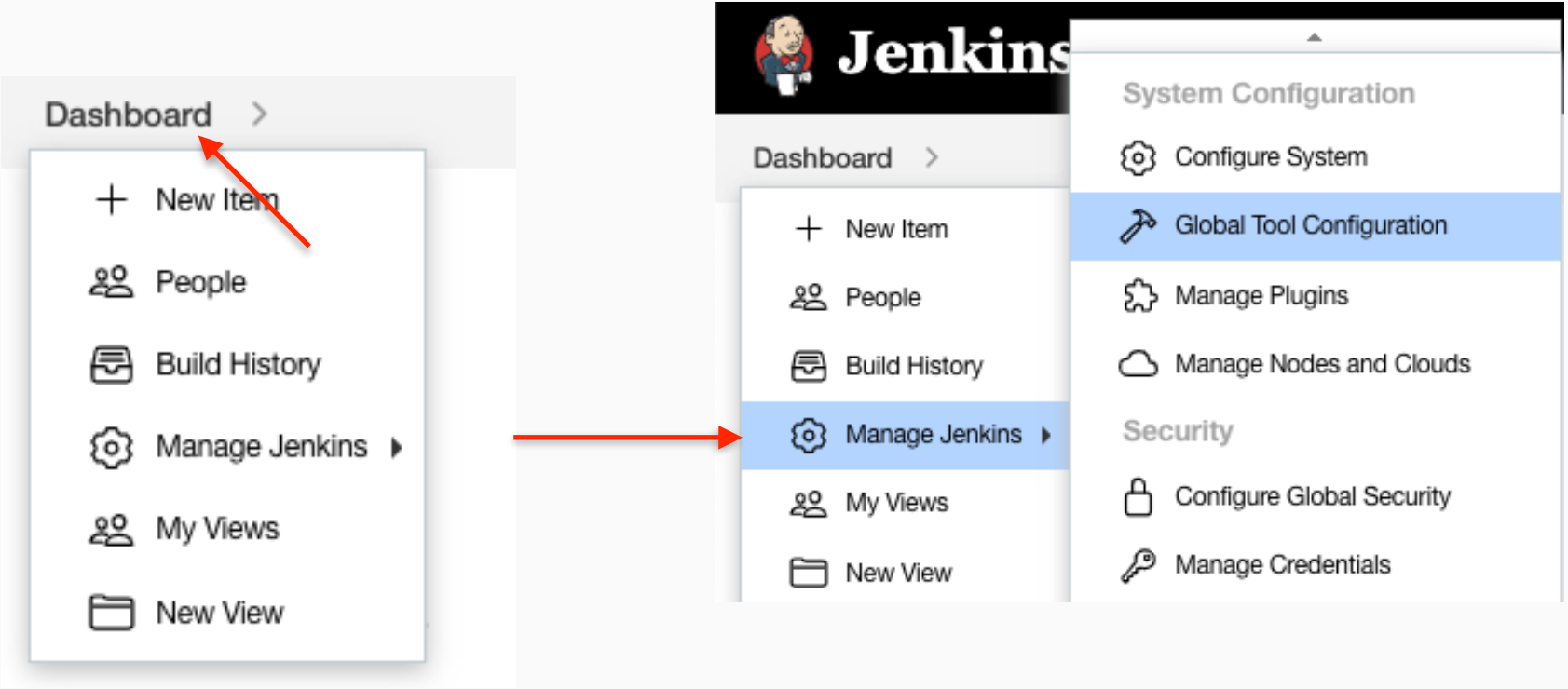
- Click Start Using JENKINS

JENKINS


Jenkins
Search
1
mehmet
log out
Dashboard
New Item
People
Build History
Manage Jenkins
My Views
New View
Build Queue
No builds in the queue.
Build Executor Status
1 Idle
2 Idle
Add description
Welcome to Jenkins!
This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.
Start building your software project
Create a job
Set up a distributed build
Set up an agent
Configure a cloud
Learn more about distributed builds

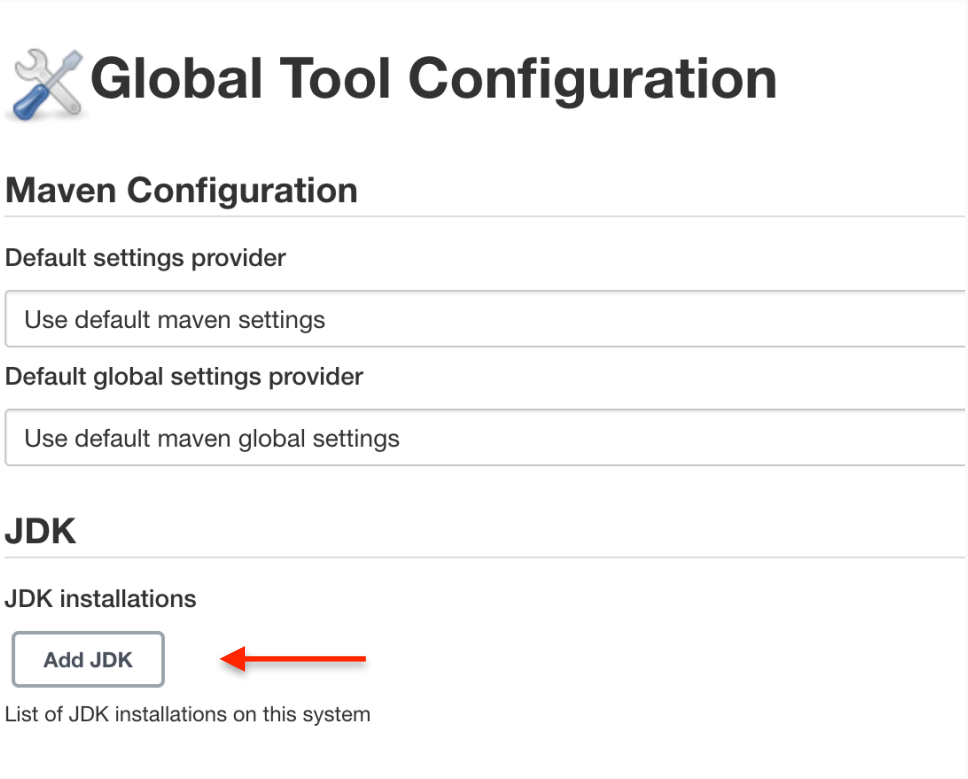
JENKINS

- Dashboard —> Manage JENKINS — > Global Tool Configuration



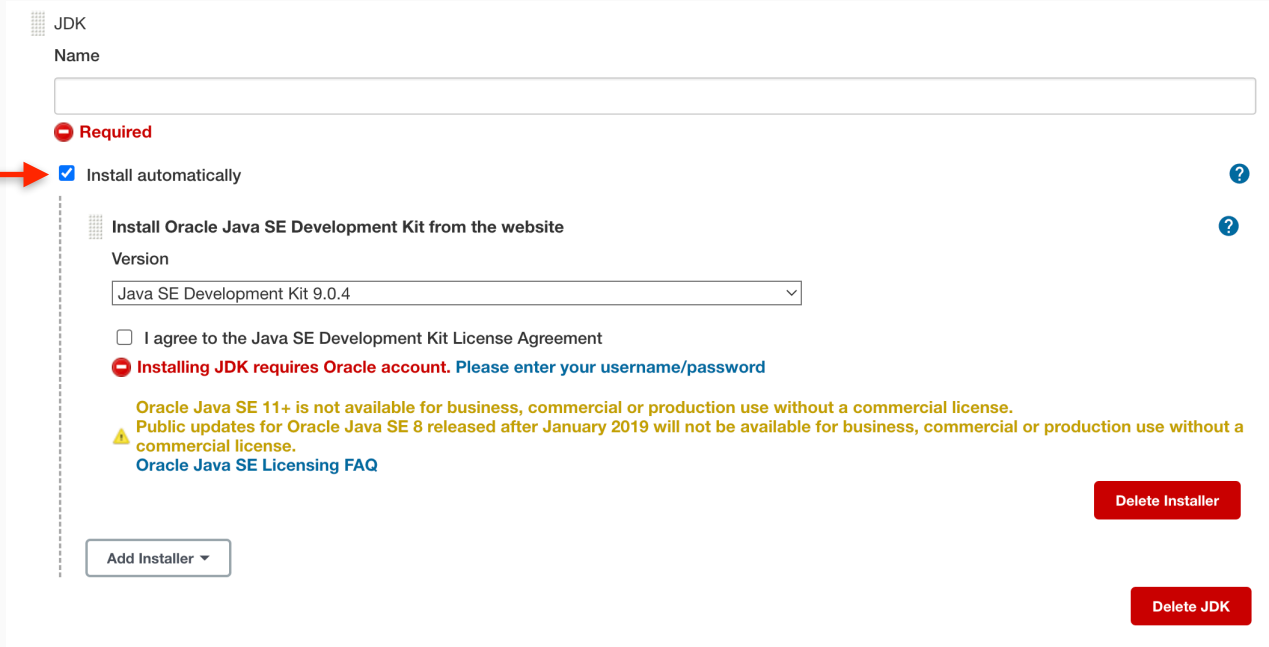
JENKINS

- Click Add JDK



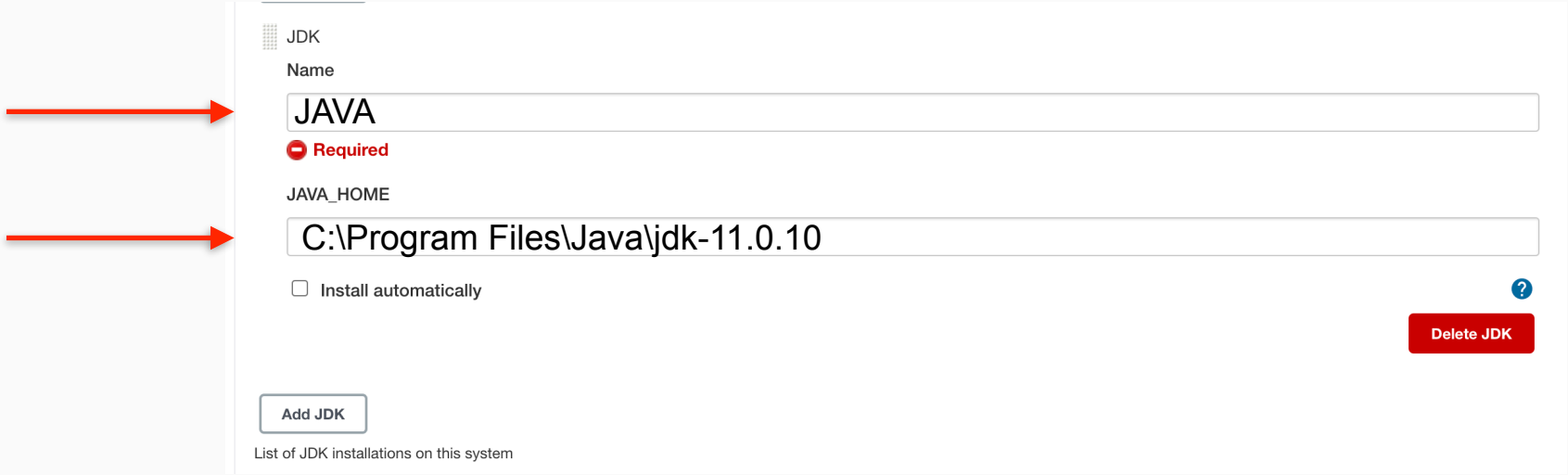
JENKINS

- Uncheck it



JENKINS

- Put this information in required fields



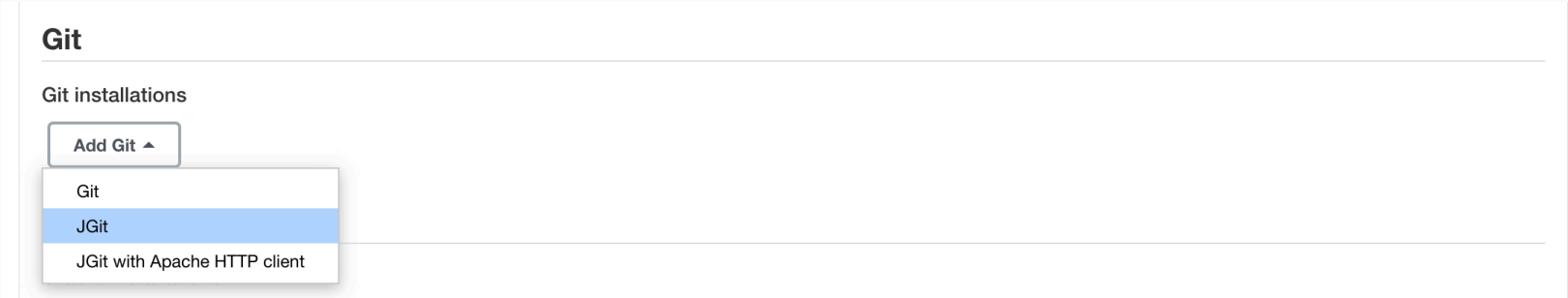
- Make sure there is no extra spaces

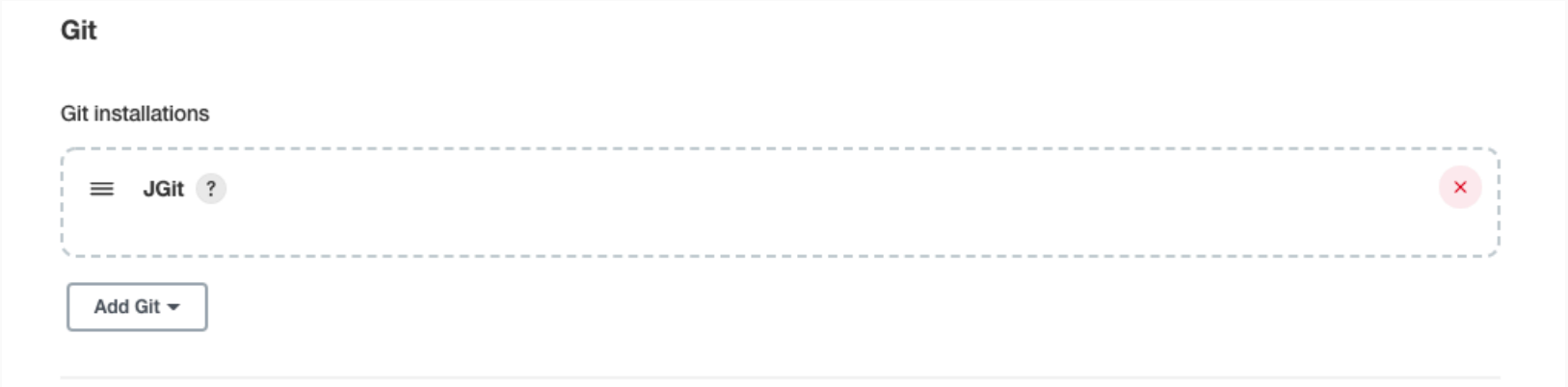
JENKINS

- Click here to remove

JENKINS

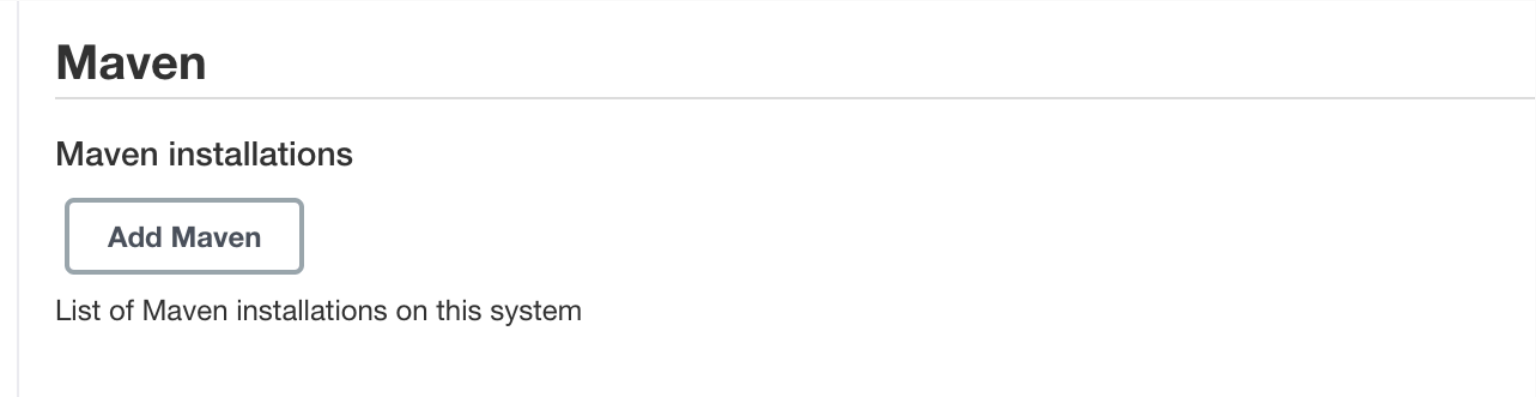
- Select JGit from list





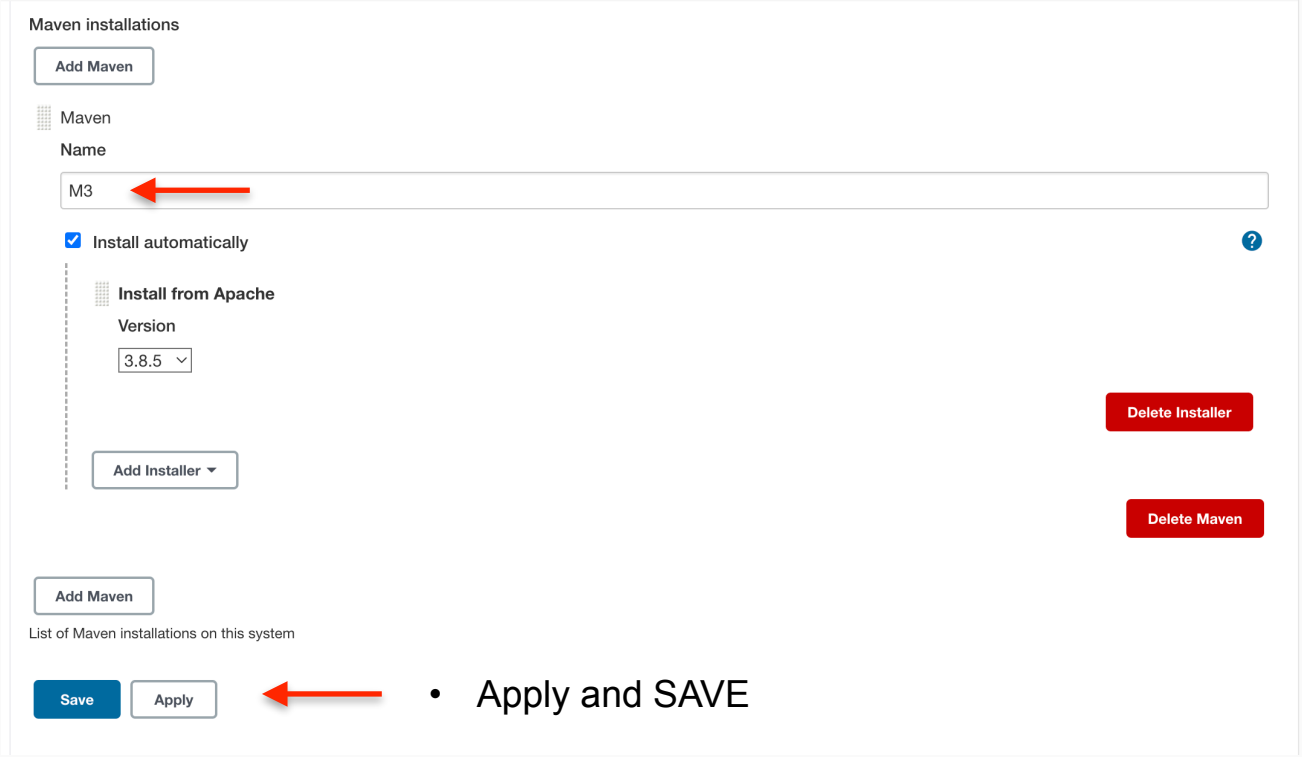
JENKINS

- Click Add Maven



JENKINS

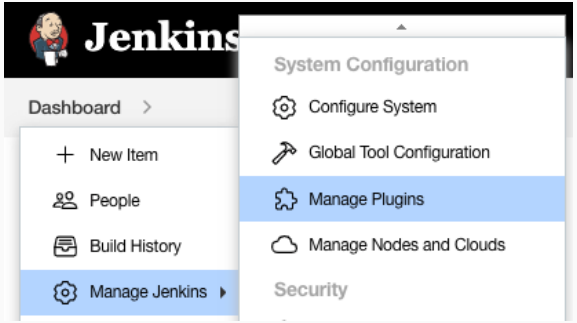
- Type here M3

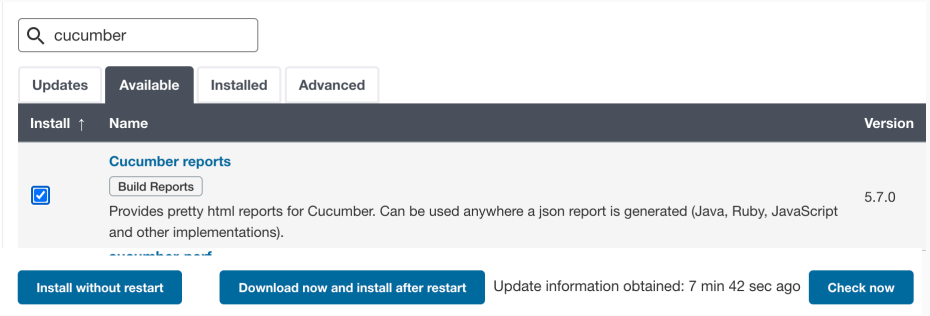


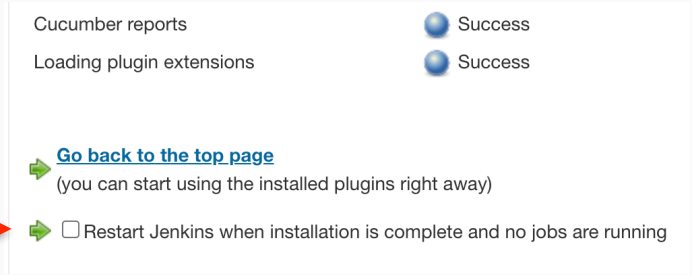
- Apply and SAVE

How to Install Cucumber Plugin to Jenkins

- Go to Jenkins Plugin Management to install
  - Select Available
  - Search for **cucumber**
  - Select **Cucumber Reports**
  - **Install without Restart**







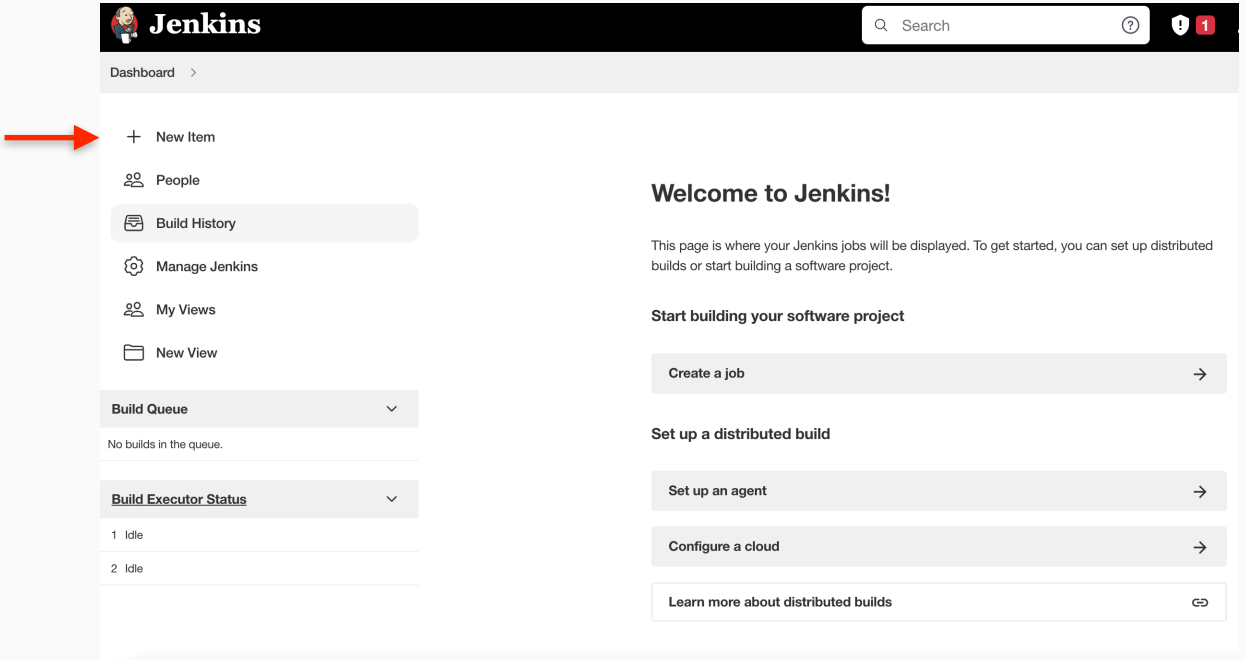
Restart Jenkins

CYDEO

My First Job

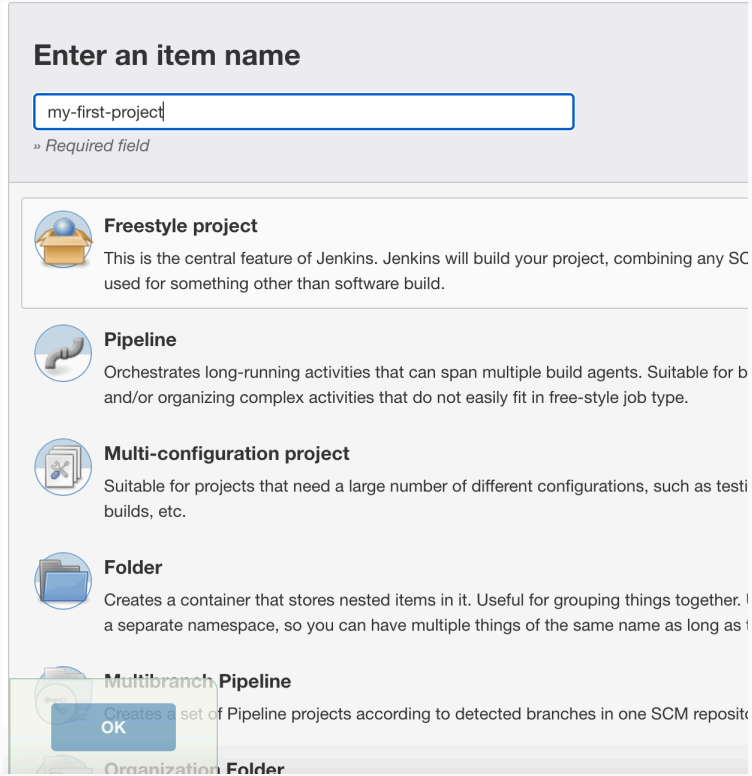
JENKINS

- Click on Dashboard
- Click New Item



JENKINS

- Give a name
- Choose Freestyle Project
- Click OK



Source Code Management

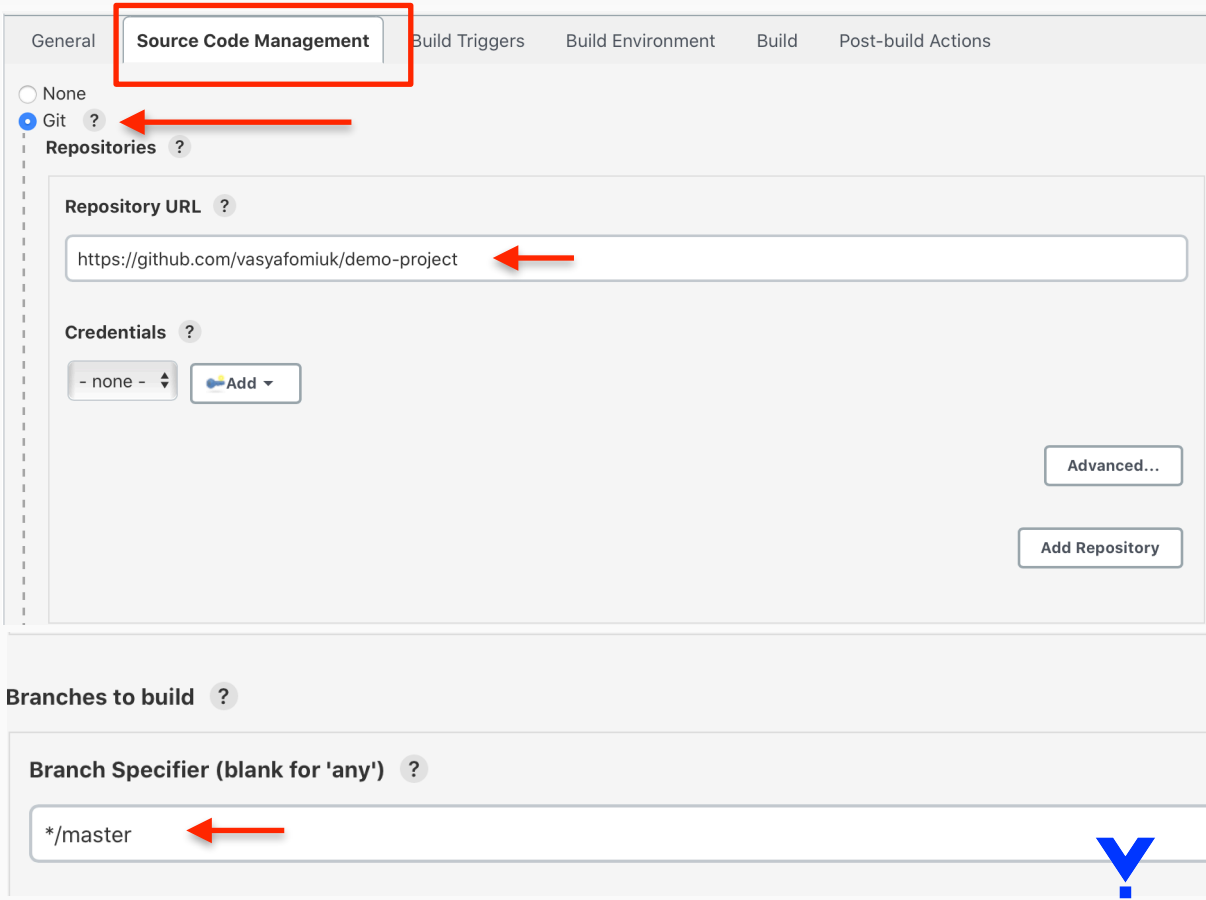
1. Select Git
2. Enter your repository url information
3. Select the branch to build

NOTE :

For private repositories you should put Credentials

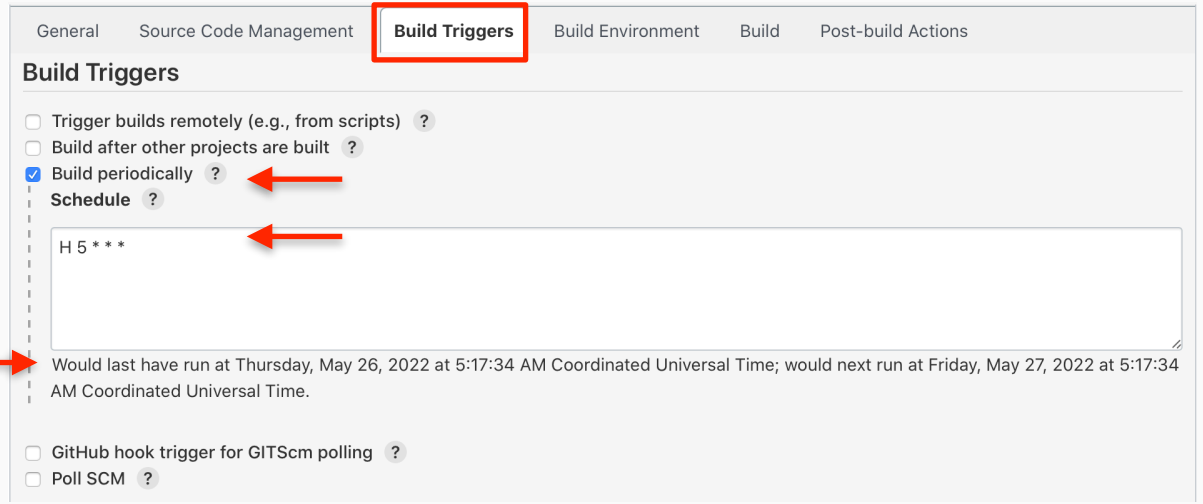
REPOSITORY URL :

Check CI/CD Notes Module **demo-project**

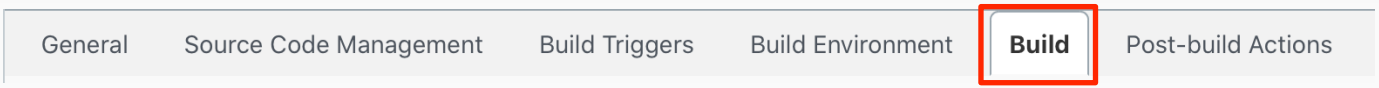


Build Triggers

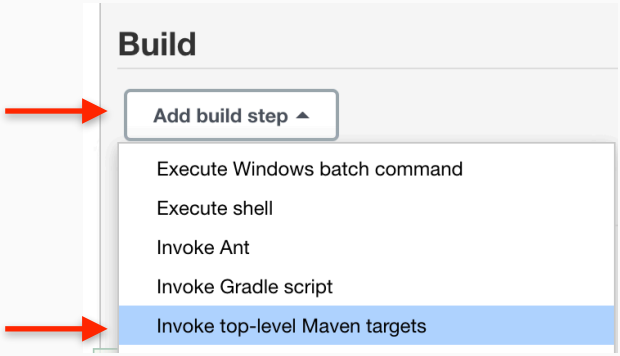
1. Click Build Triggers
2. Select Build Periodically
3. Enter schedule as cron expression

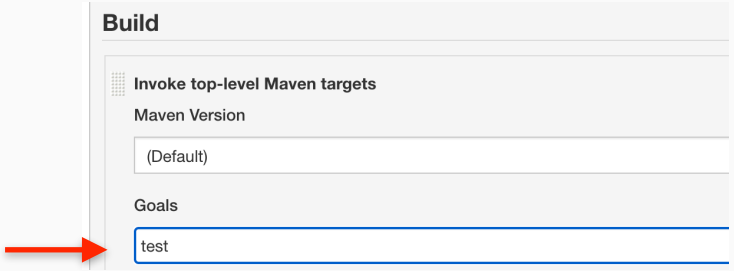


Build

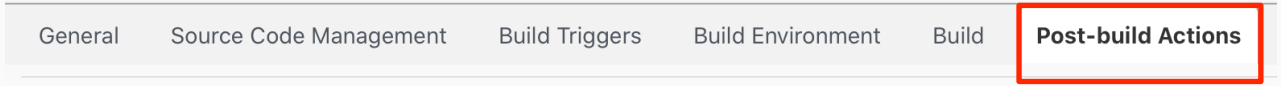


1. Click on Build
2. Click on ***Add build step***
3. Select ***Invoke top-level Maven targets***
4. Enter maven goal that executes your tests

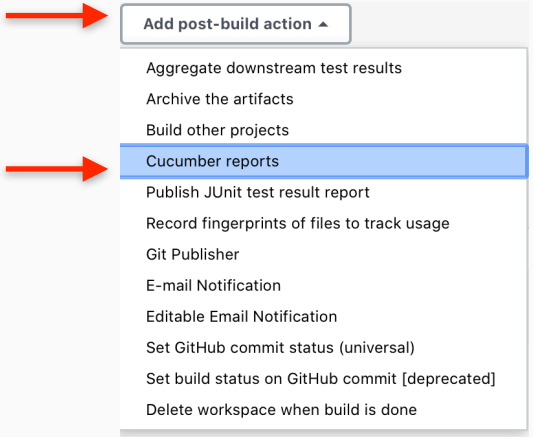


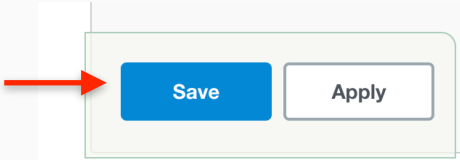


Post-Build Actions

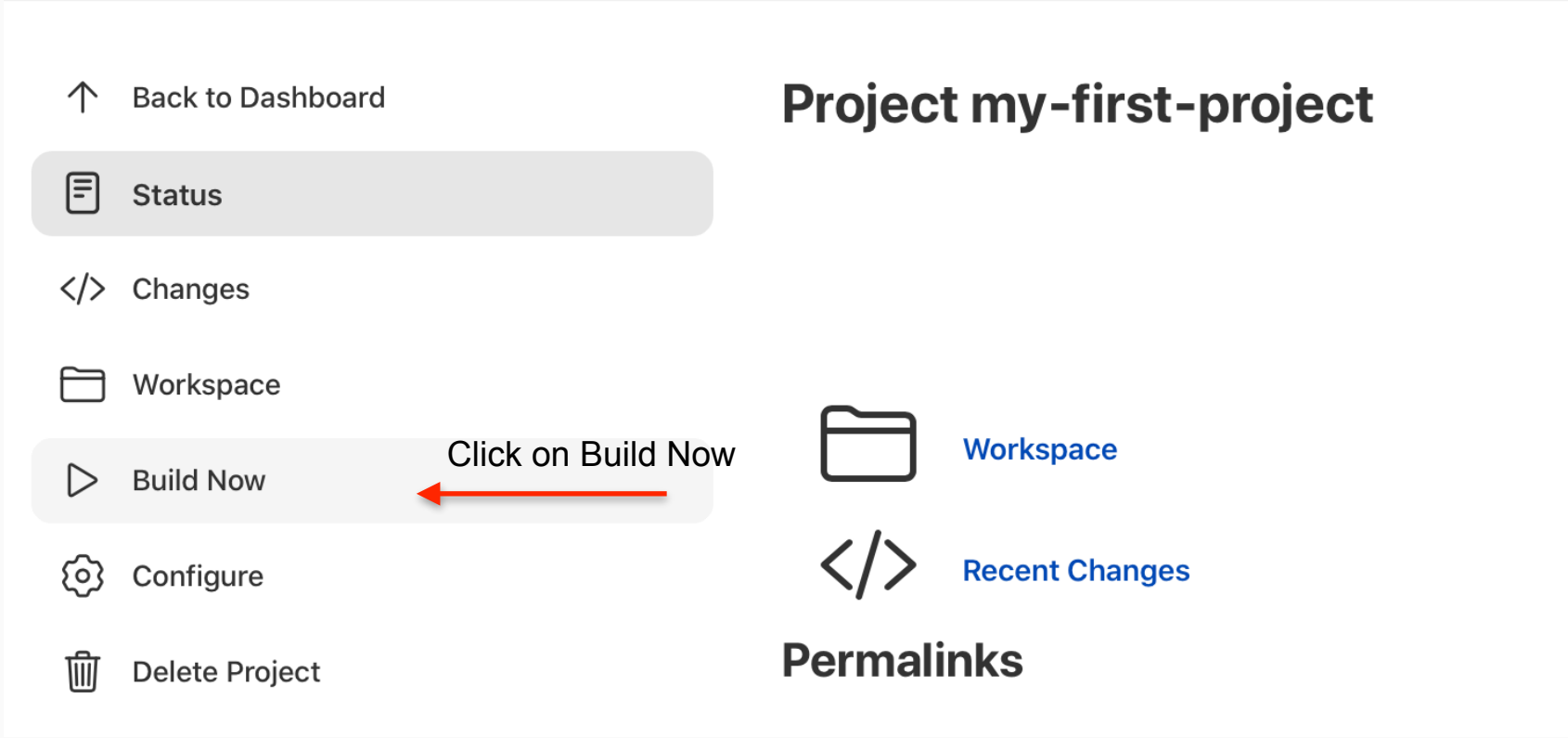


1. Click on Post-Build Actions
2. Click on ***Add post-build action***
3. Select ***Cucumber Reports***
4. ***Save***





JENKINS

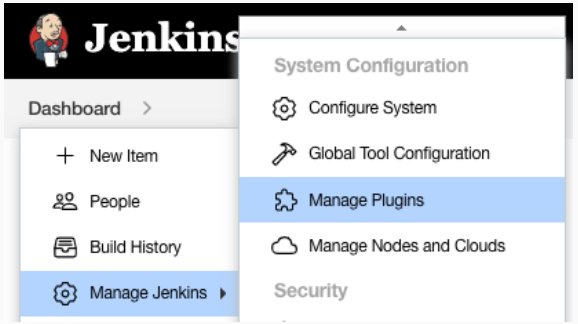

Back to Dashboard
Status
Changes
Workspace
Build Now
Configure
Delete Project
Click on Build Now
Project my-first-project
Workspace
Recent Changes
Permalinks

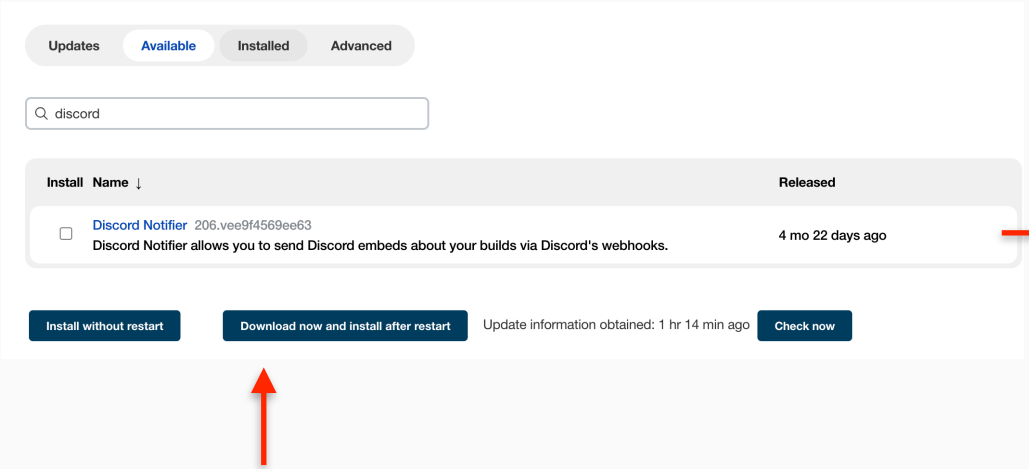
CYDEO

How to Send Discord Notifications

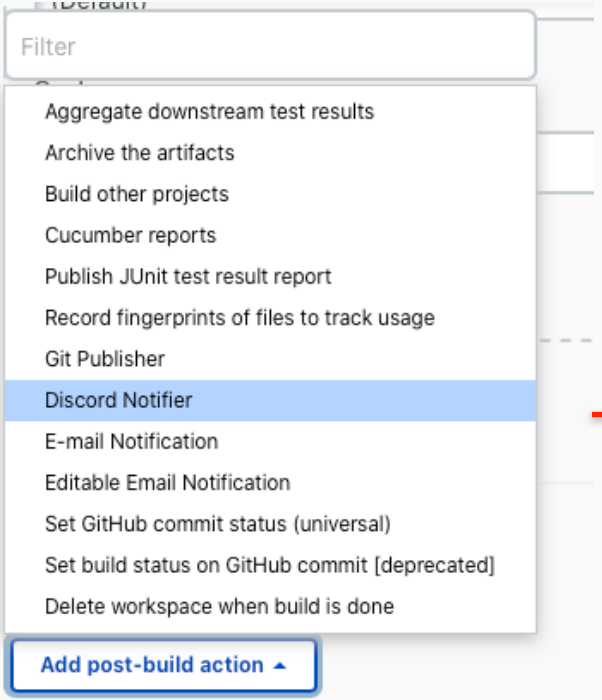
How to Install Discord Notifier Plugin to Jenkins

- Go to Jenkins Plugin Management to install
  - Select Available
  - Search for **discord**
  - Select **Discord Notifier**
  - **Download now and install after restart**

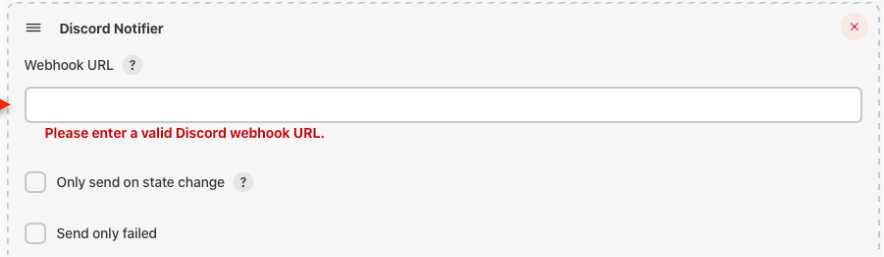


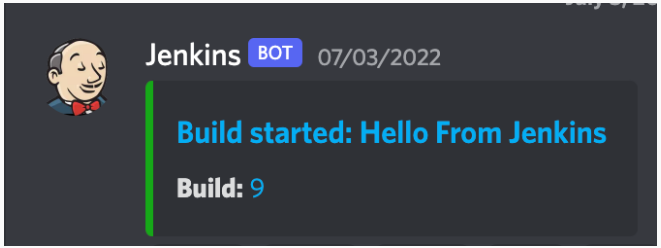


How to Add Discord Notifications to Job



- Click Add post-build action
- Choose Discord Notifier
- Take Webhook from Discord Channel
- Paste **Webhook URL** field
- Save and Apply
- Run Jenkins job to get notifications





CYDEO

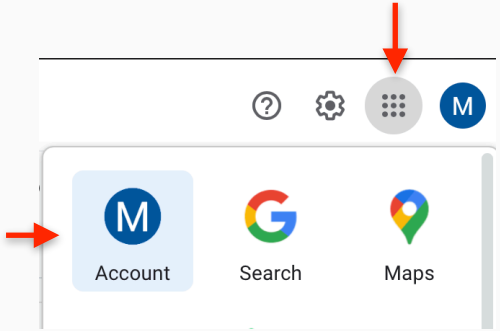
How to Send Email Notifications

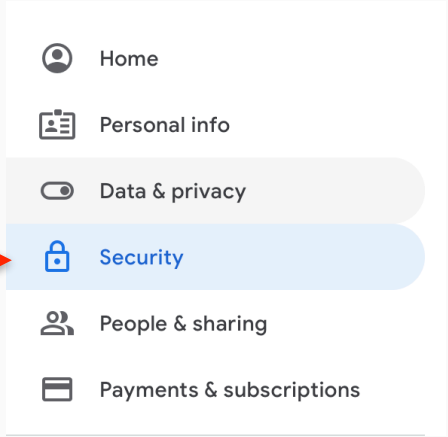
Gmail Settings

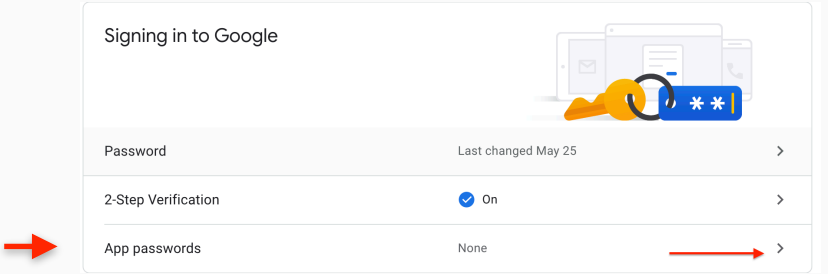
1. Log in Gmail Account
2. Click Account
3. Click Security
4. Click App Passwords

WARNING :

Don’t use your personal email account.
Since we will give access and share password with Jenkins
Create a new email to use only practice purpose.
If you can’t see App Passwords, enable 2-step verification

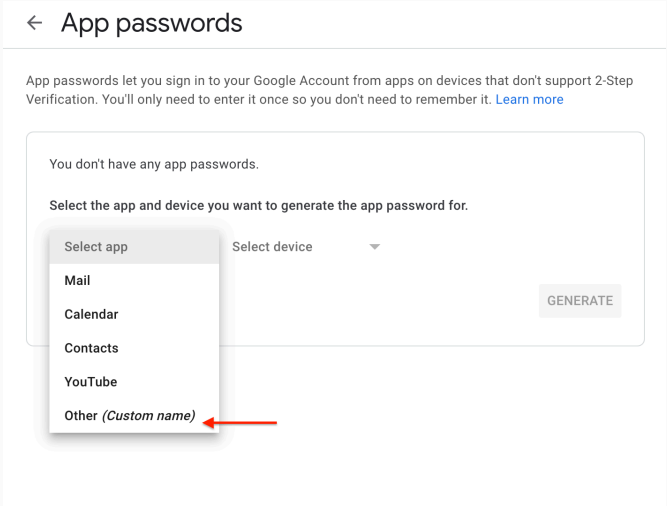


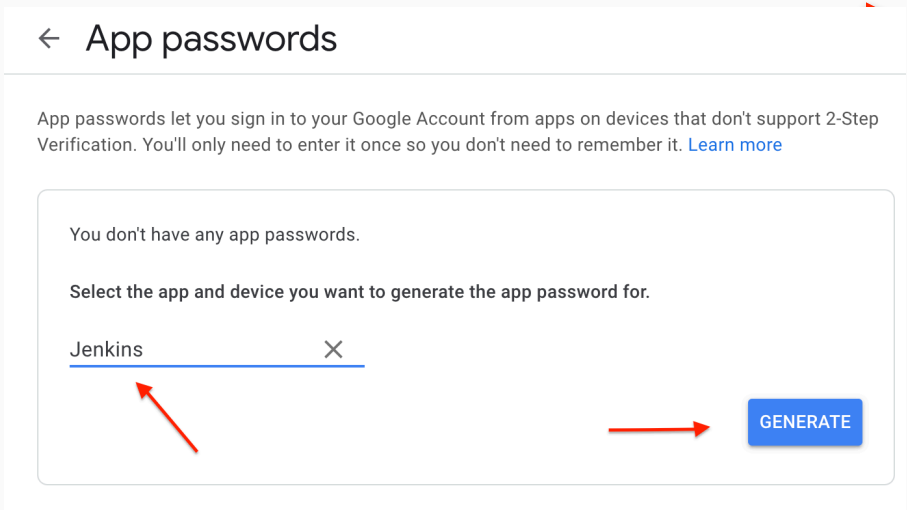




Gmail Settings

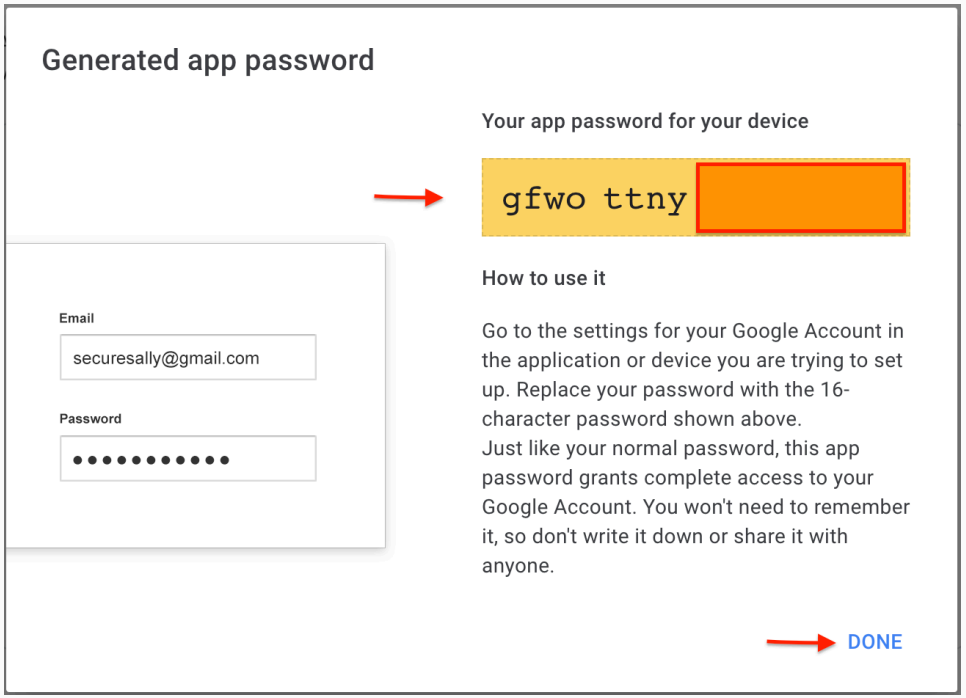
1. Click select app
2. Give a name (Jenkins)
3. Click **Generate**





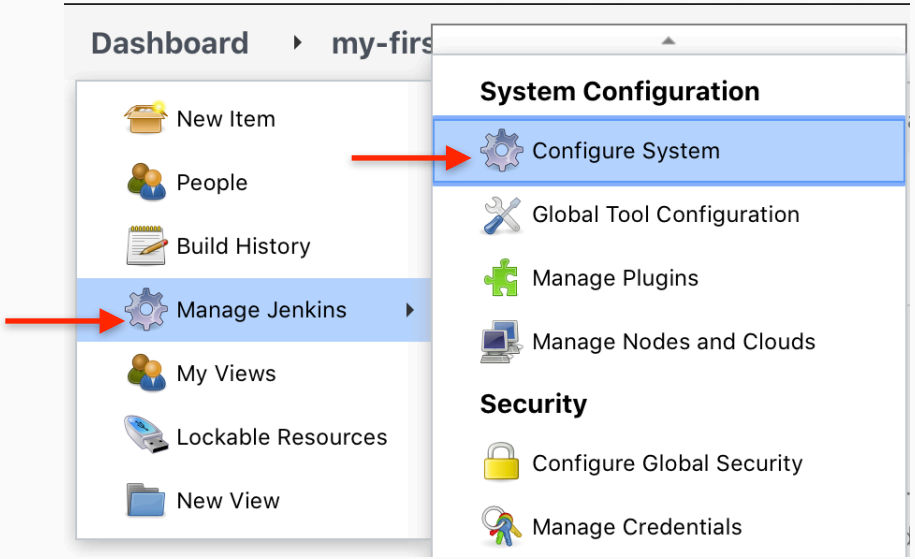
Gmail Settings

1. Copy generated password from here to use int Jenkins
2. Click DONE



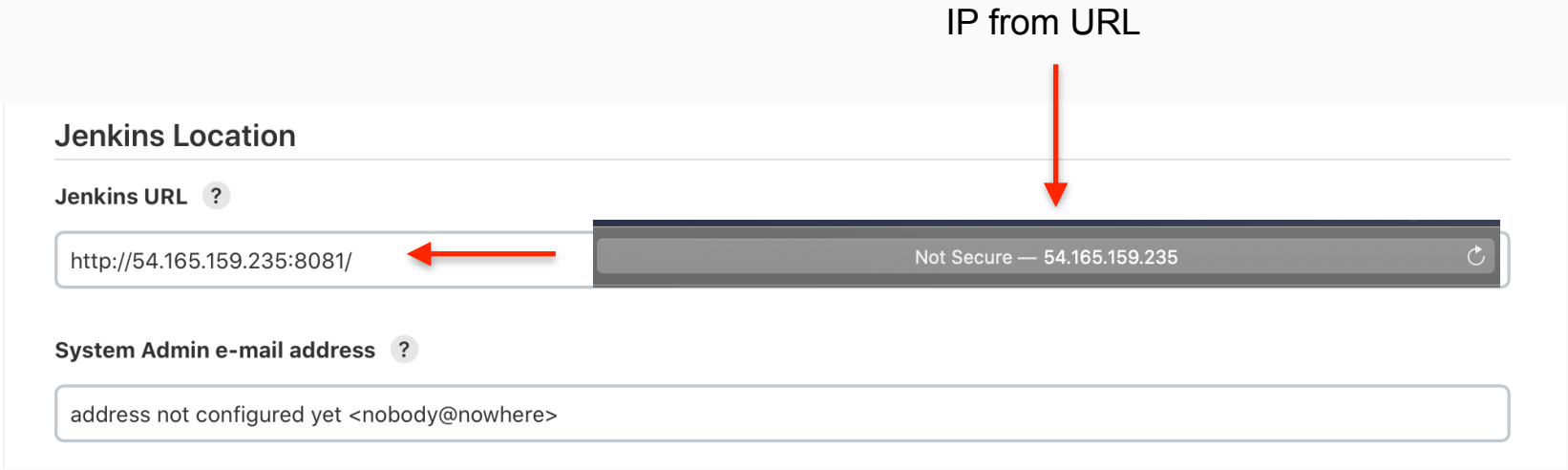
Jenkins Email Settings

1. Click dropdown from ***Dashboard***
2. Click ***Configure System***



Jenkins Email Settings

1. Scroll down to ***Jenkins Location***
2. IP should match with your IP from url

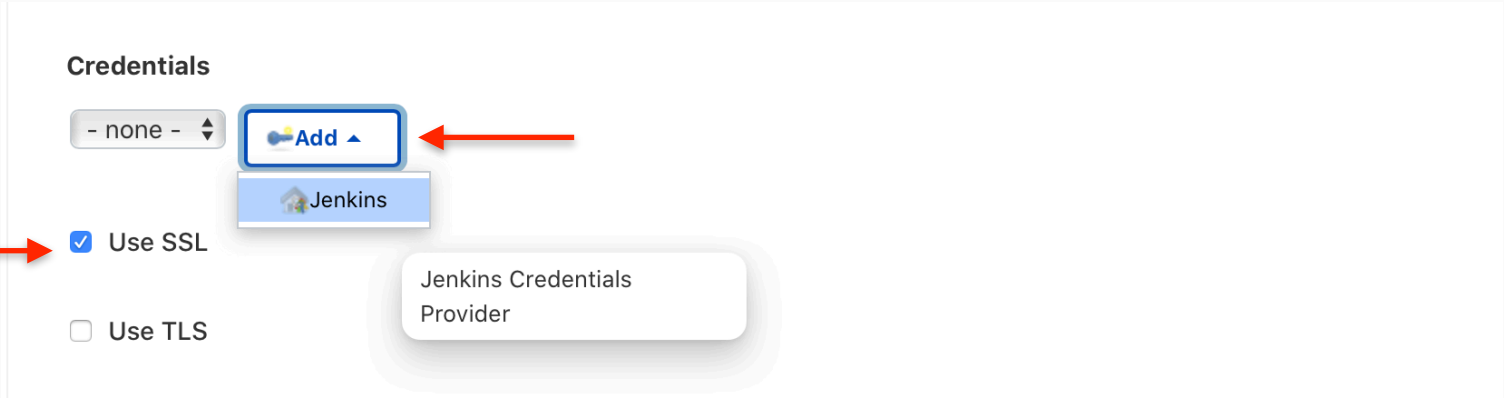


Jenkins Email Settings

1. Scroll down to ***Extended E-mail Notification***
2. **Server :** smtp.gmail.com
3. **Port** : 465
4. Click Advanced

Jenkins Email Settings

1. Select **Use SSL**
2. Click Add Jenkins

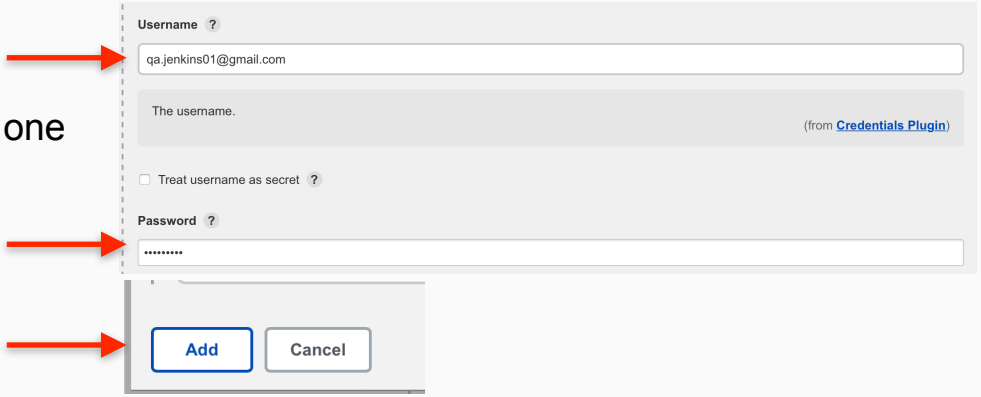


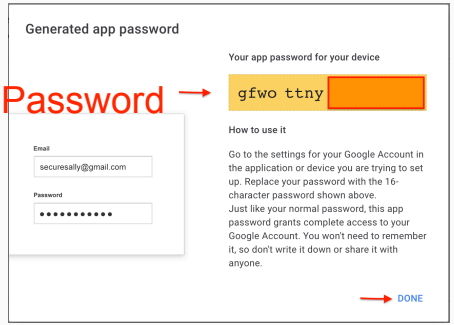
Jenkins Email Settings

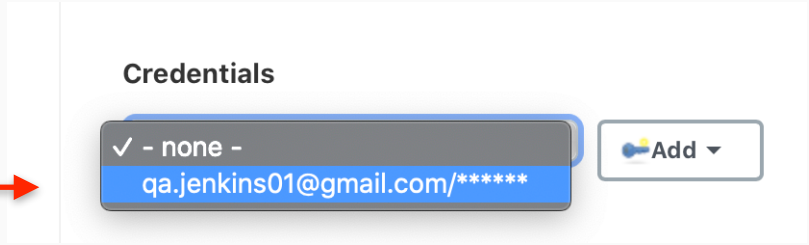
1. Put your credentials here
2. Password same with Google's generated one
3. Click Add
4. Select credentials from dropdown

NOTE :

Create a new email to use only practice purpose.

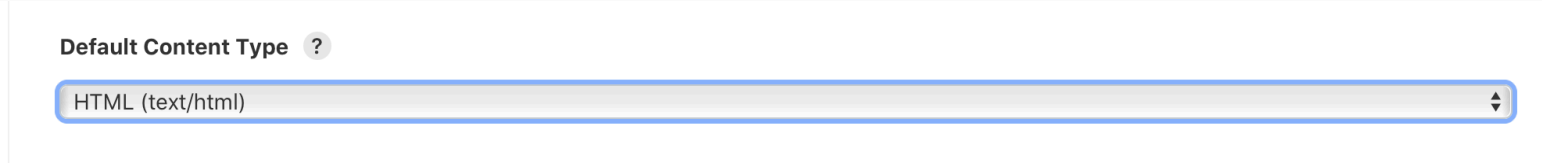






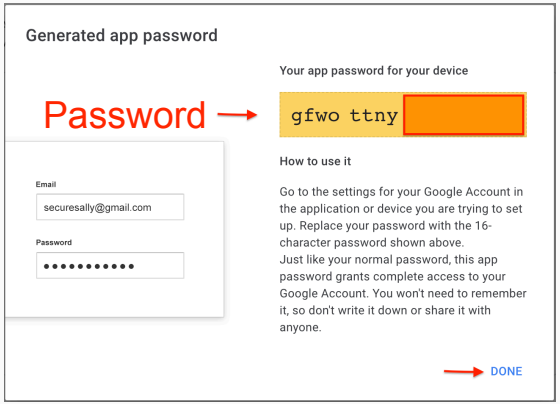
Jenkins Email Settings

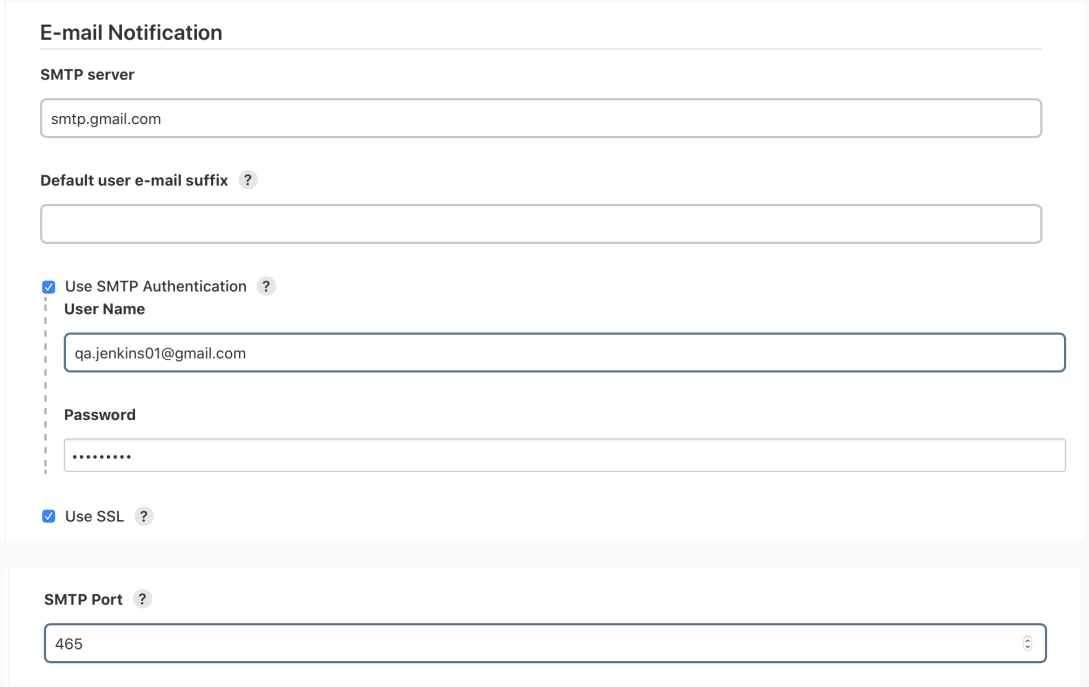
1. Scroll down to Default Content Type
2. Select HTML



Jenkins Email Settings

1. Scroll down to Email Notification
2. **Server** : smtp.gmail.com
3. **Username** : yourGmail
4. **Password** : Take it from Google
5. Select Use SSL
6. **Port** : 465



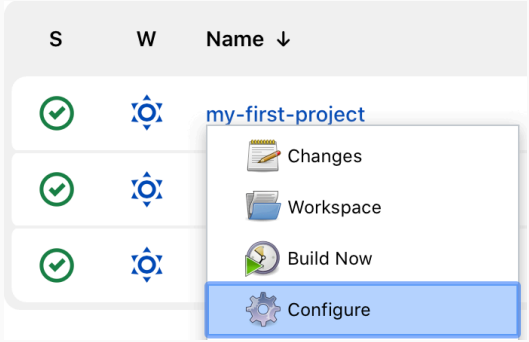


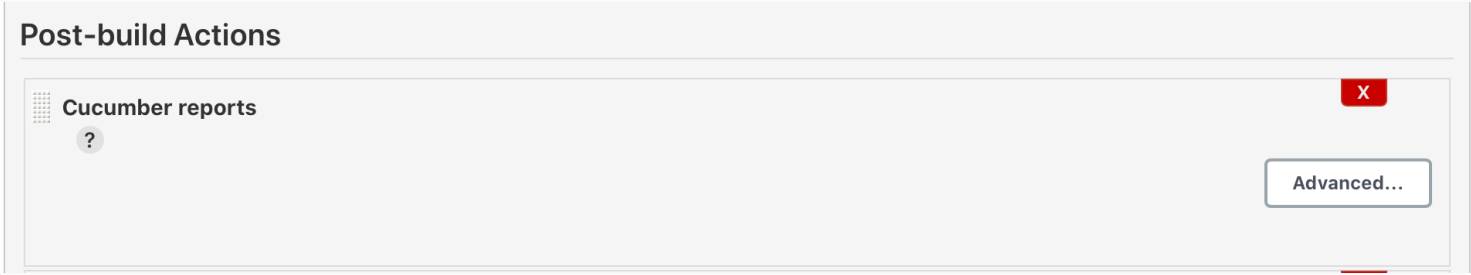
Jenkins Email Settings

1. Select **Test Configuration**
2. **Email ：**noreply@gmail.com
3. Click **Test Configuration**
4. Email was successfully sent !

Jenkins Email Settings

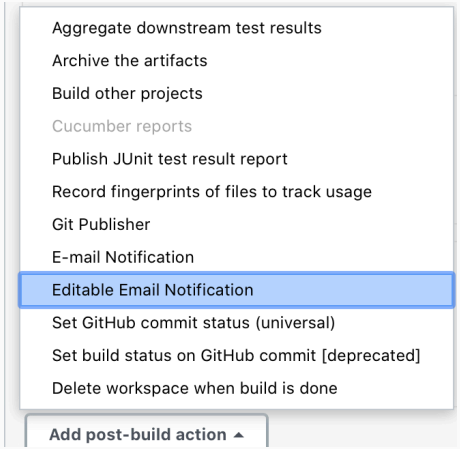
1. Click ***Configure from any Project***
2. ***Scroll down to post-build actions***
3. ***Make sure you added Cucumber Report***

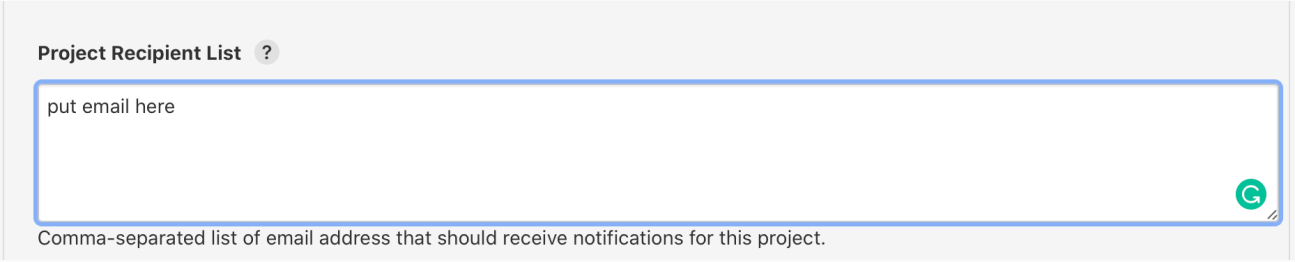


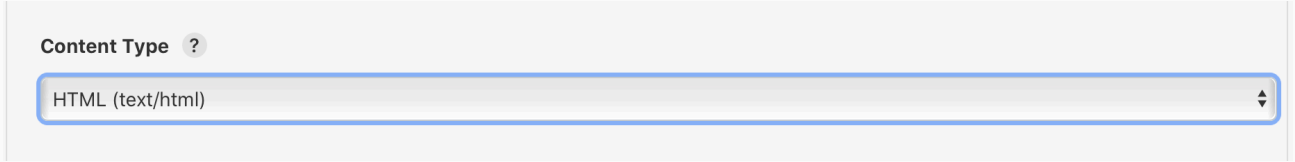


Jenkins Email Settings

1. Click ***Add post-build action***
2. Select ***Editable Email Notification***
3. Put emails into ***Recipient List***
4. **Content Type** needs to be HTML







Jenkins Email Settings

1. Click ***Advance Settings***
2. Select Trigger as **Always**
3. **Save**