

TUTORIAL JENKINS

1. Pengertian

Jenkins adalah sebuah open source automation server populer untuk mengotomatiskan tugas-tugas di dalam proses continuous integration and delivery sebuah perangkat lunak. Jenkins dapat menjalankan daftar langkah yang telah ditentukan sebelumnya, misalnya untuk mengkompilasi kode sumber Java dan membangun JAR dari kelas yang dihasilkan. Pemicu untuk eksekusi ini bisa berdasarkan waktu atau event.

2. Pre-requisite

- Memiliki Repository di Github
- Install Jenkins

Download installer di <https://www.jenkins.io/>

Jalankan installer, ikuti petunjuk yang diberikan (tips: cukup menggunakan default setting).

3. Buat Repository

Pada tutorial ini, kita sudah memiliki contoh repository pada latihan sebelumnya (Tutorial Git dan Github)

 rizaman007 / latihan3Sc3_2020

4. Download & Install Jenkin

Lakukan instalasi jenkins dengan terlebih dahulu mendownloadnya di <https://jenkins.io/download/>

Install Jenkin

Lakukan instalasi seperti biasa

Konfigurasi

Unlock Jenkin dengan memasukkan password yang berada di C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Pilih install suggested plugins

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Setelah proses instalasi selesai, buatlah akun user terlebih dahulu. Silakan gunakan username dan password yang mudah diingat

Getting Started

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:

Selanjutnya isi jenkins url dengan localhost:8080

Instance Configuration

Jenkins URL:

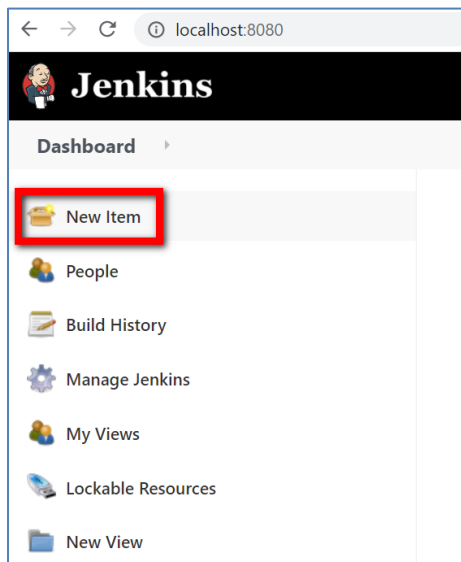
`http://localhost:8080/`

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

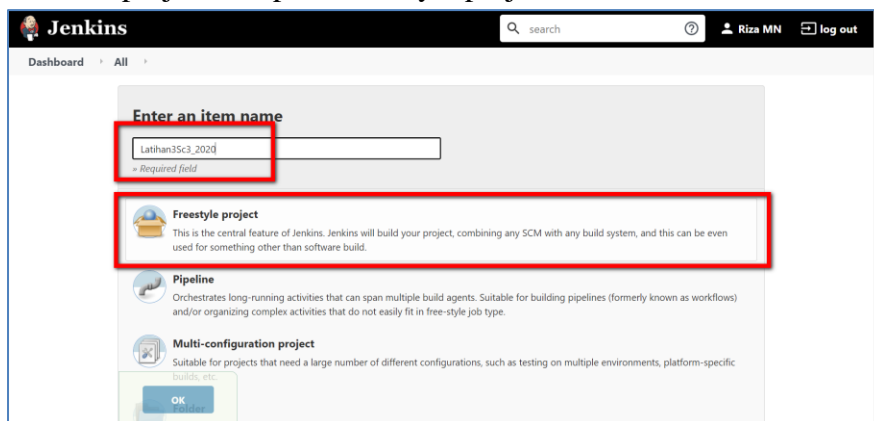
The proposed default value shown is not saved yet and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

5. Buat Project di Jenkin

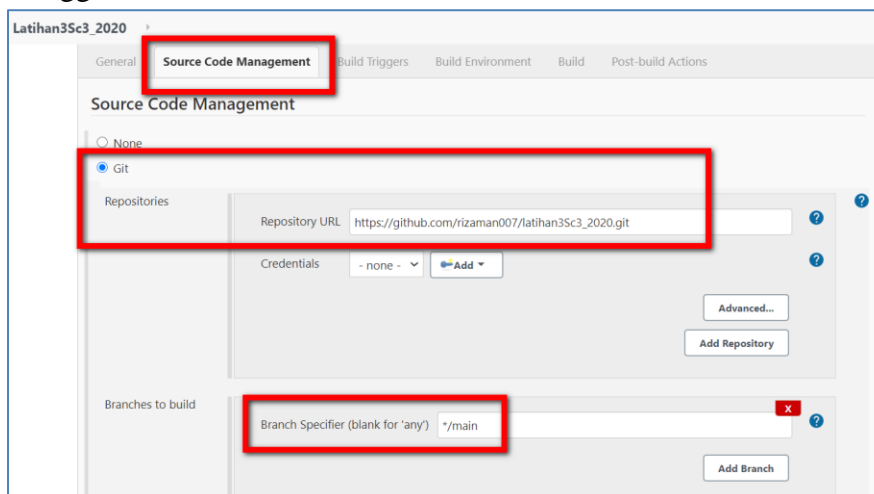
Pada halaman selanjutnya pilih menu New Item untuk membuat project baru.



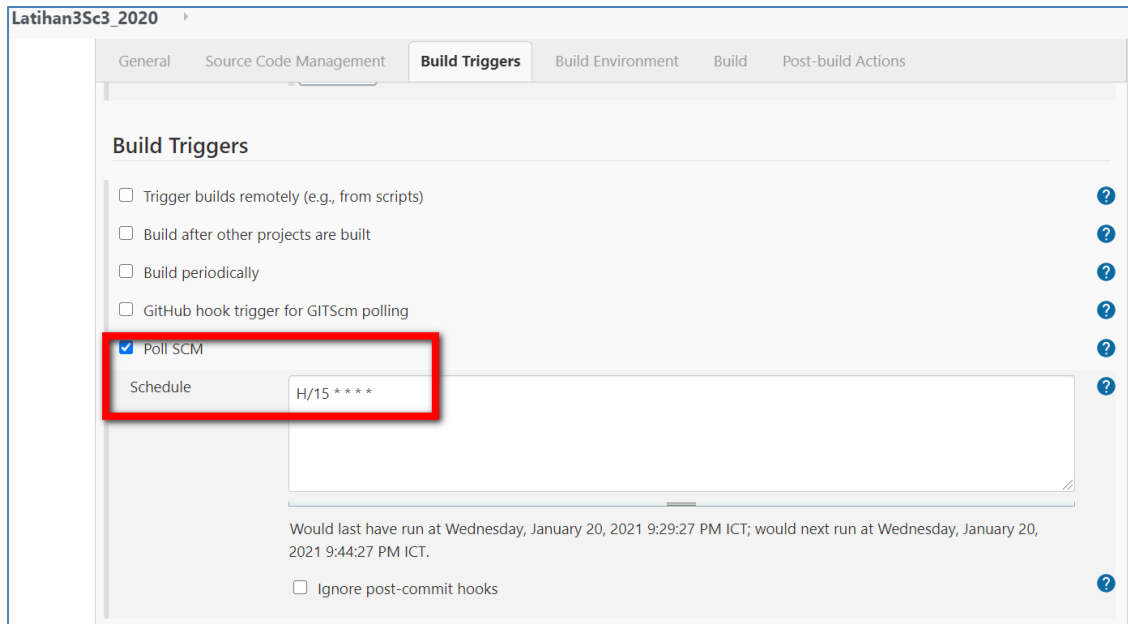
Isi nama project dan pilih Freestyle project lalu klik OK.



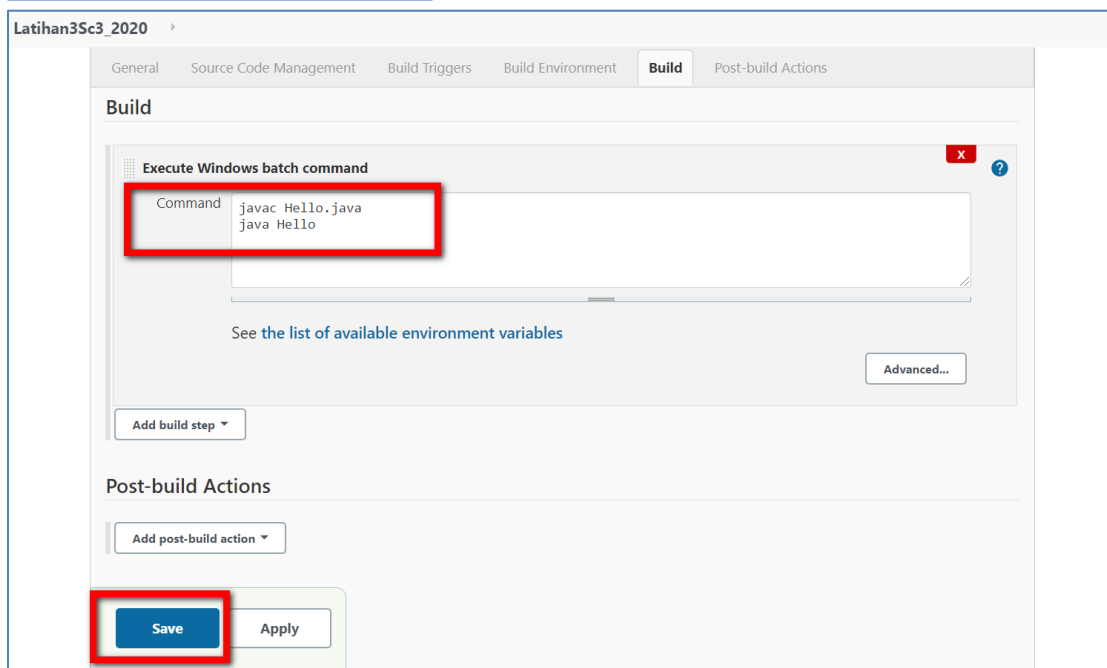
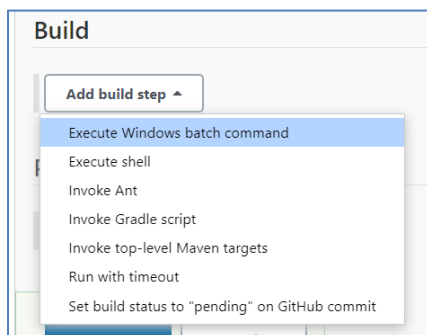
Click tab Source Code Management, lalu pilih Git dan isi dengan URL repo project. Untuk branch disesuaikan masing-masing ya. Defaultnya adalah **master**, sedangkan di sini saya menggunakan **main**. Jadi, diubah ke main.



Klik tab Build Triggers lalu pilih Poll SCM atau pilih trigger yang diinginkan. Contoh berikut polling repositori Git yang akan memicu build setiap 15 menit ketika ada sesuatu yang berubah di repo.

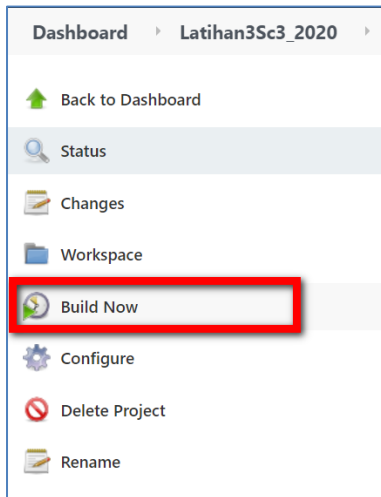


Pada tab build, Click Add build step lalu pilih Execute Windows batch command

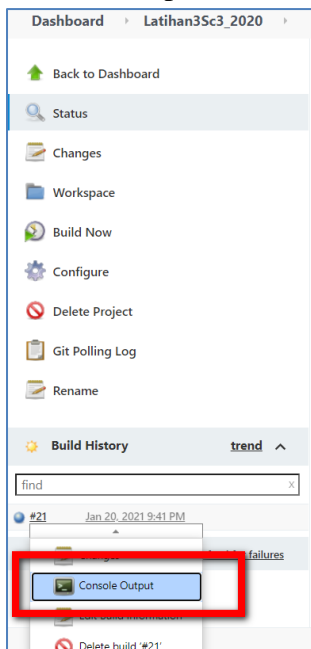


6. Build project

Pada layar utama, Klik tombol Build now di sebelah kiri untuk membuat kode sumber.



Setelah mengklik Build Now, kita dapat melihat status build yang dijalankan di Build History. Klik dropdown pada Build number yang ada lingkaran berwarna biru. Lingkaran berwarna biru artinya proses build berhasil dilakukan. Jika merah artinya proses build gagal. Kemudian pilih Console Output.



Berikut pesan sukses yang muncul. Pesan tersebut akan tampil jika kita telah mengikuti pengaturan dengan benar.



Console Output

```
Started by user Riza MN
Running as SYSTEM
Building in workspace C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Latihan3Sc3_2020
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/rizaman007/latihan3Sc3_2020.git # timeout=10
Fetching upstream changes from https://github.com/rizaman007/latihan3Sc3_2020.git
> git.exe --version # timeout=10
> git --version # 'git version 2.30.0.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/rizaman007/latihan3Sc3_2020.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision 654753f09cfac4fa0bfd42093b943b12d21f4a50 (refs/remotes/origin/main)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f 654753f09cfac4fa0bfd42093b943b12d21f4a50 # timeout=10
Commit message: "First Feature"
First time build. Skipping changelog.
[Latihan3Sc3_2020] $ cmd /c call C:\Windows\TEMP\jenkins8454709323129971568.bat

C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Latihan3Sc3_2020>javac Hello.java

C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Latihan3Sc3_2020>java Hello
Judul
RMN: Aplikasi Java 2020
--- *** ---
END
User 2
User 1

C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Latihan3Sc3_2020>exit 0
Finished: SUCCESS
```

Pada tutorial ini kita telah menjalankan program Hello yang dihosting di GitHub. Jenkins menarik kode dari remote repository dan melakukan build terus menerus sesuai dengan frekuensi yang ditentukan.