

<i>Session Date</i>	:	13 September 2019
<i>Semester</i>	:	Gasal
<i>Subject</i>	:	1132102 – Pemrograman Berorientasi Objek
<i>Week/Session</i>	:	1/3
<i>Key Topics</i>	:	<i>Environment Setting</i>
<i>Activity</i>	:	<ul style="list-style-type: none"> <li>- JDK Instalasi</li> <li>- Konfigurasi</li> <li>- Netbeans Instalasi</li> </ul>
<i>Duration</i>	:	100 minutes
<i>Objective</i>	:	Mahasiswa mampu menginstal dan melakukan konfigurasi JDK dan Netbeans
<i>Lecturer</i>	:	TMP
<i>Instructor</i>	:	SEP

## Referensi

Berikut merupakan referensi yang digunakan dalam materi:

1. [www.oracle.com](http://www.oracle.com)
2. [www-eu.apache.org](http://www-eu.apache.org)

## Pendahuluan

Java merupakan Bahasa pemrograman yang dapat dijalankan pada beberapa perangkat elektronik seperti komputer, telepon genggam, dan perangkat lain yang mendukung. Bahasa ini pertama kali lahir dari *The Green Project* yang dimotori oleh Patrick Naughton, Mike Sheridan, dan James Gosling, beserta 9 pemrogram lainnya dari Sun Microsystem.

Pada Java terdapat 2 jenis program berbeda, yaitu aplikasi dan applet. Keduanya merupakan bagian dari *execute*, dimana *execute* merupakan salah satu fase kelima dalam siklus program Java. Aplikasi adalah program yang biasanya disimpan dan dieksekusi dari komputer lokal. Applet adalah program yang biasanya disimpan pada komputer yang jauh, dikoneksikan oleh pengguna melalui *web browser*. Salah satu penggunaan terbesar Java adalah pembuatan aplikasi *native* untuk Android. Selain itu Java juga menjadi pondasi bagi berbagai bahasa pemrograman seperti Kotlin, Scala, Clojure, Groovy, JRuby, Jython, dan lainnya yang memanfaatkan Java Virtual Machine (JVM) sebagai rumahnya.

Java juga merupakan Bahasa pemrograman tingkat tinggi yang berorientasi objek dan terdiri atas beberapa kelas. Kelas-kelas tersebut terdiri atas metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya.

## Petunjuk

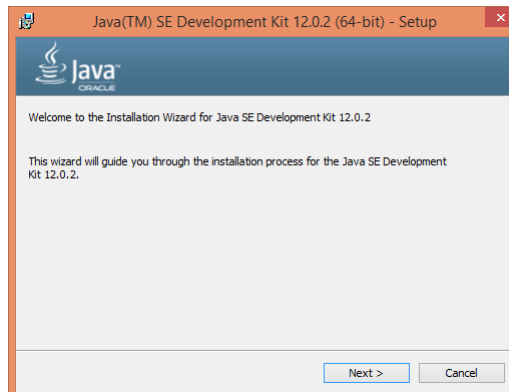
Sebelum melakukan instalasi JDK dan *text editor* ada beberapa hal yang perlu dipersiapkan sebelumnya, yaitu:

1. Mengunduh *Java Development Kit* (JDK) pada laptop atau komputer. JDK itu sendiri merupakan perangkat lunak yang digunakan untuk melakukan proses kompilasi dari kode java ke *bytecode* yang dapat dimengerti dan dapat dijalankan oleh *Java Runtime Environment* (JRE). JDK dapat diunduh melalui *link* berikut:  
<https://www.oracle.com/technetwork/java/javase/downloads/index.html>
2. Mengunduh *text editor* untuk membuat aplikasi Java. Anda dapat memilih untuk menggunakan Netbeans, Eclipse, Notepad++, Kate, Nano, Atom, Geany, Komodo Edit, Ultra Edit, JEdit, atau *text editor* lain. Pada praktikum ini disarankan mahasiswa menggunakan Netbeans pada *link* berikut:  
<https://www-eu.apache.org/dist/netbeans/netbeans/11.1/Apache-NetBeans-11.1-bin-windows-x64.exe>

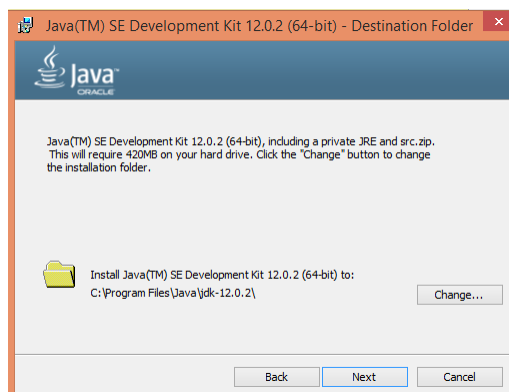
## Uraian Praktikum

### A. Instalasi JDK

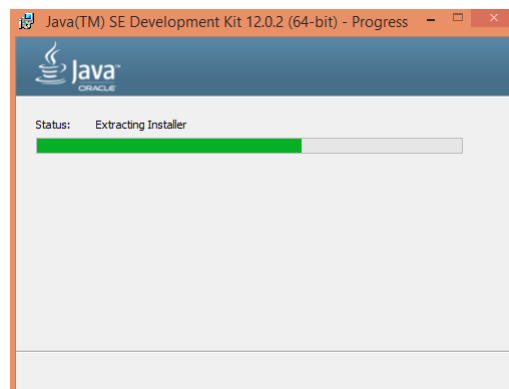
1. Lakukan *double click* pada program *installer* JDK yang telah diunduh.
2. Kemudian pilih *Next* ketika muncul *dialog installer* Java.



3. Pilih Next, kemudian Anda akan menerima lisensi.



4. Pada layar berikutnya Anda akan melihat proses instalasi Java JDK sedang berjalan dan melihat status instalasi. Tunggu hingga proses ini selesai.



5. Setelah itu akan muncul *dialog box* yang akan memastikan lokasi penyimpanan yang akan digunakan untuk instalasi, Anda dapat menentukannya sendiri atau langsung memilih Next.

6. Setelah proses sebelumnya selesai, maka akan muncul pemberitahuan bahwa Java SE Development Kit telah berhasil diinstal. Klik tombol Close untuk *finishing*.



Sampai saat ini proses instalasi Java di Windows telah berhasil. Namun Anda juga harus melakukan proses konfigurasi untuk mengatur *path* Java agar dapat digunakan pada laptop atau komputer.

Sebelum lanjut ke pembahasan konfigurasi, mungkin beberapa dari Anda menemukan pesan **“This software has already been installed on your computer. Would you like to reinstall it?”** ketika melakukan instalasi.

Bila Anda memilih **Yes** pada pilihan di atas untuk menginstall ulang Java maka akan muncul pesan peringatan berikut **“This action is only valid for products that are currently installed”**.

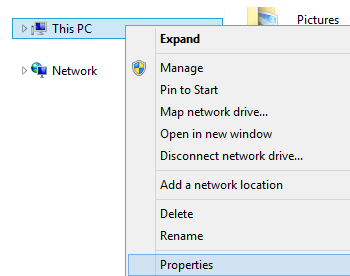
Masalah ini disebabkan oleh rusaknya informasi instalasi Java sebelumnya entah karena terhapus atau *corrupt*, tapi kunci registry yang berhubungan masih tersisa sehingga membuat installer Java mengira bahwa Java masih terinstall di computer Anda. Ada beberapa solusi yang dapat digunakan untuk mengatasi masalah ini:

- a) Menggunakan aplikasi Java MSI Fix
- b) Menggunakan *script batch* dari *link* berikut:  
(<https://cdn.utekno.com/wp-content/uploads/2013/08/JavaInstallerFix.bat>)
- c) Menggunakan *registry*

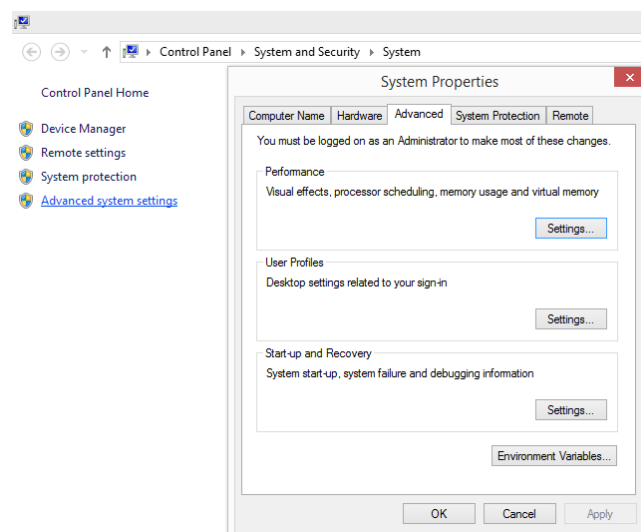
Proses ini sama dengan menggunakan *script batch*, hapus file *installer* Java di Program Files sebelum mengulangi proses instalasi kembali.

Jika tidak terdapat lagi masalah ketika melakukan instalasi, Anda dapat melakukan konfigurasi Java dengan langkah dibawah ini:

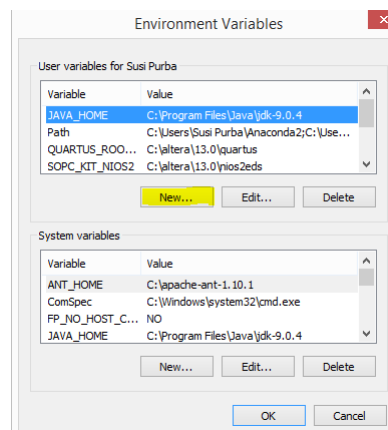
1. Setelah berhasil melakukan instalasi terhadap JDK, klik kanan pada **This PC** kemudian pilih **Properties**.



2. Selanjutnya pilih **Advanced system settings** maka akan muncul *System properties* seperti dibawah ini:



3. Selanjutnya pilih *button Environment Variables*, pilih New.



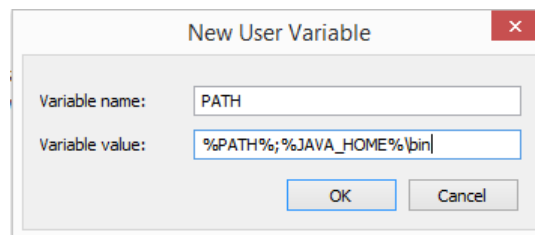
Isikan variable name dan variable value seperti dibawah ini, kemudian pilih **OK**:

**Variable name:** JAVA\_HOME

**Variable value:** C:\Program Files\Java\jdk-12.0.2

Untuk *variable value*, dapat menyesuaikan dengan jenis JDK dan lokasi penyimpanan yang dipilih ketika melakukan instalasi.

- Setelah proses nomor 3 selesai, maka Anda dapat kembali lagi ke halaman **Environment Variables** silahkan pilih **New** lagi dan isi kembali *variable name* dan *variable value* seperti gambar berikut, kemudian pilih **OK**.



- Untuk memastikan JDK sudah terintegrasi dengan laptop atau computer kita, maka dapat dilakukan pengecekan dengan menggunakan *command prompt*, lalu ketikkan **java**.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Susi Purba>java
Usage: java [options] [<mainclass>] [args...]
       or java [options] -jar [<jarfile>] [args...]
       or java [options] -n [<module>][<mainclass>] [args...]
       or java [options] --module [<module>][<mainclass>] [args...]
       (to execute the main class in a module)

Arguments following the main class, -jar <jarfile>, -n or --module
<module> [<mainclass>] are passed as the arguments to main class.

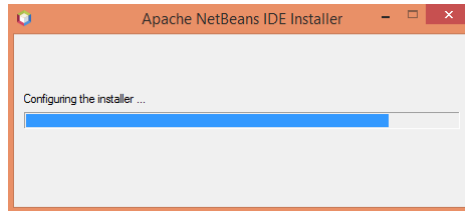
where options include:

-d32           Deprecated, will be removed in a future release
-d64           Deprecated, will be removed in a future release
-cp <class search path of directories and zip/jar files>
-classpath <class search path of directories and zip/jar files>
             A ; separated list of directories, JAR archives,
             and ZIP archives to search for class files.
-p <module path>
-module-path <module path>...
             A ; separated list of directories, each directory
             is a directory of modules.
--upgrade-module-path <module path>...
             A ; separated list of directories, each directory
             is a directory of modules that replace upgradeable
             modules in the runtime image
--add-modules <module name>[,<module name>]...
             root modules to resolve in addition to the initial module.
             <module name> can also be ALL-DEFAULT, ALL-SYSTEM,
             ALL-MODULE-PATH.
--list-modules list observable modules and exit
-d <module name>
-describe-module <module name>
             describe a module and exit
--dry-run      create VM and load main class but do not execute main method.
             The --dry-run option may be useful for validating the
             command-line options such as the module system configuration.
--validate-modules
             validate all modules and exit
             The --validate-modules option may be useful for finding
             conflicts and other errors with modules on the module path.
-D<name>=<value>
             set a system property
-verbose[:<class[:module[:gc[:jni]]>]
             enable verbose output
-version       print product version to the <stream> and exit
--version     print product version to the output stream and exit
-showversion  print product version to the <error stream> and continue
--show-version
             print product version to the output stream and continue
```

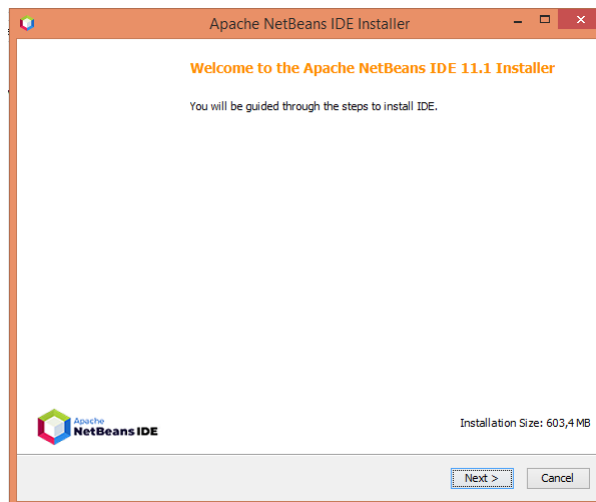
Jika hasil *command prompt* sudah seperti tampilan diatas, maka integrasi JDK dengan Windows telah berhasil dilakukan.

## B. Instalasi Netbeans

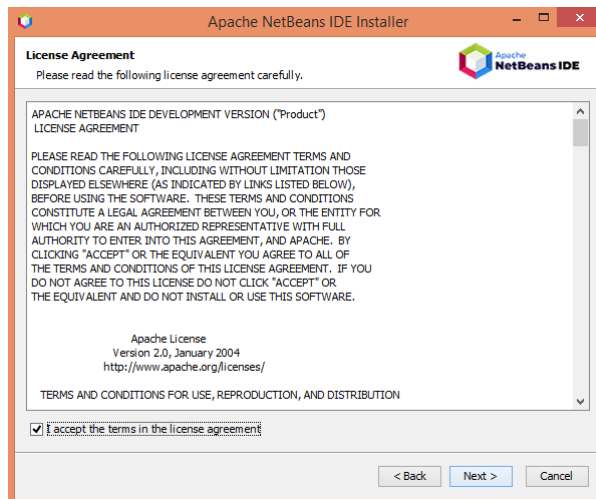
1. Setelah Anda berhasil mengunduh Netbeans Installer, maka klik dua kali *file* Netbeans (.exe) dan akan muncul proses di bawah ini.



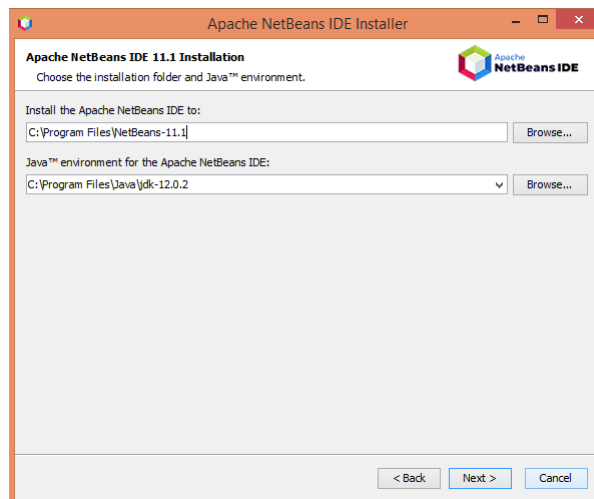
2. Klik Next untuk melanjutkan.



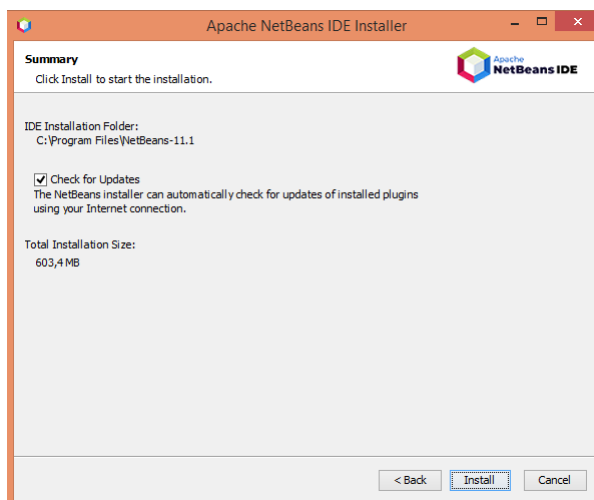
3. Beri centang pada *license agreement* kemudian Next.



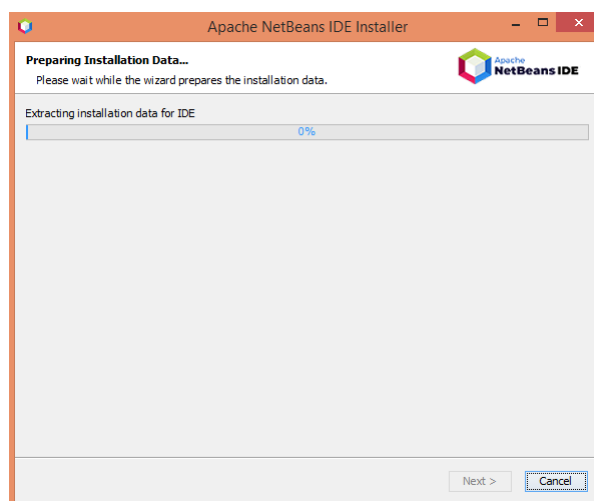
4. Kemudian pilih letak instalasi Netbeans, lalu Next.



5. Klik Install untuk memulai proses instalasi.

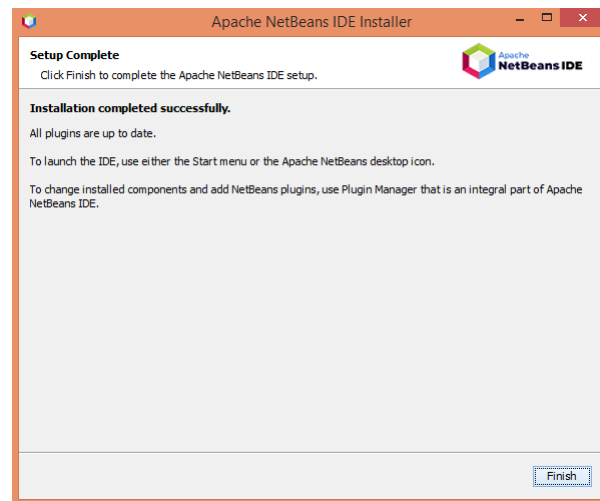


6. Tunggu proses instalasi hingga selesai.

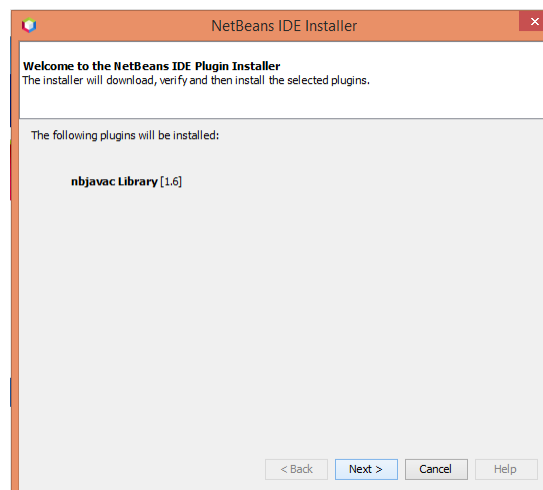




7. Kemudian klik Finish untuk menyelesaikan proses instalasi aplikasi Netbeans.

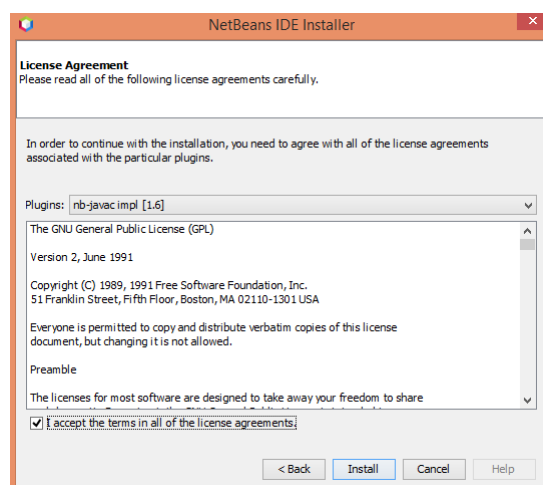


8. Ketika Anda membuka Aplikasi Apache Netbeans tersebut, maka akan muncul dialog berikut.

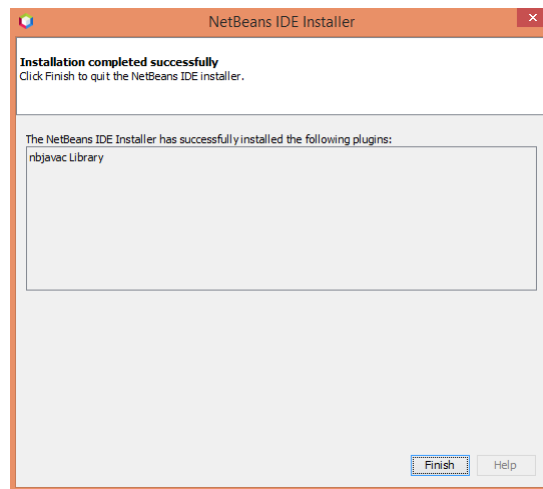


Jika Anda ingin melengkapi Netbeans Anda dengan **nbjavac plugin**, maka Anda dapat melanjutkan proses berikut dengan menekan tombol Next.

9. Pilih **I accept** dan kemudian klik Install untuk memulai instalasi *plugin*.



10. Jika proses instalasi *plugin* telah selesai, Anda dapat menekan tombol Finish untuk menyelesaikan tahap ini.



**Selamat Mengerjakan 😊**