

**LAPORAN PRAKTIKUM  
PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL I  
PENGENALAN IDE**



**Disusun Oleh:  
Nisrina Nurhaliza  
21102218**

**Dosen Pengampu :  
Dedy Agung Prabowo, S.Kom., M.Kom**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO  
PURWOKERTO  
2023**

## **BAB I**

### **TUJUAN PRAKTIKUM**

Setelah mengikuti praktikum ini, mahasiswa diharapkan dapat :

1. Memahami lingkungan dasar IntelliJ serta cara meng-*compile* dan menjalankan program

## **BAB II**

### **TOOLS**

1. IntelliJ Community Edition
2. Java SE Development Kit 19

## **BAB III**

### **DASAR TEORI**

OOP merupakan sebuah teknik membuat suatu program berdasarkan objek dan apa yang bisa dilakukan dengan objek tersebut. Object oriented program terdiri dari objek yang berinteraksi satu sama lain untuk menyelesaikan sebuah tugas. Alasan penggunaan OOP karena kode kode di-breakdown agar lebih mudah di manajemen. Breakdown berdasarkan objek atau tugas yang ada pada program tersebut. Dianjurkan digunakan untuk segala project karena dengan menggunakan OOP akan lebih mudah di manajemen dan proses debug menjadi lebih cepat.

IntelliJ IDEA adalah environment pemrograman khusus atau yang biasa disebut integrated development environment (IDE) yang sebagian besar dibuat untuk Bahasa pemrograman Java. IntelliJ IDEA, sebuah IDE yang dikembangkan oleh JetBrains. IntelliJ IDEA tersedia dalam 2 (dua) versi, yaitu versi Community Edition dengan lisensi Apache 2 Licensed yang bisa digunakan secara gratis dan versi Ultimate Edition untuk penggunaan komersial. IDE yang ditulis dengan Java dan Kotlin ini dirilis versi pertamanya pada Januari tahun 2011 dan langsung menjadi IDE pertama untuk pengembangan aplikasi berbasis Java. Ini mendukung navigasi kode tingkat lanjut dan kemampuan refactoring yang saling terintegrasi. Pada umumnya, sebuah IDE menyediakan beberapa fitur seperti text editor yang akan digunakan untuk menulis kode, tools untuk mengotomatisasi proses build dari program yang dibuat dan juga debugger yang akan membantu mendeteksi dan memperbaiki kesalahan yang terdapat pada program.

Java Development Kit (JDK) adalah perangkat pengembangan aplikasi Java. Perangkat ini mutlak diperlukan untuk membuat aplikasi Android, mengingat aplikasi Android ini berbasis java. Sebagaimana diketahui, java adalah salah satu bahasa pemrograman yang biasa digunakan untuk membuat aplikasi. Namun perlu diketahui, tidak semua pustaka di Java digunakan di Android. Sebagai contoh, Android tidak menggunakan Swing. Sebagai bahasa yang berbasas tinggi, yang menggunakan perintah-perintah yang mudah dimengerti oleh orang, Java mempunyai keunggulan yakni bersifat universal. Sebagai bahasa yang universal, Java bisa dijumpai di berbagai platform (Linux, UNIX, Windows, Mac, dll). Hal yang menarik lagi, hasil kompilasi Java yang dinamakan bytecode dapat dijalankan di berbagai platform sepanjang di sistem target memiliki Java Runtime Environment (JRE).

---

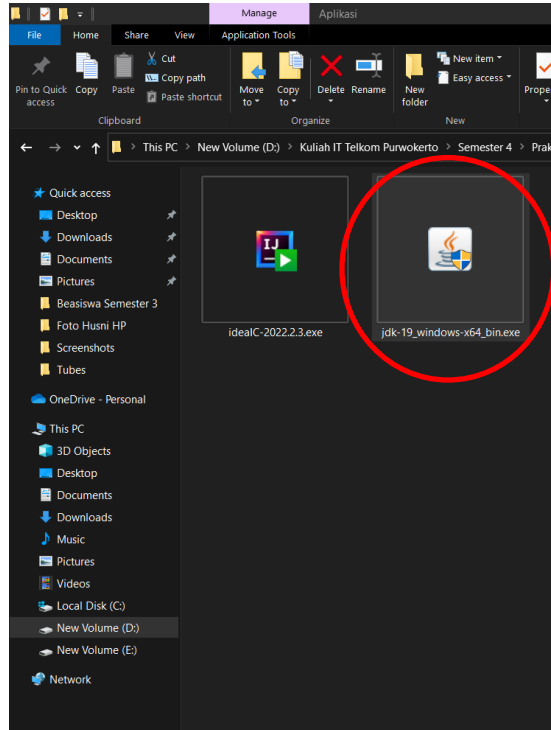
Sumber :

1. <https://www.dicoding.com/blog/berkenalan-dengan-intellij-idea/>  
Diakses pada tangga 16 Maret 2023.
2. <http://eprints.polsri.ac.id/8069/3/File%20III.pdf>  
Diakses pada tangga 16 Maret 2023.

## BAB IV

### GUIDED

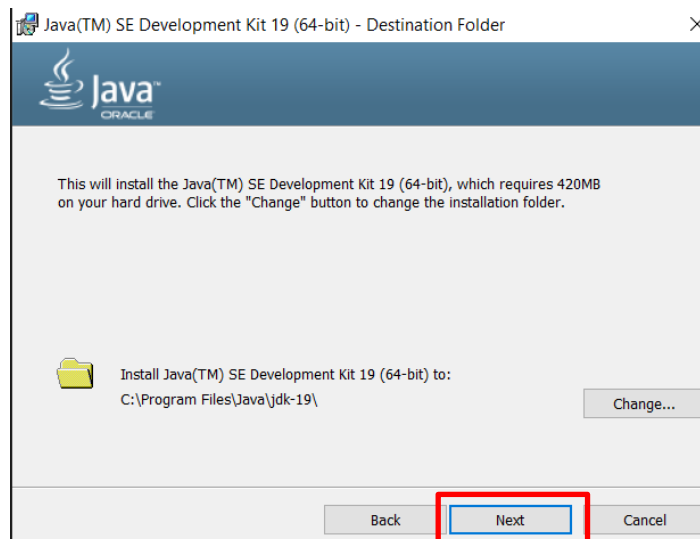
1. Langkah pertama yang dilakukan adalah meng-install aplikasi *jdk* terlebih dahulu dengan klik kanan → ***Run As Administrator***



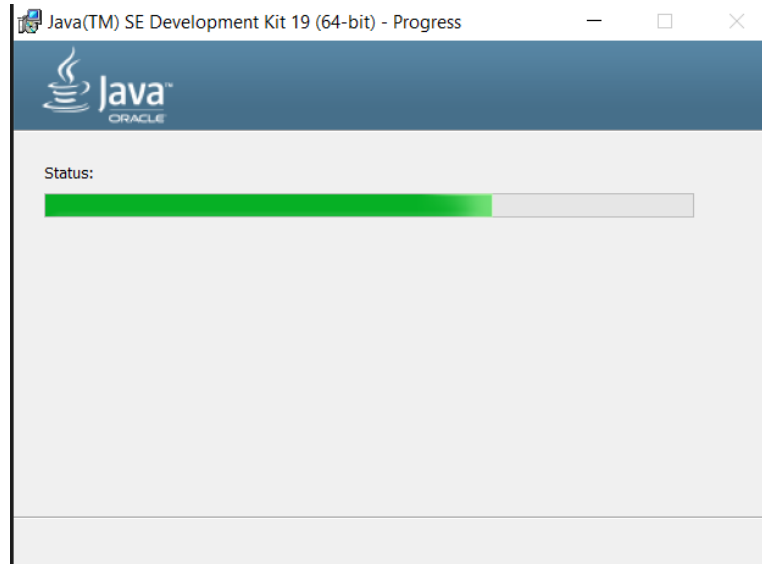
2. Setelah klik **Run As Administrator**, maka akan muncul tampilan seperti di bawah ini kemudian klik **Next**.



3. Selanjutnya kita memilih direktori penginstallan **JDK**, jika sudah memilih kemudian klik **Next**.



4. Selanjutnya tunggu **Loading** terlebih dahulu.

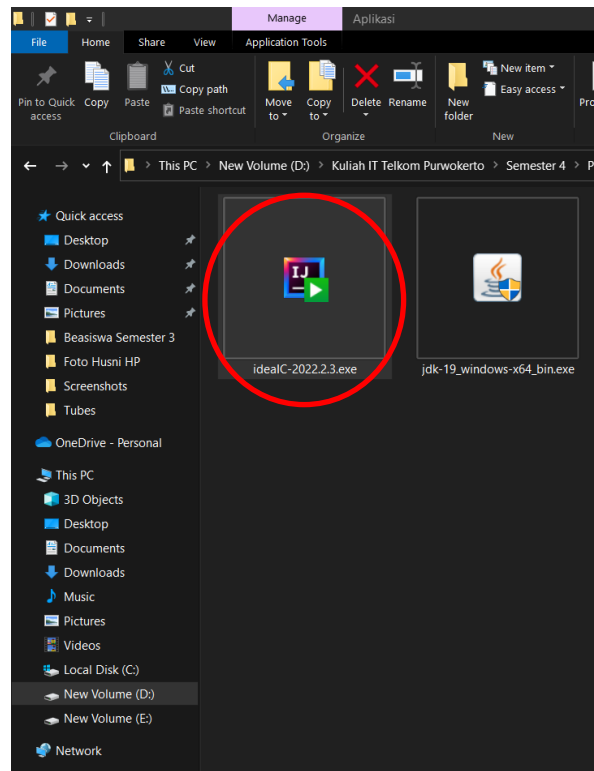


5. Instalasi aplikasi JDK sudah selesai, kemudian klik **Close**.

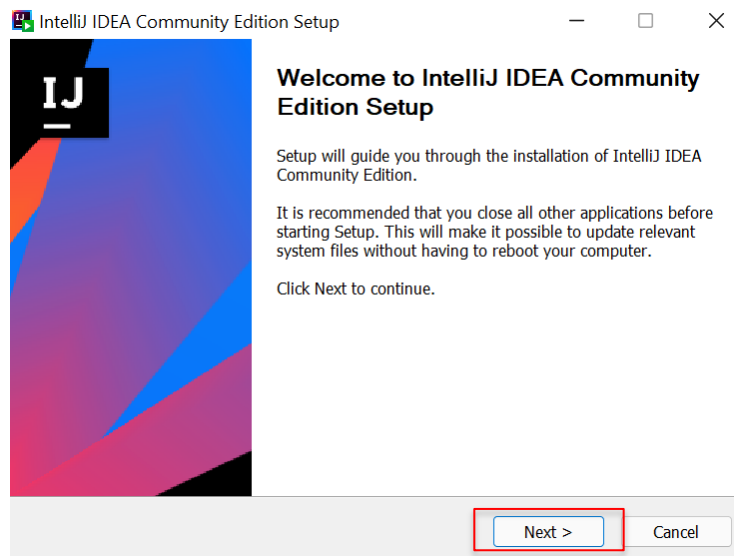




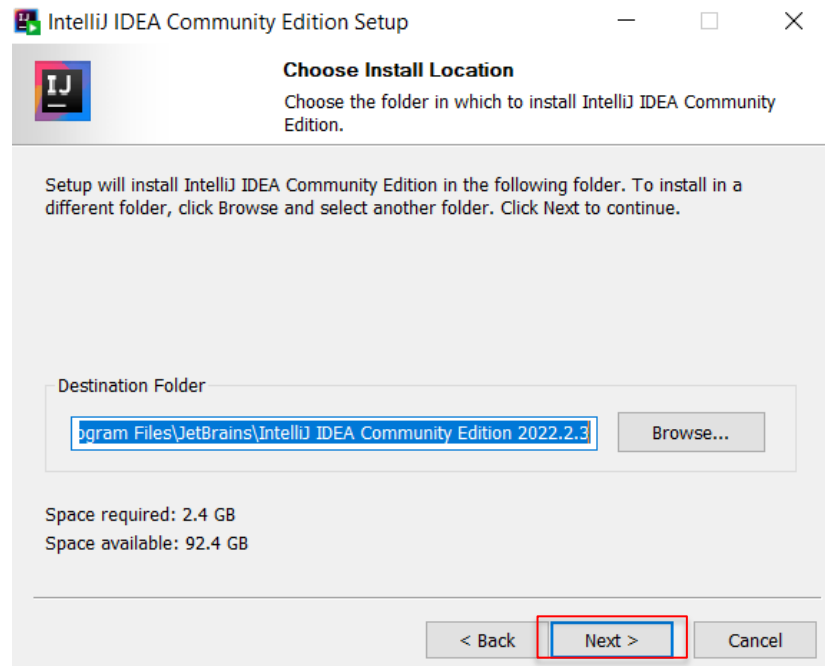
6. Setelah menginstall aplikasi JDK, selanjutnya kita akan menginstall aplikasi *IntelliJ*, dengan klik kanan → **Run As Administrator**.



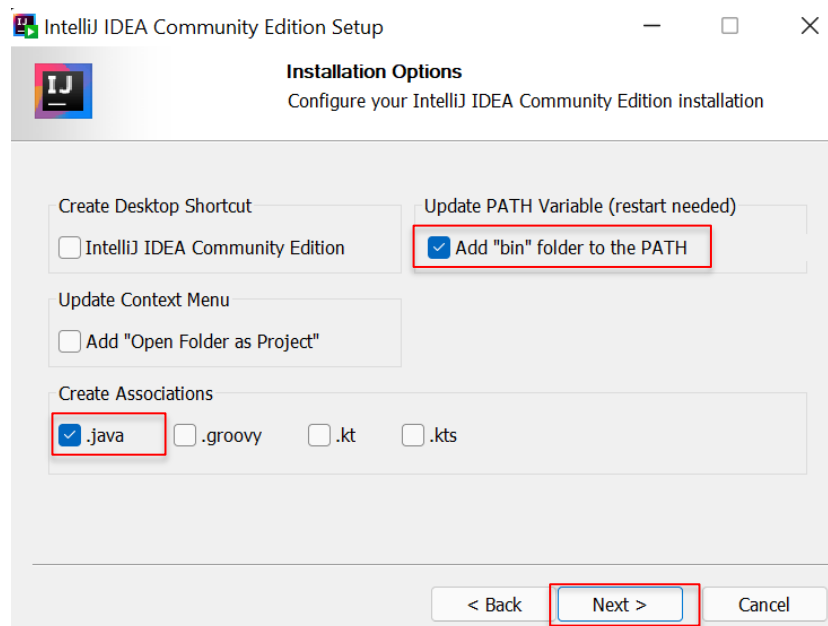
7. Pada tampilan awal penginstalan IntelliJ klik **Next**.



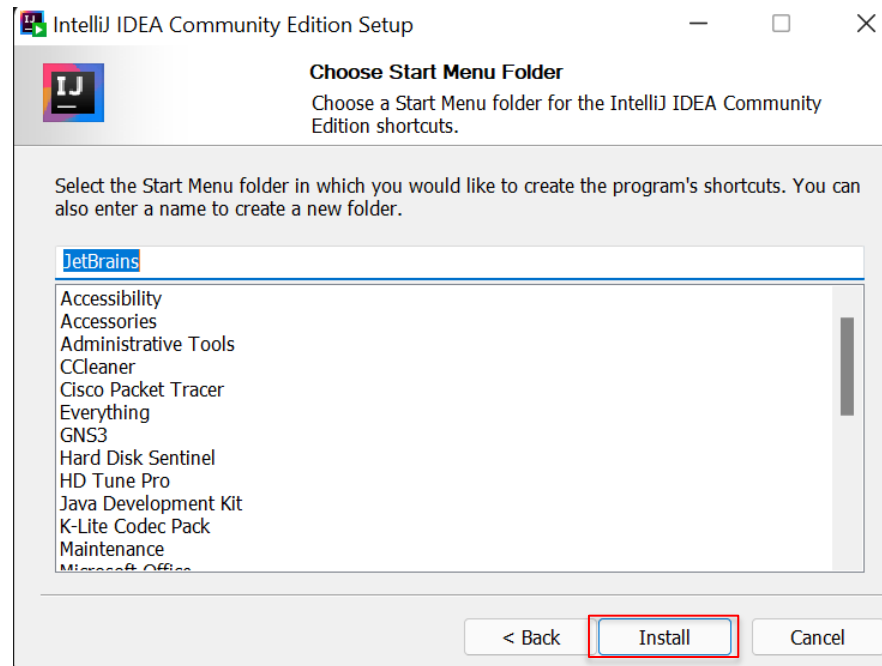
8. Klik *Next* pada tampilan berikut.



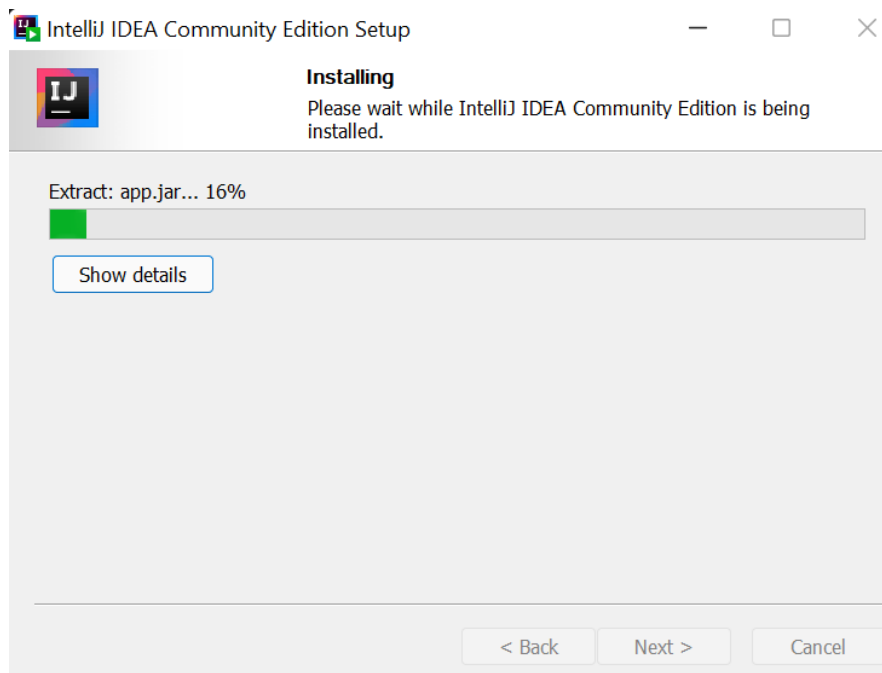
9. Pada tampilan berikut klik bagian *Update PATH Variable* dan klik *Create Associations* bagian *Java* selanjutnya kiln *Next*.



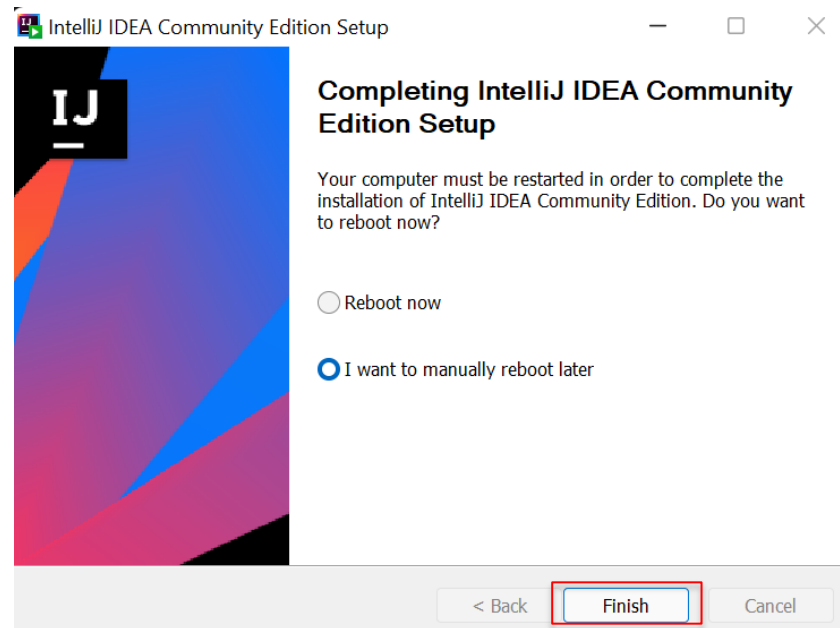
10. Selanjutnya klik ***Install*** pada tampilan berikut.



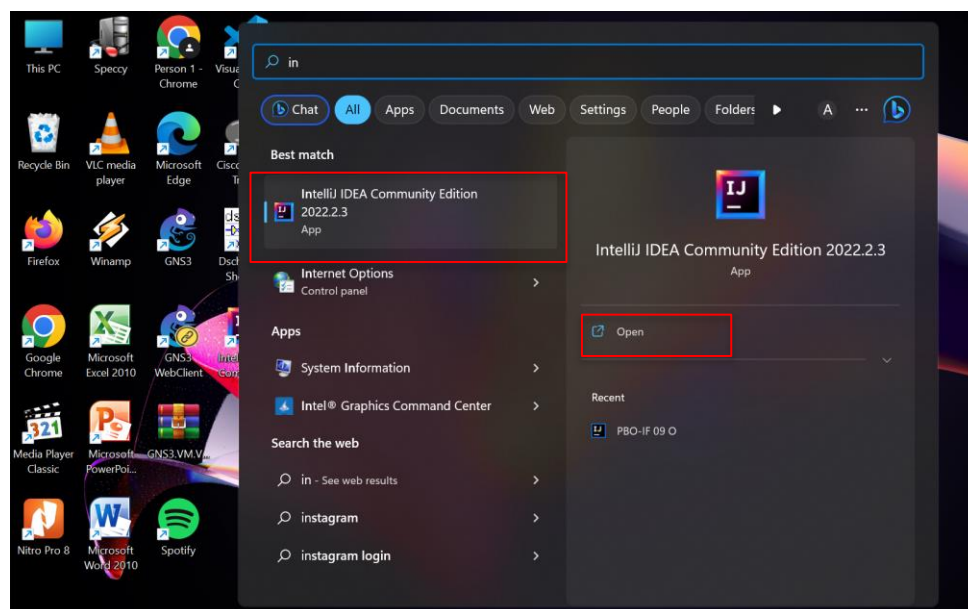
11. Tunggu proses instalasi hingga selesai.



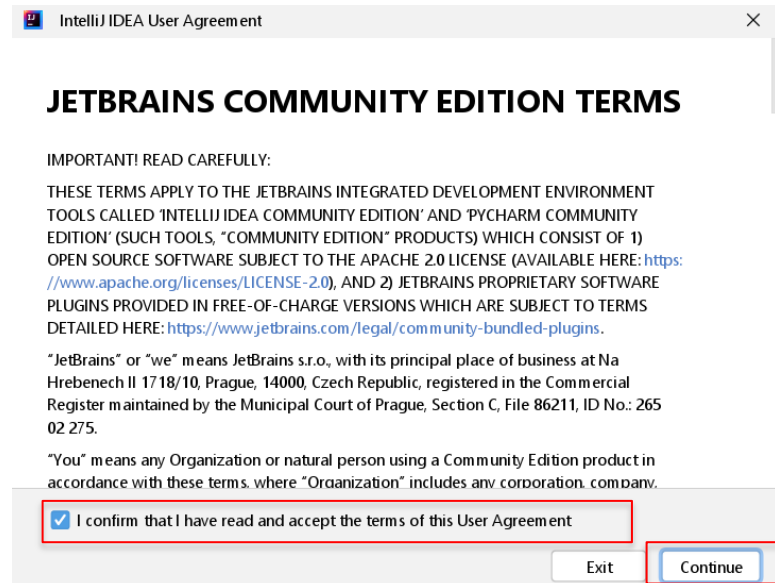
12. Ketika proses installasi telah selesai, kemudian klik ***Finish***.



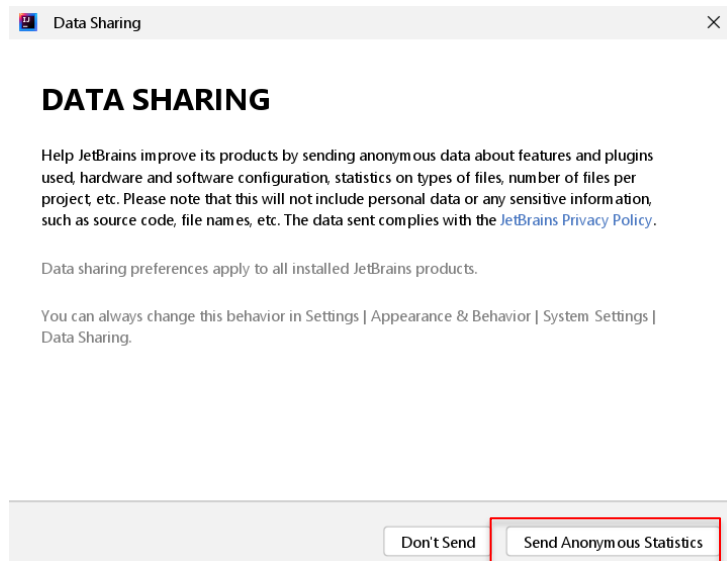
13. Jika aplikasi IntelliJ sudah teinstall, kemudian buka aplikasi tersebut.



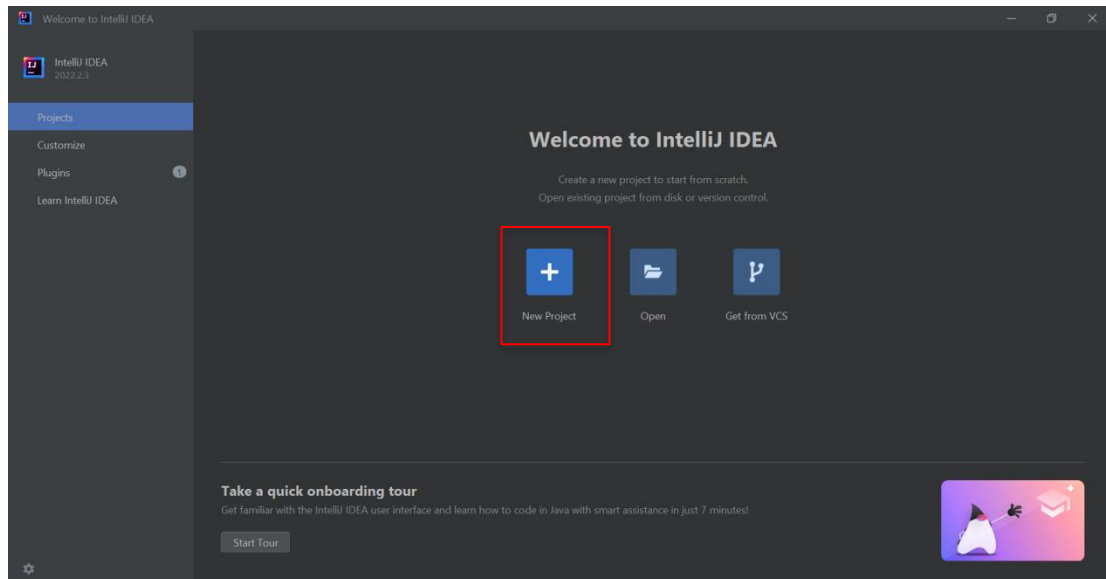
14. Pada tampilan ini, centang *checkbox* yang ada dibawah ini, kemudian klik *Continue*.



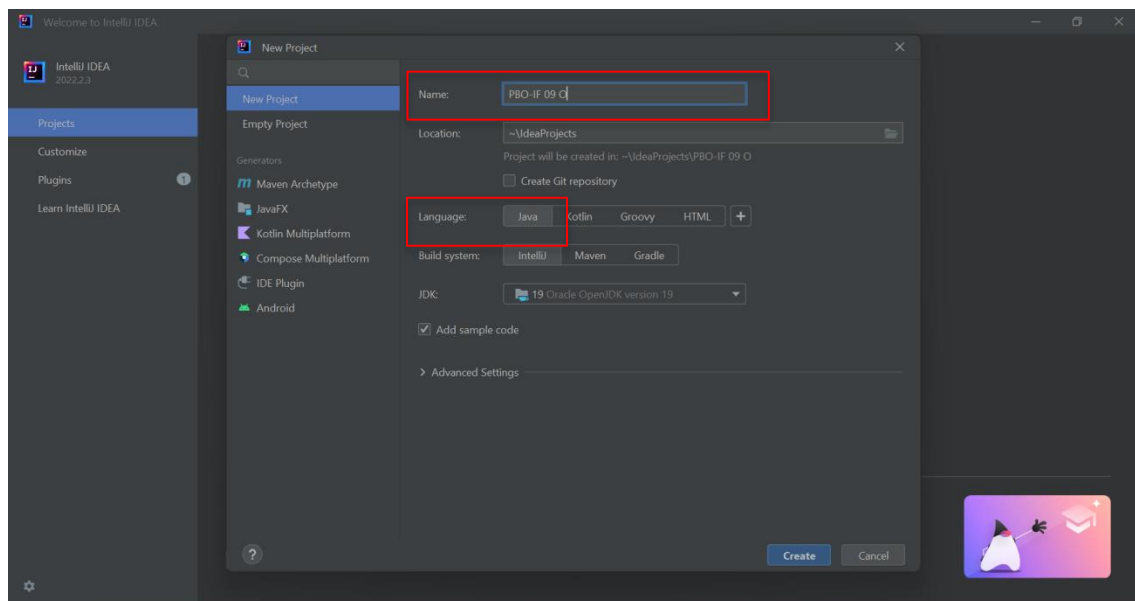
15. Klik *Send Anonymous Statistics* pada tampilan berikut.



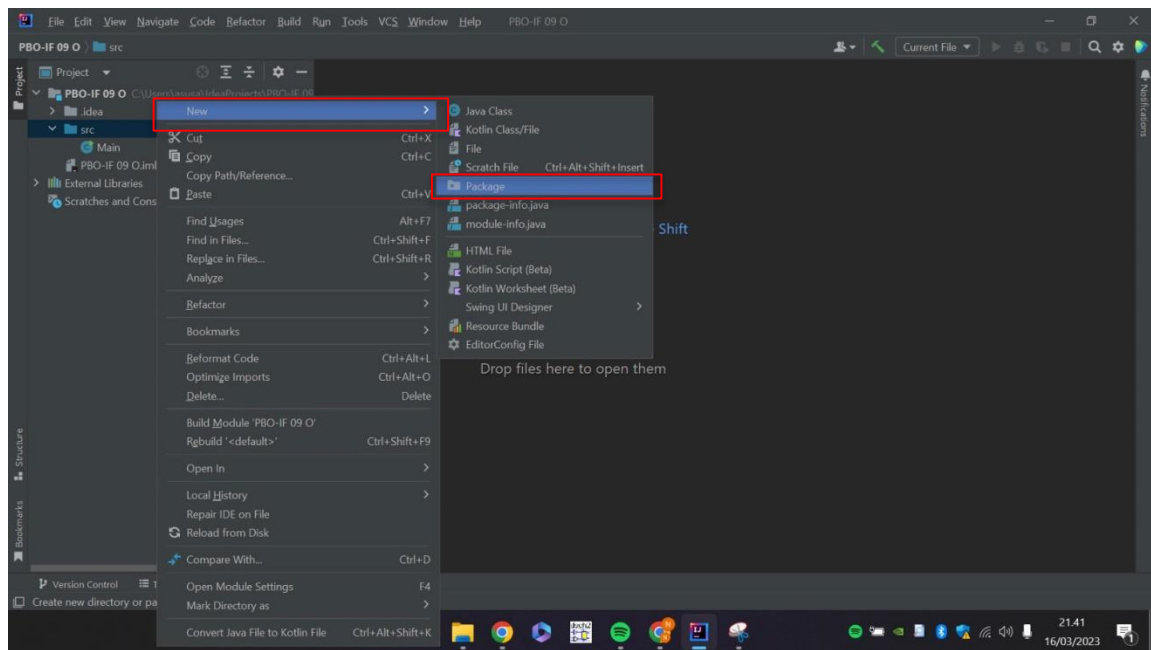
16. Pada tampilan awal aplikasi IntelliJ, klik **New Project** untuk menambah project baru.



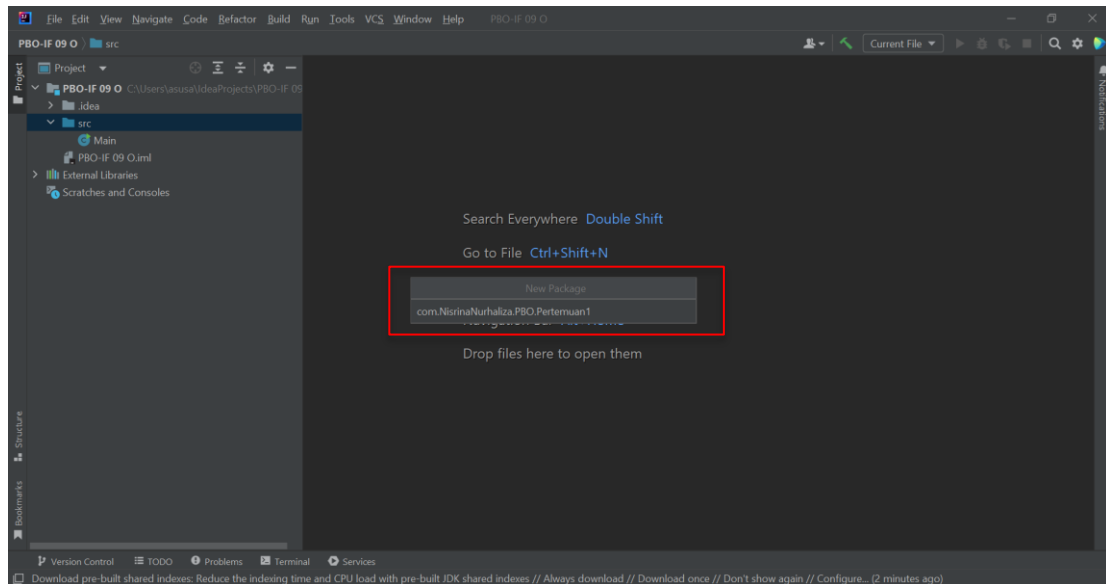
17. Setelah itu isi **nama project** dan pilih **java** pada bagian language, kemudian klik **Create**.



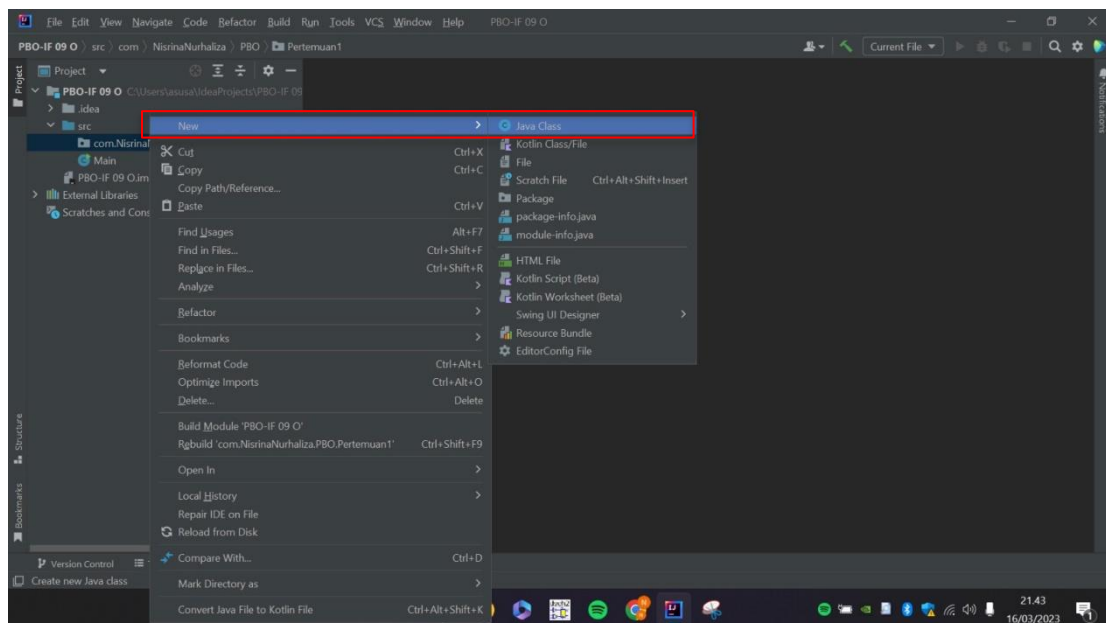
18. Setelah berhasil membuat project baru maka akan muncul tampilan workspace baru seperti pada gambar dibawah ini. Untuk mengelompokkan dan memudahkan manajemen program maka kita akan membuat satu package baru dengan klik kanan pada bagian folder *src* kemudian pilih *New* → *Package*.



19. Selanjutnya pada tampilan ini akan mengisi ***nama package*** pada bagian ***New Package***, jika sudah mengisi selanjutnya tekan enter.

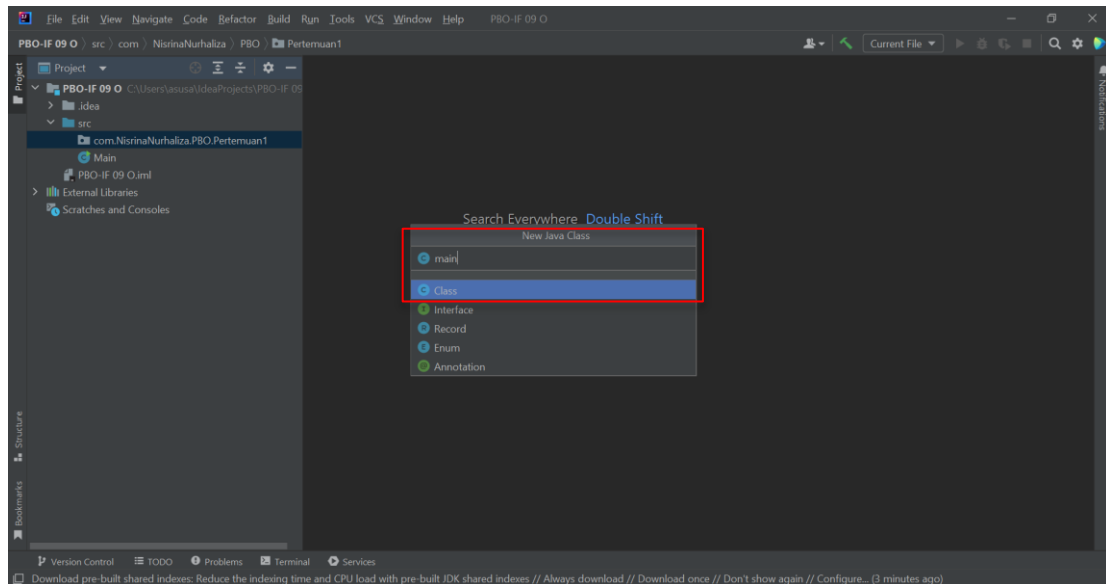


20. Selanjutnya kita akan membuat Java Class baru dengan klik kana pada ***Package*** yang sudah dibuat, kemudian pilih ***New → Java Class***.

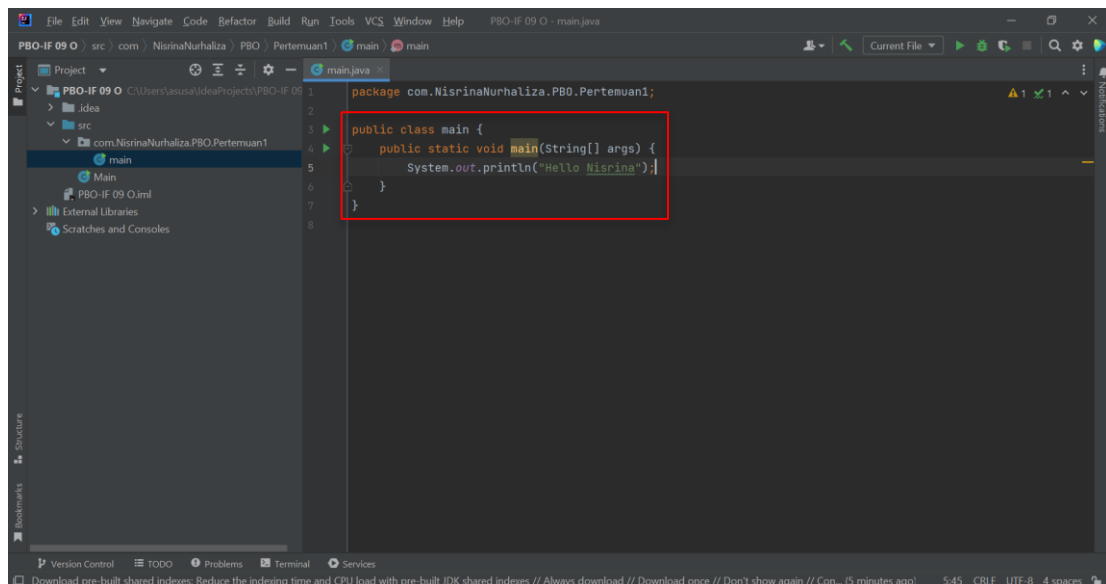




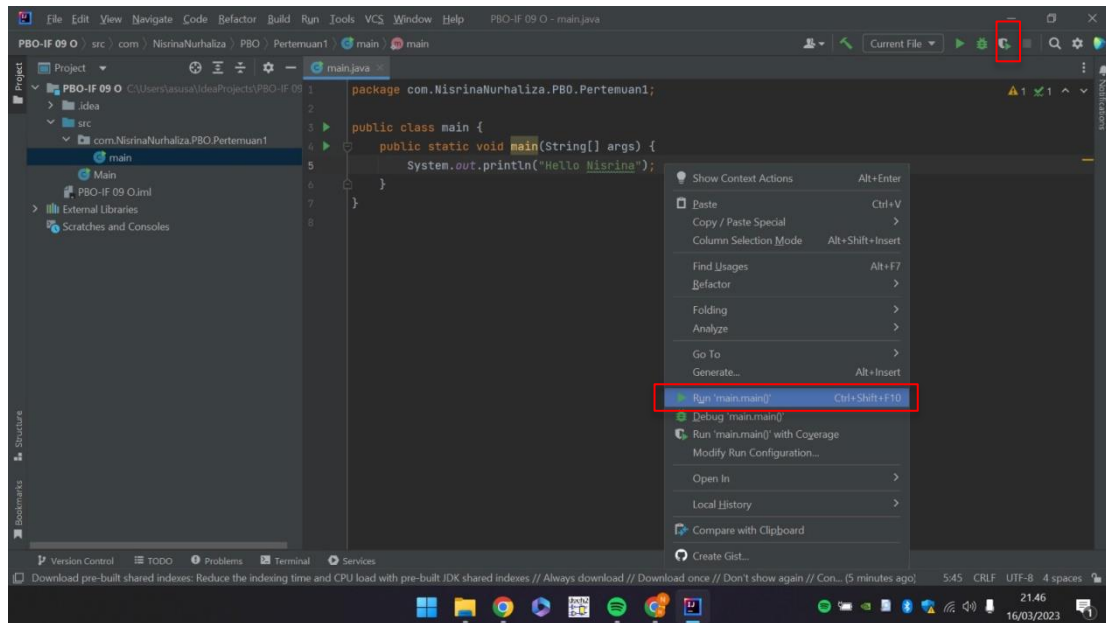
21. Pada tampilan berikut ini, kita akan mengisi nama **Java Class**, jika sudah mengisi selanjutnya tekan enter.



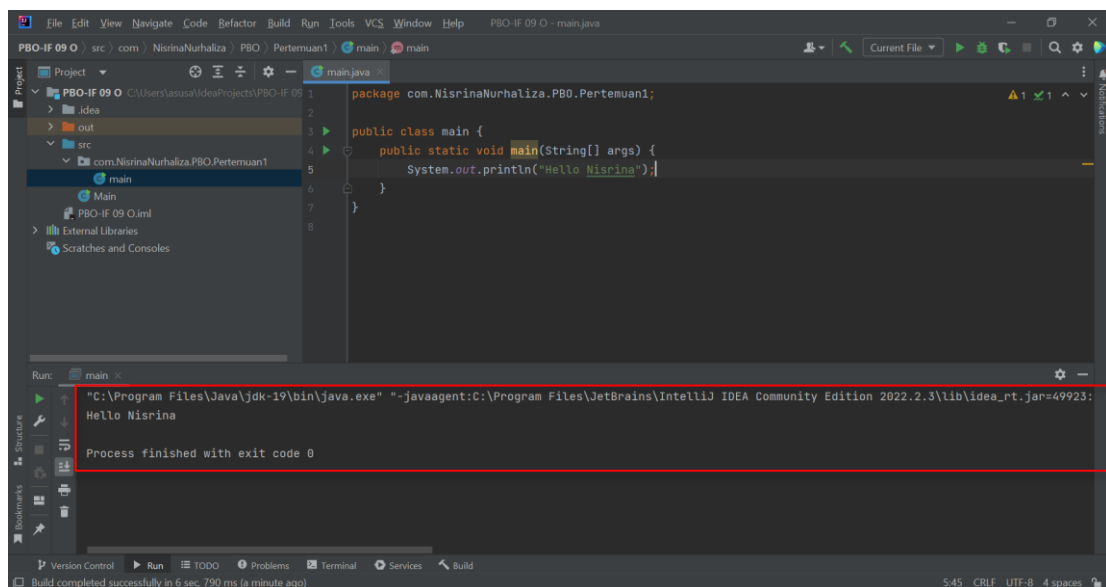
22. Selanjutnya di dalam file class main.java ketik kode seperti yang ada pada screenshot tampilan berikut ini.



23. Kita akan mencoba melakukan Running pada program dengan cara klik kanan, kemudian pilih **Run**. Dapat juga melakukan dengan klik tanda segitiga hijau pada bagian kanan atas.



24. Jika program berhasil di running dan tidak ada error, maka tampilan seperti pada screenshot yang ada dibawah ini.



## **BAB V**

### **KESIMPULAN**

Pada praktikum pertemuan pertama, diharapkan mahasiswa mampu melakukan instalasi JDK terlebih dahulu kemudian baru melakukan instalasi IntelliJ. JDK digunakan sebagai aplikasi pendukung dalam mengembangkan aplikasi berbasis java. IntelliJ IDEA berfungsi dalam menyediakan editor yang digunakan untuk mendukung berbagai penyelesaian kode, sebuah fitur yang memungkinkan untuk menulis kode dengan cepat.

Kelebihan utama dari bahasa pemrograman Java adalah bisa dijalankan di berbagai sistem operasi atau platform. Hal ini cocok dengan moto mereka: Write Once, Run Anywhere. Berarti cukup hanya dengan sekali pembuatan file Java, program bisa dijalankan di beberapa platform tanpa memerlukan perubahan. Contohnya, dapat dijalankan di Android, Linux, Windows, dan lain-lain. Alasan kenapa Java dapat dijalankan di mana saja, karena bahasa pemrograman ini memiliki sistem syntax atau kode pemrograman level tertinggi. Syntax akan disusun dengan Java Virtual Machine (JVM) menjadi kode numeric (bytecode) platform, sehingga aplikasi Java dapat dijalankan di berbagai perangkat.

Fungsi Java contohnya mendukung native method, bahasa yang digunakan sederhana, hanya focus pada objek, pengamanan yang cukup ketat, dapat dipakai di system operasi manapun, terdapat fitur GUI, menggunakan system *exception-handling*, terdapat Garbage Collector, memiliki perpustakaan yang lengkap dan tentunya bahasa Java merupakan penyempurnaan dari C++.