

**LAPORAN PRAKTIKUM  
PEMOGRAMAN BERORIENTASI OBJEK**

**MODUL I  
“PENGENALAN IDE”**



**Disusun oleh:  
Laras Wahyu Adiningsih  
21102214**

**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**

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

## BAB II

### DASAR TEORI

OOP adalah solusi untuk bisa membuat suatu program yang sesuai penerapannya dengan kehidupan sehari-hari. Bila pada umumnya pada *developer* akan merangkai deretan kode berdasarkan *function* dan *logic*, maka dengan OOP setiap *developer* bisa mengembangkan *software* yang terbagi menjadi beberapa objek khusus. OOP adalah suatu metode pemrograman yang lebih berorientasi pada objek. Berbagai program yang sudah ada adalah kombinasi dari berbagai komponen kecil yang sebelumnya sudah ada. Berbagai objek yang saling berkaitan akan disusun ke dalam satu kelompok yang dikenal dengan *class*. Berbagai objek tersebut nantinya akan saling melakukan interaksi agar bisa menyelesaikan masalah program yang rumit. Tujuan OOP adalah agar bisa mengatasi kelemahan pendekatan pemrograman konvensional.

JDK (Java Development Kit) adalah lingkungan pengembangan perangkat lunak lintas platform yang menawarkan kumpulan alat dan pustaka yang diperlukan untuk mengembangkan aplikasi dan applet perangkat lunak berbasis *Java*. JDK merupakan paket inti yang digunakan di *Java*, bersama dengan JVM (Java Virtual Machine) dan JRE (Java Runtime Environment). Perbedaannya yaitu bahwa JDK adalah paket alat untuk mengembangkan perangkat lunak berbasis *Java*, sedangkan JRE adalah paket alat untuk menjalankan kode *Java*. JDK memiliki Java Virtual Machine (JVM) personal dan beberapa sumber daya lain yang diperlukan untuk pengembangan Aplikasi *Java*.

IntelliJ IDEA adalah sebuah IDE dengan pengembangan dari JetBrains dan terdiri atas 2 versi. Pertama, versi dengan sebutan Community Edition menggunakan lisensi Apache 2 Licensed yang dapat digunakan dengan gratis oleh Anda. Kedua, versi dengan jenis Ultimate Edition yang ditujukan untuk penggunaan secara komersial. IntelliJ IDEA yang penulisannya menggunakan *Java* serta *Kotlin* tersebut telah rilis dengan versi pertamanya di bulan Januari 2011. Dengan perilisannya ini langsung menjadikan IDE pertama dengan tujuan mengembangkan aplikasi yang berbasis *Java* dan mendukung adanya navigasi kode di tingkat lanjut serta kemampuan dalam refactoring yang sudah saling terintegrasi. Secara umum, adanya IDE bisa menyediakan banyak fitur semisal text editor yang bisa digunakan dalam menulis beberapa kode. Tools yang ditujukan untuk mengotomatisasi adanya proses build pada program yang telah Anda buat serta debugger bisa membantu Anda dalam mendeteksi juga memperbaiki banyak kesalahan yang ada di dalam program.

Sumber :

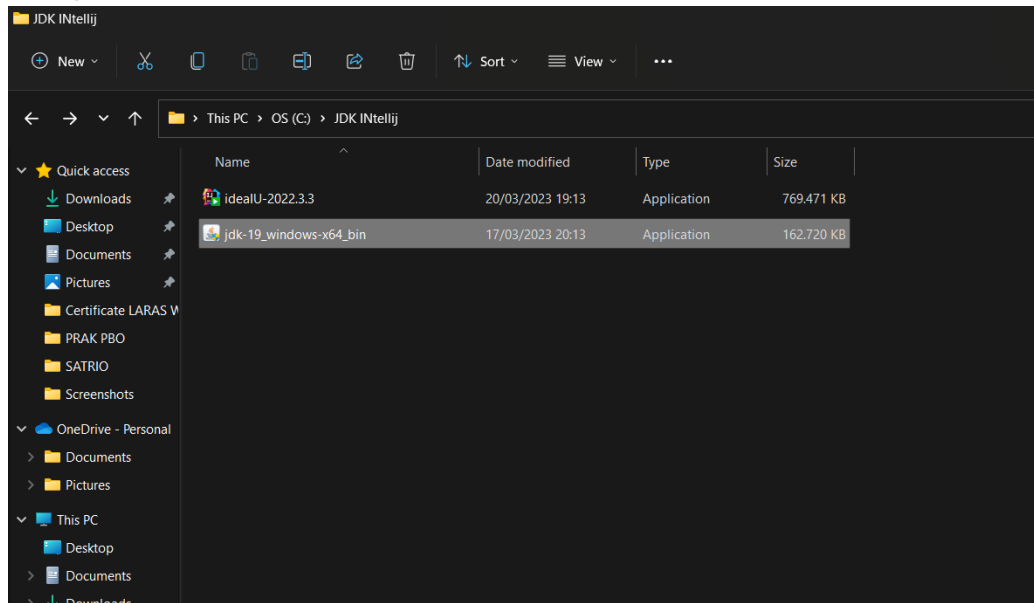
- [1] <https://accurate.id/teknologi/oop-adalah/> Diakses pada Tanggal 20 Maret 2023
- [2] <https://adammuiz.com/java-development-kit/> Diakses pada Tanggal 20 Maret 2023
- [3] <https://idcloudhost.com/mengenal-apa-itu-intellij-idea-keunggulan-kelemahan-dan-cara-kerjanya/> Diakses pada Tanggal 20 Maret 2023

## BAB III

### GUIDED

#### ▪ Langkah Praktikum

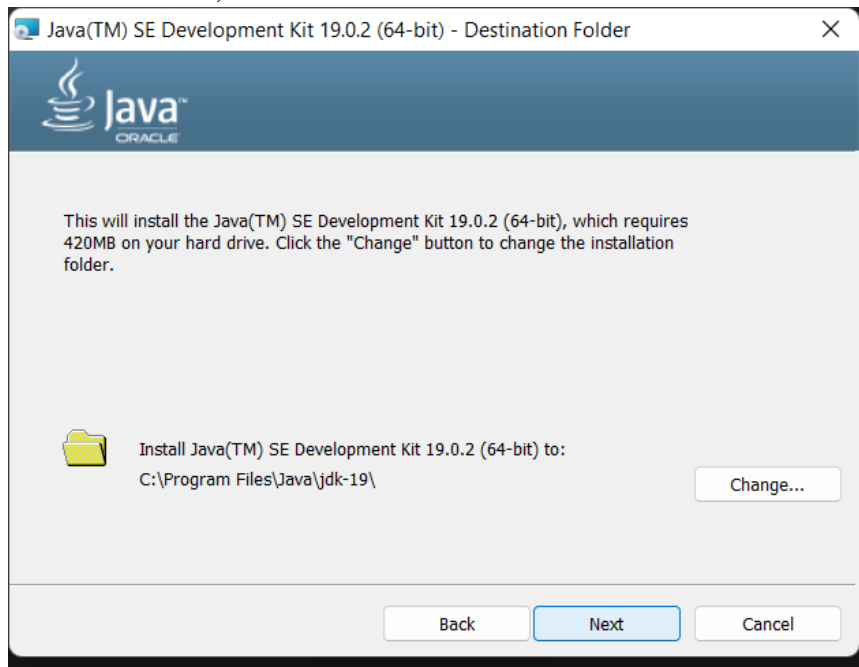
1. Langkah pertama kita install aplikasi JDK terlebih dahulu dengan meng-klik 2 kali .



2. Setelah meng-klik 2 kali, lalu akan muncul tampilan seperti dibawah ini, kemudian klik Next.



3. Selanjutnya disini kita memilih direktori penginstalan JDK, apabila telah memilih, lalu klik **Next**.



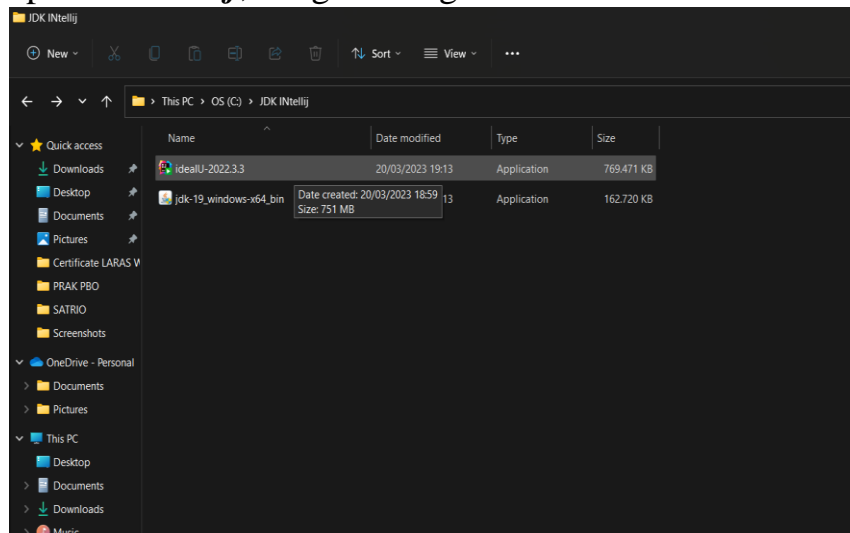
4. Lalu menunggu Loading terlebih dahulu.



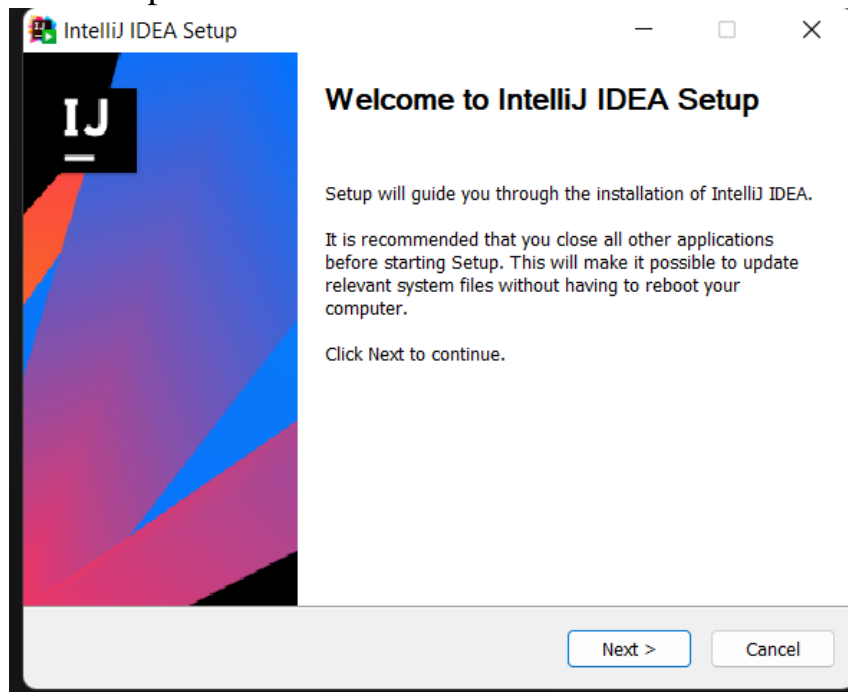
5. Selesai menginstal aplikasi JDK, lalu kita klik *Close*.



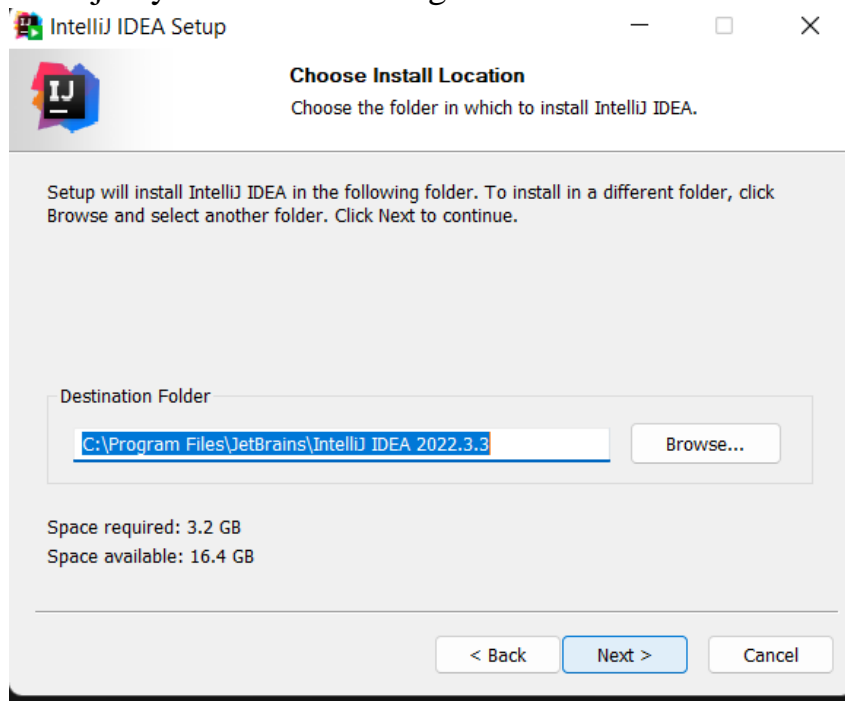
6. Setelah menginstall aplikasi JDK, selanjutnya kita akan menginstall aplikasi *IntelliJ*, dengan meng-klik 2 kali



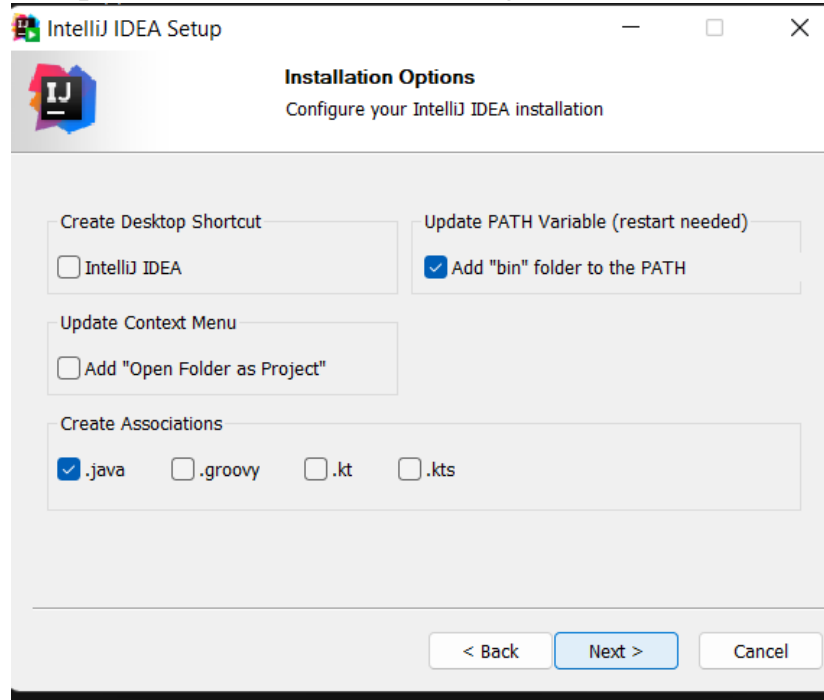
7. Pada tampilan ini kita klik *Next*



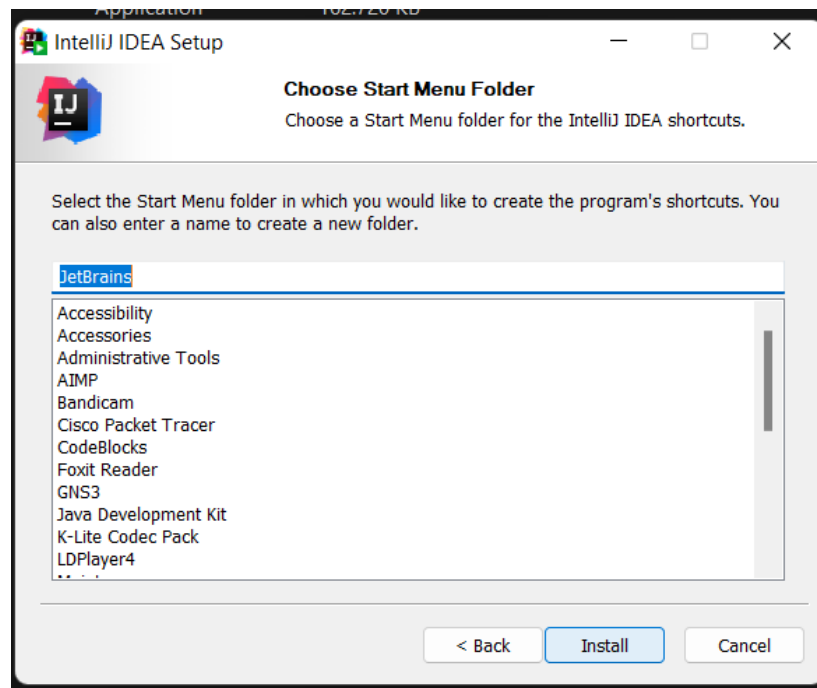
8. Selanjutnya kita klik next lagi



9. Pada bagian tersebut klik pada bagian **Update PATH Variable** dan klik pada Create Associations bagian **Java**, kemudian klik **Next**.

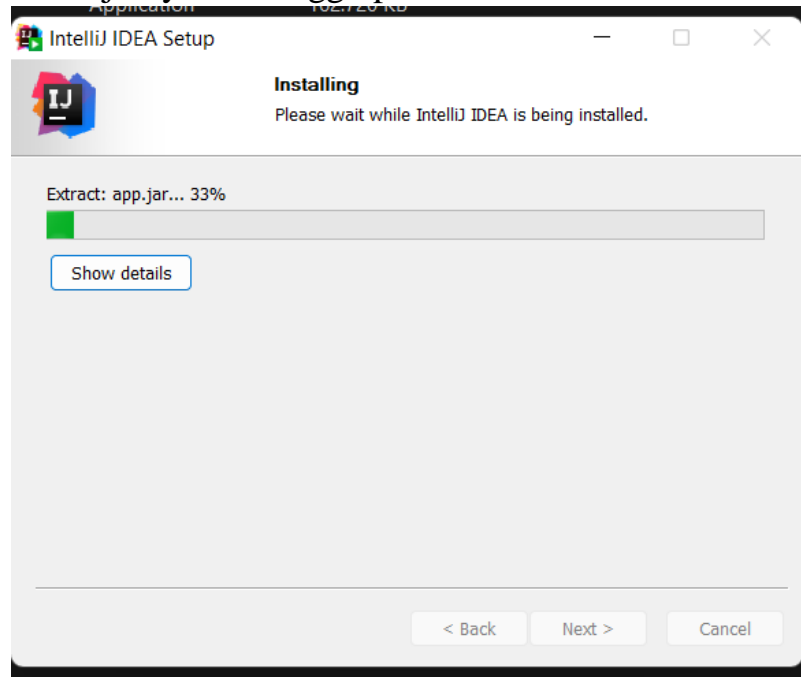


10. Lalu kita klik install

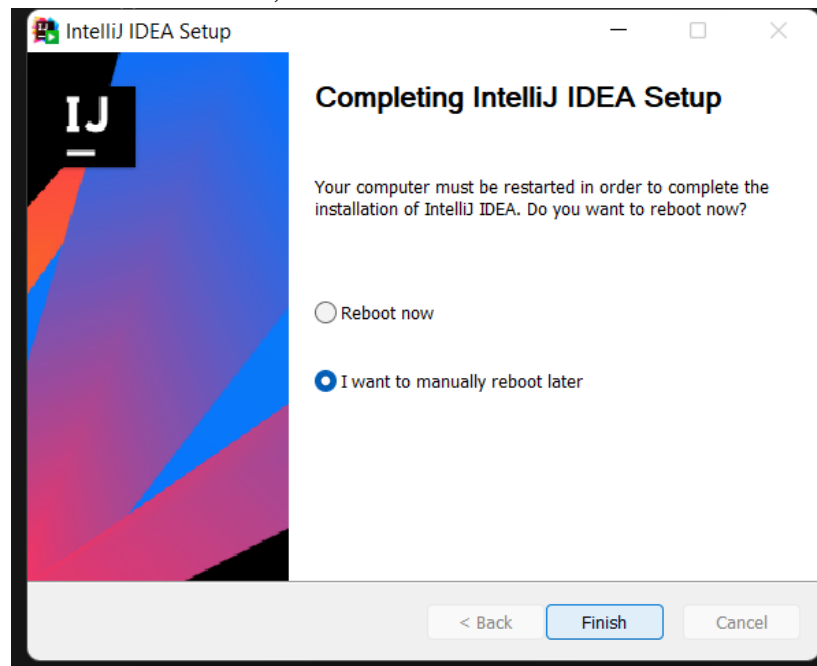




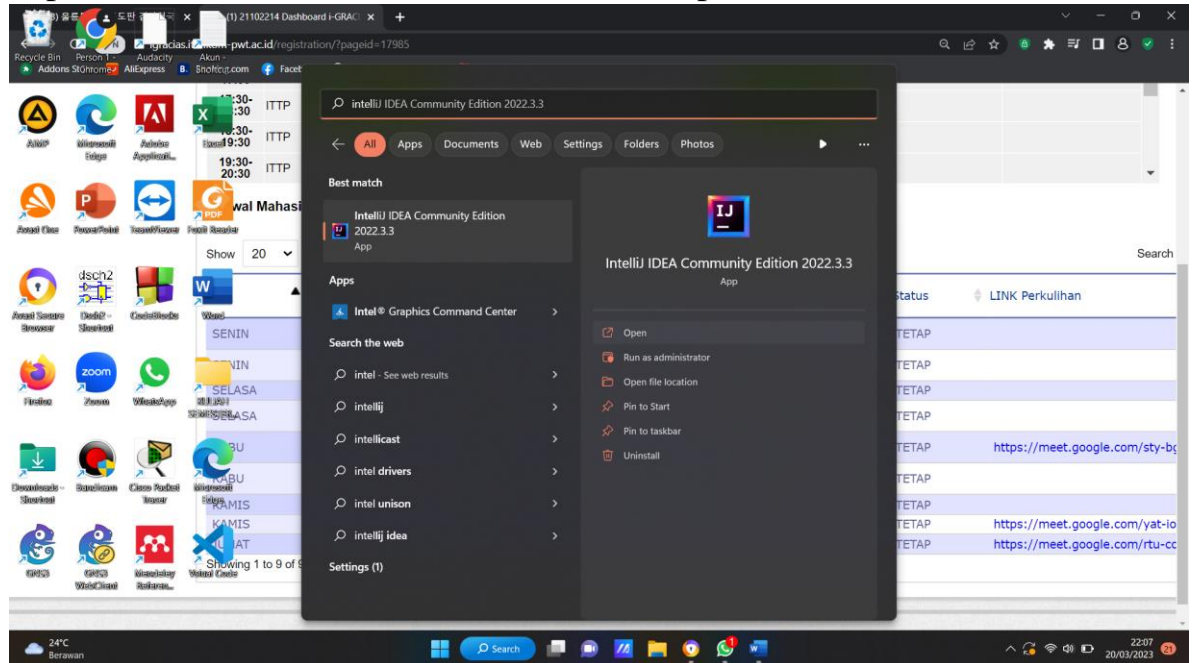
## 11. Selanjutnya menunggu proses instalasi



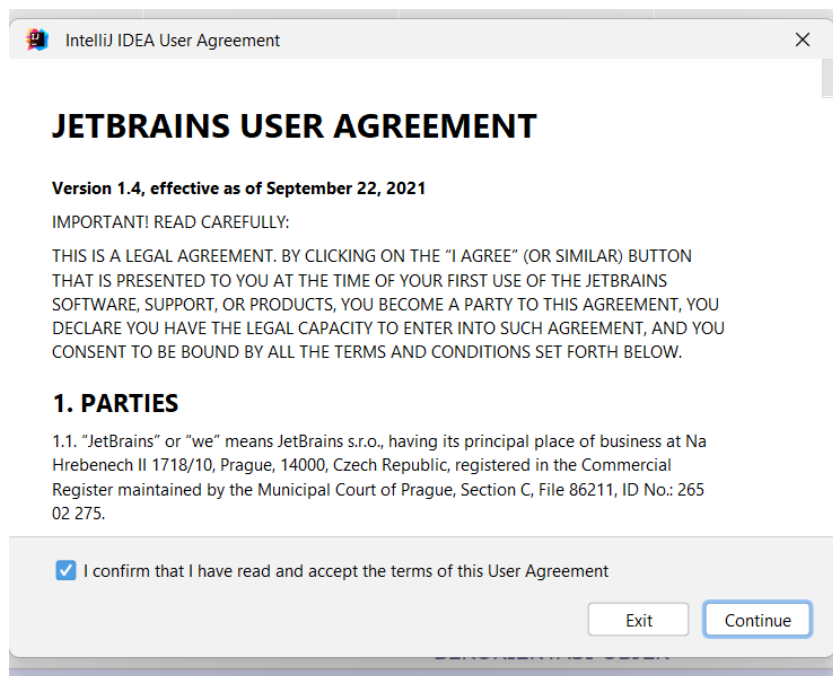
## 12. Jika telah selesai, lalu kita klik finish



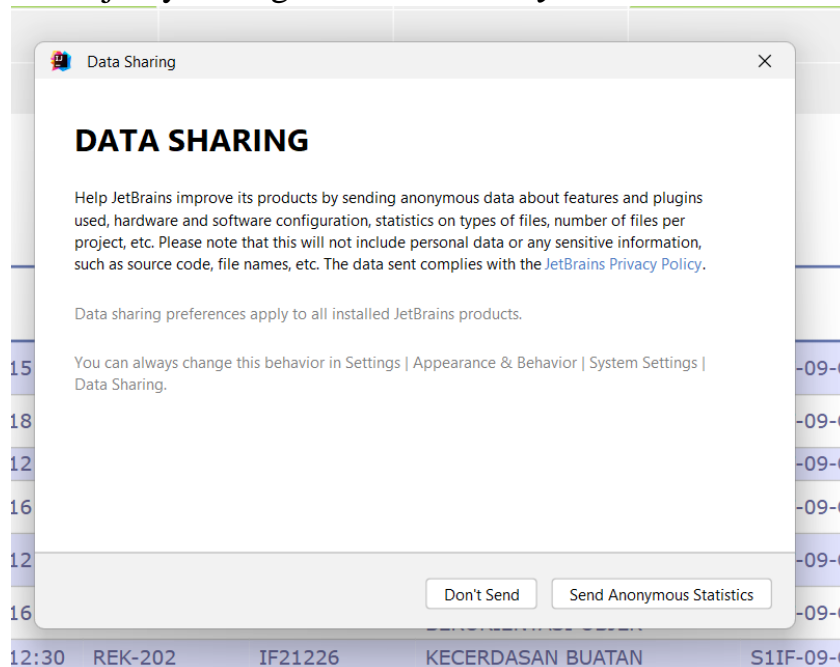
13. Apabila telah terinstall, kemudian kita buka aplikasi tersebut



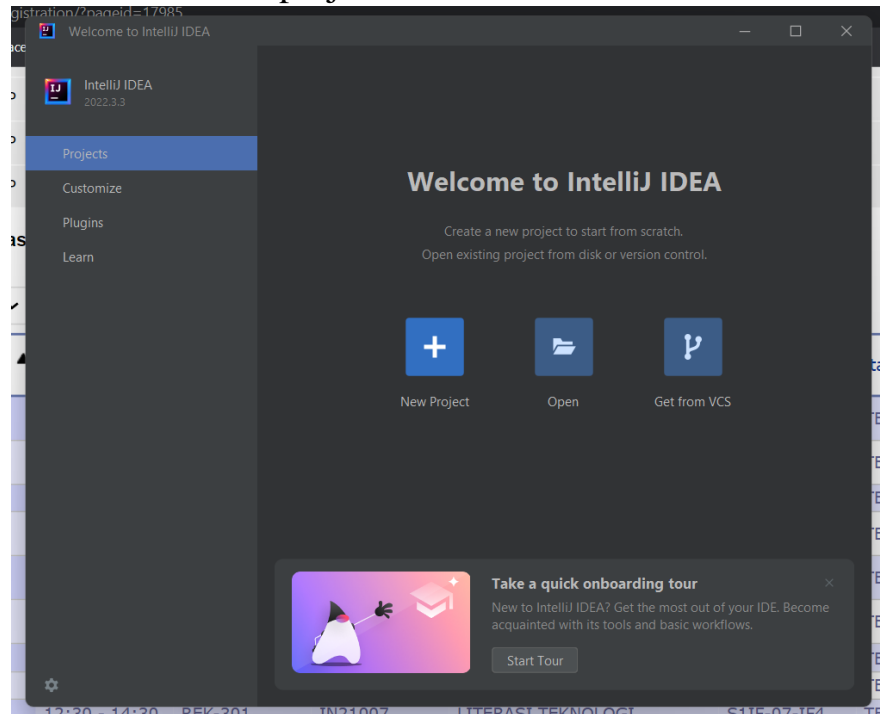
14. Pada tampilan ini, centang **checkbox** yang ada dibawah ini, kemudian klik **Continue**.



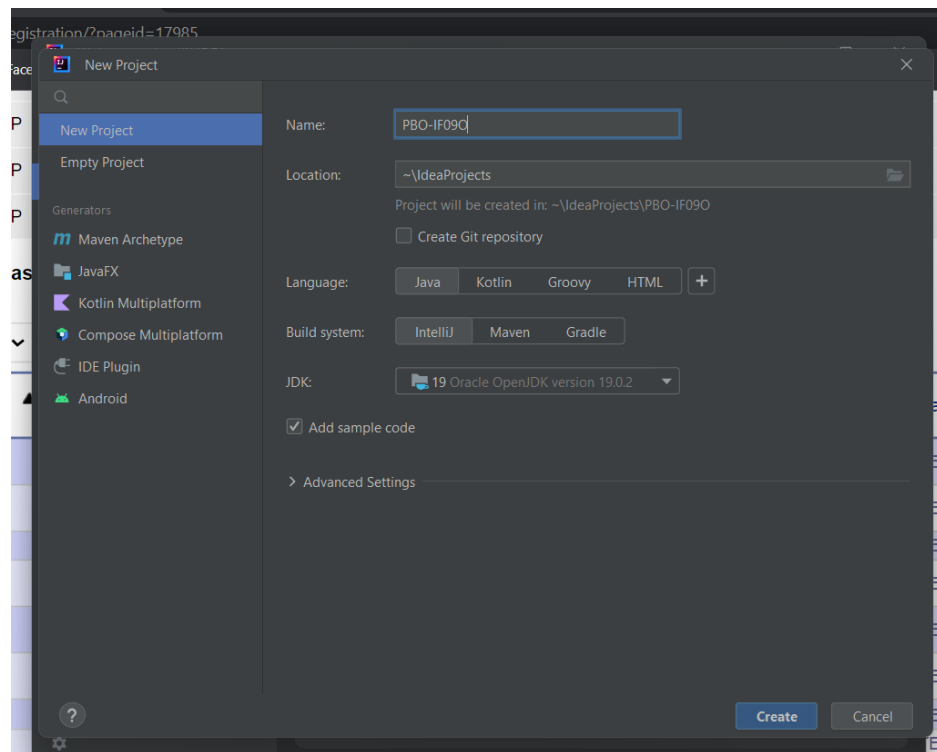
15. Selanjutnya meng-klik *Send Anonymous Statistics*.



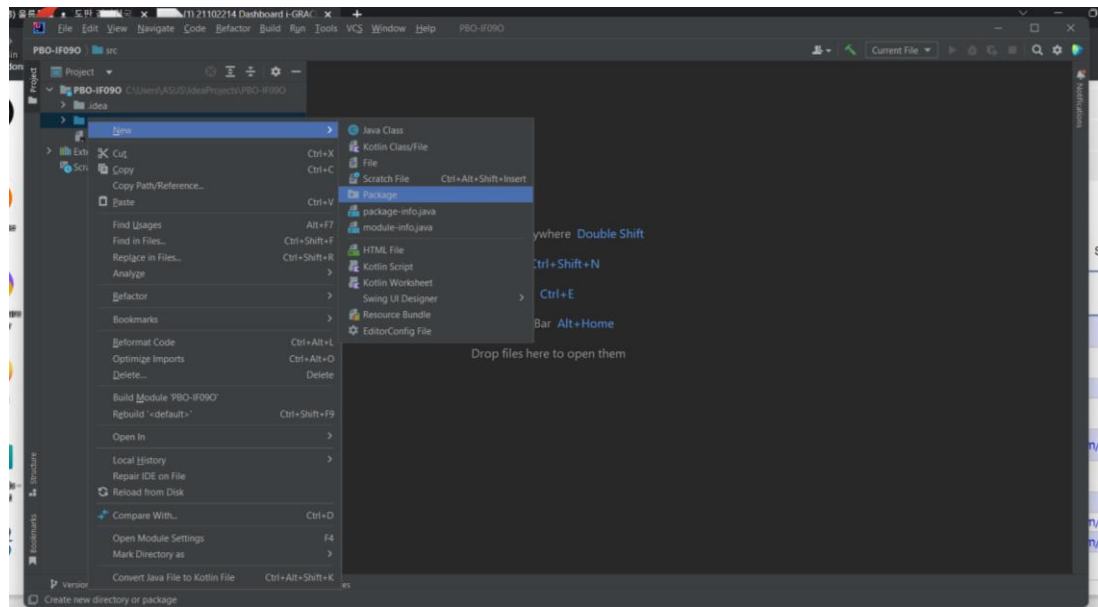
16. Selanjutnya pada tampilan awal aplikasi IntelliJ, klik *New Project* untuk menambah project baru.



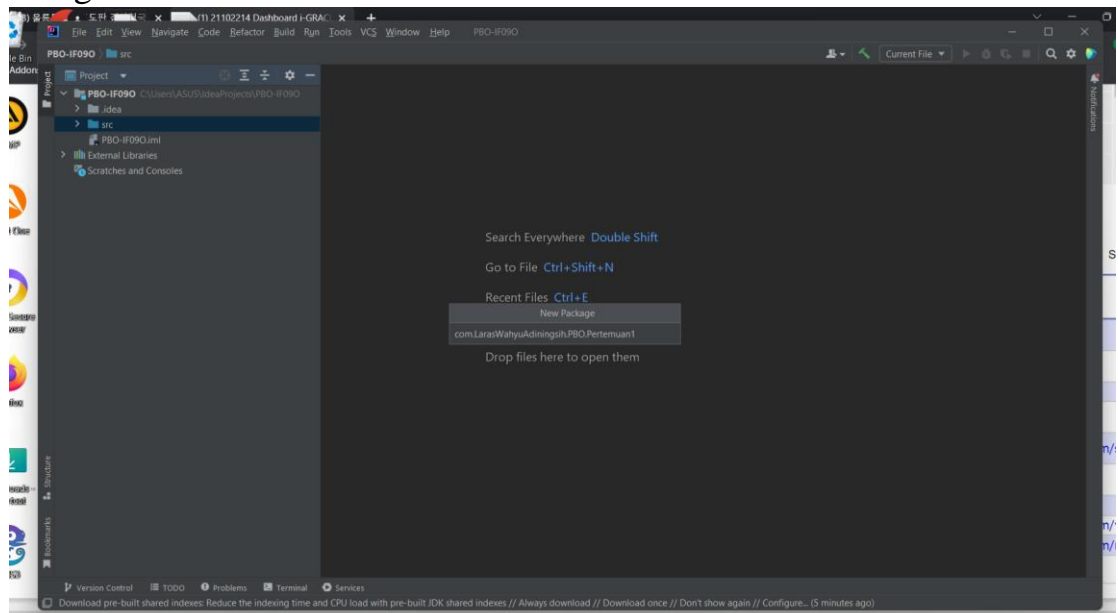
17. Setelah itu isi ***nama project*** dan pilih ***java*** pada bagian language, lalu klik ***Create***.



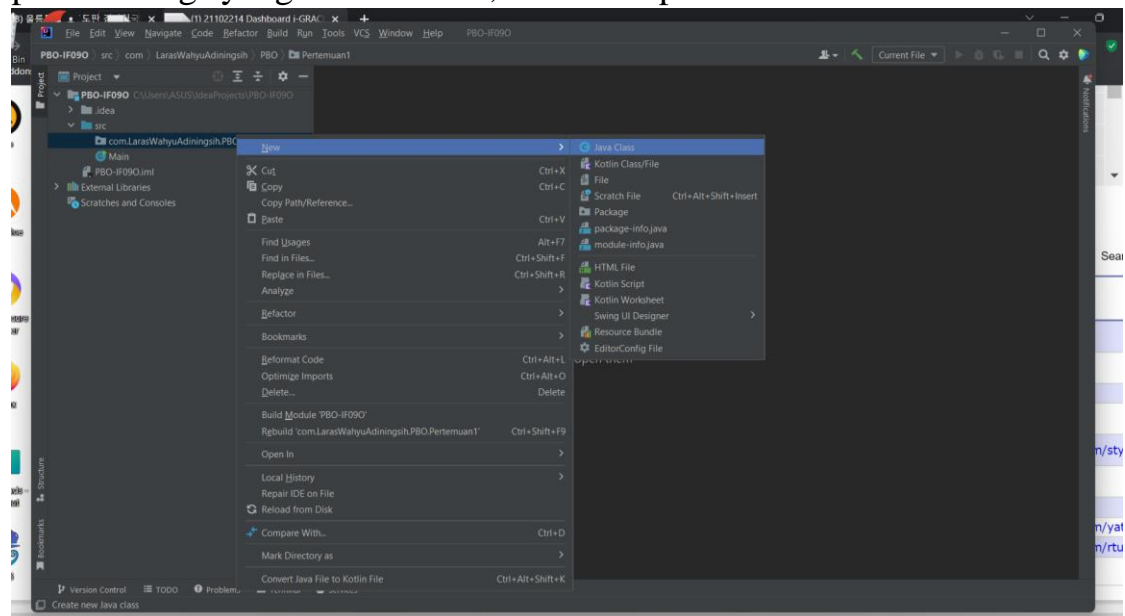
18. Setelah berhasil membuat project baru maka akan muncul tampilan workspace baru seperti pada gambar dibawah ini. Untuk mengelompokkan dan memudahkan manajemen program maka kita akan membuat satu package baru dengan klik kanan pada bagian folder src kemudian pilih new lalu package.



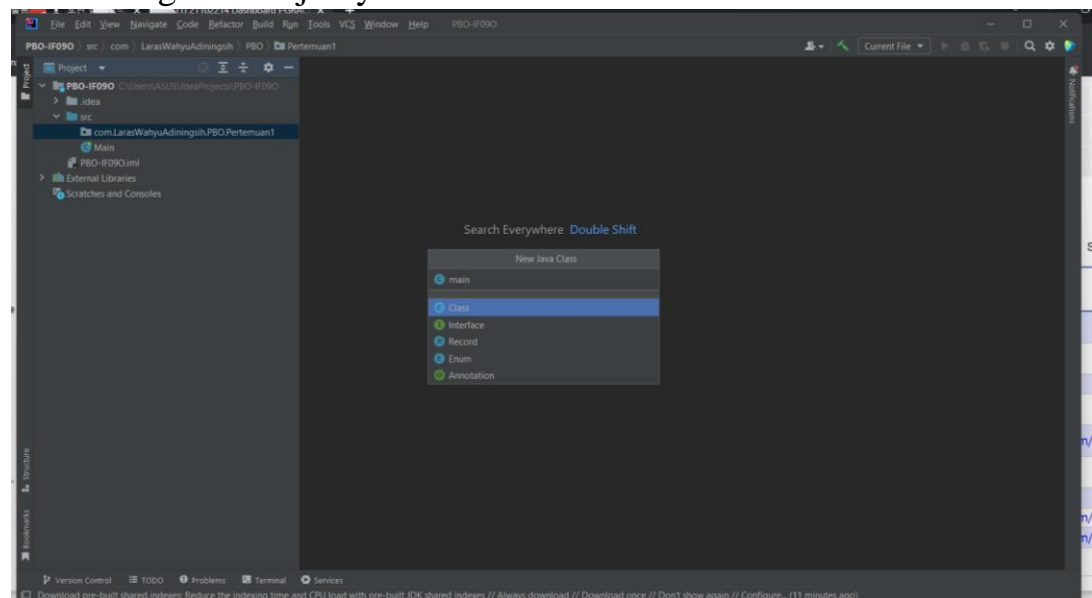
19. Kemudian pada tampilan ini akan mengisi *nama package*, jika sudah mengisi lalu tekan enter.



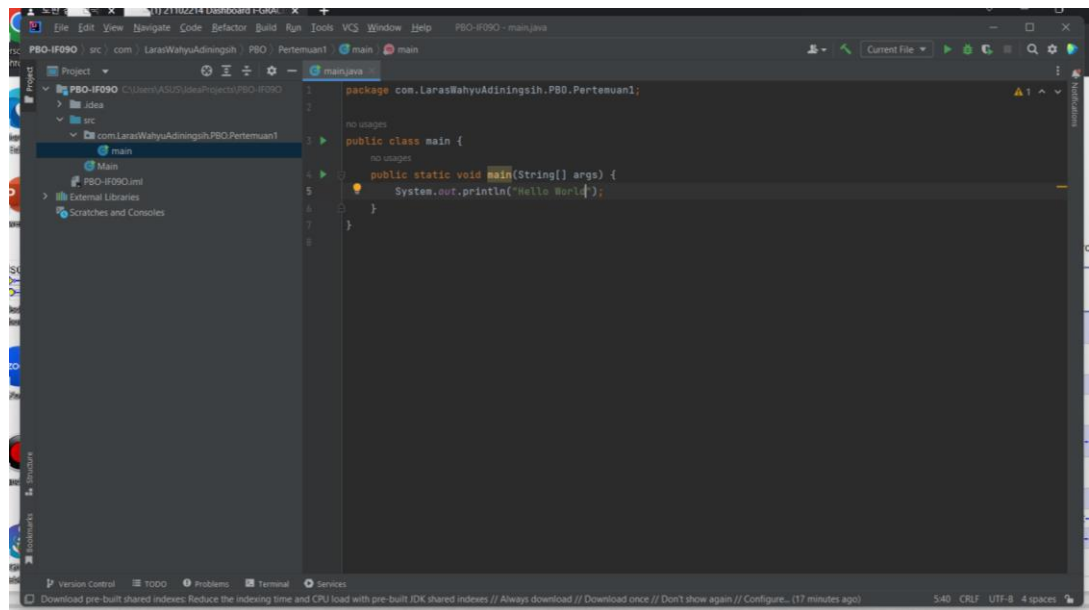
20. Kemudian kita akan membuat Java Class baru dengan klik kanan pada Package yang sudah dibuat, kemudian pilih **New** lalu **Java Class**.



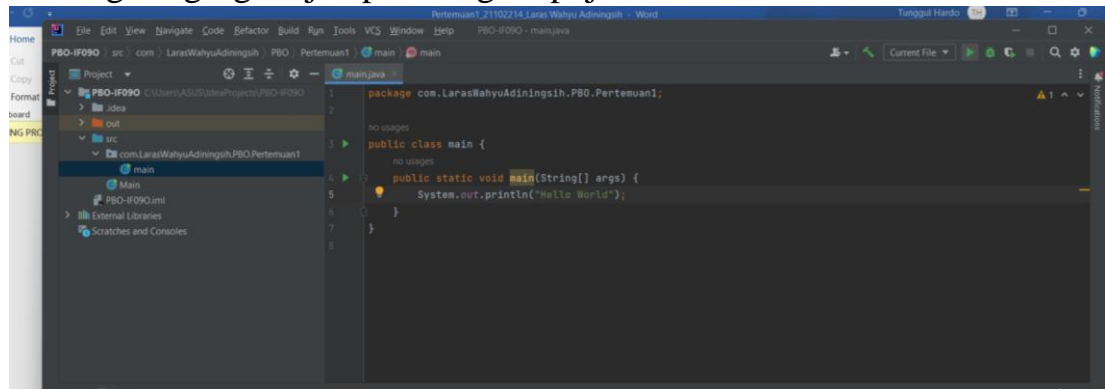
21. Kemudian di tampilan ini akan mengisi **nama Java Class**, apabila telah mengisi selanjutnya menekan enter.



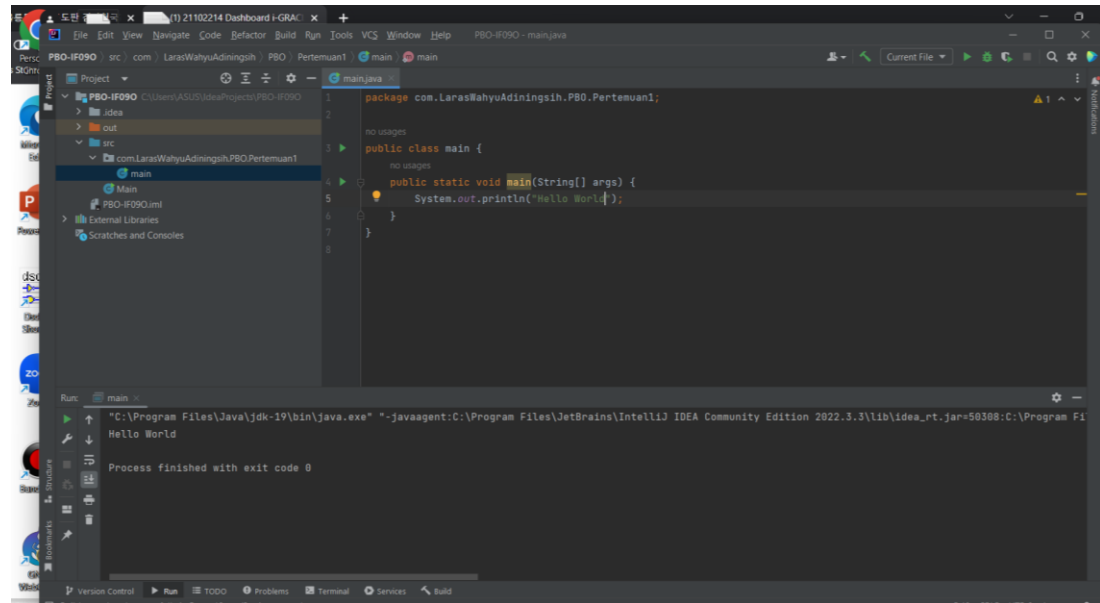
22. Lalu pada file class main java ketik kode seperti screenshot yang dibawah ini.



23. Selanjutnya kita melakukan Running pada program dengan meng-klik logo segitiga hijau pada bagian pojok kanan atas



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





## **BAB IV**

### **KESIMPULAN**

Pada praktikum kali ini kita diajarkan tentang pengenalan IDE. Mahasiswa diminta untuk melakukan instalasi JDK dan IntelliJ. JDK adalah lapisan perangkat lunak di atas JRE yang berisi kompilator, debugger, dan alat lain yang biasa ditemukan di lingkungan pengembangan perangkat lunak apa pun. IntelliJ IDEA adalah environment pemrograman khusus atau yang biasa disebut integrated development environment (IDE) yang sebagian besar dibuat untuk Bahasa pemrograman Java.

Pemrograman Berorientasi Objek (PBO) atau Object Oriented Programming (OOP) merupakan salah satu paradigma pemrograman yang berorientasi terhadap objek nyata daripada ide atau gagasan umum yang abstrak. Sampe saat ini, paradigma OOP adalah paradigma pemrograman yang paling banyak diaplikasikan, disusul oleh Pemrograman Terstruktur dan Functional Programming karena paradigma ini memang menyajikan banyak keunggulan terutama dalam mengembangkan perangkat lunak dan aplikasi yang dibutuhkan oleh industri, baik dari segi fitur, maupun jangka waktu pengerjaan. Selain itu, bahasa pemrograman yang dapat menyokong paradigma OOP ini amatlah banyak dan memiliki pengguna yang tidak sedikit pula. Beberapa bahasa pemrograman yang mendukung paradigma OOP meliputi bahasa Java, PHP, Python, C++, dll.

Java adalah bahasa pemrograman untuk berbagai tujuan (general purpose), bahasa pemrograman yang concurrent, berbasis kelas, dan berorientasi objek, yang dirancang secara khusus untuk memiliki sesedikit mungkin ketergantungan dalam penerapannya.