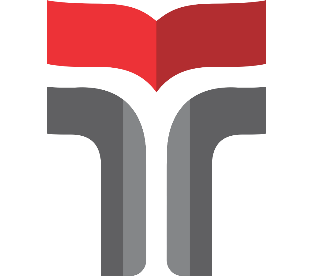
**LAPORAN PRAKTIKUM**

**MODUL VI**

**RELASI ANTAR KELAS**



**Disusun Oleh**

Firmansyah Nuralif Rohman

18102050

**Dosen**

Trihastuti Yuniati, S.Kom., M.T.

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI DAN INFORMATIKA**

**INSTITUT TEKNOLOGI TELKOM**

**PURWOKERTO**

**2020**

1. TUJUAN
2. Mahasiswa mampu memahami tentang konsep relasi antar kelas.
3. Mahasiswa mampu menerapkan konsep relasi antar kelas pada pemrograman berorientasi objek.
4. DASAR TEORI

Pemrograman berorientasi objek (PBO) mengambil realita masalah dalam kehidupan sehari-hari. Dalam PBO sering terjadi relasi antara satu objek dengan objek yang lainnya. Ada beberapa relasi yang mungkin terjadi antara suatu kelas dengan kelas yang lain, yaitu:

1. **Inheritance / Pewarisan**

Merupakan konsep pewarisan dari kelas induk ke kelas anak. Sudah dijelaskan di modul sebelumnya

1. **Exception**

Agregasi adalah relasi antara dua buah objek dengan mengatakan bahwa suatu objek memiliki atau mengandung atau berisi objek yang lain. Misalnya, sebuah mobil memiliki mesin, roda, body; sebuah rumah memiliki dapur, kamar mandi, ruang makan, dsb. Apabila suatu objek tertentu tersusun atas objek-objek lain, maka objek tersebut dikatakan sebagai objek agregat atau objek komposit (agregate, composite object). Relasi ini sering disebut dengan relasi ‘has-a’ atau relasi ‘whole-part’. Dalam diagram UML, relasi agregat ini digambarkan dengan symbol diamond. Symbol ini menunjukkan adanya inklusi structural sebuah objek terhadap objek yang lain yang diwujudkan dengan cara membuat kelas agregat yang memiliki atribut yang bertipe kelas penyusun.

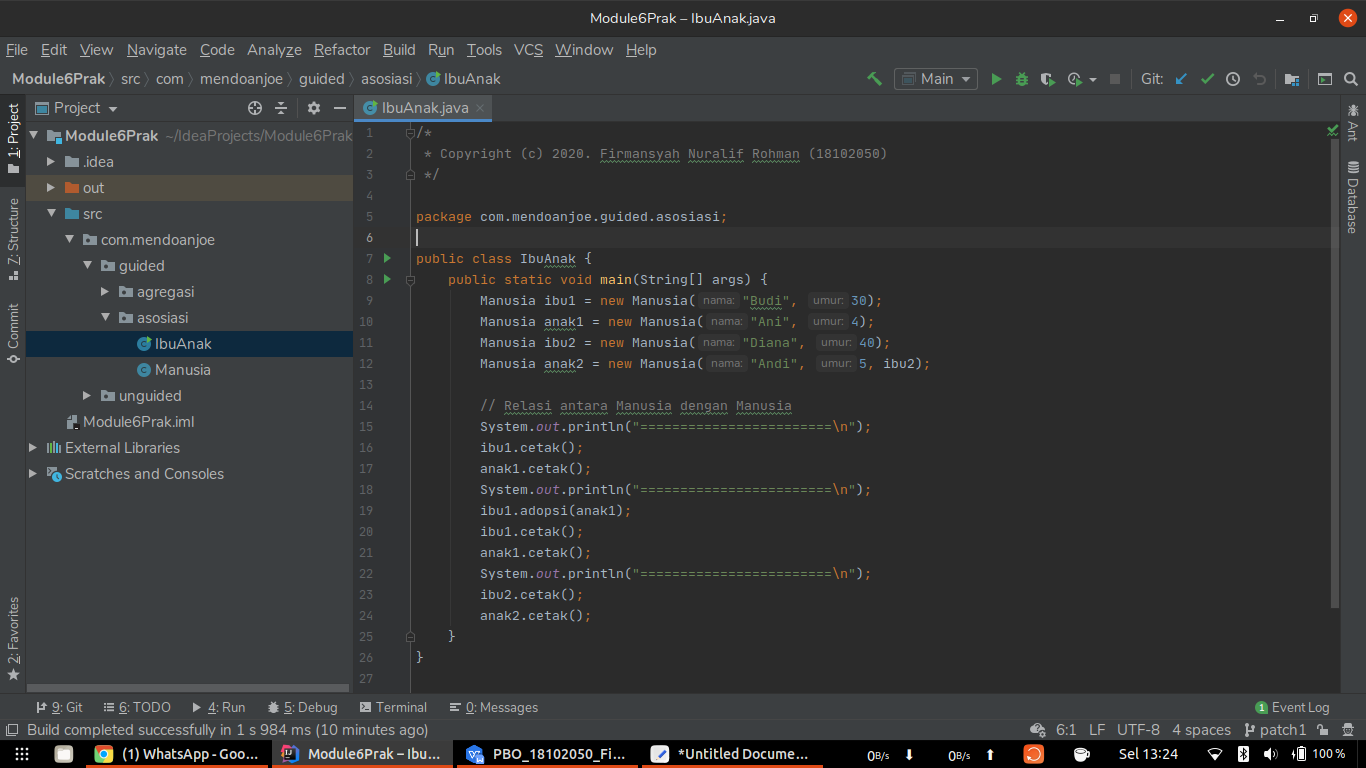
1. **Asosiasi**

Relasi asosiasi menyatakan suatu hubungan struktural antara beberapa objek yang menggambarkan objek dari suatu kelas dihubungkan dengan objek lain. Relasi ini menunjukkan variable dalam suatu kelas yang menyiman rujukan bertipe kelas lain.

1. **Dependency**

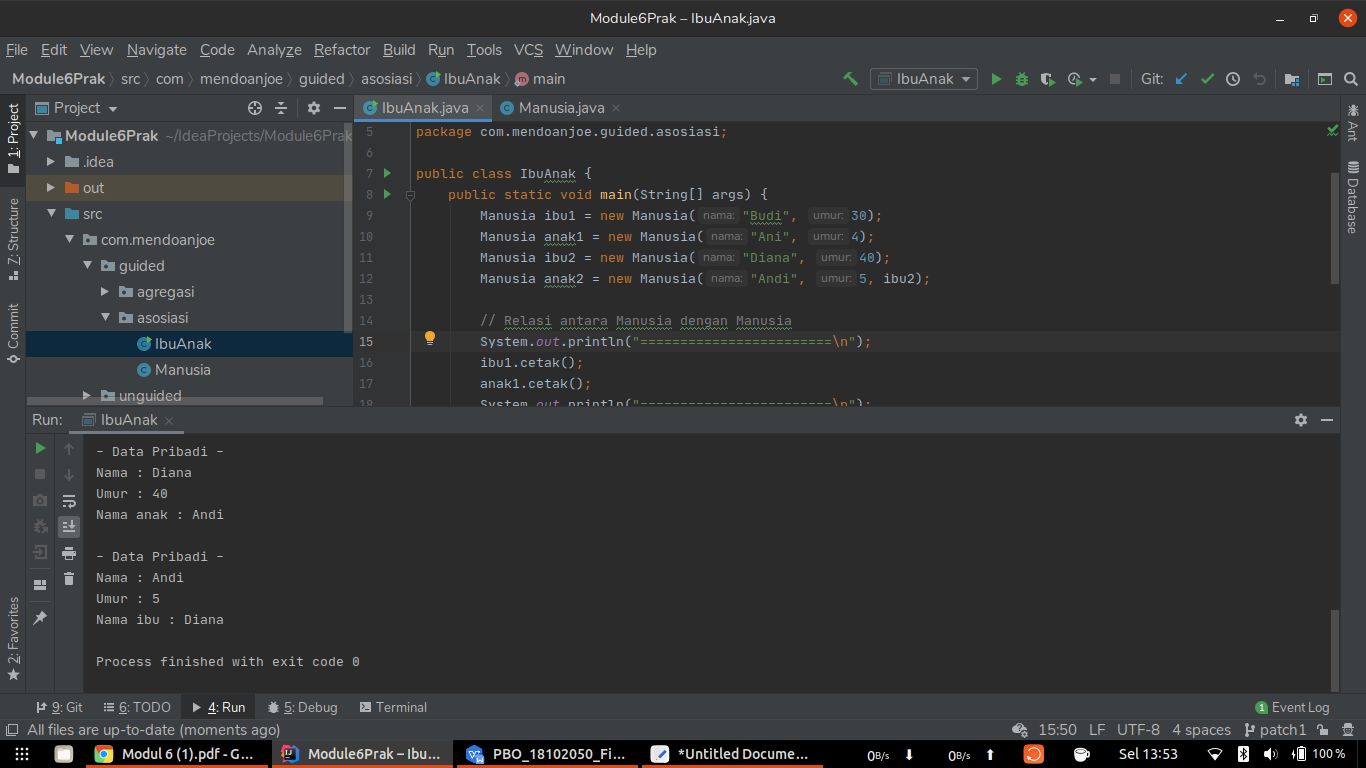
Relasi kebergantungan (dependency) menyatakan bahwa suatu kelas bergantung pada kelas yang lain, maka perubahan pada kelas yang menjadi ketergantungan dari kelas lain menyebabkan perubahan terhadap kelas yang tergantung tersebut. Misalnya, kelas tumbuhan membutuhkan kelas air. Jika kelas air mengalami perubahan, maka menyebabkan perubahan pada kelas tumbuhan.

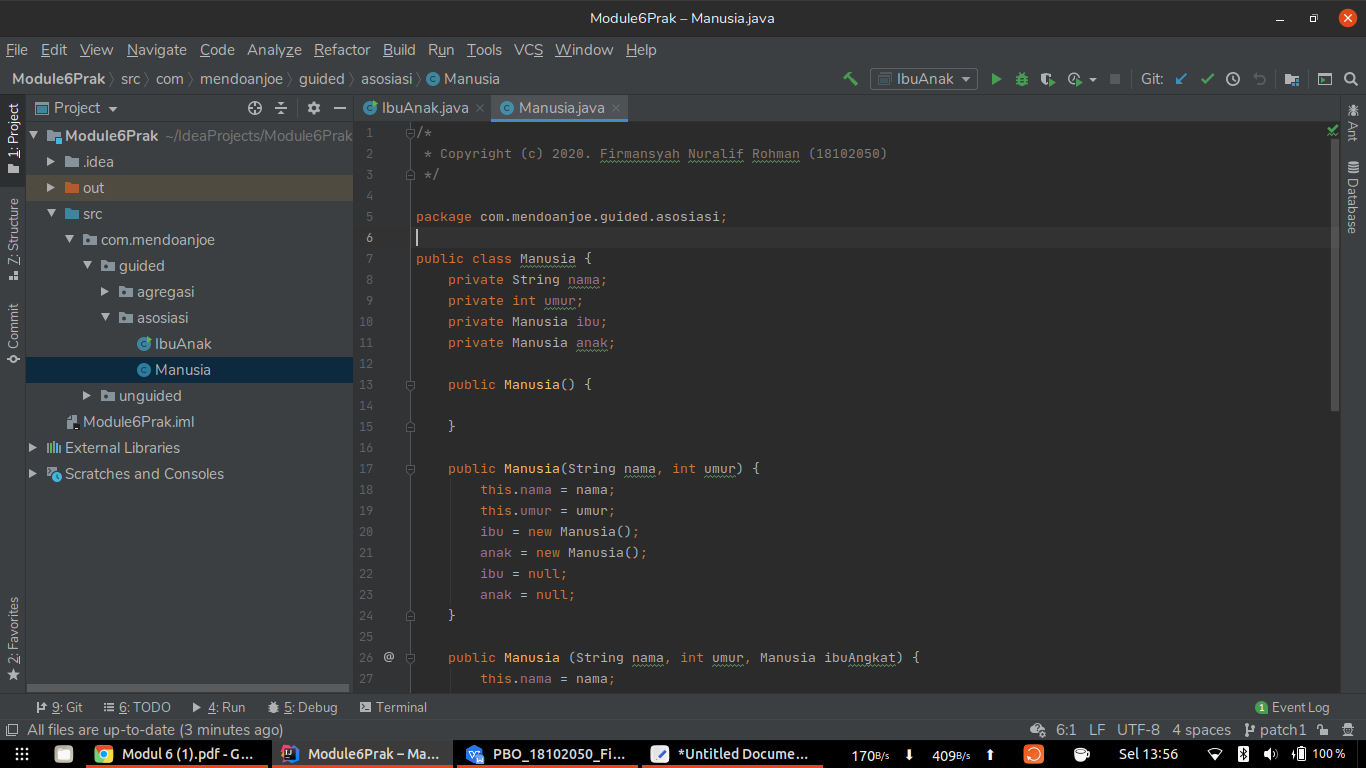
1. **PEMBAHASAN PROGRAM GUIDED**

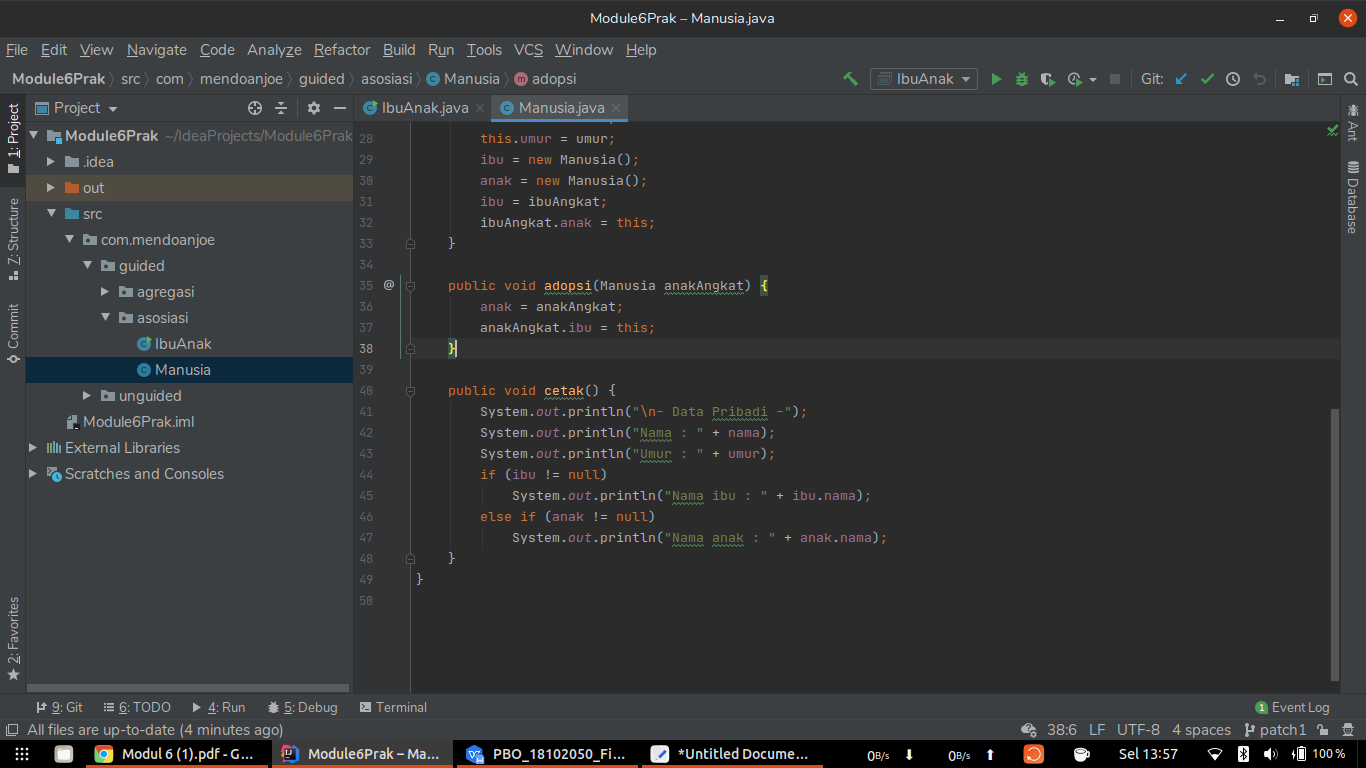


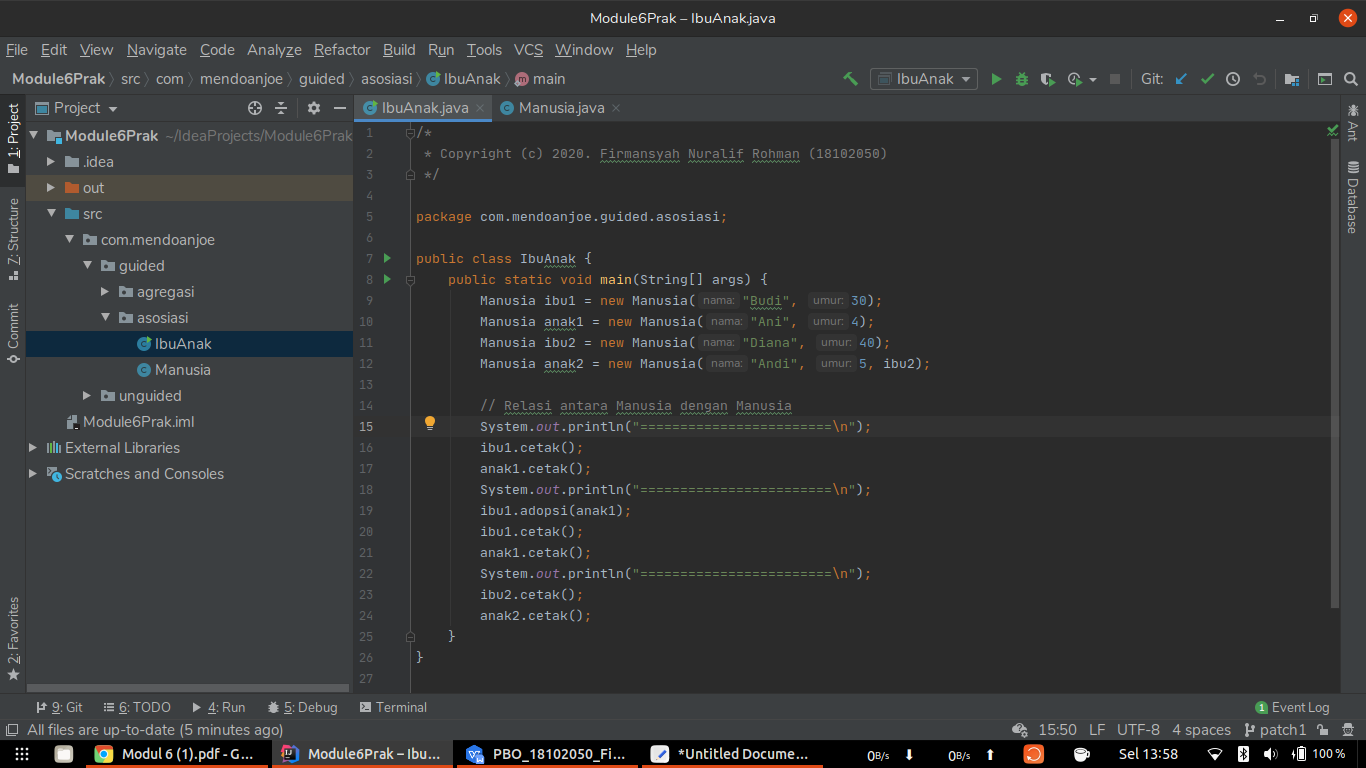
Terdapat dua contoh modul program yaitu program dengan fungsi agregasi dan asosiasi. Secara singkat asosiasi adalah model yang berkaitan dengan model itu sendiri untuk memangun model baru. Agregasi adalah model yang berkaitan dengan model lain untuk melengkapi model yang telah ada.

1. **Asosiasi**

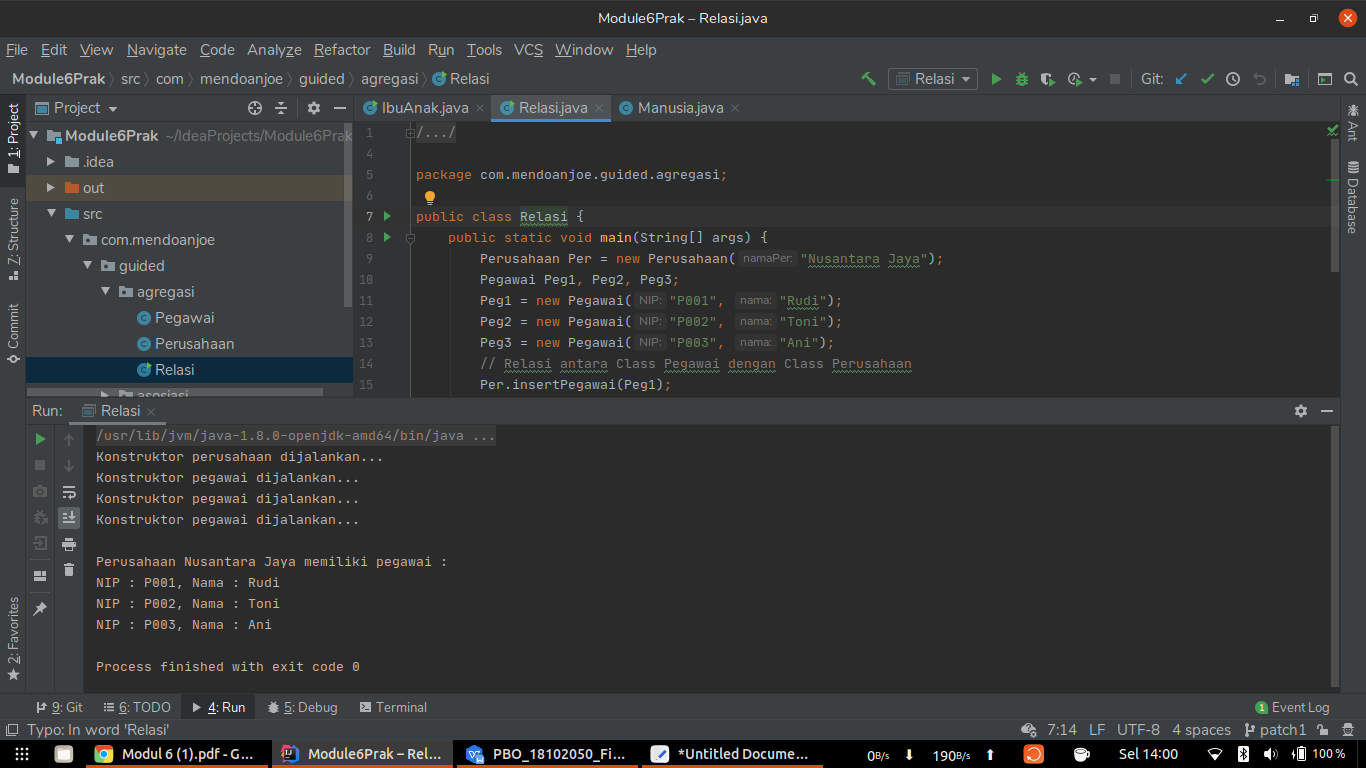
Disini terdapat 2 file java, yaitu kelas Manusia dan IbuAnak. Kelas Manusia digunakan sebagai model sedangkan kelas IbuAnak digunakan sebagai main program.



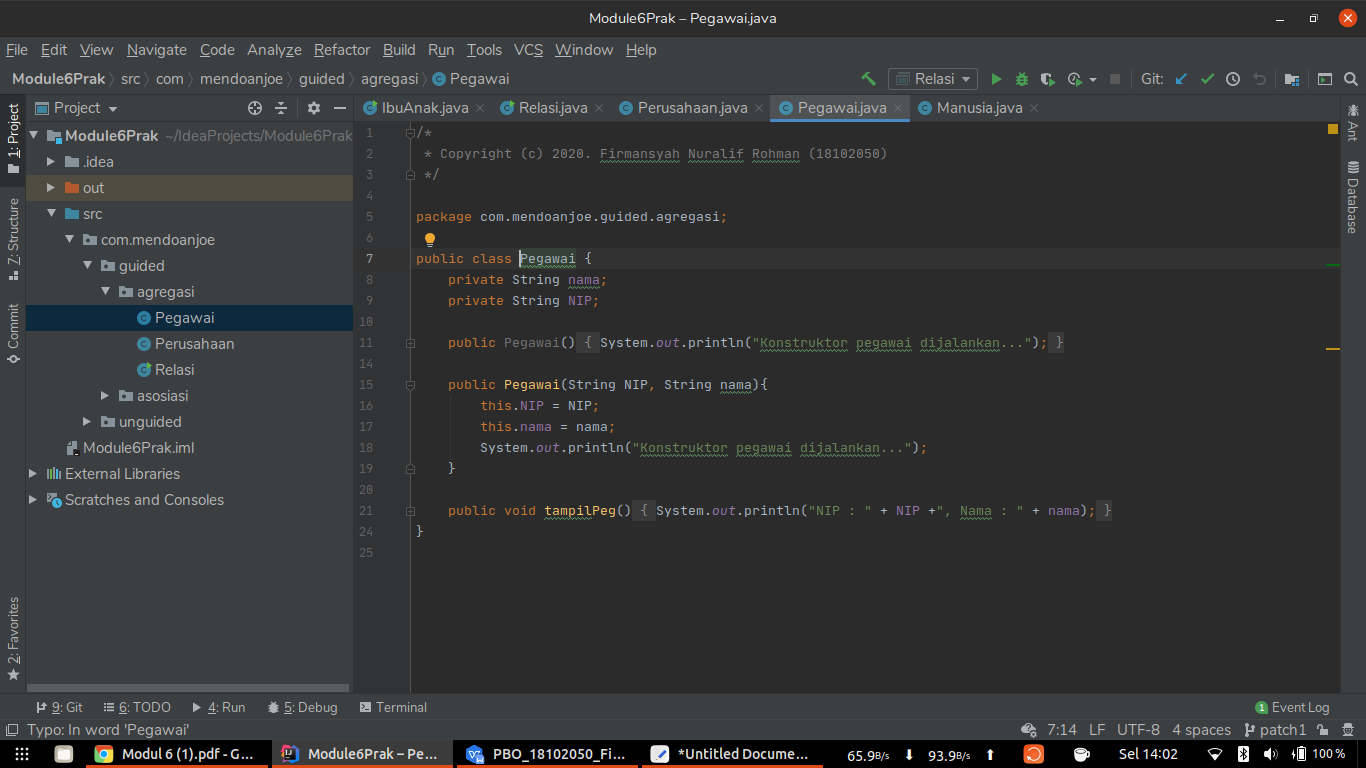


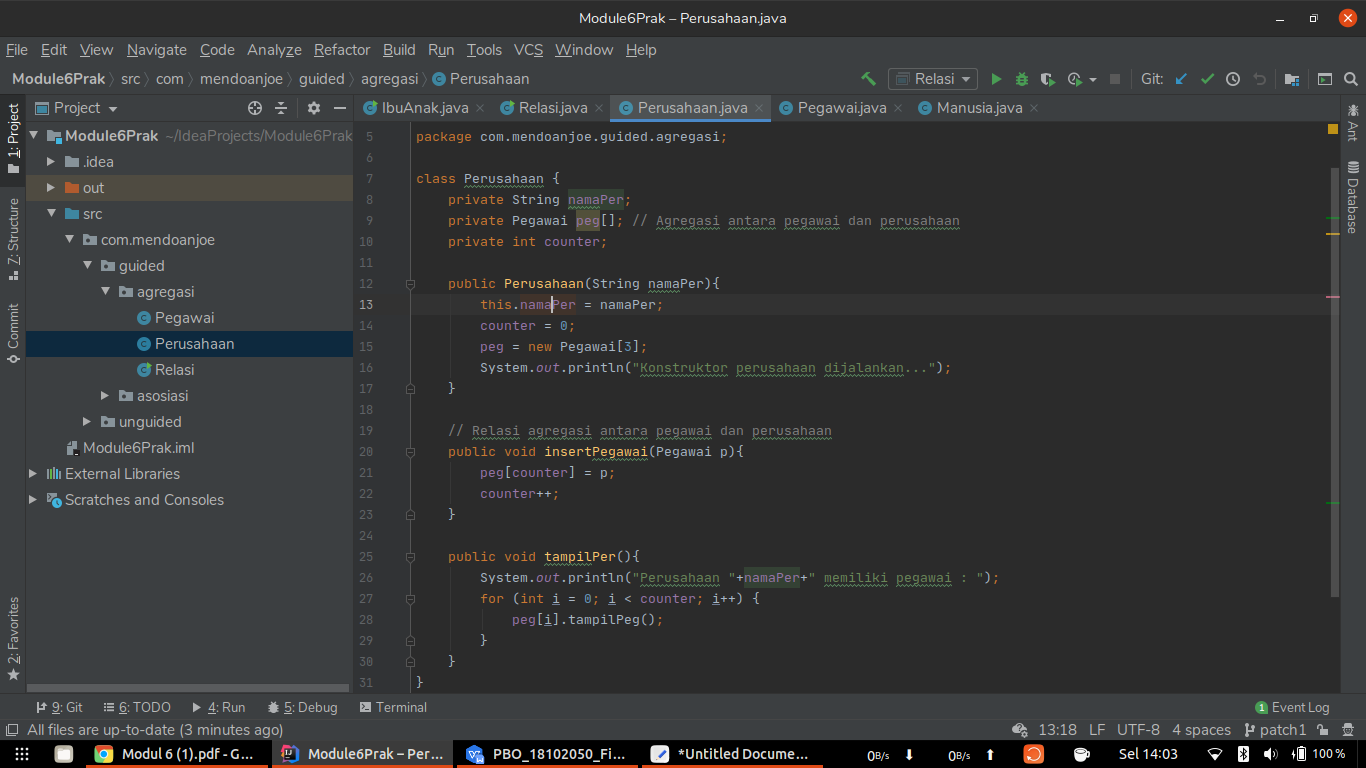


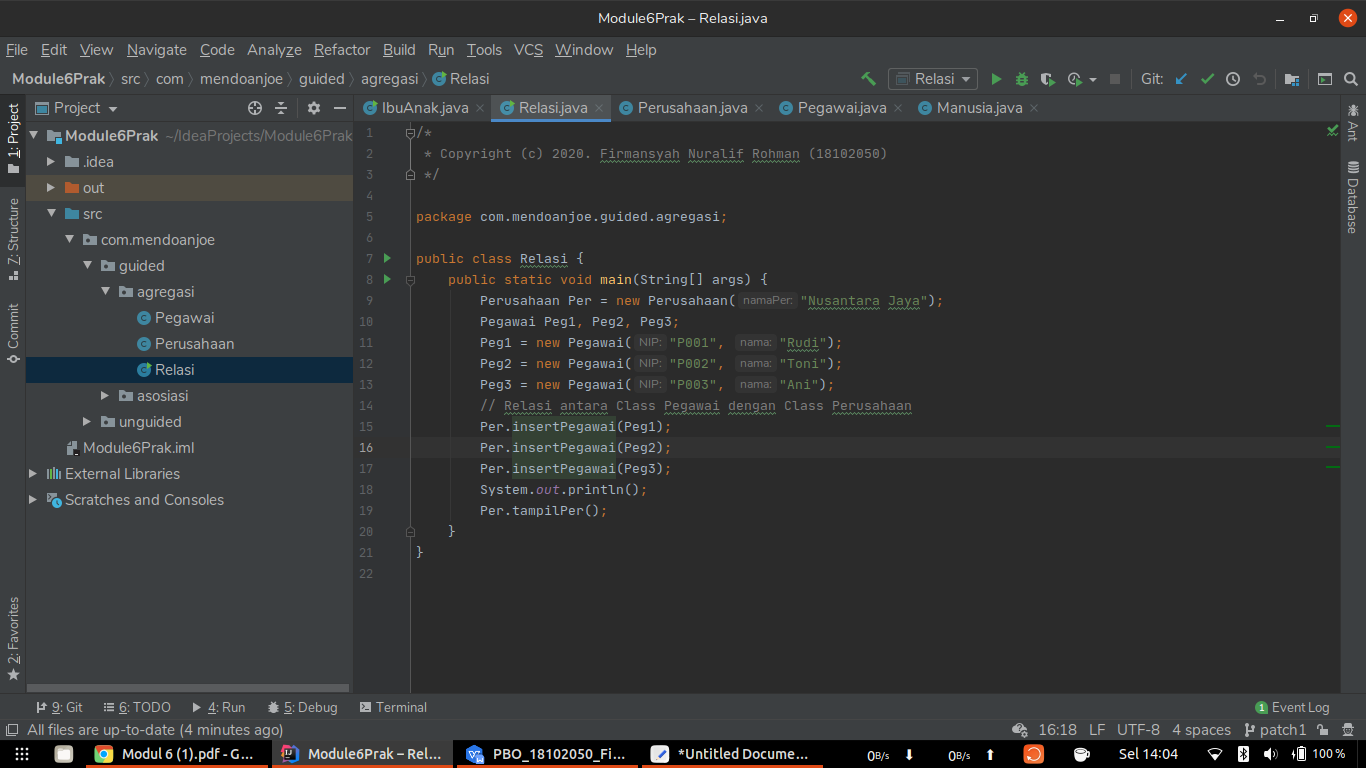
1. **Agregasi**



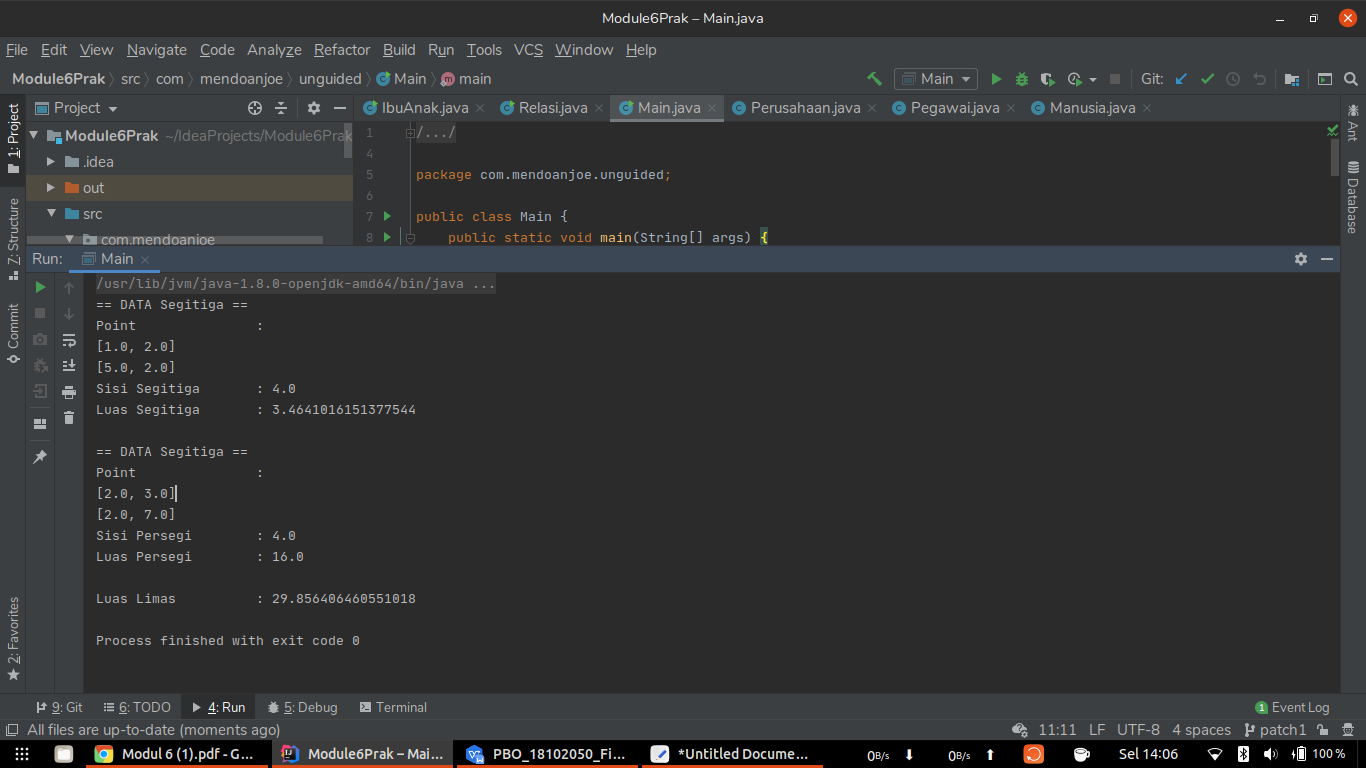
Pada module agregasi terdapat 3 file java, yaitu: Pegawai, Perusahaan dan Relasi. Relasi java digunakan sebagai main program dimana memanggil model Perusahaan dan Pegawai.



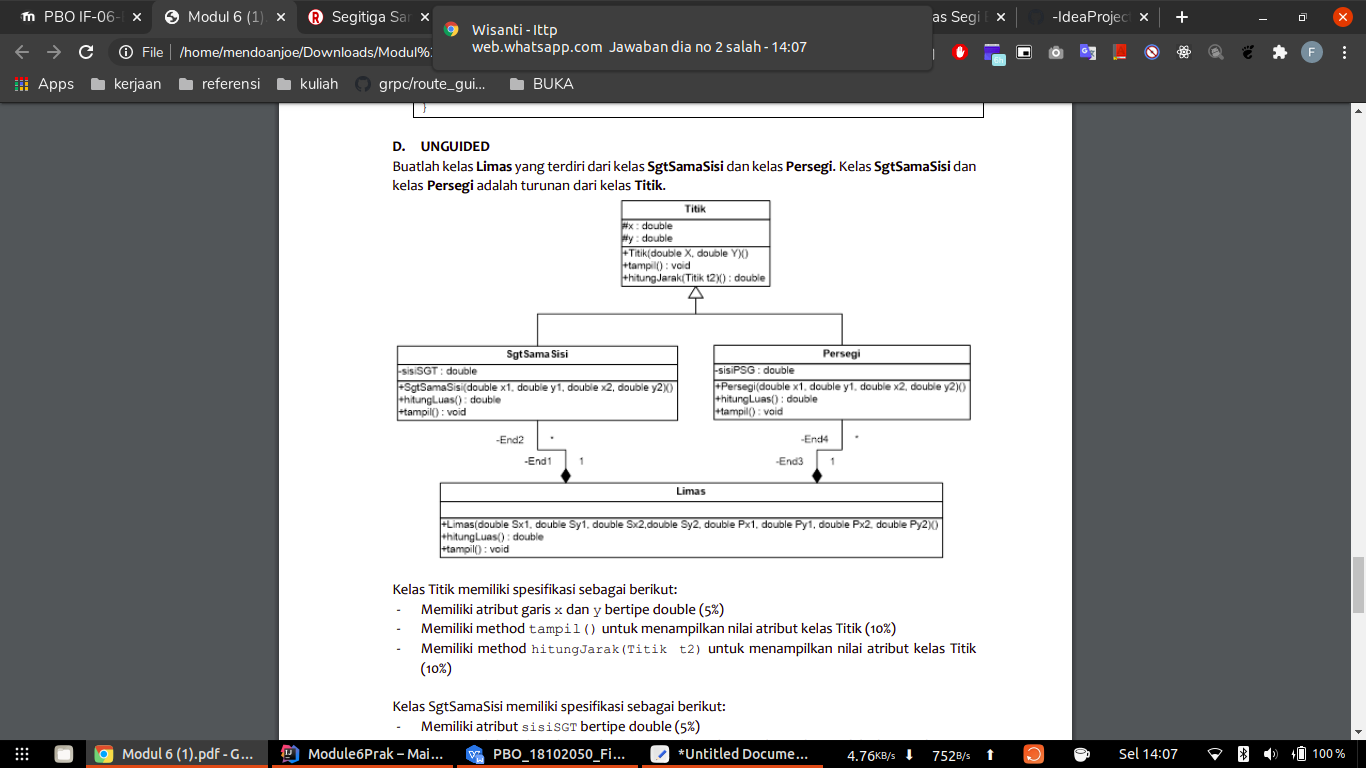




1. **PEMBAHASAN PROGRAM UNGUIDED**



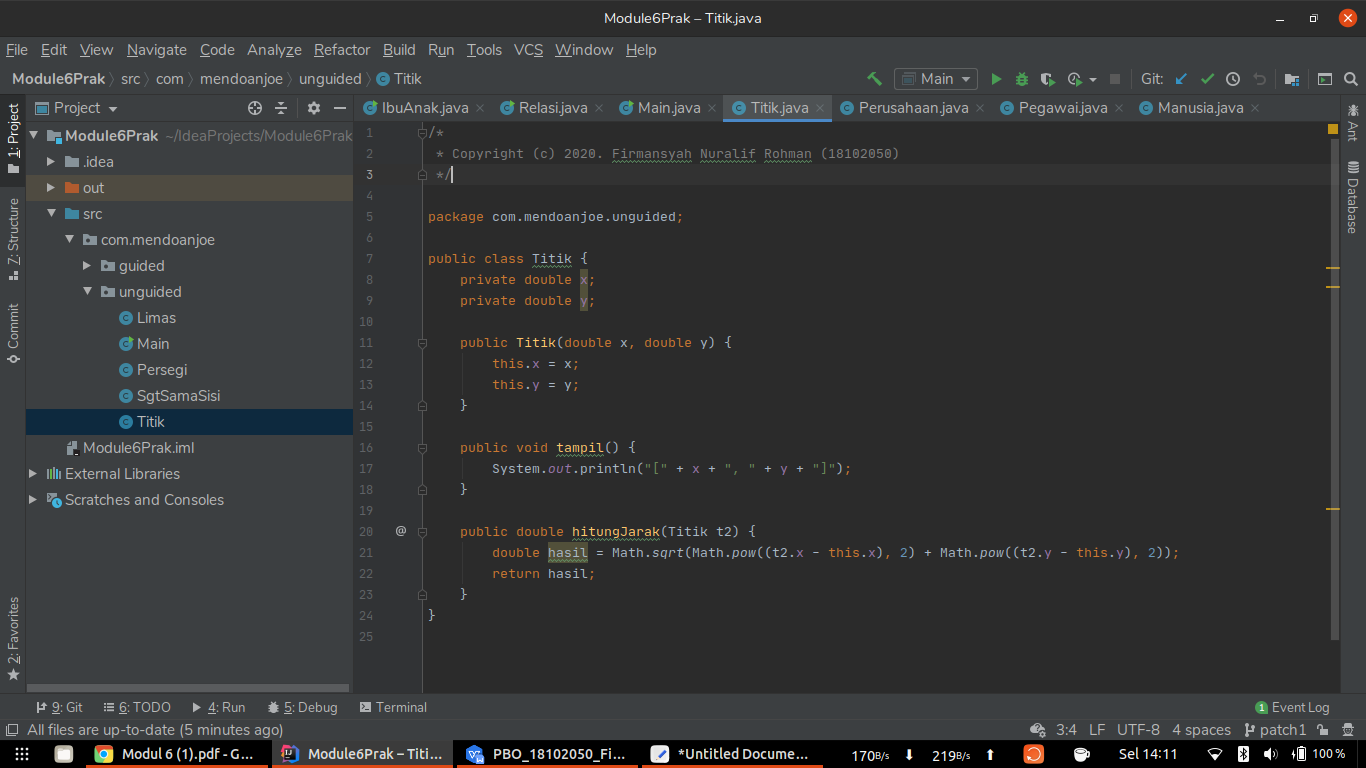
Terdapat empat kelas model dan satu kelas main, yaitu: Limas, Persegi, SgtSamaSisi, Titik, dan Main. Program di buat mengikuti map logika dibawah ini:

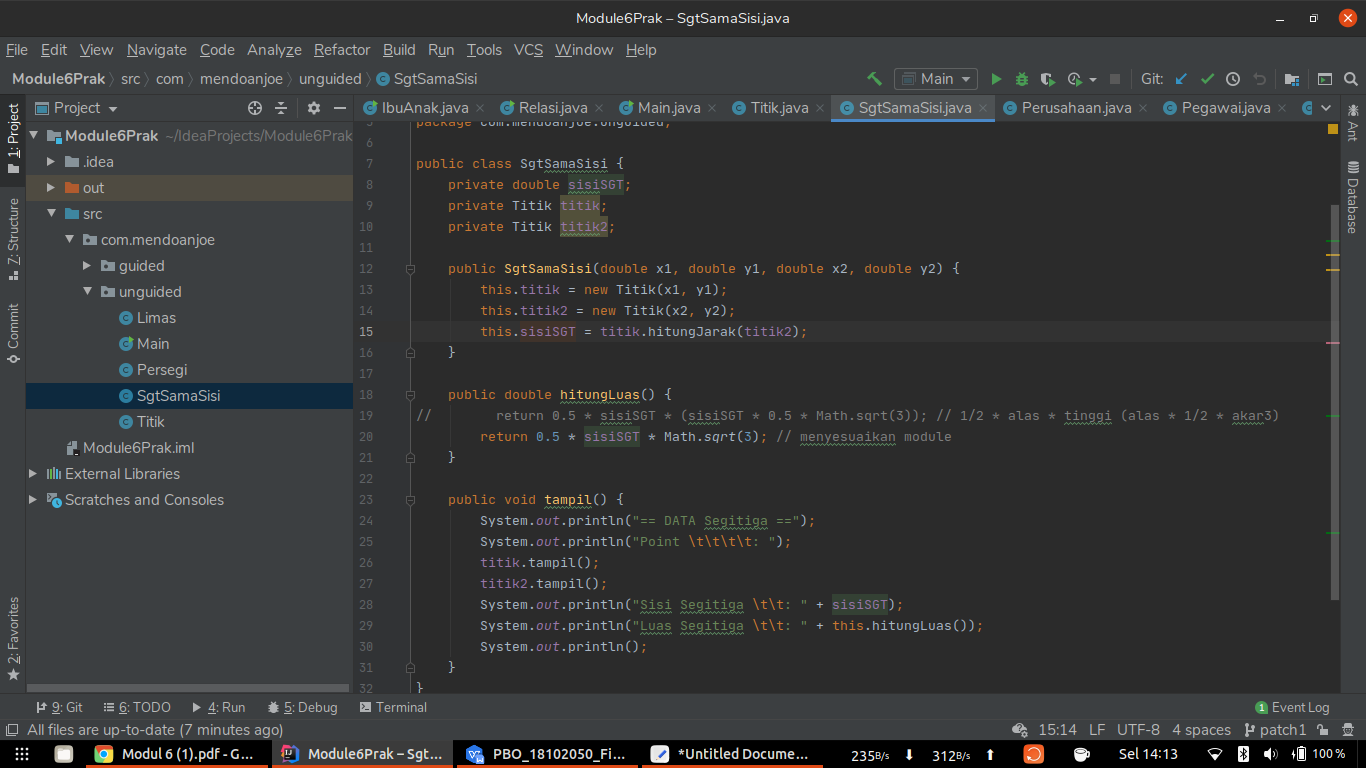


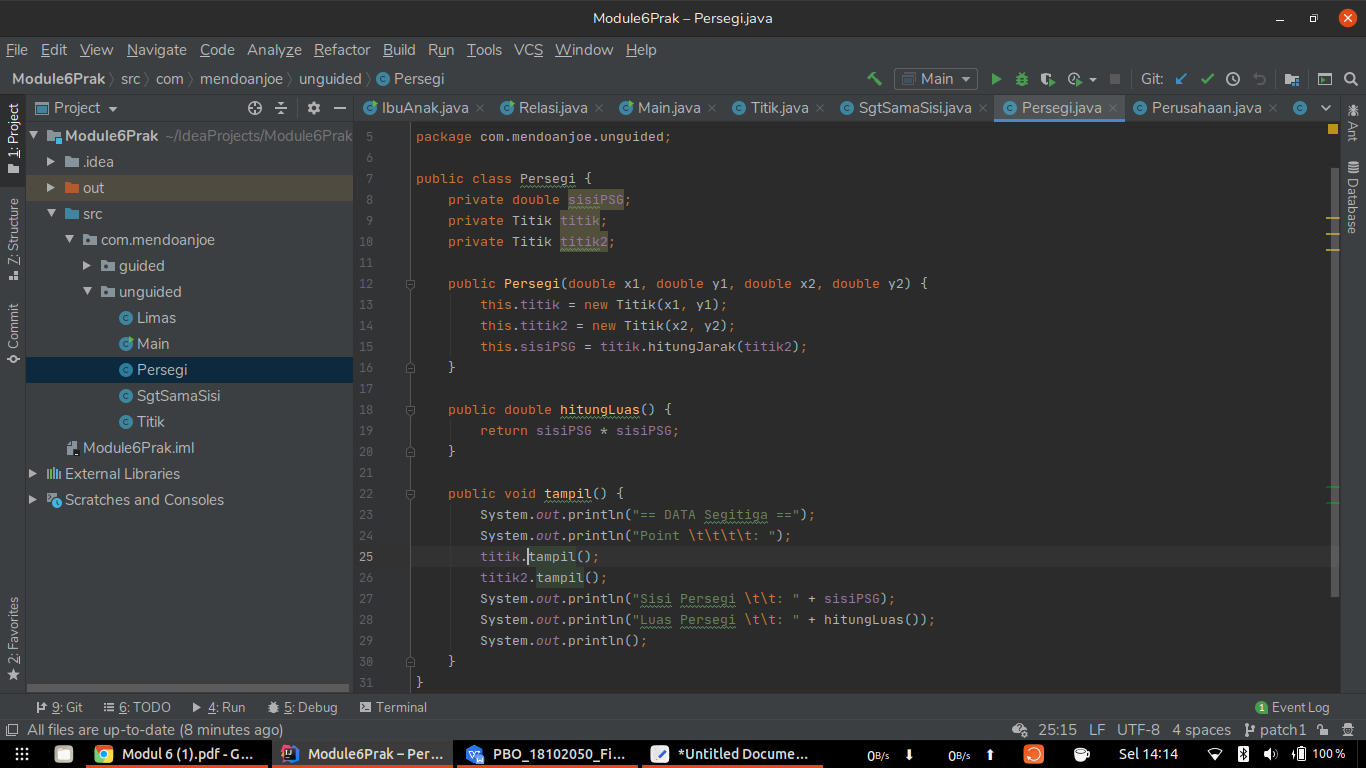
Dengan ketentuan:

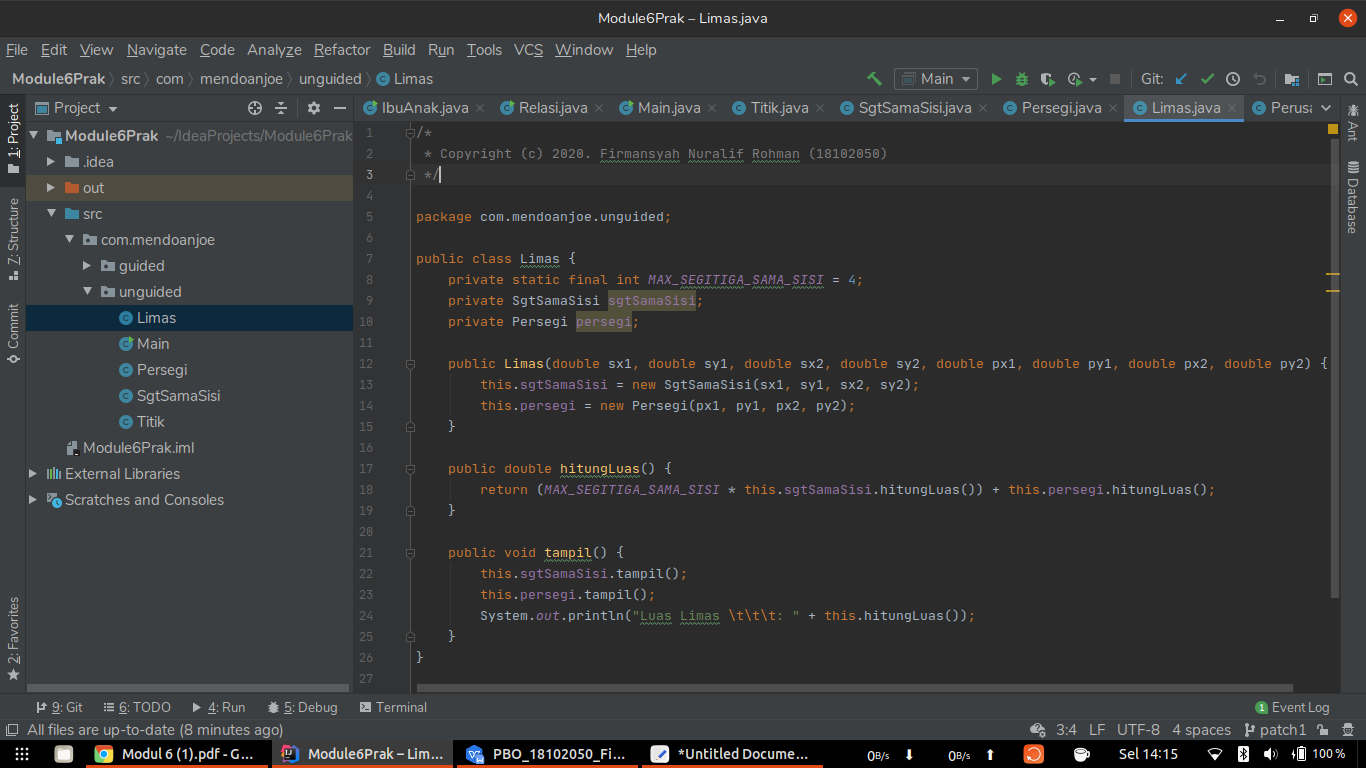
1. Program di implementasikan menggunakan metode relasi
2. Input program sudah ditentukan
3. Rumus sudah di tentukan
4. Hasil output sudah ditentukan

Berikut porgramnya yang sudah dibuat:









**Kesimpulan**

Program dengan metode relasi memang dapat menghemat penulisan kode program pada main kelas, akan tetapi dalam pengkodean dengna menggunakan metode relasi harus lebih teliti dan model bersifat dapat di akses dimana saja.