

# MODUL 2

## CLASS DAN OBJECT

### PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK

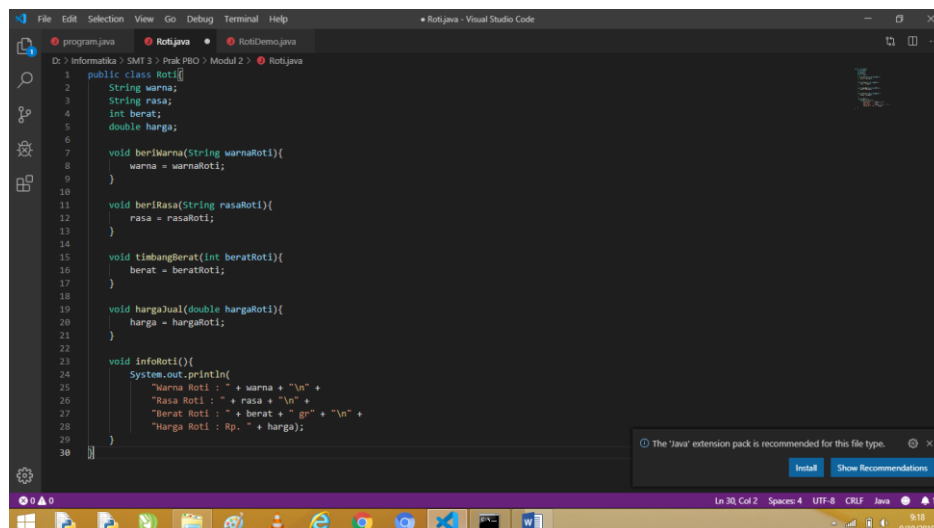
Nama : Annisa Nugraheni

NIM : L200180066

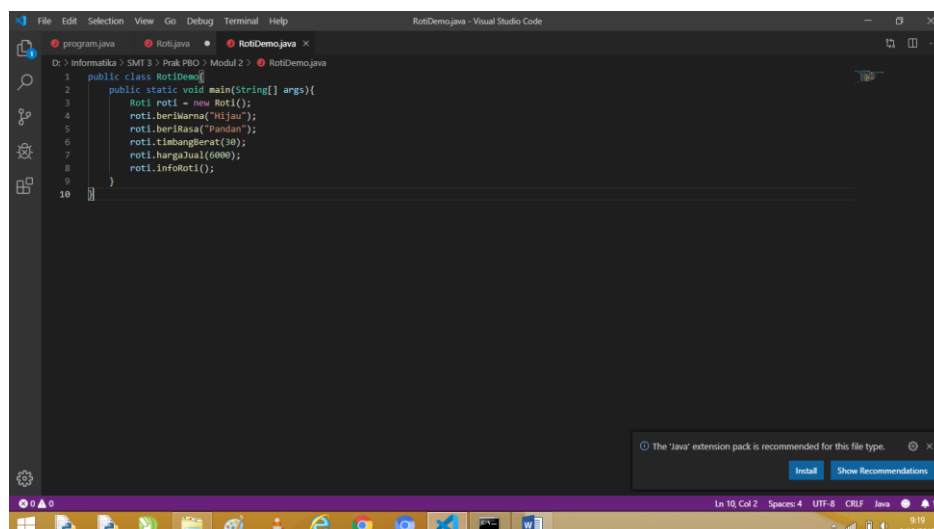
Kelas : B

#### ➤ CONTOH

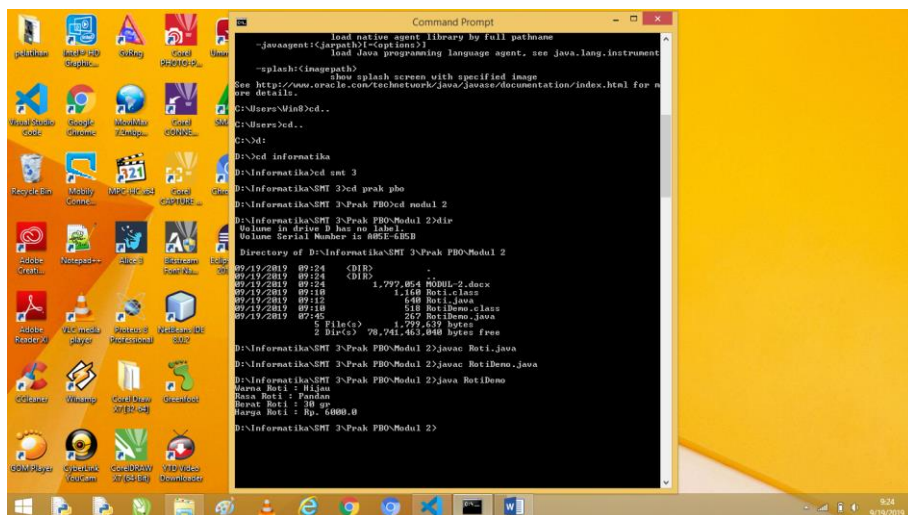
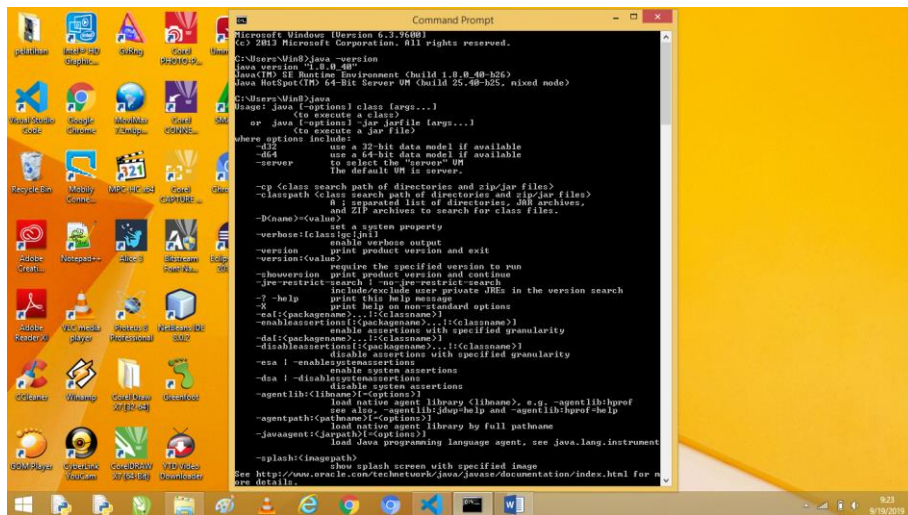
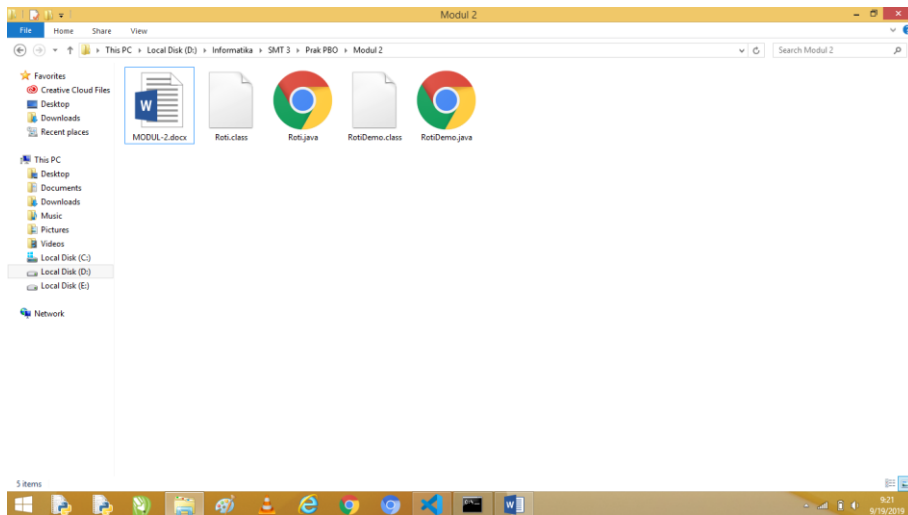
##### ❖ Screenshot class Roti dan class RotiDemo



```
1 public class Roti {
2     String warna;
3     String rasa;
4     int berat;
5     double harga;
6
7     void beriWarna(String warnaRoti){
8         warna = warnaRoti;
9     }
10
11     void beriRasa(String rasaRoti){
12         rasa = rasaRoti;
13     }
14
15     void timbangBerat(int beratRoti){
16         berat = beratRoti;
17     }
18
19     void hargaJual(double hargaRoti){
20         harga = hargaRoti;
21     }
22
23     void infoRoti(){
24         System.out.println(
25             "Warna Roti : " + warna + "\n" +
26             "Rasa Roti : " + rasa + "\n" +
27             "Berat Roti : " + berat + " gr" + "\n" +
28             "Harga Roti : Rp. " + harga);
29     }
30 }
```

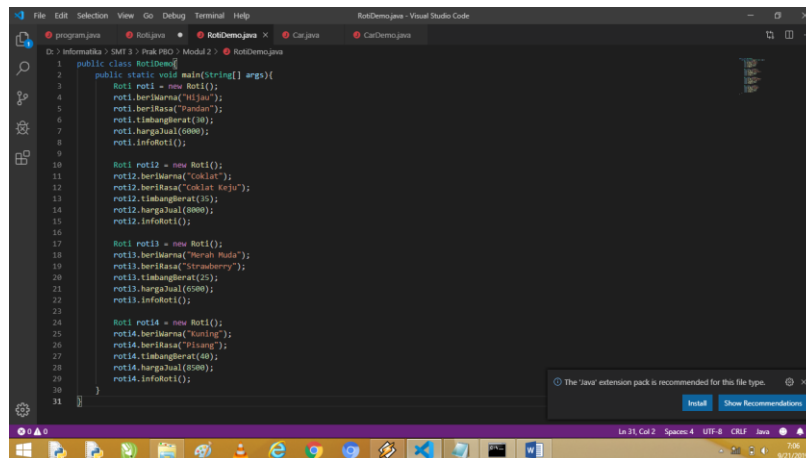


```
1 public class RotiDemo {
2     public static void main(String[] args){
3         Roti roti = new Roti();
4         roti.beriWarna("Hijau");
5         roti.beriRasa("Pandan");
6         roti.timbangBerat(30);
7         roti.hargaJual(6000);
8         roti.infoRoti();
9     }
10 }
```



## ➤ LATIHAN

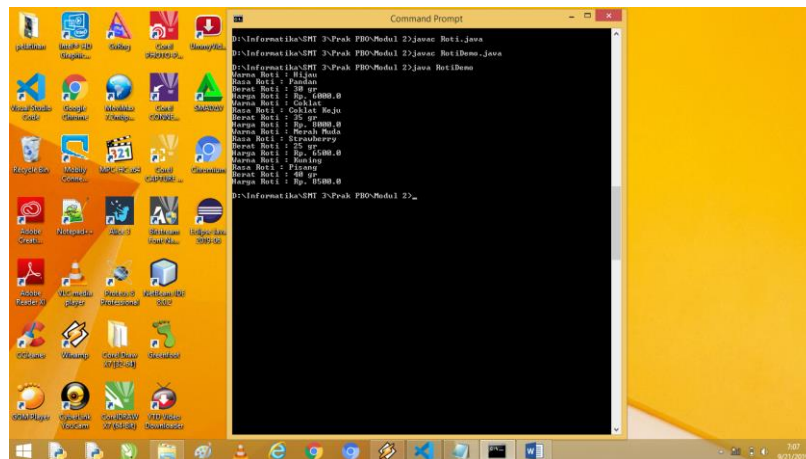
- ❖ Modifikasi class RotiDemo dan membuat 3 object baru di dalamnya



```

1 public class RotiDemo {
2     public static void main(String[] args) {
3         Roti roti = new Roti();
4         roti.bernama("Hijau");
5         roti.berRasa("Pandan");
6         roti.tambahUmur(10);
7         roti.hargaJual(5000);
8         roti.infoRoti();
9
10        Roti roti2 = new Roti();
11        roti2.bernama("Coklat");
12        roti2.berRasa("Coklat esj");
13        roti2.tambahUmur(15);
14        roti2.hargaJual(8000);
15        roti2.infoRoti();
16
17        Roti roti3 = new Roti();
18        roti3.bernama("Merah Muda");
19        roti3.berRasa("Strawberry");
20        roti3.tambahUmur(25);
21        roti3.hargaJual(6500);
22        roti3.infoRoti();
23
24        Roti roti4 = new Roti();
25        roti4.bernama("Kuning");
26        roti4.berRasa("Pisang");
27        roti4.tambahUmur(40);
28        roti4.hargaJual(8500);
29        roti4.infoRoti();
30    }
31 }

```

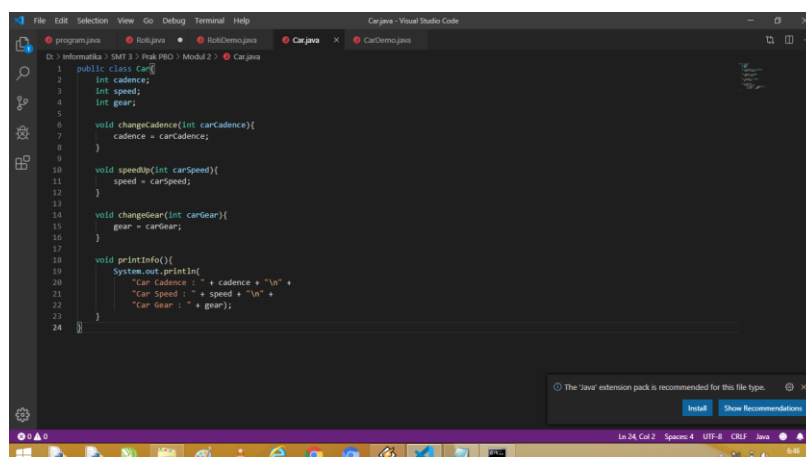


```

D:\Informatika\SMK 3>Prak PBO-Modul 2>java Roti.java
Nama Roti : Hijau
Rasa Roti : Pandan
Umur Roti : 10 yr
Harga Roti : Rp. 5000.0
Roti Roti : Coklat
Rasa Roti : Coklat esj
Umur Roti : 15 yr
Harga Roti : Rp. 8000.0
Roti Roti : Merah Muda
Rasa Roti : Strawberry
Umur Roti : 25 yr
Harga Roti : Rp. 6500.0
Roti Roti : Kuning
Rasa Roti : Pisang
Umur Roti : 40 yr
Harga Roti : Rp. 8500.0
D:\Informatika\SMK 3>Prak PBO-Modul 2>

```

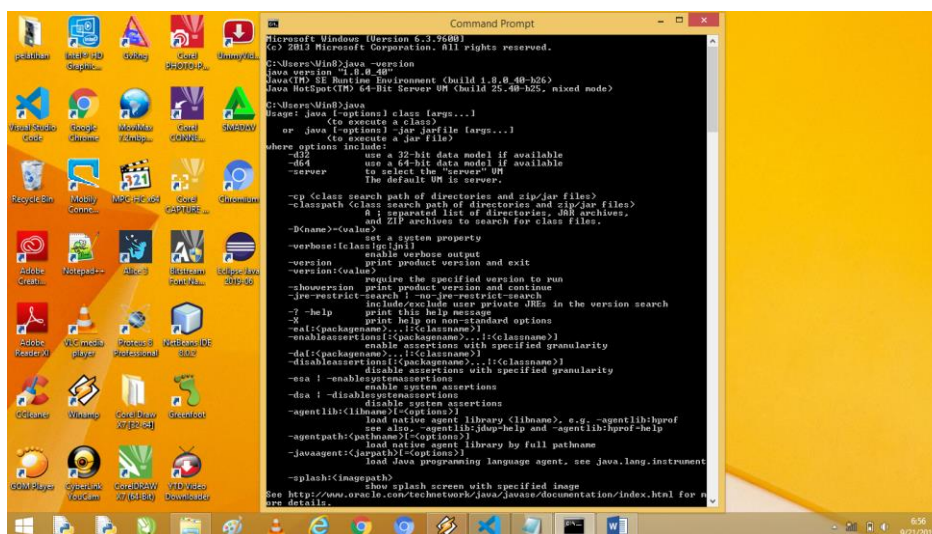
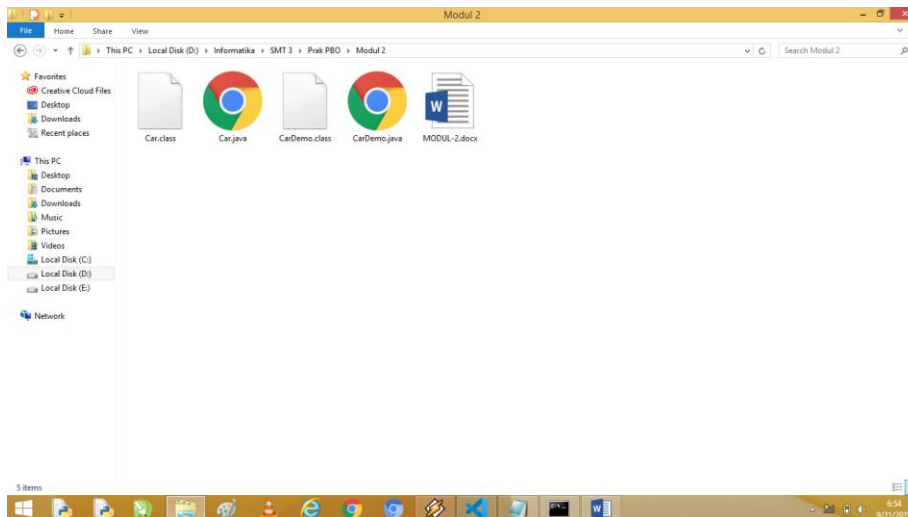
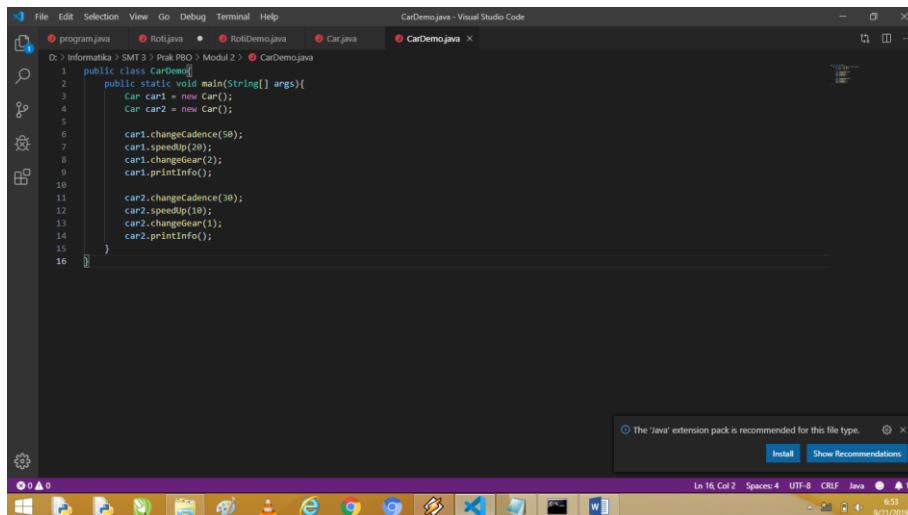
- ❖ Membuat satu class baru yang bisa digunakan sebagai template atau blueprint dari class CarDemo

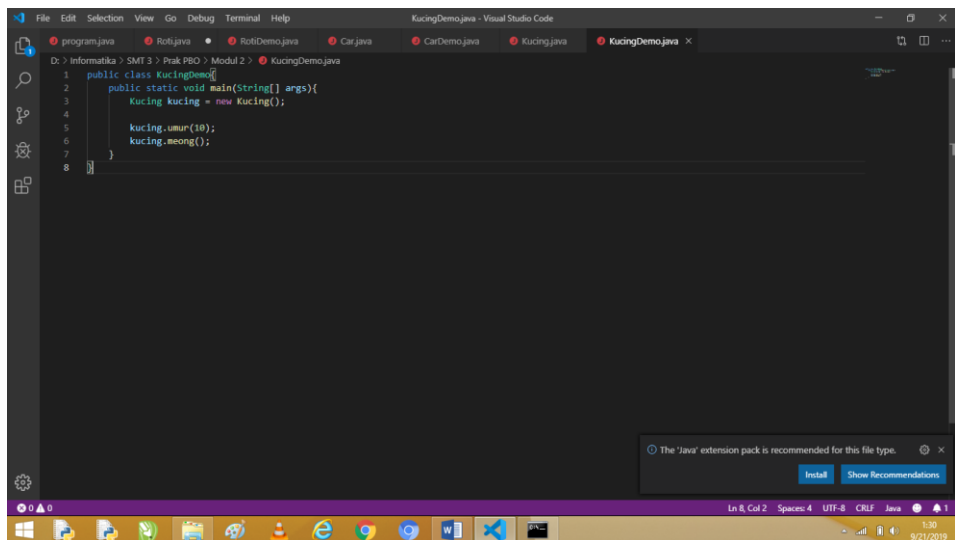


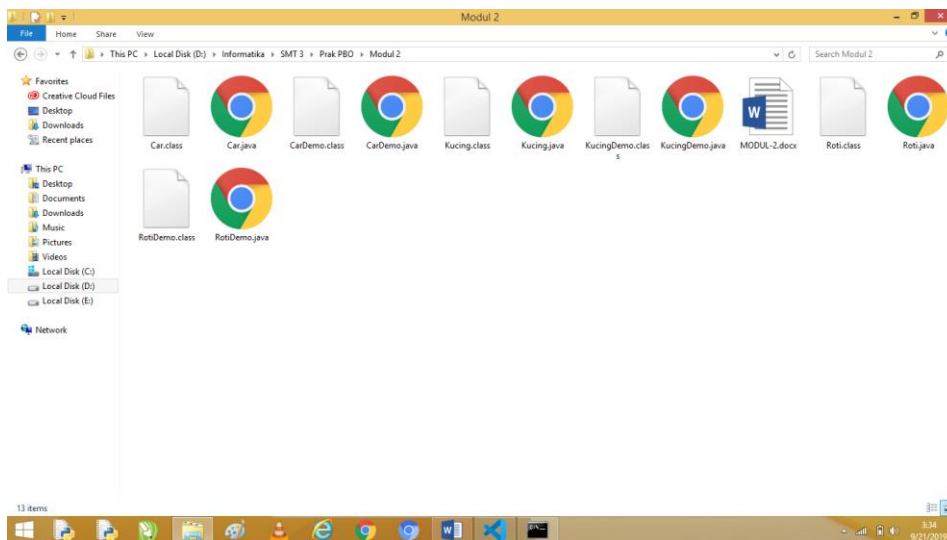
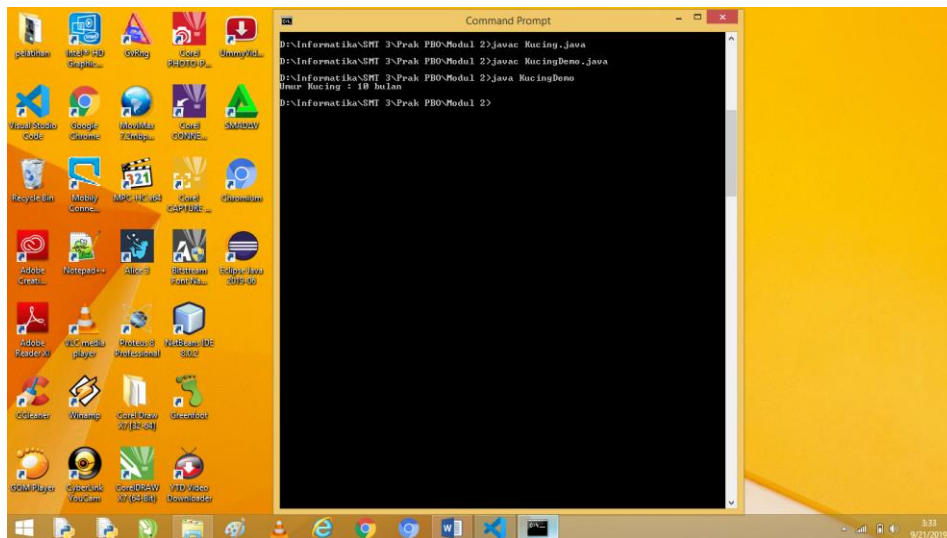
```

1 public class Car {
2     int cadence;
3     int speed;
4     int gear;
5
6     void changeCadence(int carCadence) {
7         cadence = carCadence;
8     }
9
10    void speedUp(int carSpeed) {
11        speed = carSpeed;
12    }
13
14    void changeGear(int carGear) {
15        gear = carGear;
16    }
17
18    void printInfo() {
19        System.out.println(
20            "Car Cadence : " + cadence + "\n" +
21            "Car Speed : " + speed + "\n" +
22            "Car Gear : " + gear);
23    }
24 }

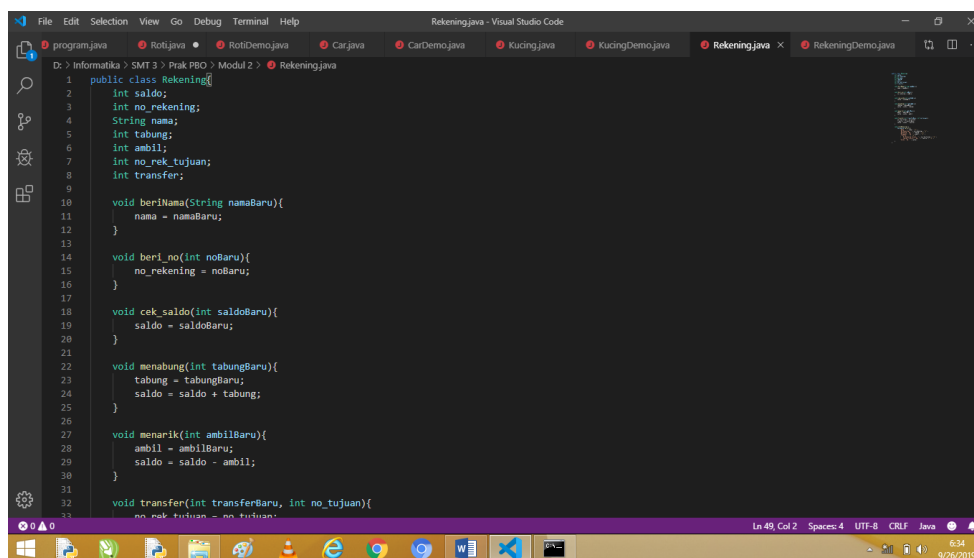
```



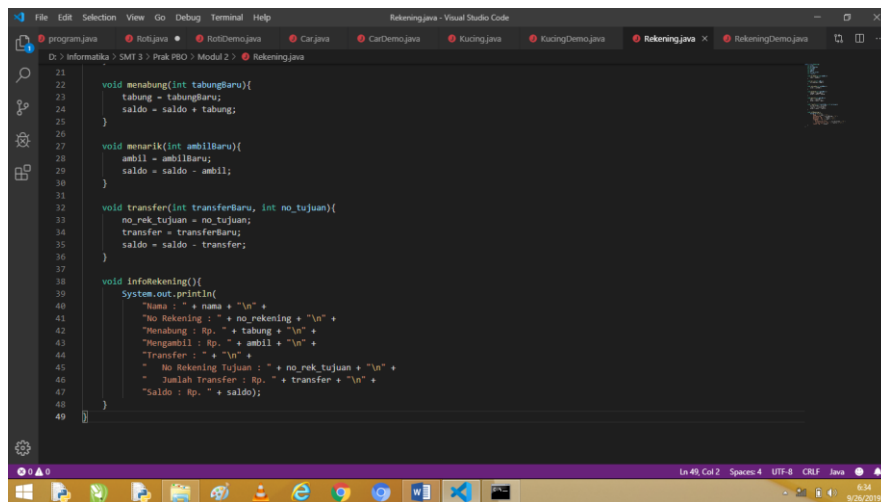




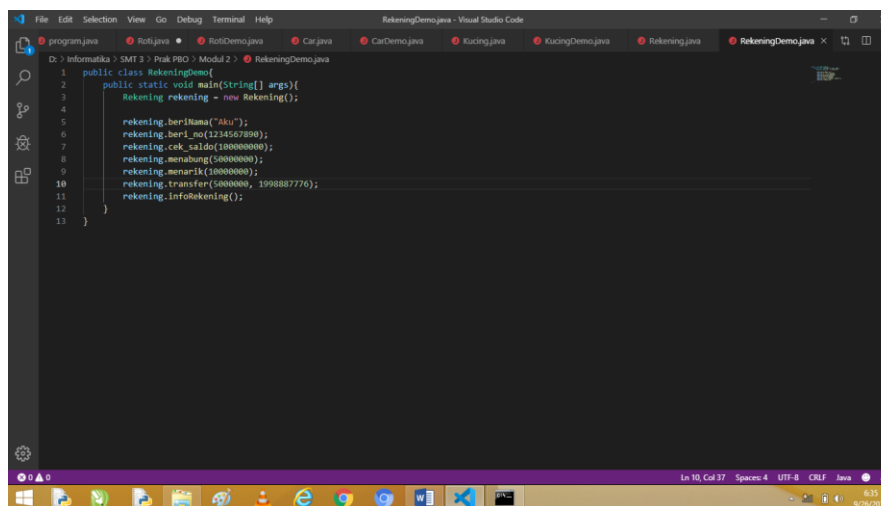
## ❖ Membuat class yang merepresentasikan rekening bank



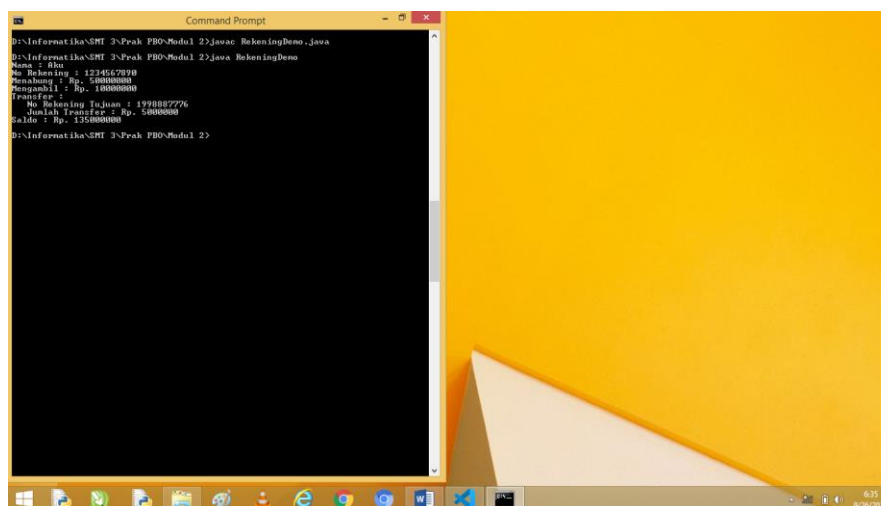




```
21 void menabung(int tabungBaru){
22     tabung = tabungBaru;
23     saldo = saldo + tabung;
24 }
25
26 void menarik(int ambilBaru){
27     ambil = ambilBaru;
28     saldo = saldo - ambil;
29 }
30
31 void transfer(int transferBaru, int no_tujuan){
32     no_rek_tujuan = no_tujuan;
33     transfer = transferBaru;
34     saldo = saldo - transfer;
35 }
36
37 void infoRekening(){
38     System.out.println(
39         "Nama : " + nama + "\n" +
40         "No Rekening : " + no_rekening + "\n" +
41         "Menabung : Rp. " + tabung + "\n" +
42         "Mengambil : Rp. " + ambil + "\n" +
43         "Transfer : " + "\n" +
44         "    No Rekening Tujuan : " + no_rek_tujuan + "\n" +
45         "    Jumlah Transfer : Rp. " + transfer + "\n" +
46         "Saldo : Rp. " + saldo);
47 }
48
49 }
```



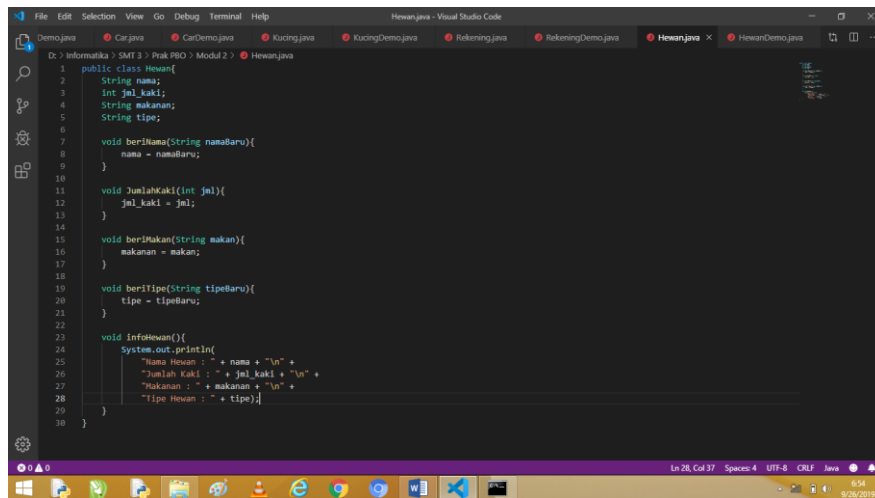
```
1 public class RekeningDemo{
2     public static void main(String[] args){
3         Rekening rekening = new Rekening();
4
5         rekening.beriNama("Aku");
6         rekening.beri_no(1234567890);
7         rekening.cek_saldo(100000000);
8         rekening.menabung(50000000);
9         rekening.menarik(10000000);
10        rekening.transfer(5000000, 1998887776);
11        rekening.infoRekening();
12    }
13 }
```



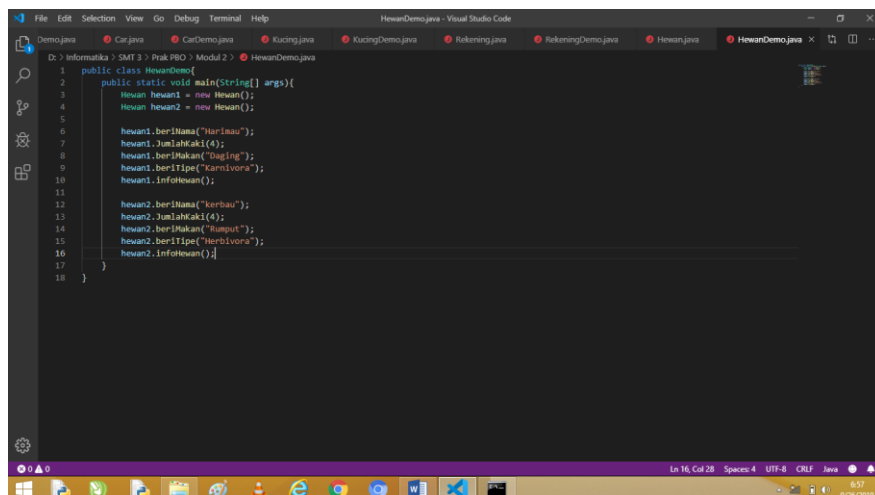
```
D:\Informatika\SMT 3\Prak PBO>Modul 2> javac RekeningDemo.java
D:\Informatika\SMT 3\Prak PBO>Modul 2> java RekeningDemo
Nama : Aku
No. Rekening : 1234567890
Menabung : Rp. 50000000
Mengambil : Rp. 10000000
Transfer :
    No Rekening Tujuan : 1998887776
    Jumlah Transfer : Rp. 5000000
Saldo : Rp. 135000000
D:\Informatika\SMT 3\Prak PBO>Modul 2>
```

## ➤ PEKERJAAN RUMAH

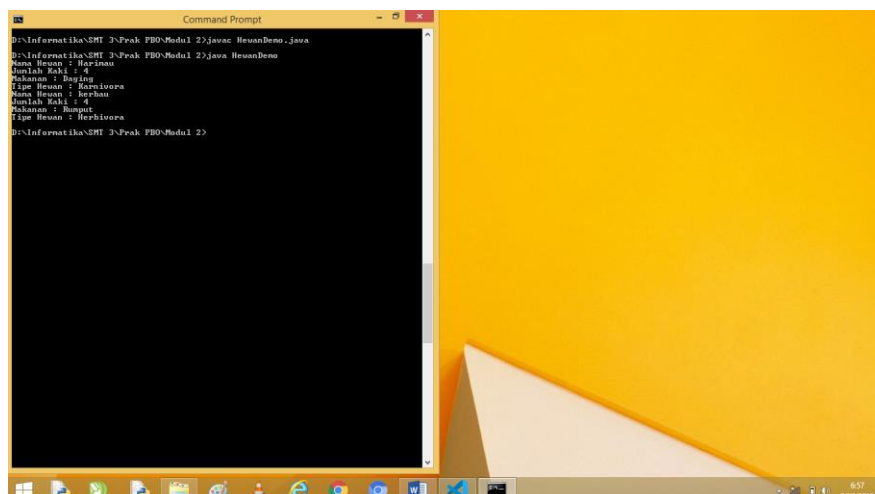
### ❖ Membuat class Hewan dan object didalamnya



```
1 public class Hewan{
2     String nama;
3     int jml_kaki;
4     String makanan;
5     String tipe;
6
7     void beriNama(String namaBaru){
8         nama = namaBaru;
9     }
10
11     void jumlahKaki(int jml){
12         jml_kaki = jml;
13     }
14
15     void beriMakan(String makan){
16         makanan = makan;
17     }
18
19     void beriTipe(String tipeBaru){
20         tipe = tipeBaru;
21     }
22
23     void infoHewan(){
24         System.out.println(
25             "Nama Hewan : " + nama + "\n" +
26             "Jumlah Kaki : " + jml_kaki + "\n" +
27             "Makanan : " + makanan + "\n" +
28             "Tipe Hewan : " + tipe);
29     }
30 }
```



```
1 public class HewanDemo{
2     public static void main(String[] args){
3         Hewan hewan1 = new Hewan();
4         Hewan hewan2 = new Hewan();
5
6         hewan1.beriNama("Harimau");
7         hewan1.jumlahKaki(4);
8         hewan1.beriMakan("Daging");
9         hewan1.beriTipe("Karnivora");
10        hewan1.infoHewan();
11
12        hewan2.beriNama("Kerbau");
13        hewan2.jumlahKaki(4);
14        hewan2.beriMakan("Rumput");
15        hewan2.beriTipe("Herbivora");
16        hewan2.infoHewan();
17    }
18 }
```

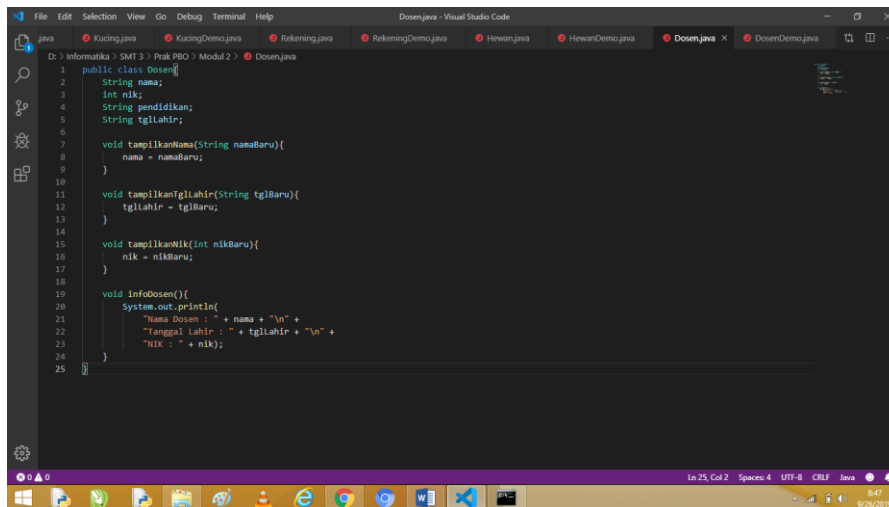


```
D:\Informatika\SMT 3\Prak PBO\Modul 2>javac HewanDemo.java
D:\Informatika\SMT 3\Prak PBO\Modul 2>java HewanDemo
Nama Hewan : Harimau
Jumlah Kaki : 4
Makanan : Daging
Tipe Hewan : Karnivora
Nama Hewan : Kerbau
Jumlah Kaki : 4
Makanan : Rumput
Tipe Hewan : Herbivora
D:\Informatika\SMT 3\Prak PBO\Modul 2>
```

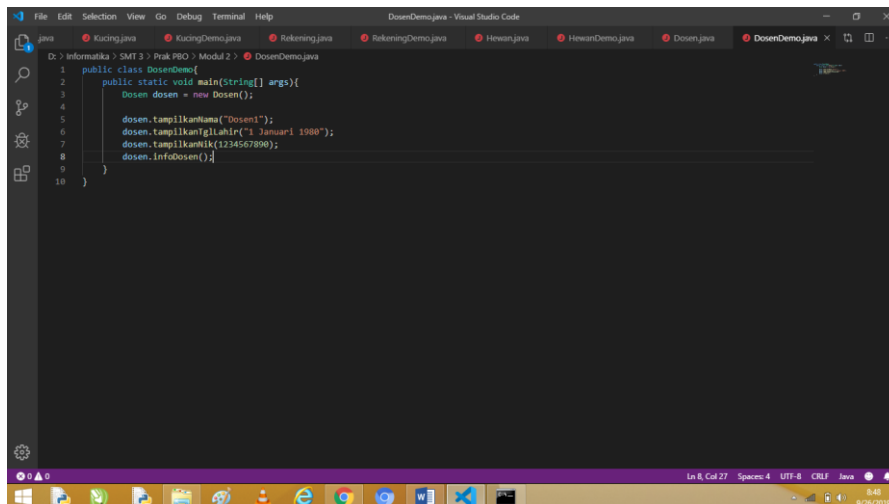


## ❖ Membuat class berdasarkan class diagram

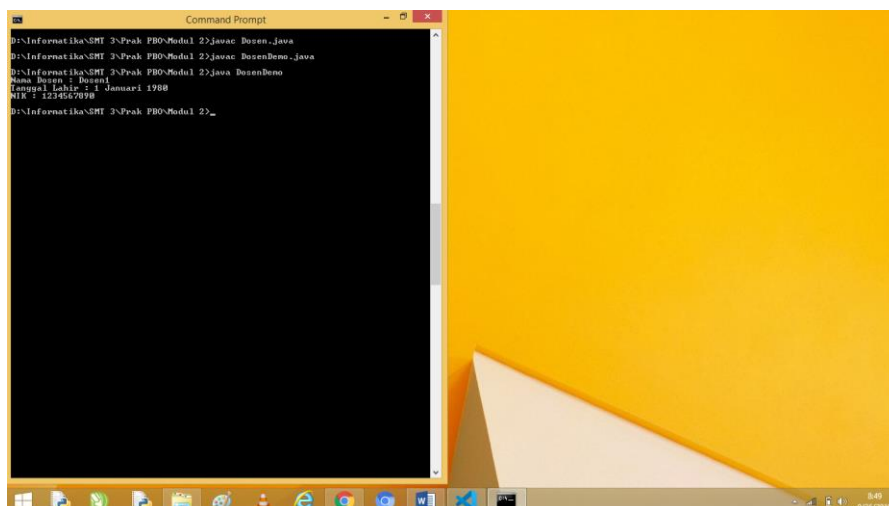
- Class Dosen



```
1 public class Dosen{
2     String nama;
3     int nik;
4     String pendidikan;
5     String tglahir;
6
7     void tampilkanNama(String namaBaru){
8         nama = namaBaru;
9     }
10
11     void tampilkanTglahir(String tglBaru){
12         tglahir = tglBaru;
13     }
14
15     void tampilkanNik(int nikBaru){
16         nik = nikBaru;
17     }
18
19     void infoDosen(){
20         System.out.println(
21             "Nama Dosen : " + nama + "\n" +
22             "Tanggal Lahir : " + tglahir + "\n" +
23             "NIK : " + nik);
24     }
25 }
```

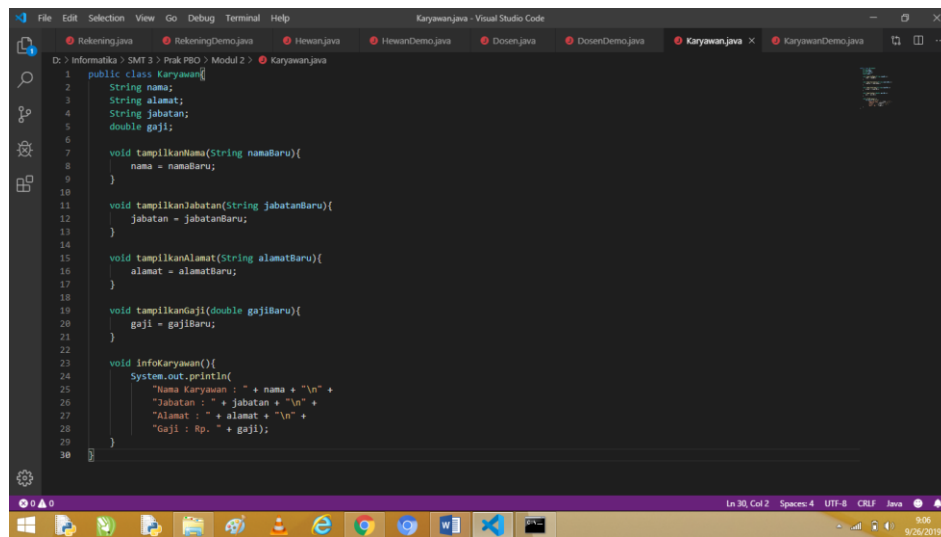


```
1 public class DosenDemo{
2     public static void main(String[] args){
3         Dosen dosen = new Dosen();
4
5         dosen.tampilkanNama("Dosen1");
6         dosen.tampilkanTglahir("1 Januari 1988");
7         dosen.tampilkanNik(1234567890);
8         dosen.infoDosen();
9     }
10 }
```

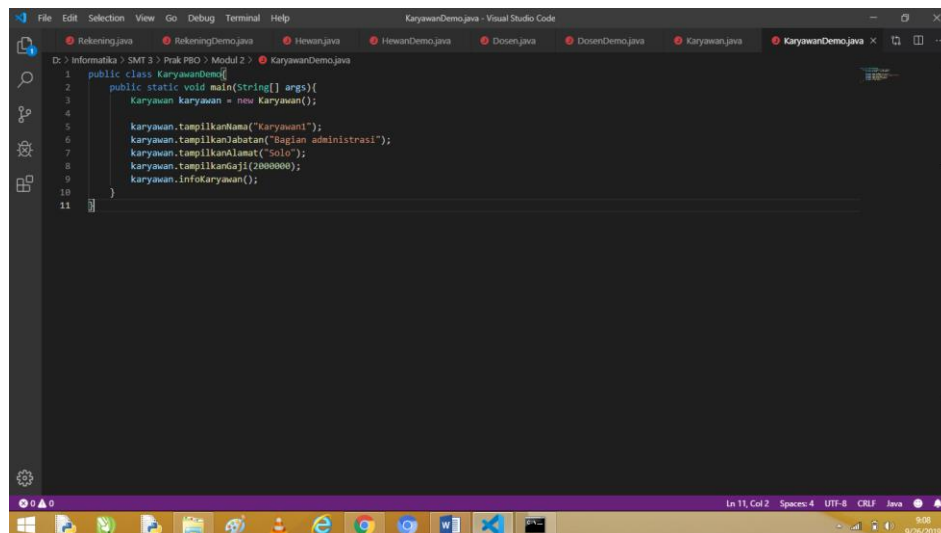


```
D:\Informatika\SM1 3>Prak PBO>Modul 2>javac Dosen.java
D:\Informatika\SM1 3>Prak PBO>Modul 2>java DosenDemo
Nama Dosen : Dosen1
Tanggal Lahir : 1 Januari 1988
NIK : 1234567890
D:\Informatika\SM1 3>Prak PBO>Modul 2>
```

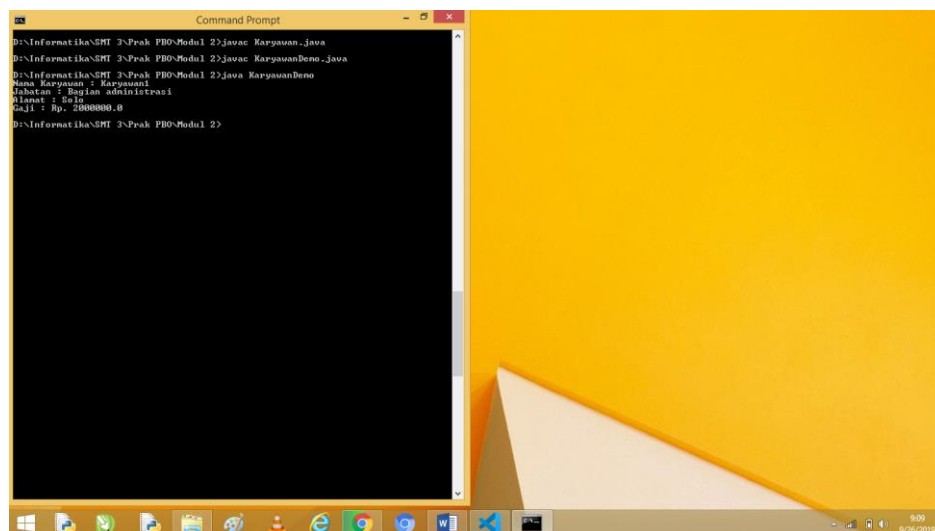
- Class Karyawan



```
1 public class Karyawan {
2     String nama;
3     String alamat;
4     String jabatan;
5     double gaji;
6
7     void tampilkanNama(String namaBaru){
8         nama = namaBaru;
9     }
10
11    void tampilkanJabatan(String jabatanBaru){
12        jabatan = jabatanBaru;
13    }
14
15    void tampilkanAlamat(String alamatBaru){
16        alamat = alamatBaru;
17    }
18
19    void tampilkanGaji(double gajiBaru){
20        gaji = gajiBaru;
21    }
22
23    void infoKaryawan(){
24        System.out.println(
25            "Nama Karyawan : " + nama + "\n" +
26            "Jabatan : " + jabatan + "\n" +
27            "Alamat : " + alamat + "\n" +
28            "Gaji : Rp. " + gaji);
29    }
30 }
```



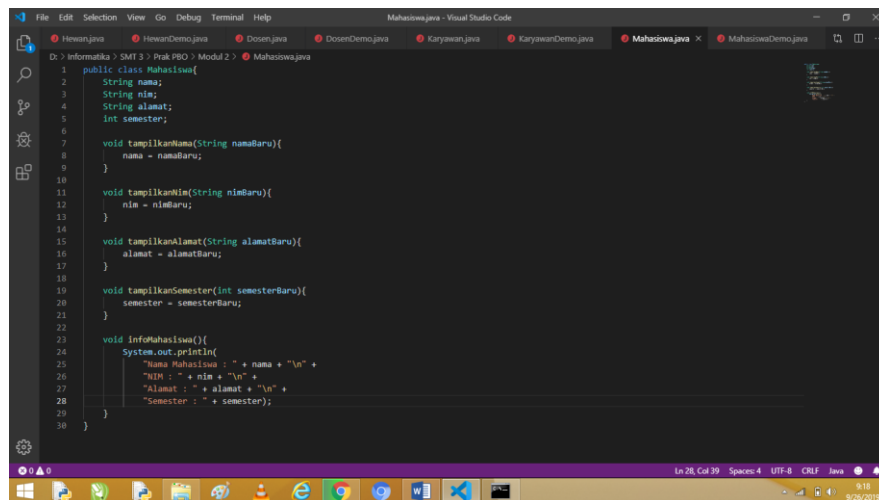
```
1 public class KaryawanDemo {
2     public static void main(String[] args){
3         Karyawan karyawan = new Karyawan();
4
5         karyawan.tampilkanNama("Karyawani");
6         karyawan.tampilkanJabatan("Bagian administrasi");
7         karyawan.tampilkanAlamat("Solo");
8         karyawan.tampilkanGaji(2000000);
9         karyawan.infoKaryawan();
10    }
11 }
```



```

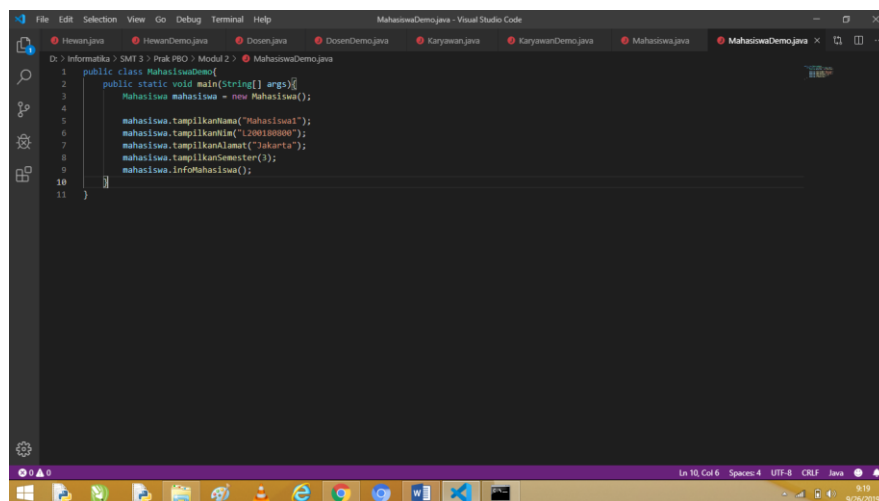
D:\Informatika\SMK 3\Prak PBO\Modul 2>javac Karyawan.java
D:\Informatika\SMK 3\Prak PBO\Modul 2>javac KaryawanDemo.java
D:\Informatika\SMK 3\Prak PBO\Modul 2>java KaryawanDemo
Nama Karyawani : Karyawani
Jabatan : Bagian administrasi
Alamat : Solo
Gaji : Rp. 2000000.0
D:\Informatika\SMK 3\Prak PBO\Modul 2>
```

- Class Mahasiswa



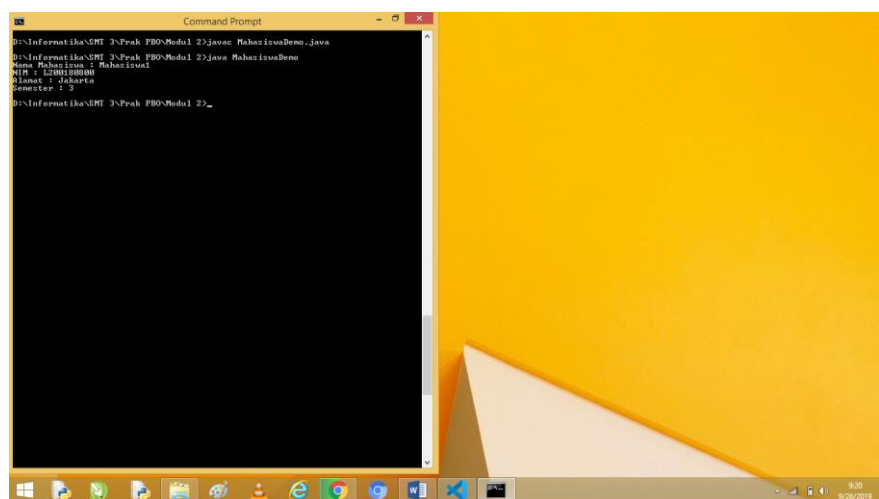
This screenshot shows the Visual Studio Code editor with the `Mahasiswa.java` file open. The code defines a `Mahasiswa` class with attributes `nama`, `nim`, `alamat`, and `semester`. It includes methods to set each attribute (`tampilkanNama`, `tampilkanNim`, `tampilkanAlamat`, `tampilkanSemester`) and a method to display all information (`infoMahasiswa`). The `infoMahasiswa` method uses `System.out.println` to print the student's details.

```
1 public class Mahasiswa{
2     String nama;
3     String nim;
4     String alamat;
5     int semester;
6
7     void tampilkanNama(String namaBaru){
8         nama = namaBaru;
9     }
10
11     void tampilkanNim(String nimBaru){
12         nim = nimBaru;
13     }
14
15     void tampilkanAlamat(String alamatBaru){
16         alamat = alamatBaru;
17     }
18
19     void tampilkanSemester(int semesterBaru){
20         semester = semesterBaru;
21     }
22
23     void infoMahasiswa(){
24         System.out.println(
25             "Nama Mahasiswa : " + nama + "\n" +
26             "NIM : " + nim + "\n" +
27             "Alamat : " + alamat + "\n" +
28             "Semester : " + semester);
29     }
30 }
```



This screenshot shows the Visual Studio Code editor with the `MahasiswaDemo.java` file open. The code defines a `MahasiswaDemo` class with a `main` method. In the `main` method, a `Mahasiswa` object is created and its attributes are set using the `tampilkan` methods. Finally, the `infoMahasiswa` method is called to display the student's information.

```
1 public class MahasiswaDemo{
2     public static void main(String[] args){
3         Mahasiswa mahasiswa = new Mahasiswa();
4
5         mahasiswa.tampilkanNama("Mahasiswa");
6         mahasiswa.tampilkanNim("1200180000");
7         mahasiswa.tampilkanAlamat("Jakarta");
8         mahasiswa.tampilkanSemester(3);
9         mahasiswa.infoMahasiswa();
10    }
11 }
```



This screenshot shows a Windows Command Prompt window where the `MahasiswaDemo.java` program has been executed. The output displays the student's information: Name, NIM, Address, and Semester.

```

D:\Informatika> SMT 3 > Prak PBO > Modul 2 > java MahasiswaDemo.java
D:\Informatika> SMT 3 > Prak PBO > Modul 2 > java MahasiswaDemo
Nama Mahasiswa : Mahasiswa
NIM : 1200180000
Alamat : Jakarta
Semester : 3
D:\Informatika> SMT 3 > Prak PBO > Modul 2 >
```