

**LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK**  
**MODUL 2**  
**LATIHAN 2.4 “CLASS DAN OBJECT”**



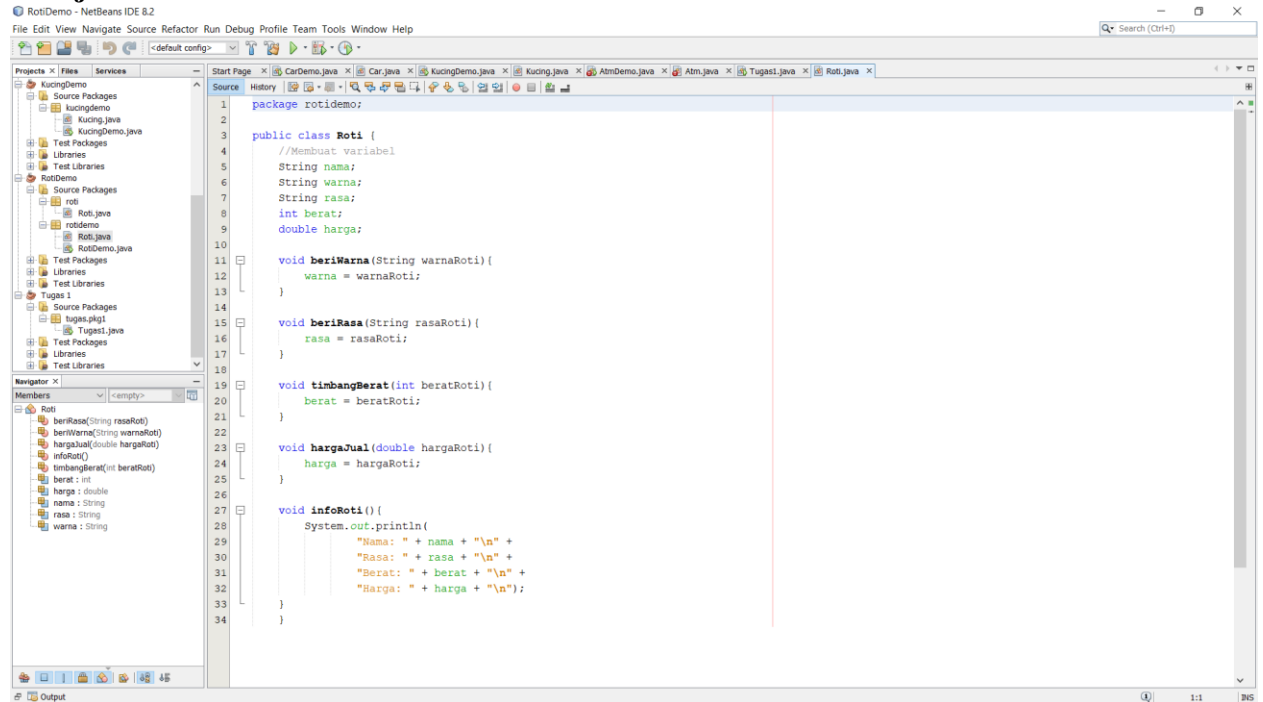
Disusun Oleh :  
Nama: Yoga Pramudita  
NIM : L200200182  
Kelas : D  
Matkul : Praktikum Pemrograman Berorientasi Obyek

**Fakultas komunikasi dan Informatika**  
**Universitas Muhammadiyah Surakarta**  
**Tahun Ajaran 2021/2022**

## 2.4 LATIHAN

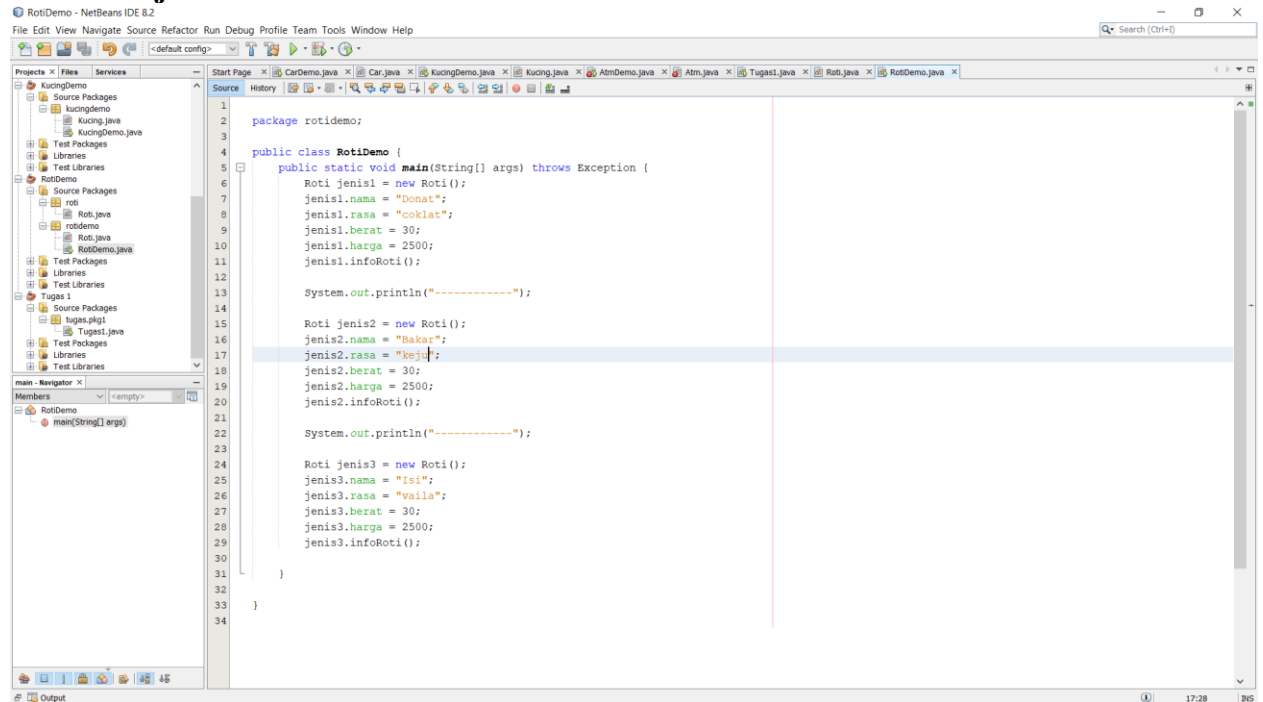
1. Silakan modifikasi class RotiDemo dan buatlah 3 object baru di dalamnya

### Roti.java



```
1 package rotidemo;
2
3 public class Roti {
4     //Membuat variabel
5     String nama;
6     String warna;
7     String rasa;
8     int berat;
9     double harga;
10
11     void beriWarna(String warnaRoti){
12         warna = warnaRoti;
13     }
14
15     void beriRasa(String rasaRoti){
16         rasa = rasaRoti;
17     }
18
19     void timbangBerat(int beratRoti){
20         berat = beratRoti;
21     }
22
23     void hargaJual(double hargaRoti){
24         harga = hargaRoti;
25     }
26
27     void infoRoti(){
28         System.out.println(
29             "Nama: " + nama + "\n" +
30             "Rasa: " + rasa + "\n" +
31             "Berat: " + berat + "\n" +
32             "Harga: " + harga + "\n");
33     }
34 }
```

### RotiDemo.java



```
1 package rotidemo;
2
3 public class RotiDemo {
4     public static void main(String[] args) throws Exception {
5         Roti jeni1 = new Roti();
6         jeni1.nama = "Donat";
7         jeni1.rasa = "coklat";
8         jeni1.berat = 30;
9         jeni1.harga = 2500;
10        jeni1.infoRoti();
11
12        System.out.println("-----");
13
14        Roti jeni2 = new Roti();
15        jeni2.nama = "Baklar";
16        jeni2.rasa = "keju";
17        jeni2.berat = 30;
18        jeni2.harga = 2500;
19        jeni2.infoRoti();
20
21        System.out.println("-----");
22
23        Roti jeni3 = new Roti();
24        jeni3.nama = "Isi";
25        jeni3.rasa = "valla";
26        jeni3.berat = 30;
27        jeni3.harga = 2500;
28        jeni3.infoRoti();
29
30    }
31
32 }
33
34 }
```

## Output

Output - RotiDemo (run)

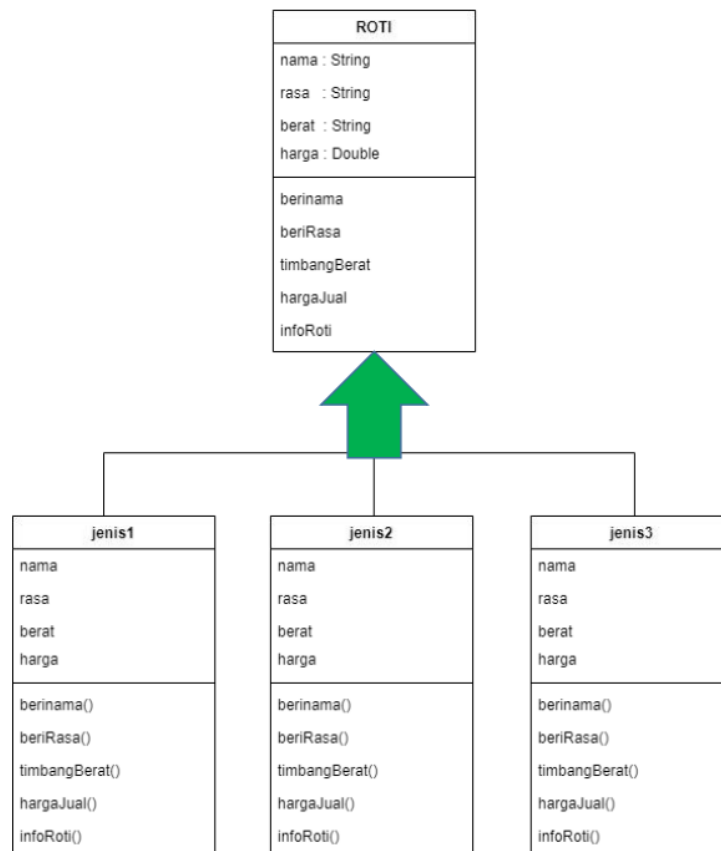
```
run:
Nama: Donat
Rasa: coklat
Berat: 30
Harga: 2500.0

-----
Nama: Bakar
Rasa: keju
Berat: 30
Harga: 2500.0

-----
Nama: Isi
Rasa: vaila
Berat: 30
Harga: 2500.0

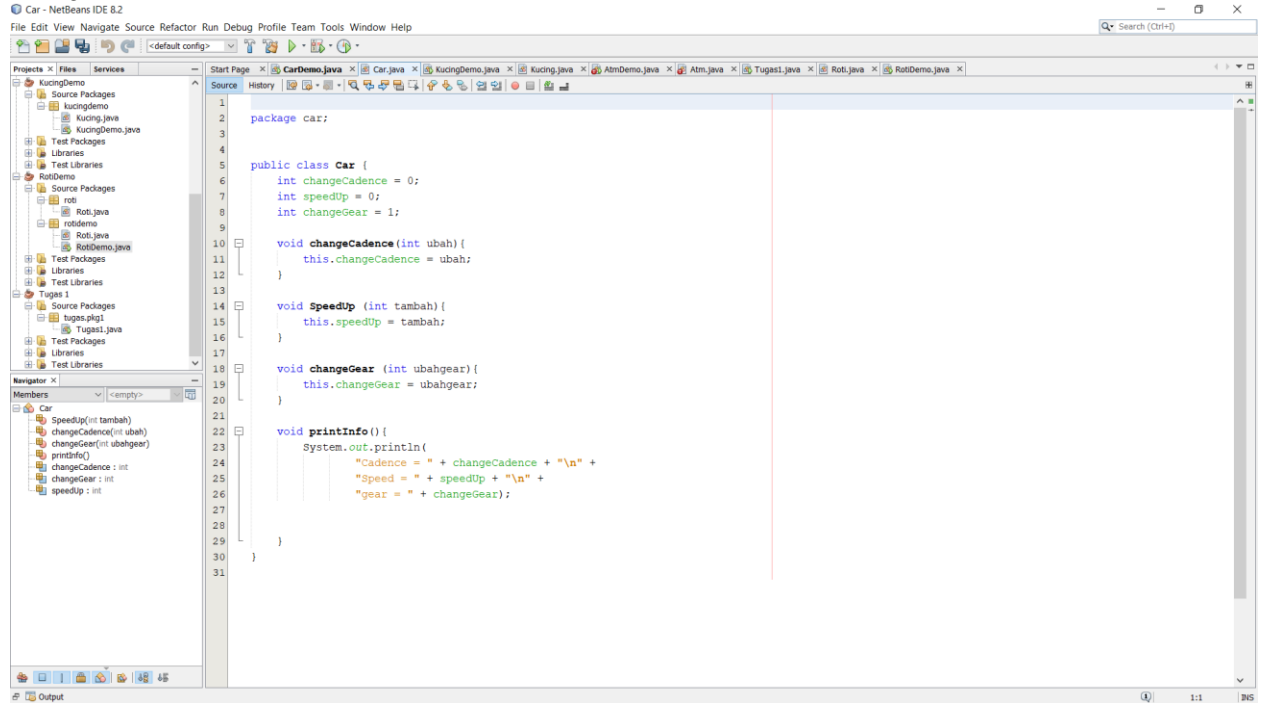
BUILD SUCCESSFUL (total time: 0 seconds)
|
```

## 2. Gambarkan class diagram dari class Roti Demo



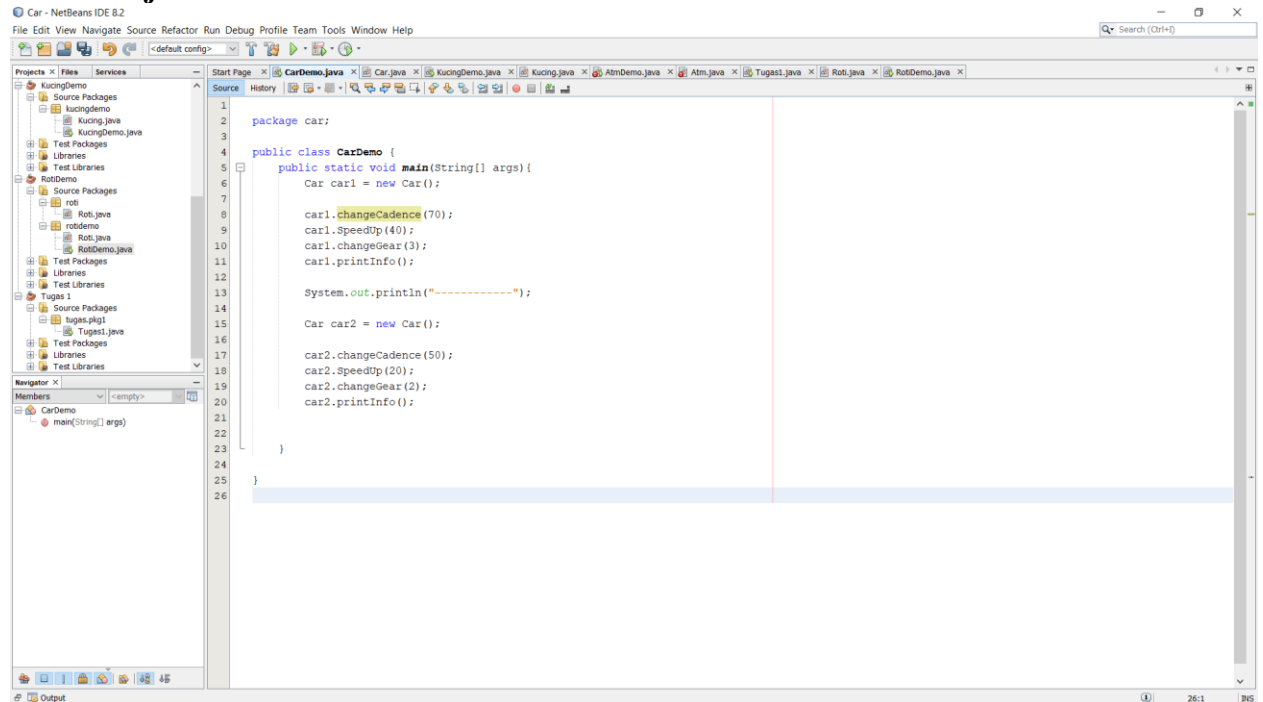
3. Buatlah satu class baru yang bisa digunakan sebagai template/ blueprint dari class **CarDemo** seperti terlihat pada Program 3. Class baru tersebut tidak memiliki fungsi **main()**.

## Car.java



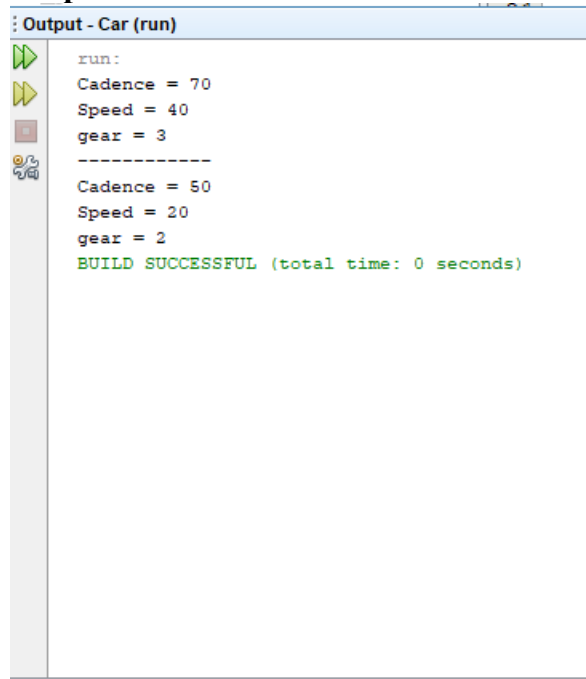
```
1 package car;
2
3
4
5 public class Car {
6     int changeCadence = 0;
7     int speedUp = 0;
8     int changeGear = 1;
9
10    void changeCadence(int ubah){
11        this.changeCadence = ubah;
12    }
13
14    void SpeedUp (int tambah){
15        this.speedUp = tambah;
16    }
17
18    void changeGear (int ubahgear){
19        this.changeGear = ubahgear;
20    }
21
22    void printInfo(){
23        System.out.println(
24            "Cadence = " + changeCadence + "\n" +
25            "Speed = " + speedUp + "\n" +
26            "gear = " + changeGear);
27    }
28
29
30 }
31
```

## Cardemo.java



```
1 package car;
2
3
4 public class Cardemo {
5     public static void main(String[] args){
6         Car car1 = new Car();
7
8         car1.changeCadence(70);
9         car1.SpeedUp(40);
10        car1.changeGear(3);
11        car1.printInfo();
12
13        System.out.println("-----");
14
15        Car car2 = new Car();
16
17        car2.changeCadence(50);
18        car2.SpeedUp(20);
19        car2.changeGear(2);
20        car2.printInfo();
21
22    }
23
24 }
25
26
```

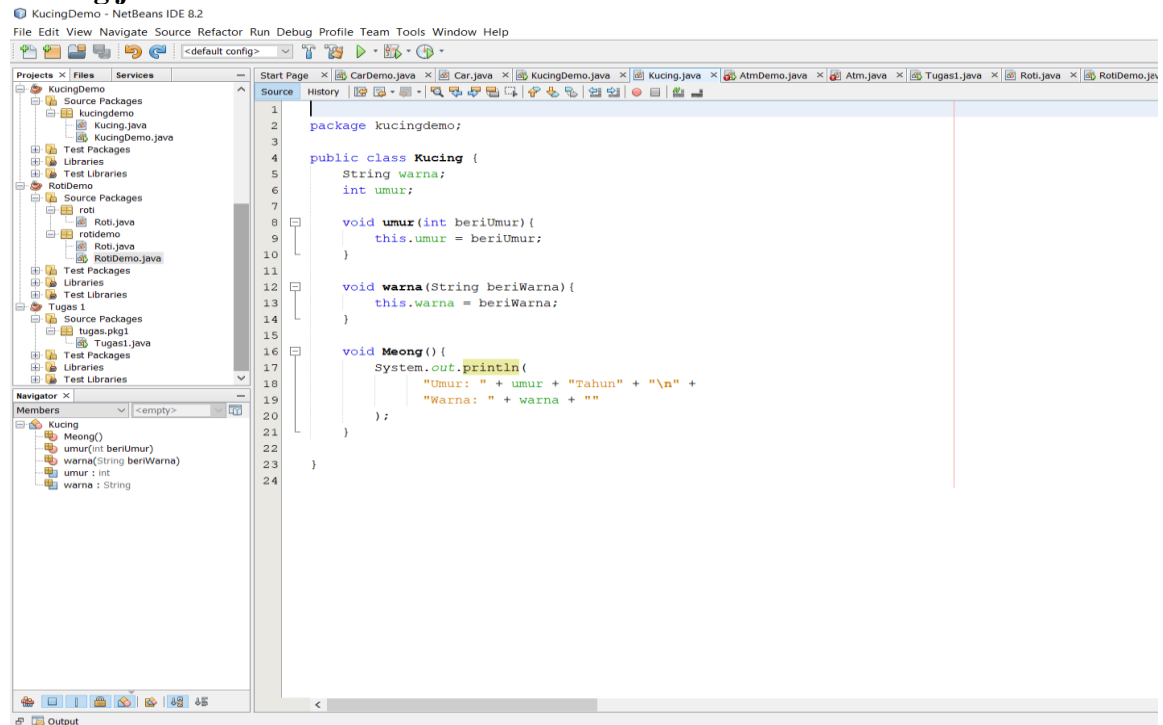
## Output



```
run:
Cadence = 70
Speed = 40
gear = 3
-----
Cadence = 50
Speed = 20
gear = 2
BUILD SUCCESSFUL (total time: 0 seconds)
```

4. Buatlah suatu class yang dapat merepresentasikan sifat-sifat dari object Kucing. Object ini memiliki field/variable/properties berupa umur, warna bulu dan method berupa meong() dan umur()

## Kucing.java

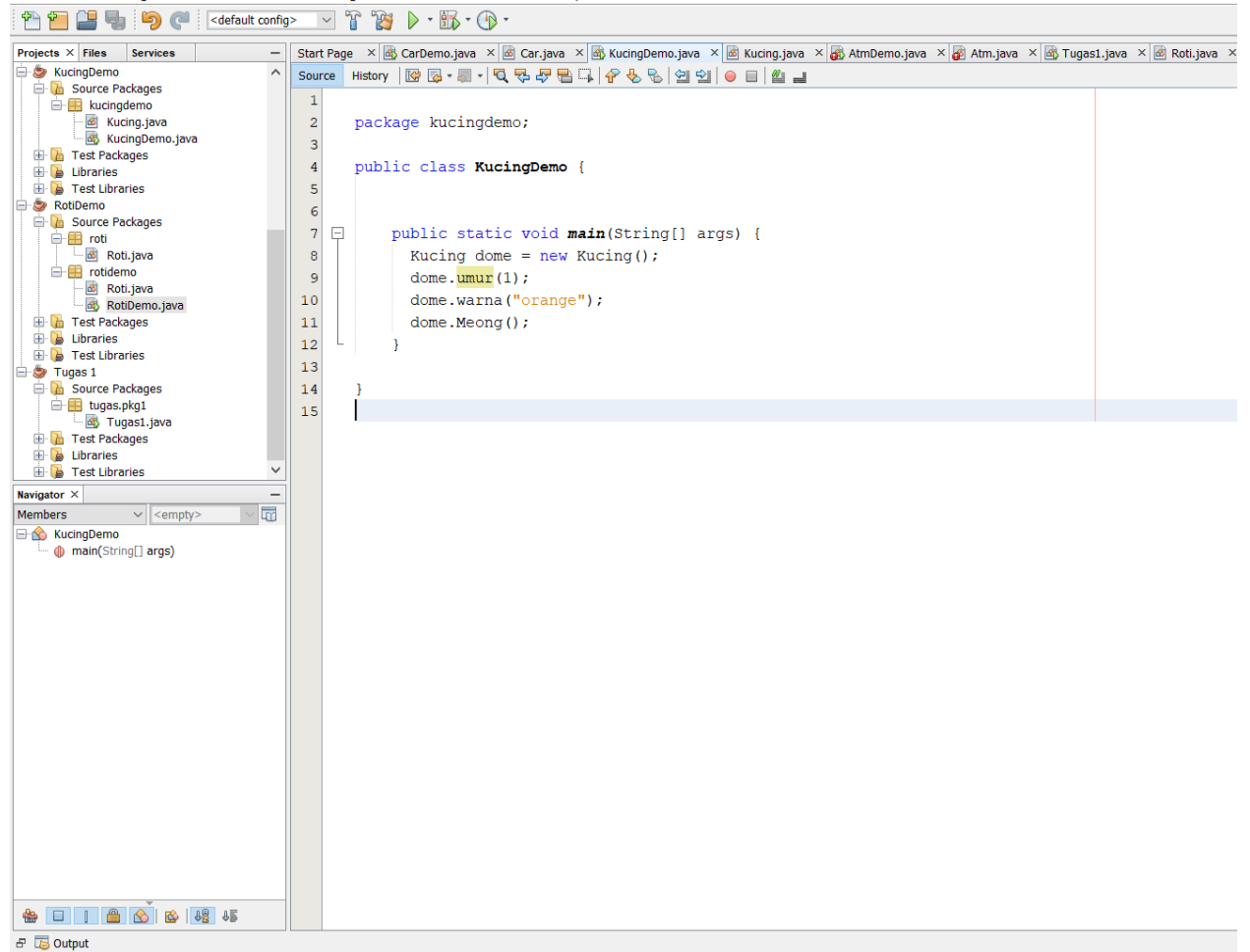


```
KucingDemo - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Projects Files Services
KucingDemo
  Source Packages
    kucingdemo
      Kucing.java
      KucingDemo.java
  Test Packages
  Libraries
  Test Libraries
  RotiDemo
    Source Packages
      roti
      Roti.java
      rotidemo
      Roti.java
      RotiDemo.java
    Test Packages
    Libraries
    Test Libraries
  Tugas1
    Source Packages
      tugas.pkg1
      Tugas1.java
    Test Packages
    Libraries
    Test Libraries
Navigator
Members
Kucing
  meong()
  umur(int beriUmur)
  warna(String beriWarna)
  umur : int
  warna : String
Source
1
2 package kucingdemo;
3
4 public class Kucing {
5     String warna;
6     int umur;
7
8     void umur(int beriUmur){
9         this.umur = beriUmur;
10    }
11
12    void warna(String beriWarna){
13        this.warna = beriWarna;
14    }
15
16    void meong(){
17        System.out.println(
18            "Umur: " + umur + "Tahun" + "\n" +
19            "Warna: " + warna + ""
20        );
21    }
22
23 }
24
```

# Kucingdemo.java

KucingDemo - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help



```
1
2 package kucingdemo;
3
4 public class KucingDemo {
5
6     public static void main(String[] args) {
7         Kucing dome = new Kucing();
8         dome.umur(1);
9         dome.warna("orange");
10        dome.Meong();
11    }
12
13 }
14
15
```

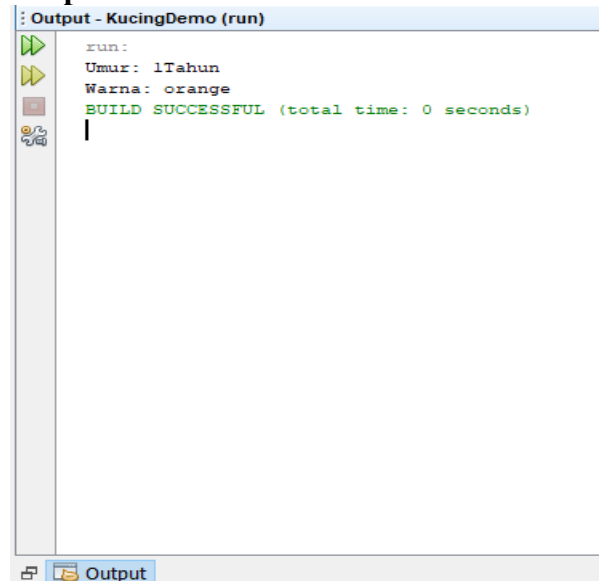
Navigator

Members

- KucingDemo
  - main(String[] args)

Output

## Output



```
run:
Umur: 1Tahun
Warna: orange
BUILD SUCCESSFUL (total time: 0 seconds)
```

Output

5. Salah satu aplikasi PBO yang sangat umum adalah berupa aplikasi keuangan. Bank Account (Rekening Bank) adalah salah satu hal yang dapat dijadikan sebagai suatu object di dalam PBO.

- a. Buatlah suatu class yang dapat merepresentasikan Object Rekening tersebut. Variabel dari object ini adalah saldo, no\_rekening, nama dan method berupa cek\_saldo(), menabung(), menarik(), dan transfer().

### Atm.java

```
1 package Bank;
2 import java.util.Scanner;
3
4 public class Atm {
5     String nama, norekening;
6     int saldo, tabung, tarik, menu, transfer;
7     Scanner masukan = new Scanner(System.in);
8     //*****
9
10    void pilihmenu() {
11        while (true) {
12            System.out.println("Pilih Menu: "+ "\n"
13 + "1.Menabung"+ "\n"
14 + "2.Menarik"+ "\n"
15 + "3.Transfer"+ "\n"
16 + "4.Info Atm anda"+ "\n"
17 + "5.Exit");
18            menu = masukan.nextInt();
19
20
21            if (menu == 1) {
22                this.menabung();
23            }
24            else if (menu == 2) {
25                this.menarik();
26            }
27            else if (menu == 3) {
28                this.transfer();
29            }
30            else if (menu == 4) {
31                this.infoatm();
32            }
33            else if (menu == 5) {
34                System.out.println("Anda Berhasil Keluar dari ATM");
35                break;
36            }
37            else {
38                System.out.println("Menu Salah");
39            }
40        }
41    }
42 }
```

```
Atm.java x AtmDemo.java
45 //#####
46
47
48 void inputnama() {
49     System.out.print("Masukan Nama :");
50     nama = masukan.nextLine();
51
52 }
53 void input_norekening() {
54     System.out.print("Masukan No Rekening Anda: ");
55     norekening = masukan.nextLine();
56 }
57 void cek_saldo() {
58     System.out.println("Saldo Anda: ");
59     saldo = masukan.nextInt();
60 }
61 void menabung() {
62     System.out.println("Masukan Nominal Tabung: ");
63     tabung = masukan.nextInt();
64     saldo = saldo + tabung;
65 }
66 void menarik() {
67     System.out.println("Masukan Nominal Tarik: ");
68     tarik = masukan.nextInt();
69     saldo = saldo - tarik;
70 }
71 void transfer() {
72     System.out.println("Masukan Nominal Transfer: ");
73     transfer = masukan.nextInt();
74     saldo = saldo - transfer;
75 }
76
77
78 void infoatm() {
79     System.out.println(
80         "Nama: " + nama + "\n" +
81         "Saldo: " + saldo
82     );
83     System.out.println("#####");
84 }
85
86
```



## Atmdemo.java

```
Atm.java  AtmDemo.java
1 package Bank;
2
3 public class AtmDemo {
4     public static void main (String[] args) {
5         Atm orangnya = new Atm();
6         orangnya.inputnama();
7         orangnya.input_norekening();
8         orangnya.pilihmenu();
9
10
11     }
12 }
```

6. Perhatikan Class String yang ada di dalam dokumentasi Java. Sebutkan daftar variable dan fungsi/method yang dimiliki oleh Class String tersebut.

- **Variabel**  
Nama, Norekening(String)  
Saldo, tabung, tarik, menu, transfer(Int)
- **Method**  
Void inputnama()  
Void input\_norekening()  
Void cek\_saldo()  
Void menabung()  
Void menarik()  
Void transfer()  
Void infoatm()