# LAPORAN PRAKTIKUM

# PEMROGRAMAN BERORIENTASI OBJECT

# JOBSHEET 3



# Sahrul Ramadhani

# 244107020058

# 2H

# D-IV Teknik Informatika

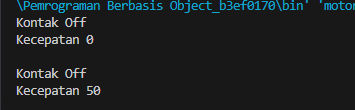
# Teknologi Informasi

# Politeknik Negeri Malang

# 2025

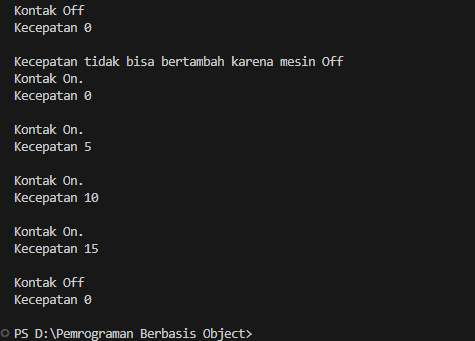
# Percobaan 1

Hasil RUN:



# Percobaan 2

Hasil RUN:



# Pertanyaan Percobaan 2

## Pada class TestMobil?? motorDemo ya?, saat kita menambah kecepatan untuk pertama kalinya, mengapa muncul peringatan “Kecepatan tidak bisa bertambah karena Mesin Off!”?

## Mengapat atribut kecepatan dan kontakOn diset private?

## Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

# Jawaban:

## Karena di Class Motor ada If Else

public void tambahKeceptan (){

if (kontakOn == true) {

kecepatan += 5;

}

else {

System.out.println("Kecepatan tidak bisa bertambah karena mesin Off ");

}

}

Artinya jika kontakOn == False maka akan ada peringatan Kecepatan tidak bisa bertambah karena mesin Off

## Ya agar User ketika menambahkan kecepatan tidak bisa langsung instan, misalnya dari 0-100 jadi secara bertahap dengan method tambahKecepatan setiap memanggil method akan bertambah +=5

## Hanya perlu ubah pada method tambahKecepatan sebagaiberikut:

public void tambahKecepatan (){

if (kontakOn == true) {

if (kecepatan == 100) {

System.out.println("Kecepatan maksimal 100! \n");

}

else {kecepatan += 5;

}

}

else {

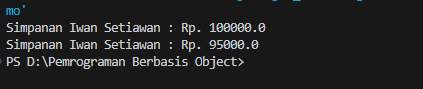
System.out.println("Kecepatan tidak bisa bertambah karena mesin Off");

}

}

# Percobaan 3

Hasil RUN:

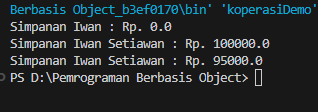


# Percobaan 4

Hasil RUN:

## 

## Ditambahkan kontruktor dengan access modifier default



# Pertanyaan Percobaan 3 & Percobaan 4

## 1. Apa yang dimaksud getter dan setter?

## Apa kegunaan dari method getSimpanan()?

## Method apa yang digunakan untk menambah saldo?

## Apa yand dimaksud konstruktor?

## Sebutkan aturan dalam membuat konstruktor?

## Apakah boleh konstruktor bertipe private?

## Kapan menggunakan parameter dengan passsing parameter?

## Apa perbedaan atribut class dan instansiasi atribut?

## Apa perbedaan class method dan instansiasi method?

JAWABAN:

## - Getter = method untuk mengambil/melihat isi data (misalnya saldo, nama).

- Setter = method untuk mengubah atau mengisi data.

## Untuk melihat berapa jumlah saldo/tabungan yang dimiliki anggota saat ini.

## Method setor() digunakan untuk menambah saldo/simpanan.

## Konstruktor adalah method khusus yang otomatis dipanggil saat membuat objek baru.Fungsinya untuk mengisi data awal.

## Nama konstruktor harus sama dengan nama class.

* Tidak boleh punya tipe kembalian (void, int, dll).
* Boleh punya parameter (opsional).

## Boleh, tapi sangat jarang dipakai. Biasanya digunakan untuk membatasi pembuatan objek, misalnya di Singleton Pattern (supaya hanya ada 1 objek).

## Saat kita ingin mengisi data awal langsung dari luar class ketika objek dibuat. Contoh: new Anggota("Budi", "Malang", 100000);

## - Atribut class (static) → data milik bersama semua objek (contoh: tarif bunga koperasi).

- Atribut instansiasi (non-static) → data milik masing-masing objek (contoh: saldo tiap anggota berbeda-beda).

## - Class method (static) → bisa dipanggil langsung pakai nama class, tanpa bikin objek.

- Instansiasi method (non-static) → harus lewat objek yang sudah dibuat.

# TUGAS

## Hasil RUN kode :



## Pada program diatas, pada class EncapTest kita mengeset age dengan nilai 35, namun pada saat ditampilkan ke layar nilainya 30, jelaskan mengapa?

Karena encapDemo ada If Else yang fungsinya jika memasukan Age > 30 maka akan di simpan 30

if (newAge > 30){

            age = 30;

        }

## Kode saya ubah Jika <17 atau >30 maka akan ada peringatan minimum maximum dan otomatis keluar program

public void setAge (int newAge){

if (age <18 || age>30) {

System.out.println("Minimum age is 18 and maximum age is 30");

System.exit(0);

}

else {

age = newAge;

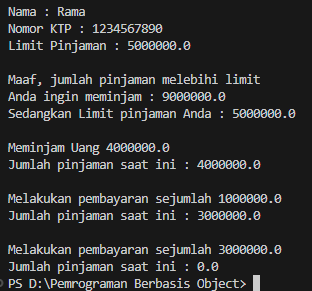
}

}

## Hasil RUN :



## Berikut adalah hasil RUN dari Tugas untuk membuat sebuah sistem peminjaman, yang bisa di lihat kodenya pada folder Koperasi



## Hasil Modify Kode pada method membayar:

public void membayar (float uang){

if (uang < 0.1\*pinjaman) {

System.out.println("jumlah nominal angsuran minimal 10% dari totalpinjaman");

System.out.println("anda harus membayar minimal : " + 0.1\*pinjaman);

} else {

pinjaman -= uang;

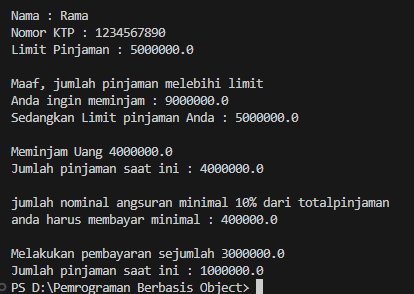
System.out.println("Melakukan pembayaran sejumlah " + uang);

System.out.println("Jumlah pinjaman saat ini : " + pinjaman);

}

}

Hasil RUN :



## Hasil Modify Code dengan menambahkan Scanner dan Syntax Inputnya nya :

import java.util.Scanner;

public class testKoperasi {

public static void main(String[] args) {

float pinjam;

float bayar;

Scanner input= new Scanner(System.in);

koperasi Rama = new koperasi ("1234567890", "Rama", 5000000);

System.out.println("");

System.out.println("Nama : " + Rama.getNama());

System.out.println("Nomor KTP : " + Rama.getNomorKTP());

System.out.println("Limit Pinjaman : " + Rama.getLimitPinjaman());

System.out.println("");

System.out.println("");

System.out.print("masukkan jumlah pinjaman : ");

pinjam = input.nextFloat();

Rama.meminjam(pinjam);

System.out.println("");

System.out.print("masukkan jumlah pinjaman : ");

pinjam = input.nextFloat();

Rama.meminjam(pinjam);

System.out.println("");

System.out.print("masukkan jumlah angsuran : ");

bayar = input.nextFloat();

Rama.membayar(bayar);

System.out.println("");

System.out.print("masukkan jumlah angsuran : ");

bayar = input.nextFloat();

Rama.membayar(bayar);

}

}

HASIL RUN :

