**Oracle Academy Objected Oriented Programming**

Here I just create main.java as the main executor of the code and all the practice .java files and ignore other classes that trigger to become the main one. Mostly exercise question/asking, using Indonesia language.

**Jfo\_2\_1**

Latihan 1, Bagian 1 dan 2

* Sahabat Anda, Clinton, memiliki rencana untuk akhir Pekan.
* Lihat emailnya dan pikirkan tentang langkah apa yang perlu dilakukan untuk mewujudkan rencana tersebut:

*Hai, bro*

*Ada pameran Computer History spesial di City Museum bulan*

*ini. Kami berencana akan pergi di hari Jumat ini, jam 5 sore.*

*Kamu mau ikut juga? Menurutku palign tepat kalau ke sana*

*naik kereta bawah tanah.*

*-Clinton*

* Selesaikan bagan dengan menulis setidaknya satu item untuk setiap bagian.

1. Pernyataan. Apa yang ditanyakan pada email Clinton ?

= Clinton menanyakan apakah saya ingin ikut pergi ke pameran Computer History di City Museum pada hari Jumat jam 5 sore dengan menggunakan kereta bawah tanah.

1. Mendesain Rencana. Apa yang perlu di pertimbangkan sebelum pergi ?
2. Menyusun jadwal agar bisa selesai sekolah atau bekerja sebelum pukul 5 sore.
3. Mengetahui lokasi dan jadwal keberangkatan kereta bawah tanah.
4. Mengatur titik pertemuan dengan Clinton dan teman-teman lainnya.
5. Menyediakan uang untuk tiket masuk pameran dan transportasi.
6. Memastikan pameran masih berlangsung pada hari dan jam tersebut.
7. Menginformasikan kepada orang tua atau wali jika diperlukan.
8. Pengujian. Bagaimana anda tahu rencana tersebut berhasil ?
9. Semua orang yang berencana ikut berhasil sampai di City Museum tepat waktu.
10. Bisa masuk ke pameran dan menikmati acara sesuai harapan.
11. Tidak ada kendala besar seperti keterlambatan kereta atau pameran ditutup.
12. Semua peserta merasa senang dan puas dengan kegiatan tersebut.
13. Menerapkan rencana. Tindakan apa yang anda lakukan ?
14. Membalas email Clinton dan mengonfirmasi keikutsertaan.
15. Mengecek jadwal kereta dan rute menuju City Museum.
16. Menyiapkan perlengkapan dan uang secukupnya.
17. Datang ke titik kumpul tepat waktu dan berangkat bersama.
18. Menikmati acara di pameran dan menjaga komunikasi dengan kelompok.

Latihan 2, bagian 1 dan 2

* Lihat kembali email Clinton, jika Anda membutuhkannya untuk latihan

*Hai, bro*

*Ada pameran Computer History spesial di City Museum bulan ini. Kami berencana akan pergi di hari Jumat ini, jam 5 sore. Kamu mau ikut juga? Menurutku palign tepat kalau ke sana naik kereta bawah tanah. -Clinton*

* Selesaikan bagan, Bayangkan apa yang mungkin terjadi pada malam Anda di

museum jika Anda lupa langkah tertentu:

1. Pernyataan. Apa yang ditanyakan pada email Clinton ?

Isi:

Clinton bertanya apakah saya ingin ikut ke pameran Computer History di City Museum pada hari Jumat pukul 5 sore, menggunakan kereta bawah tanah.

Jika dilupakan:

Saya mungkin tidak tahu tentang rencana tersebut dan akhirnya tidak ikut, atau malah salah memahami waktu dan tempat acara.

1. Mendesain Rencana. Apa yang perlu di pertimbangkan sebelum pergi ?

Isi:

Mengetahui lokasi dan waktu pameran.

Menyusun jadwal pribadi agar bebas jam 5 sore.

Mencari tahu jadwal kereta bawah tanah.

Menyiapkan uang untuk transportasi dan tiket masuk.

Menyusun rencana pertemuan dengan Clinton dan teman-teman.

Memastikan museum buka dan pameran berlangsung.

Jika dilupakan:

Saya bisa datang terlambat atau ke tempat yang salah.

Bisa kehabisan tiket atau tidak punya cukup uang.

Kehilangan rombongan dan harus mencari mereka sendirian.

Museum bisa saja tutup atau acaranya sudah selesai.

1. Pengujian. Bagaimana anda tahu rencana tersebut berhasil ?

Isi:

Sampai di museum tepat waktu bersama teman-teman.

Dapat masuk dan mengikuti pameran sesuai rencana.

Tidak ada kendala besar selama perjalanan.

Semua orang merasa senang dan puas.

Jika dilupakan:

Tidak bisa mengevaluasi apakah rencana sukses atau tidak.

Tidak belajar dari pengalaman jika terjadi kesalahan.

Tidak bisa memperbaiki rencana untuk kegiatan serupa di masa depan.

1. Menerapkan rencana. Tindakan apa yang anda lakukan ?

Isi:

Membalas email Clinton untuk konfirmasi.

Mencatat waktu dan lokasi keberangkatan.

Mengecek jadwal kereta dan membeli tiket.

Berangkat tepat waktu dan bertemu dengan kelompok.

Menikmati pameran bersama teman-teman.

Jika dilupakan:

Bisa tidak ikut karena lupa konfirmasi.

Tertinggal kereta dan tidak sampai ke museum.

Tidak tahu harus pergi ke mana atau dengan siapa.

Kegiatan jadi berantakan atau bahkan batal bagi saya.

Latihan 3, Bagian 1 dan 2 (**Tidak bisa dilakukan karena Acces Denied**)

Unduh OldGameVersions.zip, buka zip, dan

putar rekaman beberapa versi game selama

pengembangannya:

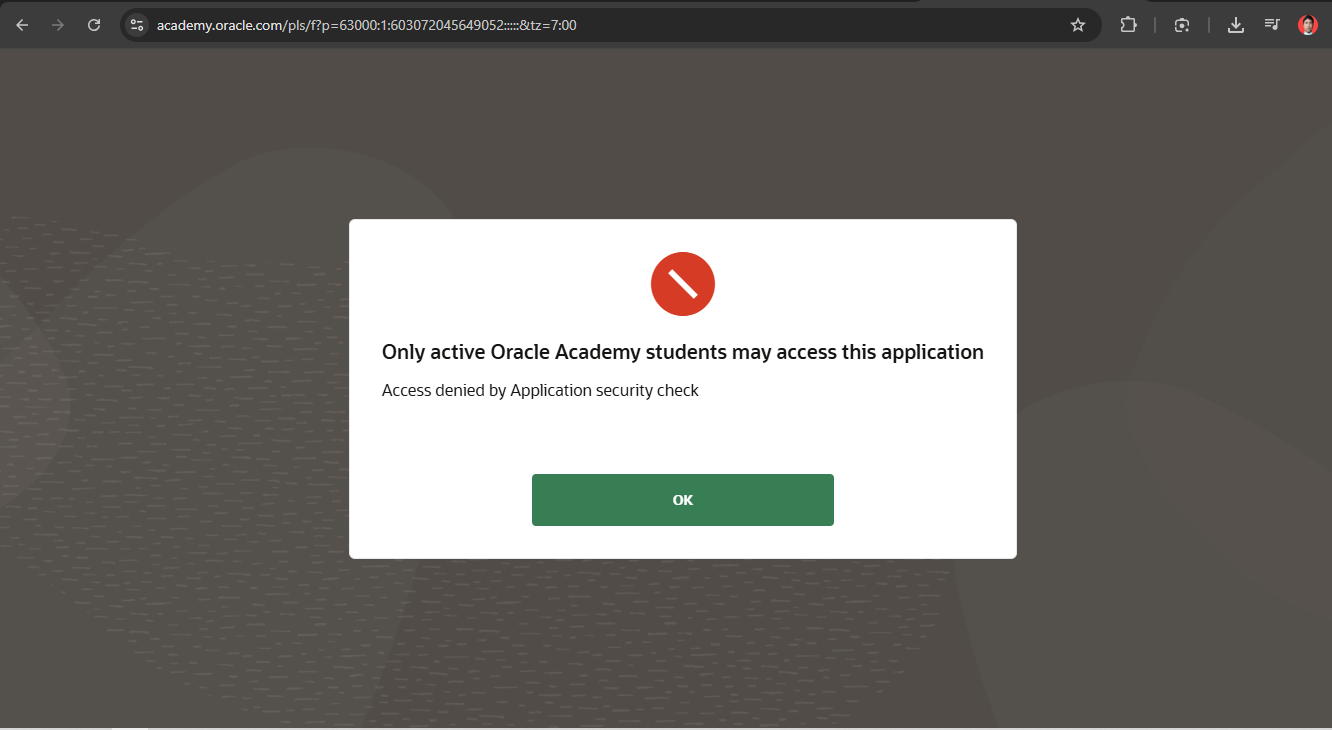
−16 Agustus 2013 (08-16-13.mp4)

−22 Agustus 2013 (08-22-13.mp4)

−27 September 2013 (09-27-13.mp4)

−16 Oktober 2013 (10-16-13.mp4)

−21 November 2013 (11-21-13.mp4)



**Jfo\_3\_1**

Latihan 1

• Jalankan JavaLibs.jar

• Pertimbangkan jenis data yang diminta oleh program

ini

import javax.swing.JOptionPane;

public class javlibs {

public static void javlibsederhana() {

// ambil input si user

String name = JOptionPane.showInputDialog("Masukkan nama kamu:");

String ageStr = JOptionPane.showInputDialog("Masukkan umur kamu:");

int age = Integer.parseInt(ageStr); // konversi ke integer

String color = JOptionPane.showInputDialog("Masukkan warna favorit:");

String hobby = JOptionPane.showInputDialog("Apa hobi kamu?");

String food = JOptionPane.showInputDialog("Masakan favorit kamu:");

String dream = JOptionPane.showInputDialog("Apa impian terbesar kamu?");

String adjective = JOptionPane.showInputDialog("Masukkan kata sifat (contoh: ceria, cepat, kuat):");

// cerita yang dihasilkan berdasarkan masukan si usernya

String story = "Pada suatu hari, ada seorang bernama " + name + " yang berumur " + age + " tahun.\n" +

name + " sangat menyukai warna " + color + " dan punya hobi " + hobby + ".\n" +

"Setiap sore, " + name + " menikmati masakan favoritnya, yaitu " + food + ".\n" +

"Dengan semangat yang " + adjective + ", " + name + " terus mengejar impiannya menjadi " + dream + ".\n" +

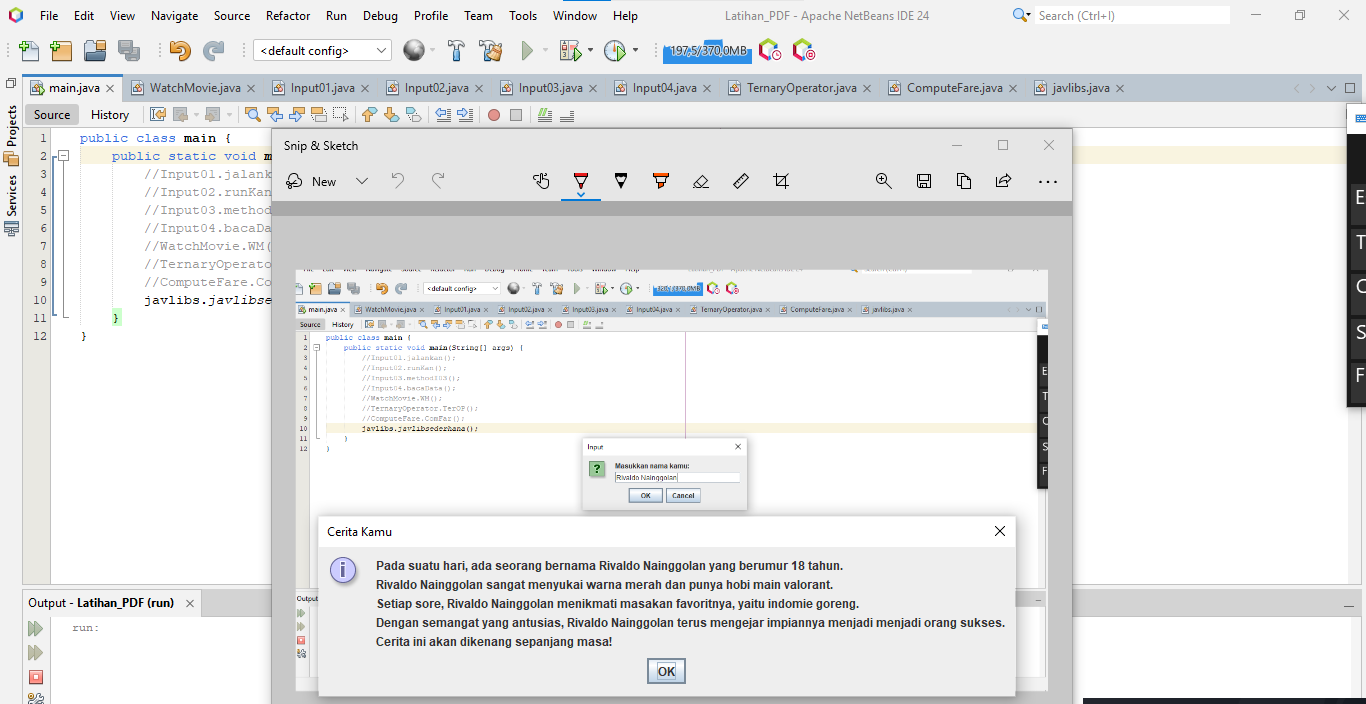
"Cerita ini akan dikenang sepanjang masa!";

// tampilkan hasil cerita

JOptionPane.showMessageDialog(null, story, "Cerita Kamu", JOptionPane.INFORMATION\_MESSAGE);

}

}



Latihan 2

* Buat proyek baru dan tambahkan file Variables02.java ke proyek
* Ikuti langkah-langkah dalam latihan
* Jalankan program antara setiap langkah dan amati Outputnya
* Program Anda akan menghasilkan output berikut:

−Setelah Langkah 1) :

puppy

puppy

−Setelah Langkah 2):

kitty

Kitty

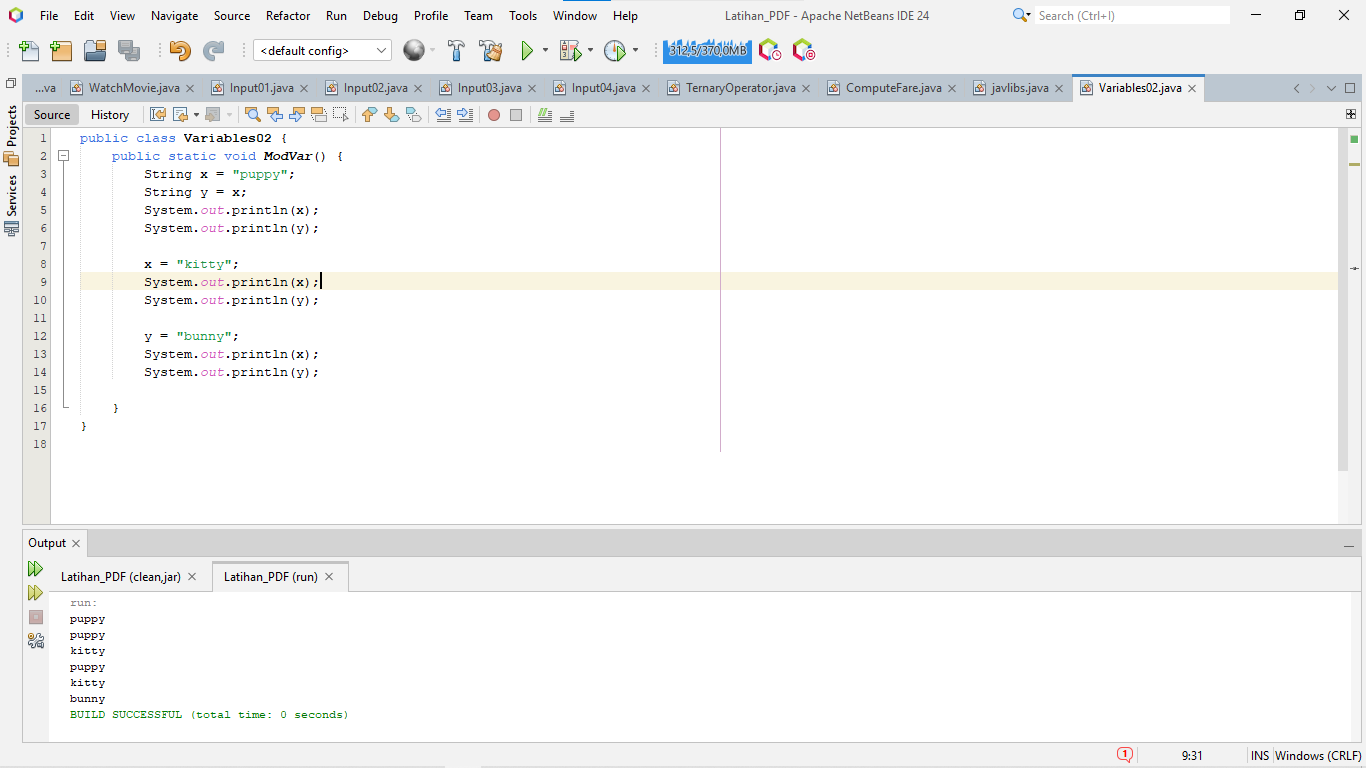
−Setelah Langkah 3):

kitty

bunny

Sifat Program Baris demi Baris

* Dari baris 8 dan selanjutnya, x selalu sama dengan "kitty" hingga …
* •Baris 14 dan selanjutnya di mana x selalu sama dengan "bunny"



public class Variables02 {

public static void ModVar() {

String x = "puppy";

String y = x;

System.out.println(x);

System.out.println(y);

x = "kitty";

System.out.println(x);

System.out.println(y);

y = "bunny";

System.out.println(x);

System.out.println(y);

}

}

Latihan 3, Bagian 1, 2 dan 3

* Buat proyek dan tambahkan file Variables03.java ke proyek

−Ada enam kesalahan dalam program ini

−Dapatkah Anda memperbaiki kesalahan tersebut agar program menghasilkan output berikut?

bool = true

intVar1 = 1

intVar2 = 2

intVar3 = 3

doubleVar1 = 1.1

doubleVar2 = 2.1

doubleVar3 = 3.1

doubleVar4 = 4.1

stringVar1 = 11

stringVar2 = 22

Petunjuk 1, IDE Anda menggarisbawahi kode problematik

−Tahan kursor pada kode atau ikon di margin kiri untuk detailnya

−IDE Anda mungkin menyoroti kemungkinan solusi

−Klik ikon di margin kiripublic class Variables03 {

public static void main(String() args) {

int intVarl = true;

int intVar2 = 2;

intVar3 = 3;

double doubleVarl, doubleVar2, doubleVar3, doubleVar4;

doubleVar1 = 1.1;

doublevAr2 = 2.1;

double doubleVar3 = 3.1;

}

}

Petunjuk 2, Solusi yang disarankan IDE Anda terkadang kurang baik

−Jangan sepenuhnya mengandalkan solusi yang disoroti IDE Anda, Keterampilan pemecahan masalah Anda dapat menjadi sumber daya yang luar biasa

Udah di perbaiki :

public class Variables03 {

public static void FixVarNama() {

boolean bool = true;

int intVar1 = 1;

int intVar2 = 2;

int intVar3 = 3;

double doubleVar1 = 1.1;

double doubleVar2 = 2.1;

double doubleVar3 = 3.1;

double doubleVar4 = 4.1;

String stringVar1 = "11";

String stringVar2 = "22";

System.out.println("bool = " + bool);

System.out.println("intVar1 = " + intVar1);

System.out.println("intVar2 = " + intVar2);

System.out.println("intVar3 = " + intVar3);

System.out.println("doubleVar1 = " + doubleVar1);

System.out.println("doubleVar2 = " + doubleVar2);

System.out.println("doubleVar3 = " + doubleVar3);

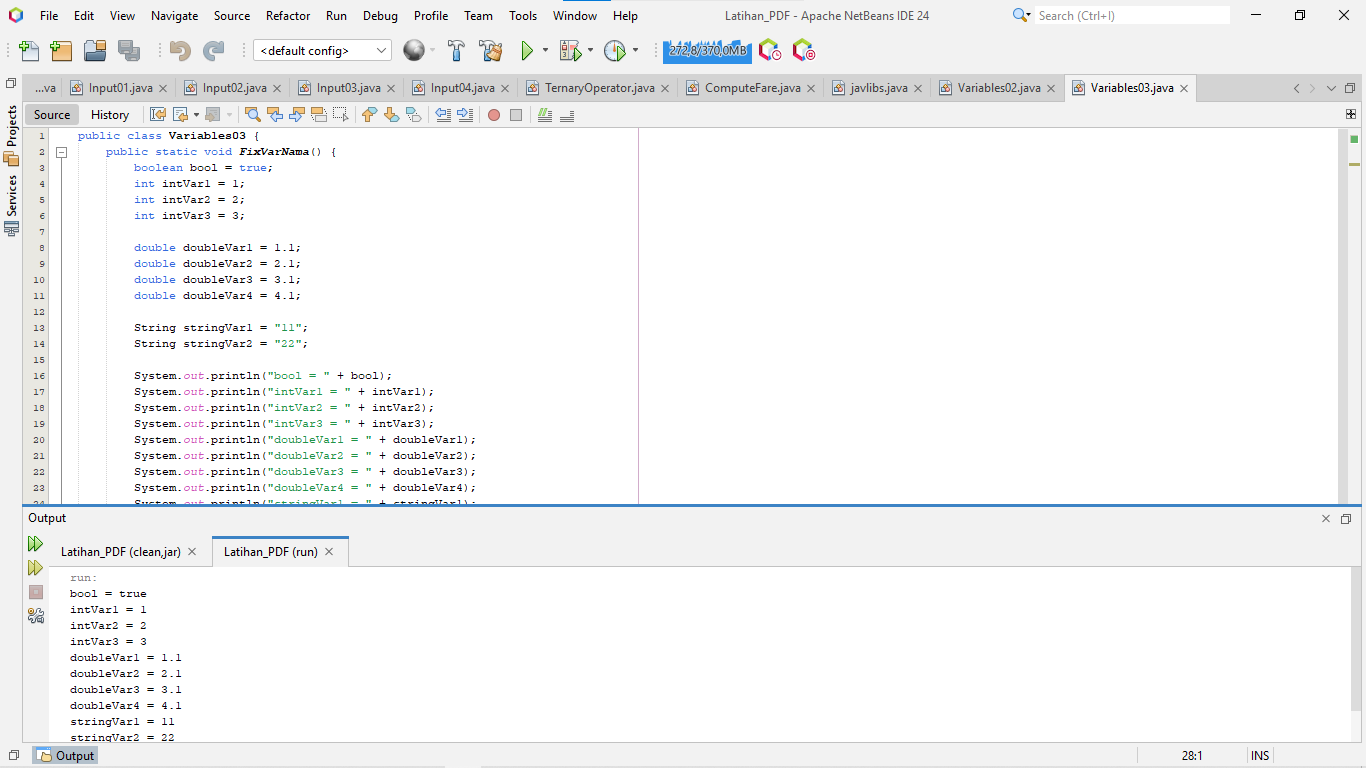
System.out.println("doubleVar4 = " + doubleVar4);

System.out.println("stringVar1 = " + stringVar1);

System.out.println("stringVar2 = " + stringVar2);

}

}



**Jfo\_4\_1**

Latihan 1, Bagian 1,2 dan 3

* Mari kita lihat skenario dan lihat bagaimana kita dapat membuat model komponen yang terlibat:
* Ini Ulang Tahun Alex! Anda telah mengatur satu kelompok yang terdiri dari delapan teman untuk merayakannya di restoran lokal. Saat pesta Anda menerima tagihan, tidak ada yang tahu pasti berapa jumlah yang harus mereka bayar. Anda hanya tahu total setiap orang sebelum pajak (5%) dan tip (15%). Tapi Anda beruntung! Anda membawa laptop dan diminta untuk menulis program yang menghitung total setiap orang
* Buat proyek baru dan tambahkan file Tip01.java ke proyek
* Berikut yang harus dibayar oleh setiap orang sebelum pajak (5%) dan tip (15%):

Person 1: $10 Person 5: $7

Person 2: $12 Person 6: $15 (Alex)

Person 3: $9 Orang yang 7: $11

Person 4: $8 Person 8: $30

* Program Anda akan menghasilkan output berikut:

person1: $12.0

person2: $14.4

person3: $10.8

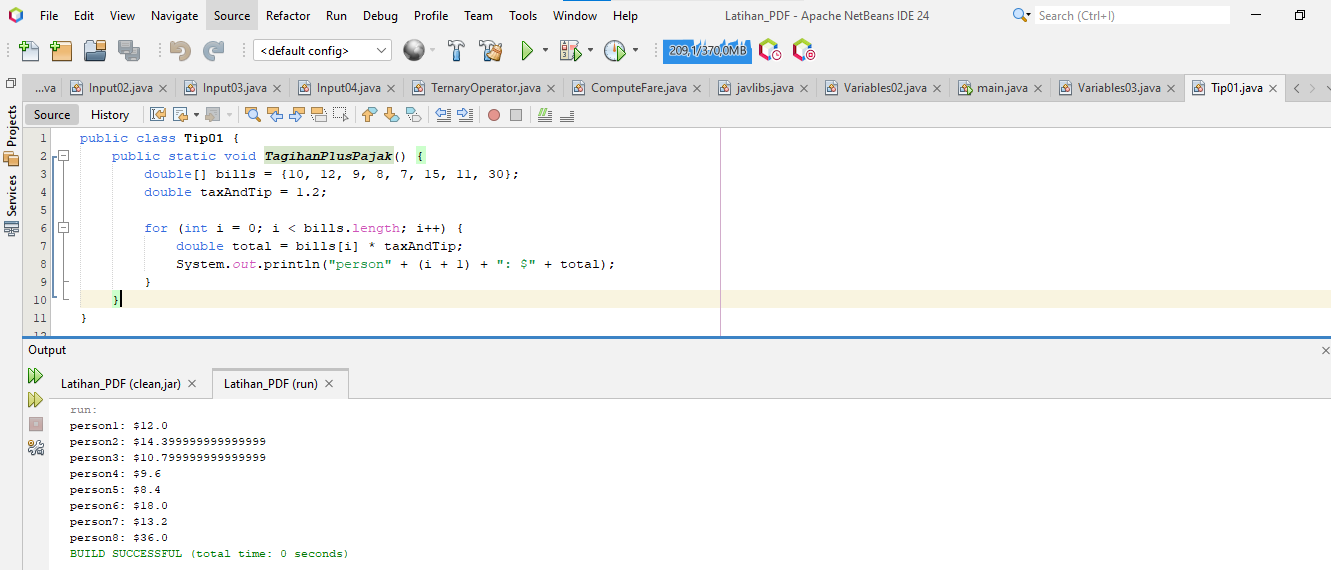
person4: $9.6

person5: $8.4

person6: $18.0

person7: $13.2

person8: $36.0



public class Tip01 {

public static void TagihanPlusPajak() {

double[] bills = {10, 12, 9, 8, 7, 15, 11, 30};

double taxAndTip = 1.2;

for (int i = 0; i < bills.length; i++) {

double total = bills[i] \* taxAndTip;

System.out.println("person" + (i + 1) + ": $" + total);

}

}

}

Latihan 2, Bagian 1 dan 2

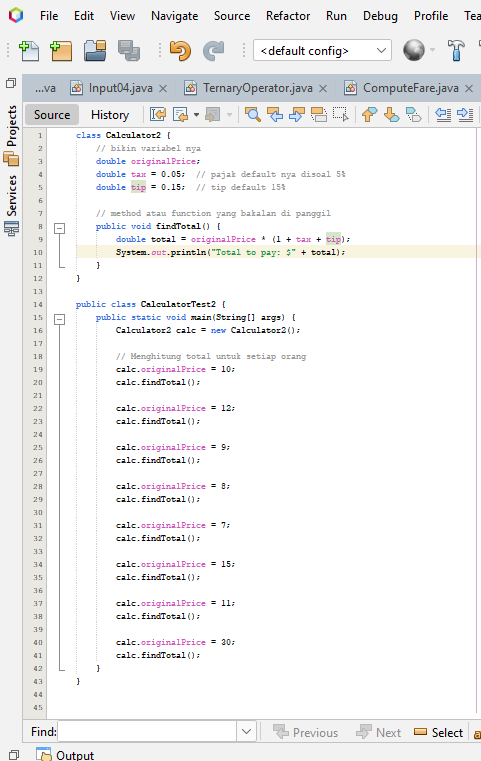
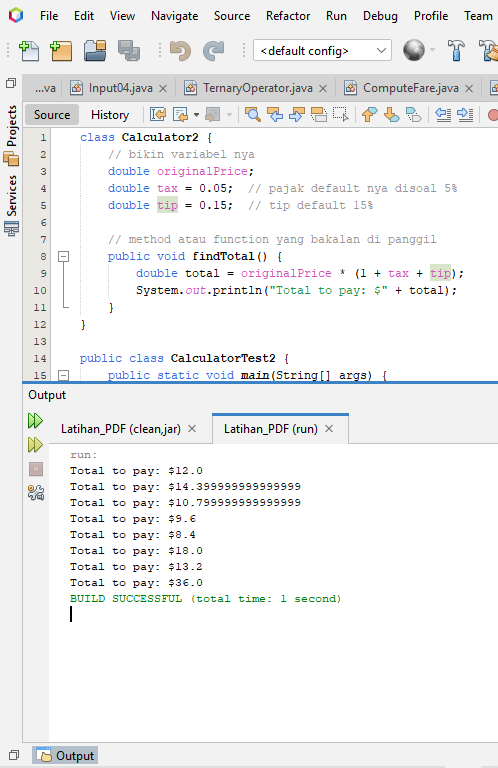
* Buat proyek baru dan tambahkan file CalculatorTest2.java dan Calculator2.java ke proyek
* Lengkap metode findTotal(), yang seharusnya:
* Menghitung total berdasarkan bidang pajak, tip, dan originalPrice
* Mencetak total seseorang
* Dari metode utama:

−Mulai objek Kalkulator yang diberi nama calc

−Amati IDE setelah mengetik "calc"

−Akses bidang dan metode objek ini untuk mencetak total bagi setiap orang di pesta ulang tahun

* Ubah tip dan pajak jika Anda memilih nilai yang berbeda

class Calculator2 {

// bikin variabel nya

double originalPrice;

double tax = 0.05; // pajak default nya disoal 5%

double tip = 0.15; // tip default 15%

// method atau function yang bakalan di panggil

public void findTotal() {

double total = originalPrice \* (1 + tax + tip);

System.out.println("Total to pay: $" + total);

}

}

public class CalculatorTest2 {

public static void main(String[] args) {

Calculator2 calc = new Calculator2();

// ngitung total untuk setiap orang

calc.originalPrice = 10;

calc.findTotal();

calc.originalPrice = 12;

calc.findTotal();

calc.originalPrice = 9;

calc.findTotal();

calc.originalPrice = 8;

calc.findTotal();

calc.originalPrice = 7;

calc.findTotal();

calc.originalPrice = 15;

calc.findTotal();

calc.originalPrice = 11;

calc.findTotal();

calc.originalPrice = 30;

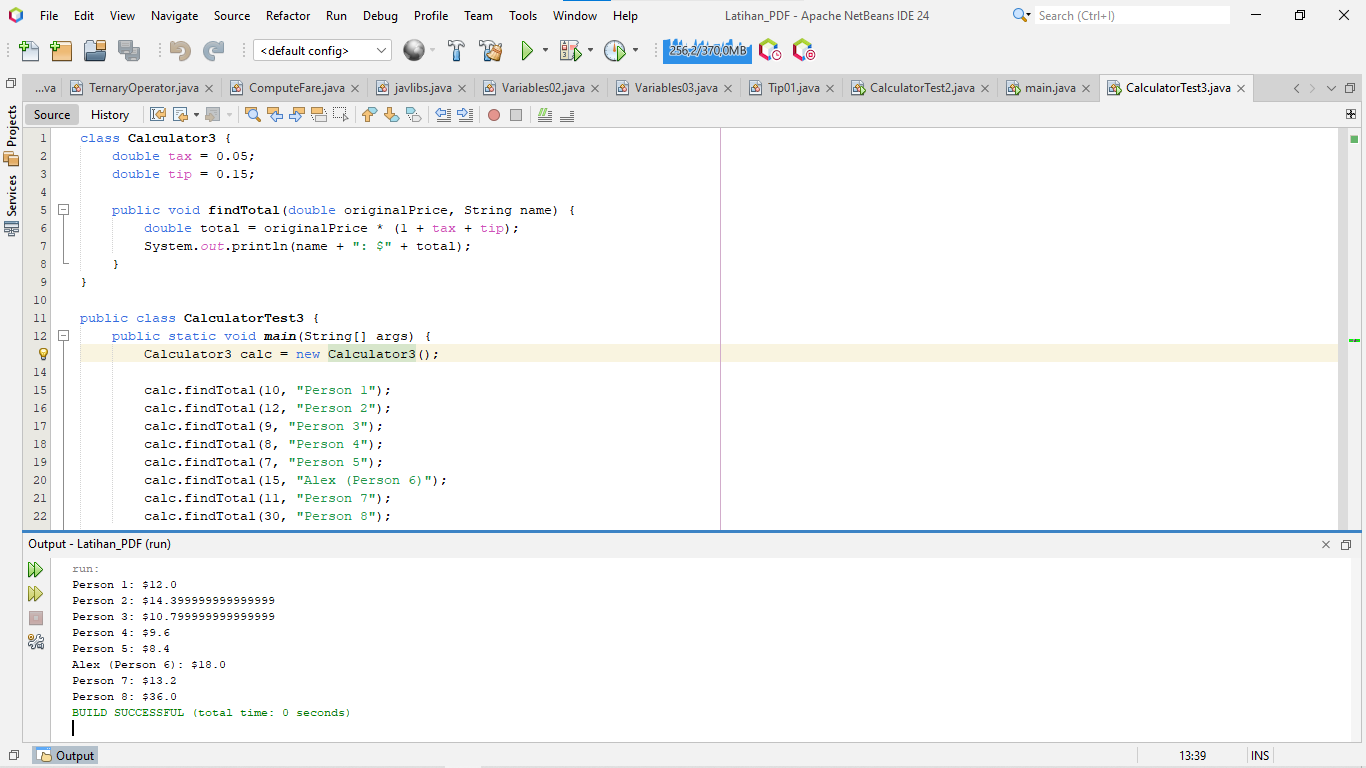
calc.findTotal();

}

}

Latihan 3, Bagian 1 dan 2

* Buat proyek baru dan tambahkan file CalculatorTest3.java dan Calculator3.java ke proyek
* Dari metode utama:
* Gunakan instance objek Kalkulator dan berikan argumen ke findTotal() untuk mencetak total bagi setiap orang
* Petunjuk: Amati metode findTotal() di class Calculator untuk mengetahui berapa banyak argumen yang diterima oleh metode ini
* Siapa pemilik dari setiap total?
* Ubah metode findTotal() untuk menerima argumen nama String tambahan
* Gabungkan pernyataan cetak agar menyertakan nama
* Amati complain IDE di metode utama dan revisi panggilan metode findTotal() Anda



class Calculator3 {

double tax = 0.05;

double tip = 0.15;

public void findTotal(double originalPrice, String name) {

double total = originalPrice \* (1 + tax + tip);

System.out.println(name + ": $" + total);

}

}

public class CalculatorTest3 {

public static void main(String[] args) {

Calculator3 calc = new Calculator3();

calc.findTotal(10, "Person 1");

calc.findTotal(12, "Person 2");

calc.findTotal(9, "Person 3");

calc.findTotal(8, "Person 4");

calc.findTotal(7, "Person 5");

calc.findTotal(15, "Alex (Person 6)");

calc.findTotal(11, "Person 7");

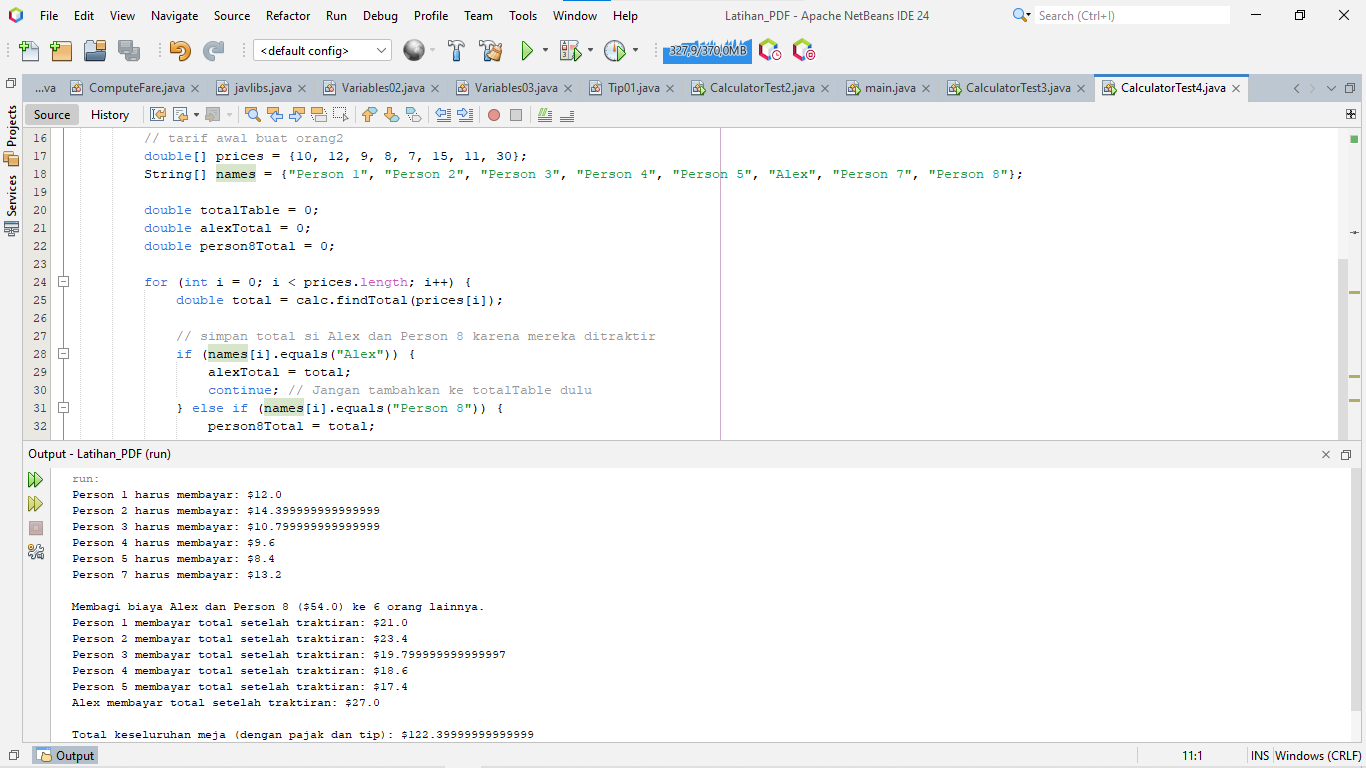
calc.findTotal(30, "Person 8");

}

}

Latihan 4, Bagian 1 dan 2

* Edit solusi Exercise3
* Atau buat proyek baru dan tambahkan file CalculatorTest4.java dan Calculator4.java ke proyek
* Temukan dan cetak total seluruh tabel, termasuk pajak dan tip
* Anda harus mengedit findTotal() yang telah ditentukan agar menghasilkan nilai terhitung
* Person 8 lupa membawa dompet
* Makanan Alex juga dimaksudkan menjadi traktiran ulang tahun
* Modifikasi findTotal() agar biaya makanan mereka dibagi sama rata dengan semua tamu pesta lainnya
* Hitung ulang total seluruh tabel
* Angka ini tidak boleh diubah



class Calculator4 {

// Pajak dan tip

double tax = 0.05;

double tip = 0.15;

// Metode yang menghitung dan mengembalikan total

public double findTotal(double originalPrice) {

return originalPrice \* (1 + tax + tip);

}

}

public class CalculatorTest4 {

public static void main(String[] args) {

Calculator4 calc = new Calculator4();

// tarif awal buat orang2

double[] prices = {10, 12, 9, 8, 7, 15, 11, 30};

String[] names = {"Person 1", "Person 2", "Person 3", "Person 4", "Person 5", "Alex", "Person 7", "Person 8"};

double totalTable = 0;

double alexTotal = 0;

double person8Total = 0;

for (int i = 0; i < prices.length; i++) {

double total = calc.findTotal(prices[i]);

// simpan total si Alex dan Person 8 karena mereka ditraktir

if (names[i].equals("Alex")) {

alexTotal = total;

continue; // Jangan tambahkan ke totalTable dulu

} else if (names[i].equals("Person 8")) {

person8Total = total;

continue;

}

// Tambahkan yang lain ke totalTable

System.out.println(names[i] + " harus membayar: $" + total);

totalTable += total;

}

// bagikan total Alex dan Person 8 ke 6 orang lainnya

double sharedTotal = (alexTotal + person8Total) / 6.0;

// tambahin bagian traktiran ke total tiap orang

System.out.println("\nMembagi biaya Alex dan Person 8 ($" + (alexTotal + person8Total) + ") ke 6 orang lainnya.");

for (int i = 0; i < 6; i++) {

double personTotal = calc.findTotal(prices[i]) + sharedTotal;

System.out.println(names[i] + " membayar total setelah traktiran: $" + personTotal);

}

// itung total keseluruhan meja

totalTable += alexTotal + person8Total;

System.out.println("\nTotal keseluruhan meja (dengan pajak dan tip): $" + totalTable);

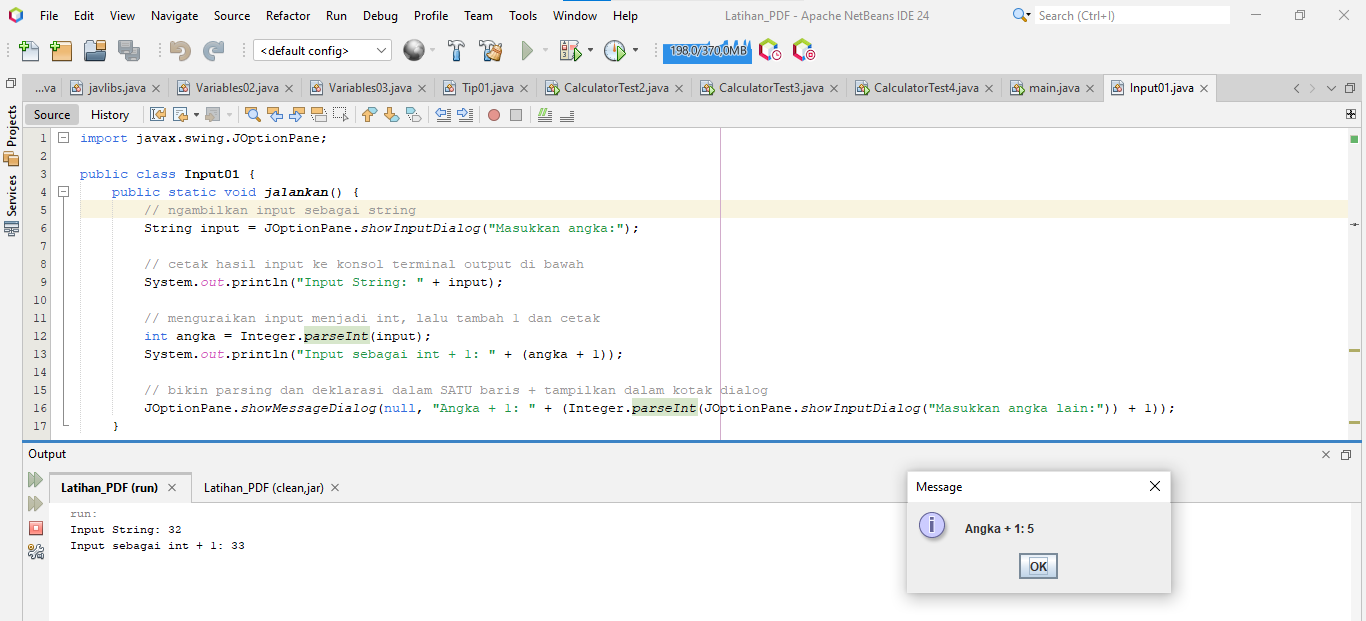
}

}

**Jfo 3\_5**

Latihan 1, Bagian 1 dan 2

* Buat proyek baru dan tambahkan file Input01.java ke proyek
* Buat JOptionPane:
* NetBeans akan mengirim keluhan
* Ikuti saran NetBeans untuk mengimpor javax.swing.JOptionPane
* Kita akan membahas pengimporan di bagian lain
* Simpan input ini sebagai String
* Cetak Variabel String
* Uraikan String sebagai variabel int terpisah
* Anda perlu memasukkan nilai yang dapat diuraikan
* Cetak nilai ini +1
* Cobalah membuat kotak dialog, menguraikannya, dan memulai int dalam satu baris. Anda hanya akan memiliki titik-koma (;)



import javax.swing.JOptionPane;

public class Input01 {

public static void jalankan() {

// ngambilkan input sebagai string

String input = JOptionPane.showInputDialog("Masukkan angka:");

// cetak hasil input ke konsol terminal output di bawah

System.out.println("Input String: " + input);

// menguraikan input menjadi int, lalu tambah 1 dan cetak

int angka = Integer.parseInt(input);

System.out.println("Input sebagai int + 1: " + (angka + 1));

// bikin parsing dan deklarasi dalam SATU baris + tampilkan dalam kotak dialog

JOptionPane.showMessageDialog(null, "Angka + 1: " + (Integer.parseInt(JOptionPane.showInputDialog("Masukkan angka lain:")) + 1));

}

}

Latihan 2

* Buat proyek baru dan tambahkan file Input02.java ke proyek
* Bereksperimen dengan kode dan coba mengubah …

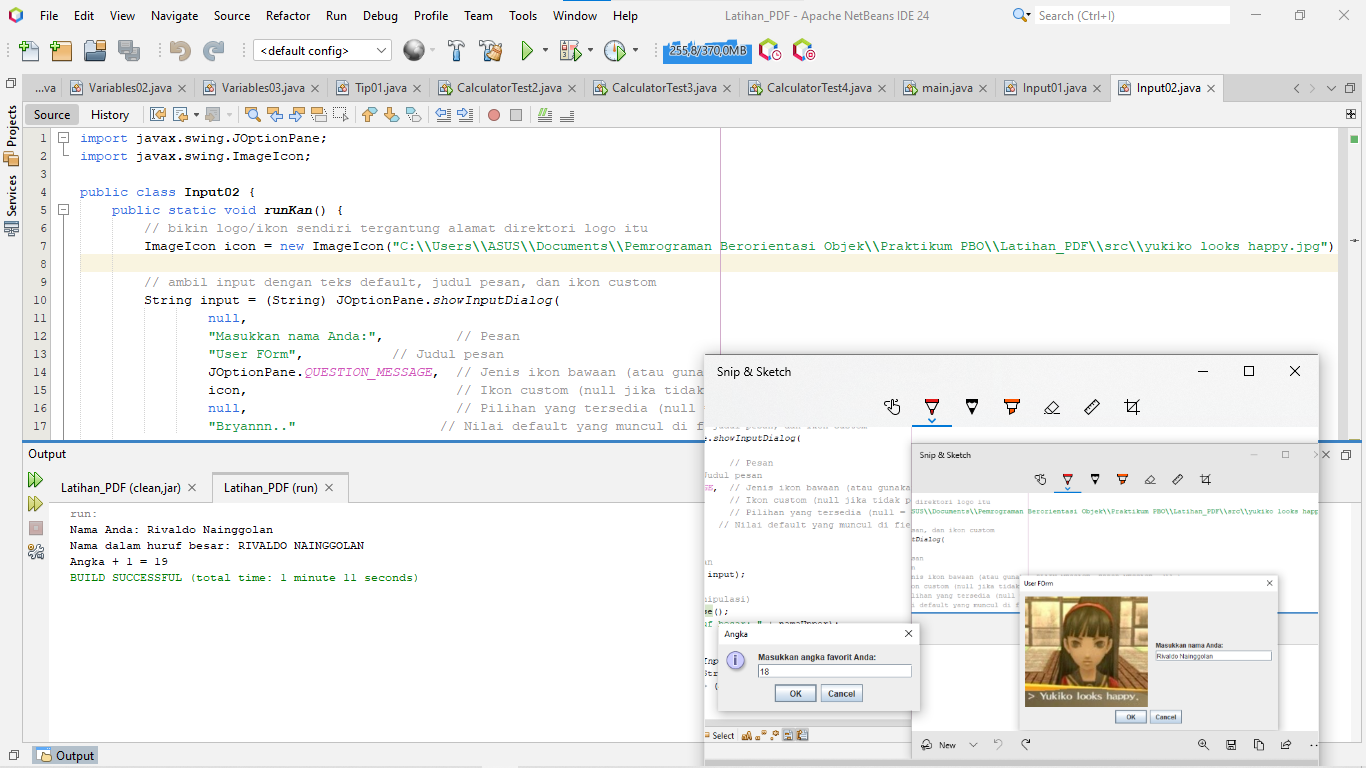
−Judul pesan

−Pesan

−Semua teks input default

−Ikon kotak dialog

* Uraikan, manipulasi, dan cetak input



import javax.swing.JOptionPane;

import javax.swing.ImageIcon;

public class Input02 {

public static void runKan() {

// bikin logo/ikon sendiri tergantung alamat direktori logo itu

ImageIcon icon = new ImageIcon("C:\\Users\\ASUS\\Documents\\Pemrograman Berorientasi Objek\\Praktikum PBO\\Latihan\_PDF\\src\\yukiko looks happy.jpg"); // Kosongkan atau isi dengan path gambar

// ambil input dengan teks default, judul pesan, dan ikon custom

String input = (String) JOptionPane.showInputDialog(

null,

"Masukkan nama Anda:", // Pesan

"User FOrm", // Judul pesan

JOptionPane.QUESTION\_MESSAGE, // Jenis ikon bawaan (atau gunakan PLAIN\_MESSAGE, ERROR\_MESSAGE, dll.)

icon, // Ikon custom (null jika tidak pakai)

null, // Pilihan yang tersedia (null = input bebas)

"Bryannn.." // Nilai default yang muncul di field input

);

// menampilkan input yang dimasukkan

System.out.println("Nama Anda: " + input);

// ubahkan menjadi huruf besar (manipulasi)

String namaUpper = input.toUpperCase();

System.out.println("Nama dalam huruf besar: " + namaUpper);

// ngambil input angka

String angkaStr = JOptionPane.showInputDialog(null, "Masukkan angka favorit Anda:", "Angka", JOptionPane.INFORMATION\_MESSAGE);

int angka = Integer.parseInt(angkaStr); // diuraikan ke int

System.out.println("Angka + 1 = " + (angka + 1));

}

}

Asd

Latihan 3

* Buat proyek baru dan tambahkan file Input03.java ke proyek
* Buat Pemindai:

− IDE Anda akan membuat komplain

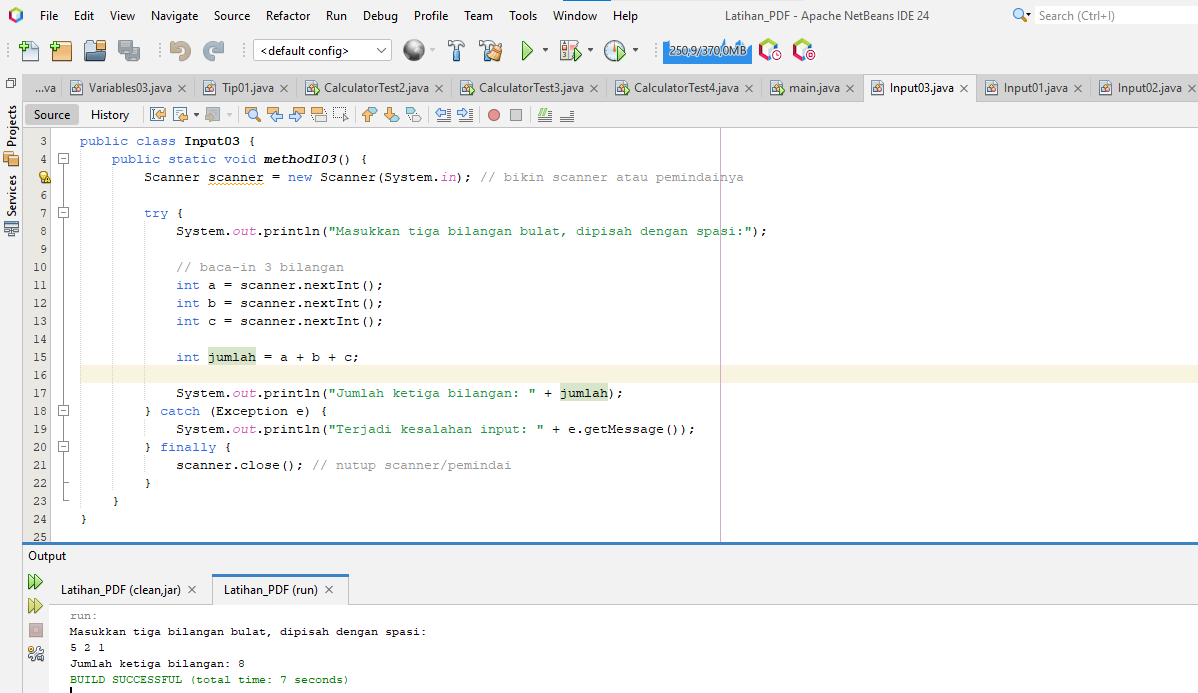
− Ikuti saran IDE untuk mengimpor java.util.Scanner

− Ingat untuk menutup Pemindai

* Gunakan Pemindai dan Sistem.in untuk menulis program yang …

−Menemukan dan mencetak jumlah tiga bilangan bulat yang dimasukkan oleh pengguna

* Coba masukkan kurang dari tiga token
* Coba masukkan token yang tidak dapat diuraikan sebagai token



import java.util.Scanner;

public class Input03 {

public static void methodI03() {

Scanner scanner = new Scanner(System.in); // bikin scanner atau pemindainya

try {

System.out.println("Masukkan tiga bilangan bulat, dipisah dengan spasi:");

// baca-in 3 bilangan

int a = scanner.nextInt();

int b = scanner.nextInt();

int c = scanner.nextInt();

int jumlah = a + b + c;

System.out.println("Jumlah ketiga bilangan: " + jumlah);

} catch (Exception e) {

System.out.println("Terjadi kesalahan input: " + e.getMessage());

} finally {

scanner.close(); // nutup scanner/pemindai

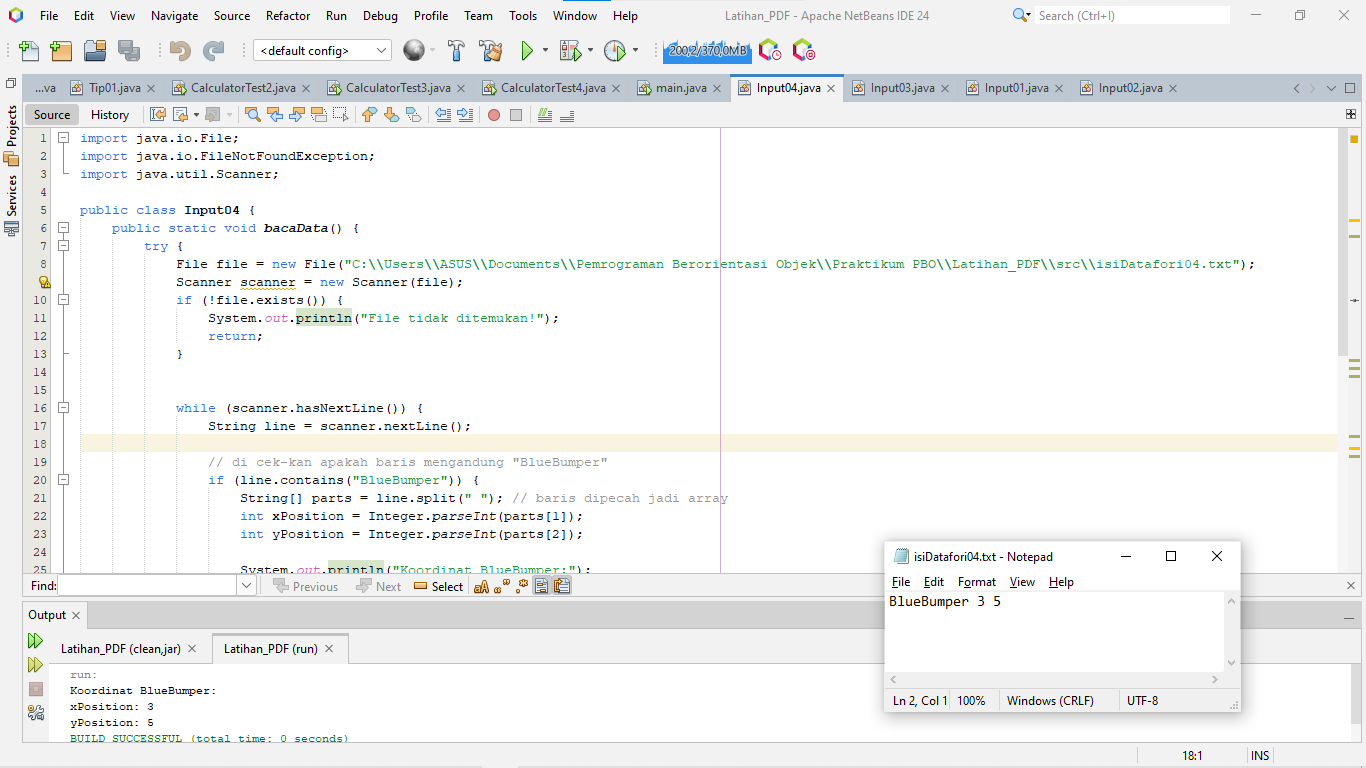
}

}

}

Latihan 4, Bagian 1

* Buat proyek baru dan tambahkan file Input04.java ke proyek
* Jalankan kode dan periksa output
* Baca setiap baris berikutnya hingga Anda menemukan "BlueBumper"
* Dua angka di belakang "BlueBumper" merupakan xPosition dan yPosition objek. Simpan koordinat ini sebagai bilangan bulat dan cetak
* Periksa input04text.txt, jika perlu



import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Input04 {

public static void bacaData() {

try {

File file = new File("C:\\Users\\ASUS\\Documents\\Pemrograman Berorientasi Objek\\Praktikum PBO\\Latihan\_PDF\\src\\isiDatafori04.txt");

Scanner scanner = new Scanner(file);

if (!file.exists()) {

System.out.println("File tidak ditemukan!");

return;

}

while (scanner.hasNextLine()) {

String line = scanner.nextLine();

// di cek-kan apakah baris mengandung "BlueBumper"

if (line.contains("BlueBumper")) {

String[] parts = line.split(" "); // baris dipecah jadi array

int xPosition = Integer.parseInt(parts[1]);

int yPosition = Integer.parseInt(parts[2]);

System.out.println("Koordinat BlueBumper:");

System.out.println("xPosition: " + xPosition);

System.out.println("yPosition: " + yPosition);

break; // siap, keluar dari loop

}

}

scanner.close(); // nutup scanner

} catch (FileNotFoundException e) {

System.out.println("File tidak ditemukan: " + e.getMessage());

} catch (Exception e) {

System.out.println("Kesalahan saat membaca file: " + e.getMessage());

}

}

}

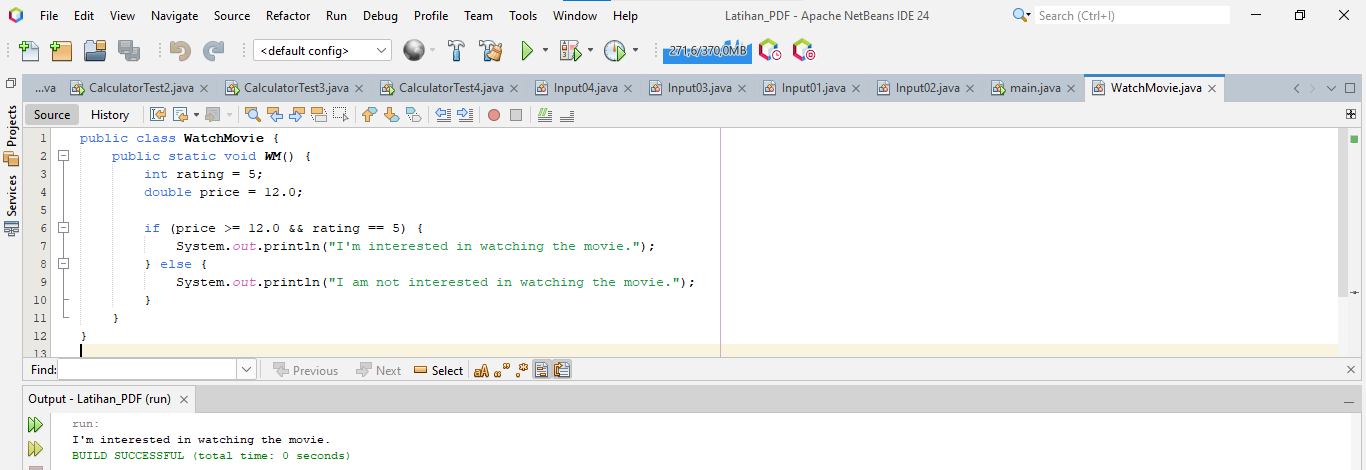
**Jfo 5\_2**

Latihan 1

* Buat proyek baru dan tambahkan file WatchMovie.java ke proyek
* Modifikasi WatchMovie.java untuk menonton film yang memenuhi dua kondisi berikut:
* Harga film lebih besar dari atau sama dengan $12
* Peringkat film sama dengan 5

−Tampilkan output sebagai “I’m interested in watching the movie.”

−Atau tampilkan output sebagai “I am not interested in watching the movie”



public class WatchMovie {

public static void WM() {

int rating = 5;

double price = 12.0;

if (price >= 12.0 && rating == 5) {

System.out.println("I'm interested in watching the movie.");

} else {

System.out.println("I am not interested in watching the movie.");

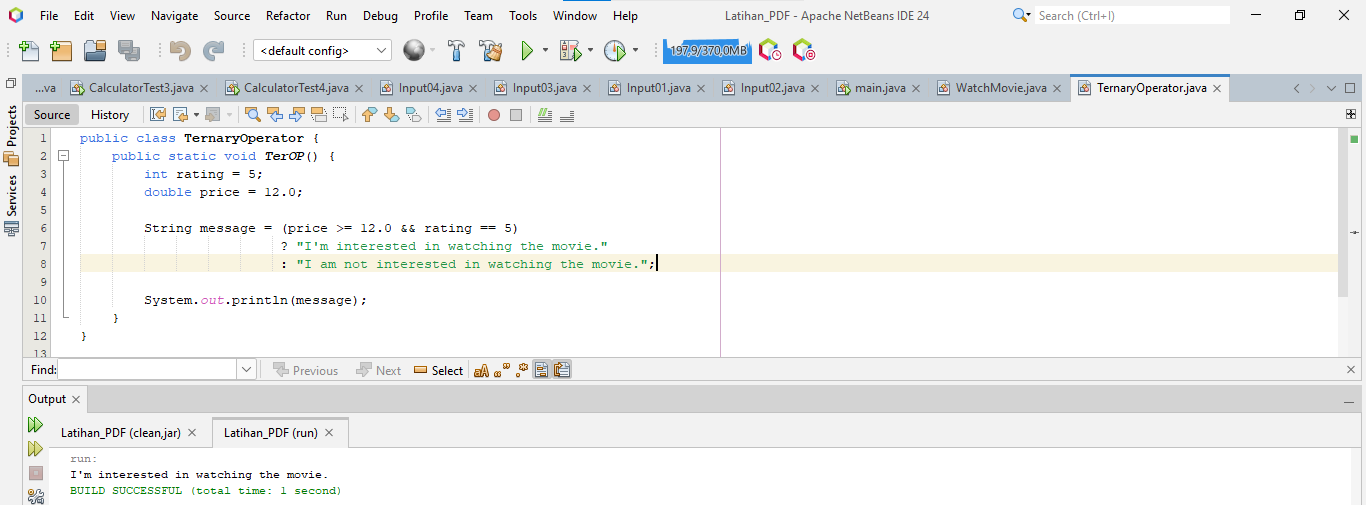
}

}

}

Latihan 2

* Tambahkan file TernaryOperator.java ke proyek yang Anda buat untuk latihan 1
* Modifikasi TernaryOperator.java untuk menduplikasi logika yang diberikan dalam pernyataan if/else dengan menggunakan operator belah-tiga



public class TernaryOperator {

public static void TerOP() {

int rating = 5;

double price = 12.0;

String message = (price >= 12.0 && rating == 5)

? "I'm interested in watching the movie."

: "I am not interested in watching the movie.";

System.out.println(message);

}

}

Latihan 3

* Tambahkan file ComputeFare.java ke proyek yang Anda buat untuk latihan 1
* Teliti CompueFare.java
* Terapkan hal berikut menggunakan construct if/else:

− Nyatakan variabel bilangan bulat, usia

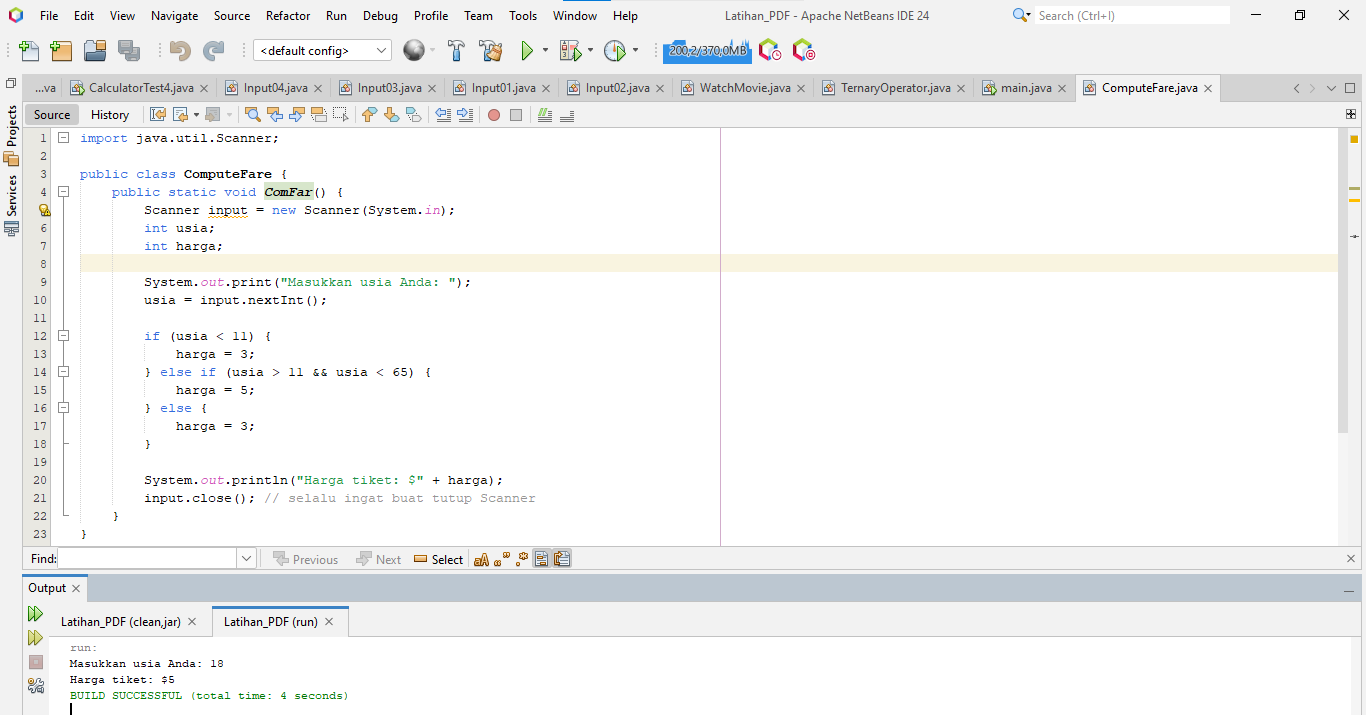
− Minta pengguna memasukkan nilai untuk usia

* Gunakan construct if berantai, komputasikan biaya berdasarkan usia sesuai dengan kondisi berikut:

−Jika usia di bawah 11 tahun, maka harga=3$

−Jika usia di atas 11 tahun dan di bawah 65, maka harga=5$

−Atau, untuk semua usia lainnya, maka harga=3$



import java.util.Scanner;

public class ComputeFare {

public static void ComFar() {

Scanner input = new Scanner(System.in);

int usia;

int harga;

System.out.print("Masukkan usia Anda: ");

usia = input.nextInt();

if (usia < 11) {

harga = 3;

} else if (usia > 11 && usia < 65) {

harga = 5;

} else {

harga = 3;

}

System.out.println("Harga tiket: $" + harga);

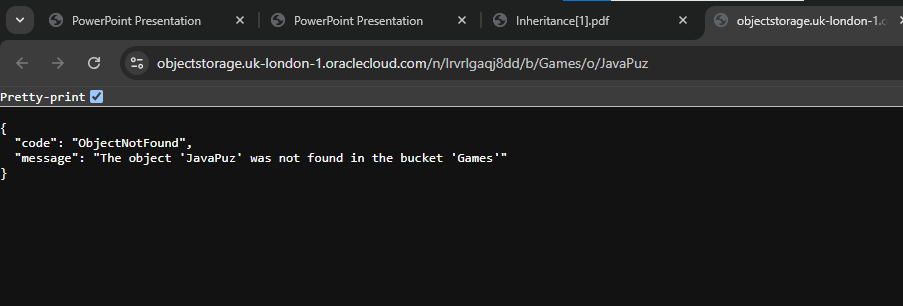
input.close(); // selalu ingat buat tutup Scanner

}

}

**Jfo 2\_3**

Latihan 1 ERROR LINK



* Buka   
  <https://objectstorage.uk-london-1.oraclecloud.com/n/Irvrlgaqj8dd/b/Games/o/JavaPuzzleBall/index.html>
* Mainkan Basic Puzzles 1 sampai 5
* -Sasaran Anda: Mendesain solusi yang membelokkan bola ke Duke
* Pertimbangkan yang berikut ini:
* -Objek apa yang terdapat dalam bidang permainan?
* -Apa yang terjadi bila Anda memasukkan dinding segitiga atau ikon dinding sederhana di roda biru?

Latihan 2, Bagian 1 dan 2

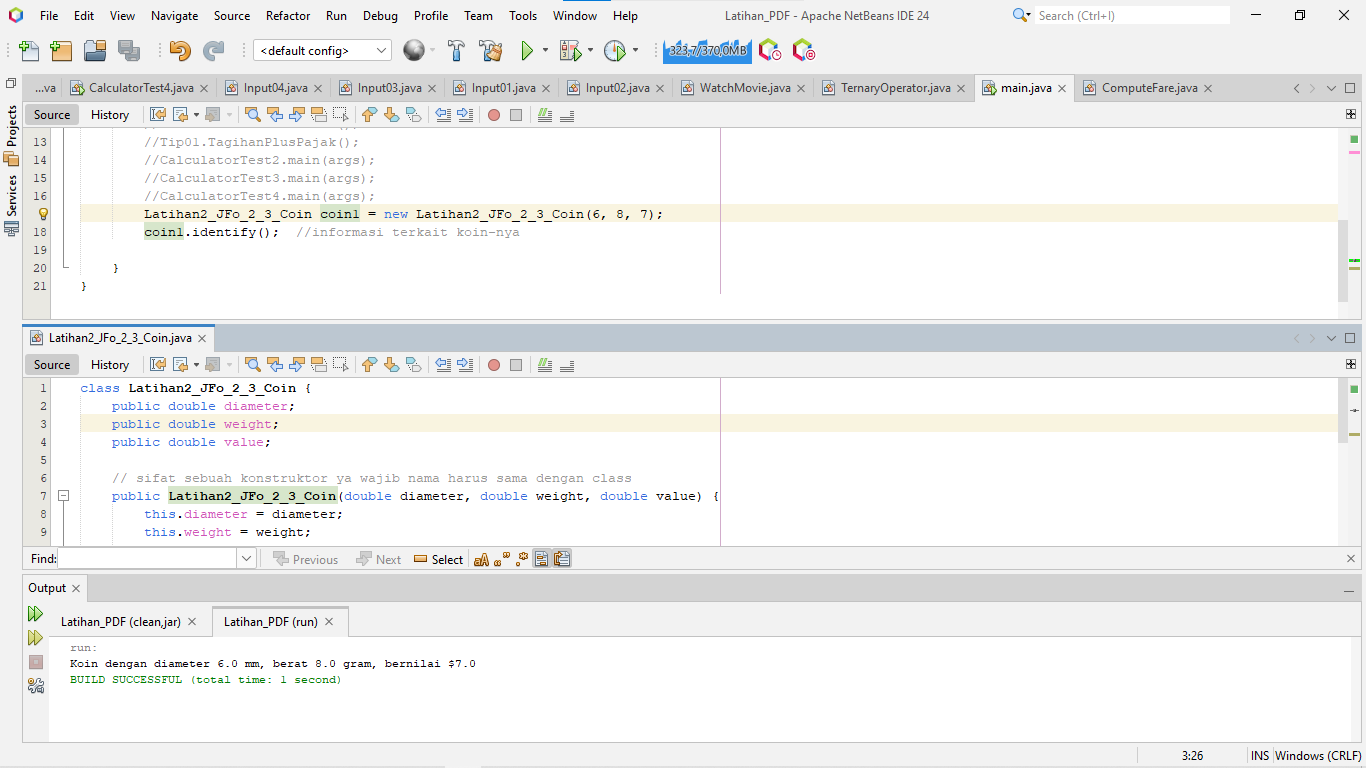
* Dengan skenario berikut, objek apa yang mungkin Anda buat model untuk menyelesaikan program Anda?

- Desain program untuk mesin penyortiran koin

- Mesin ini harus mengukur, menghitung, dan menyortir koin berdasarkan ukuran atau nilainya

- Mesin ini juga harus mencetak tanda terima

* Cantumkan setidaknya 3 objek yang akan Anda butuhkan untuk memodel contoh ini:
* Koin
* Mesin Sortir
* Kertas Tanda terima
* Pilih objek dari Bagian 1
* Properti dan perilaku apa dari objek yang dapat Anda sertakan dalam program?
* Properti :
* double diameter → ukuran koin
* double weight → berat koin
* double value → nilai uang koin (misalnya 0.1, 0.25)
* Perilaku :
* void identify() → mengidentifikasi jenis koin berdasarkan ukuran dan berat
* void printDetails() → menampilkan informasi koin
* boolean isValid() → mengecek apakah koin sesuai spesifikasi yang sah



public class Coin {

public double diameter; // mm

public double weight; // gram

public double value; // nilai dalam mata uang

public Coin(double diameter, double weight, double value) {

this.diameter = diameter;

this.weight = weight;

this.value = value;

}

public boolean isValid() {

return diameter > 0 && weight > 0 && value > 0;

}

public void identify() {

System.out.println("Koin dengan diameter " + diameter + " mm, berat " + weight + " gram, bernilai $" + value);

}

}

**Jfo 3\_2**

Latihan 1, Bagian 1 dan 2

* Buat proyek baru dan tambahkan file Chickens01.java ke proyek
* Baca cerita ini dan hitung/cetak totalEggs yang dikumpulkan antara hari Senin dan Rabu:
* Ayam Petani Brown selalu bertelur eggsPerChicken telur tepat di siang hari, yang dikumpulkannya di hari itu
* Pada hari Senin, Petani Brown memiliki punya chickenCount ayam.
* Pada Selasa pagi, Petani Brown mendapatkan 1 ayam
* Pada Rabu pagi, seekor hewan liar memangsa setengah jumlah ayam!
* Berapa banyak telur yang dikumpulkan oleh Petani Brown jika ia memulai dengan

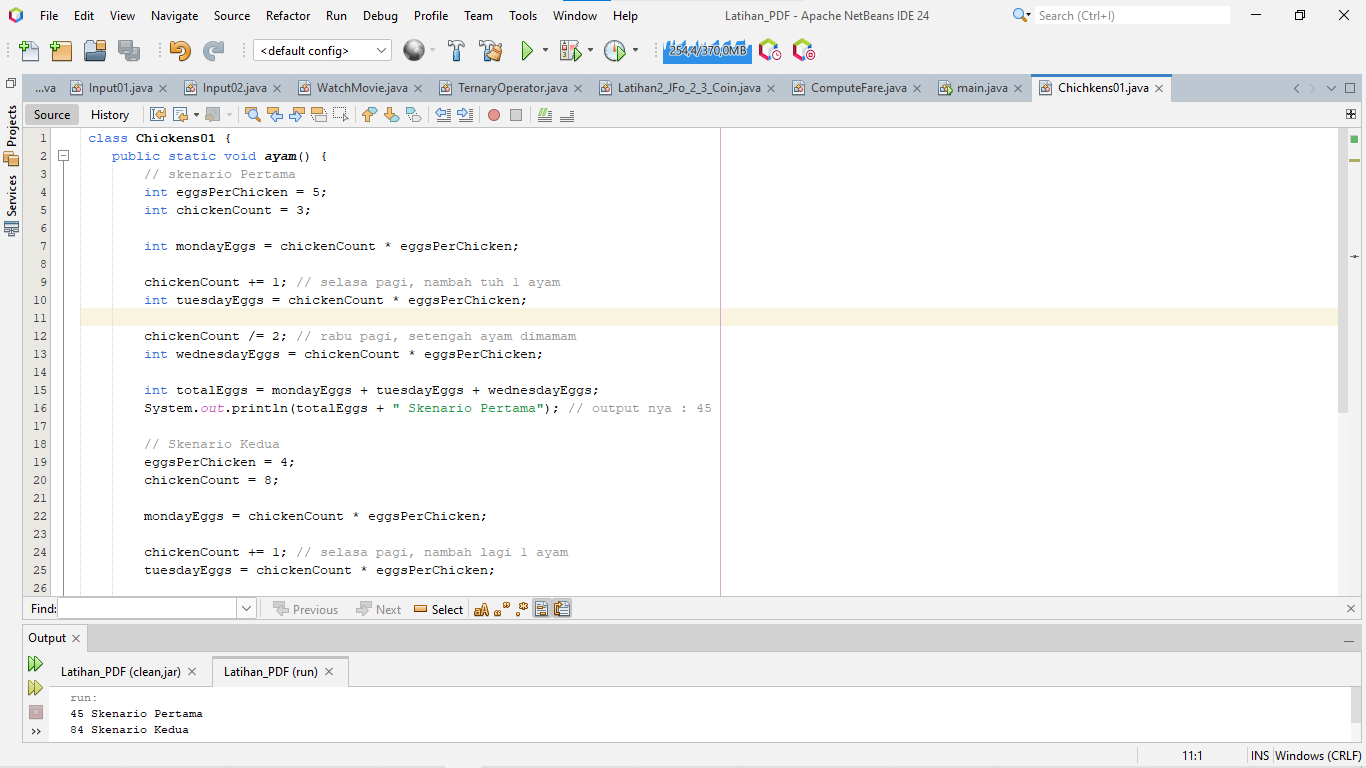
eggsPerChicken = 5, chickenCount = 3

eggsPerChicken = 4, chickenCount = 8

* Program Anda akan menghasilkan output berikut:

45 Skenario Pertama

84 Skenario Kedua



class Chickens01 {

public static void ayam() {

// skenario Pertama

int eggsPerChicken = 5;

int chickenCount = 3;

int mondayEggs = chickenCount \* eggsPerChicken;

chickenCount += 1; // selasa pagi, nambah tuh 1 ayam

int tuesdayEggs = chickenCount \* eggsPerChicken;

chickenCount /= 2; // rabu pagi, setengah ayam dimamam

int wednesdayEggs = chickenCount \* eggsPerChicken;

int totalEggs = mondayEggs + tuesdayEggs + wednesdayEggs;

System.out.println(totalEggs + " Skenario Pertama"); // output nya : 45

// Skenario Kedua

eggsPerChicken = 4;

chickenCount = 8;

mondayEggs = chickenCount \* eggsPerChicken;

chickenCount += 1; // selasa pagi, nambah lagi 1 ayam

tuesdayEggs = chickenCount \* eggsPerChicken;

chickenCount /= 2; // rabu pagi, setengah ayam dimangsa

wednesdayEggs = chickenCount \* eggsPerChicken;

totalEggs = mondayEggs + tuesdayEggs + wednesdayEggs;

System.out.println(totalEggs + " Skenario Kedua"); // output: 84

}

}

Latihan 2, Bagian 1 dan 2

* Buat proyek baru dan tambahkan file Chickens02.java ke proyek
* Baca cerita ini dan hitung/cetak nilai yang dibutuhkan:

- Pada hari Senin, Petani Fred mengumpulkan 100 telur

- Pada hari Selasa, Petani Fred mengumpulkan 121 telur

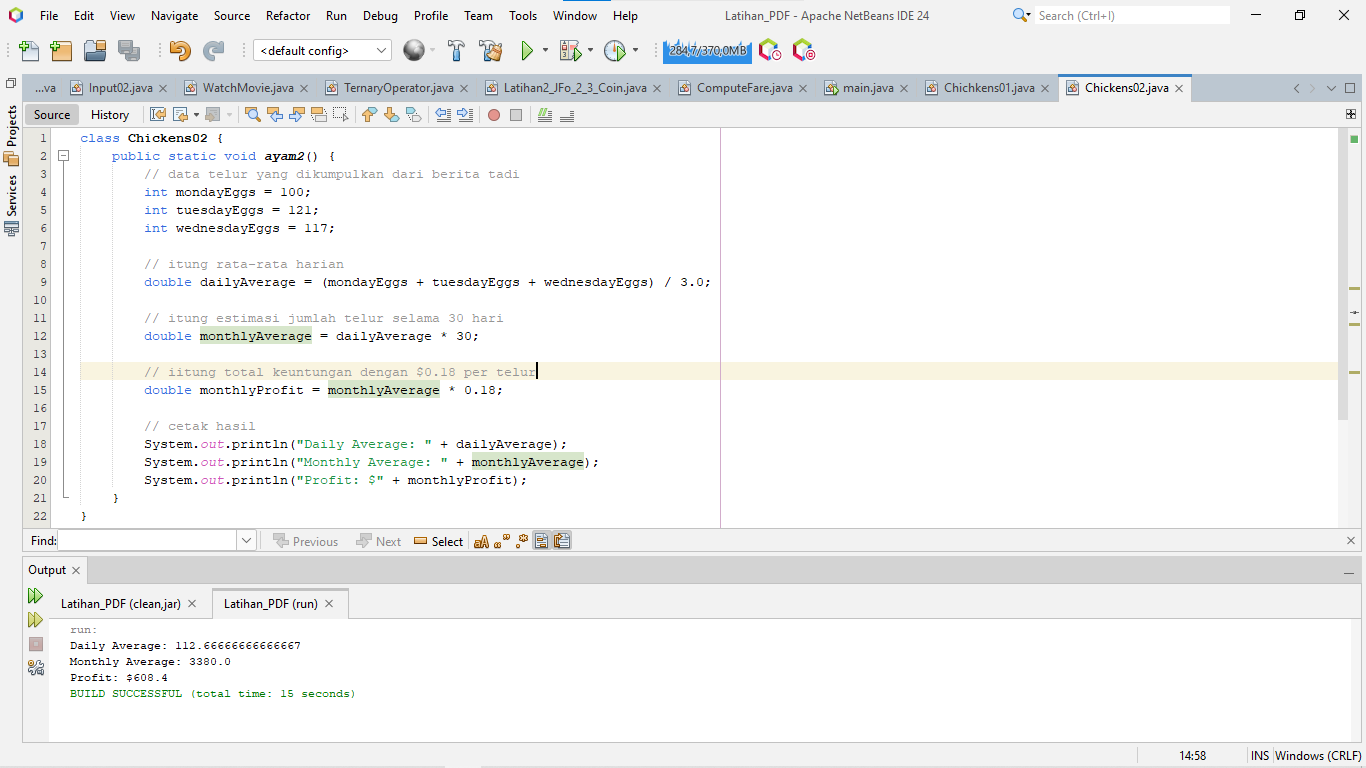
- Pada hari Rabu, Petani Fred mengumpulkan 117 telur

* Berapa dailyAverage telur yang dikumpulkan?
* Berapa jumlah telur yang dapat diharapkan dalam monthlyAverage 30 hari?
* Jika telur dapat dijual untuk mendapatkan laba sebesar $0,18, berapa total monthlyProfit Petani Fred untuk semua telur?
* Program Anda akan menghasilkan output berikut:

Daily Average: 112.66666666666667

Monthly Average: 3380.0

Profit: $608.4



class Chickens02 {

public static void ayam2() {

// data telur yang dikumpulkan dari berita tadi

int mondayEggs = 100;

int tuesdayEggs = 121;

int wednesdayEggs = 117;

// itung rata-rata harian

double dailyAverage = (mondayEggs + tuesdayEggs + wednesdayEggs) / 3.0;

// itung estimasi jumlah telur selama 30 hari

double monthlyAverage = dailyAverage \* 30;

// iitung total keuntungan dengan $0.18 per telur

double monthlyProfit = monthlyAverage \* 0.18;

// cetak hasil

System.out.println("Daily Average: " + dailyAverage);

System.out.println("Monthly Average: " + monthlyAverage);

System.out.println("Profit: $" + monthlyProfit);

}

}

**Jfo 5\_1**

Latihan 1

* Buat proyek baru dan tambahkan file AgeValidity.java ke proyek
* Modifikasi AgeValidity.java untuk menerapkan berikut:

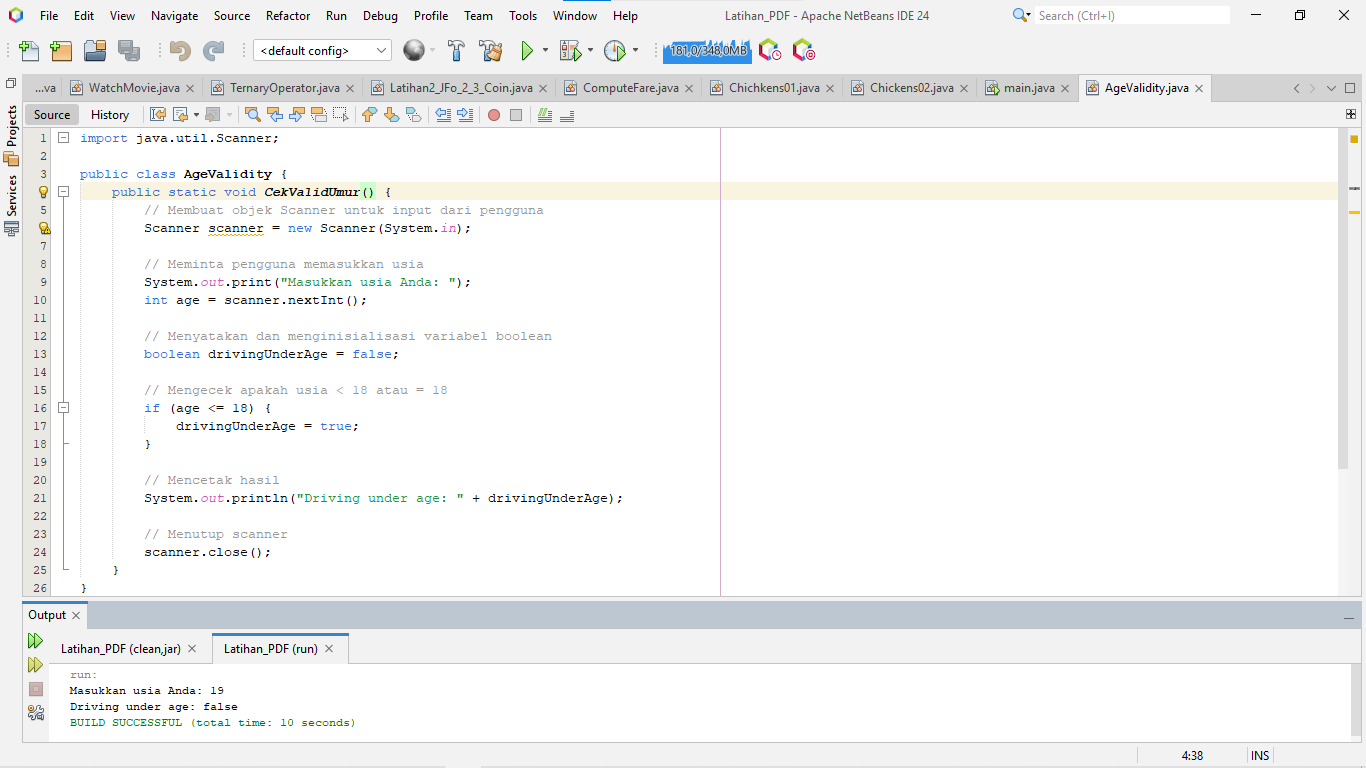
- Meminta pengguna memasukkan usia

- Menyatakan variabel boolean, drivingUnderAge

- Memulai driving UnderAge ke false

- Menulis boolean memeriksa usia yang dimasukkan <18 atau =18, lalu drivingUnderAge ke true

- Mencetak nilai driving UnderAge



import java.util.Scanner;

public class AgeValidity {

public static void CekValidUmur() {

// Membuat objek Scanner untuk input dari pengguna

Scanner scanner = new Scanner(System.in);

// Meminta pengguna memasukkan usia

System.out.print("Masukkan usia Anda: ");

int age = scanner.nextInt();

// Menyatakan dan menginisialisasi variabel boolean

boolean drivingUnderAge = false;

// Mengecek apakah usia < 18 atau = 18

if (age <= 18) {

drivingUnderAge = true;

}

// Mencetak hasil

System.out.println("Driving under age: " + drivingUnderAge);

// Menutup scanner

scanner.close();

}

}

Latihan 2

* Tambahkan file ChkOddEven.java ke proyek yang Anda buat untuk latihan 1
* Modifikasi ChkOddEven.java untuk menerapkan hal berikut:

-Masukkan angka antara 1 dan 10

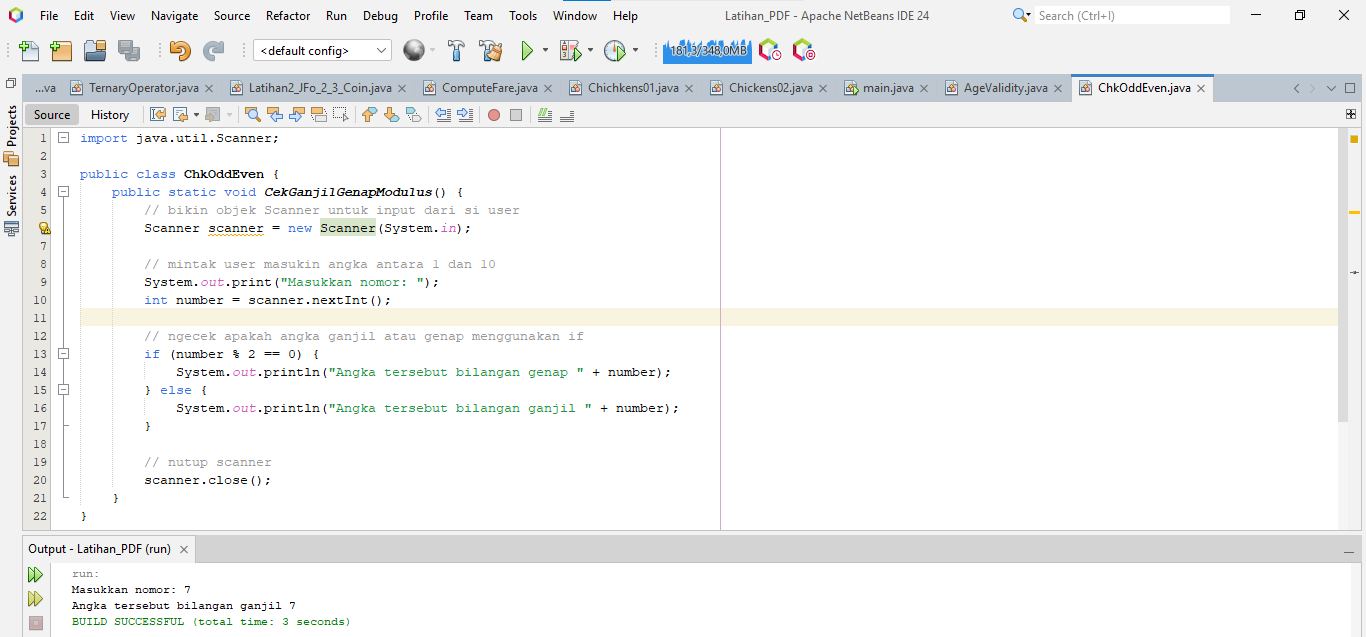
-Gunakan pernyataan if

- Uji apakah angka tersebut ganjil atau genap

* Program harus menghasilkan output berikut:

-Masukkan nomor: 7

-Angka tersebut bilangan ganjil 7



import java.util.Scanner;

public class ChkOddEven {

public static void CekGanjilGenapModulus() {

// bikin objek Scanner untuk input dari si user

Scanner scanner = new Scanner(System.in);

// mintak user masukin angka antara 1 dan 10

System.out.print("Masukkan nomor: ");

int number = scanner.nextInt();

// ngecek apakah angka ganjil atau genap menggunakan if

if (number % 2 == 0) {

System.out.println("Angka tersebut bilangan genap " + number);

} else {

System.out.println("Angka tersebut bilangan ganjil " + number);

}

// nutup scanner

scanner.close();

}

}

Latihan 3

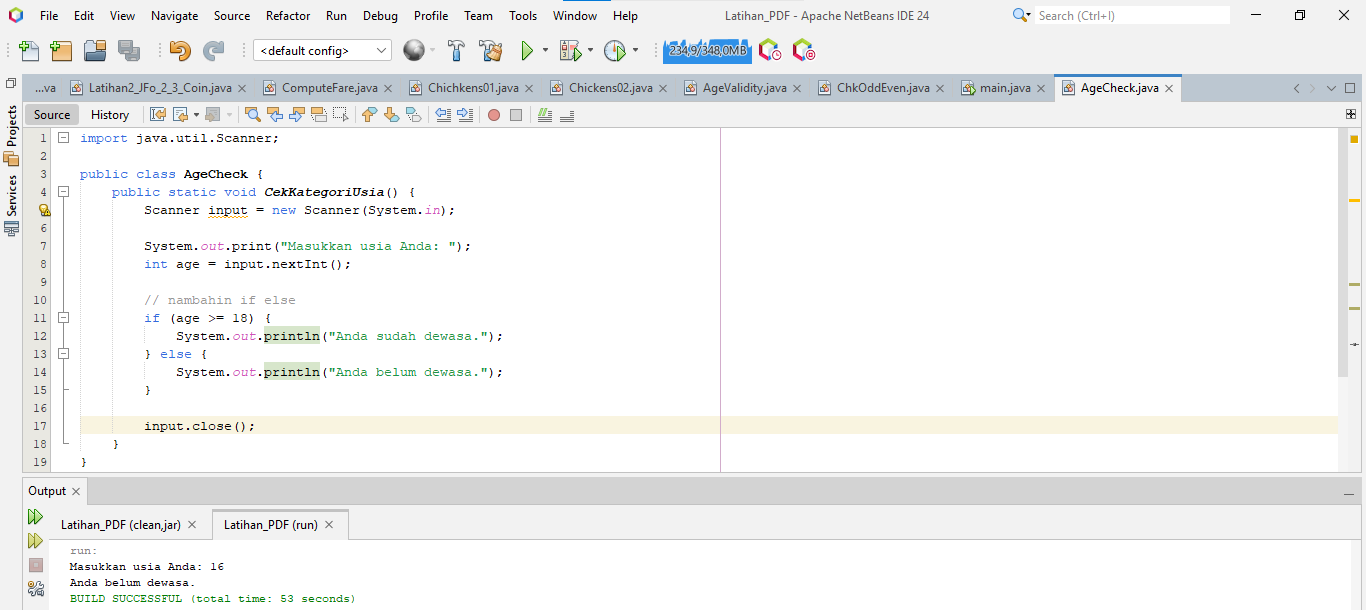
* Tambahkan file AgeCheck.java ke proyek yang Anda buat untuk latihan 1
* Teliti AgeCheck.java:

- Program tersebut memiliki masalah logika

- Bagi beberapa nilai, program akan mencetak jawaban yang salah

- Menemukan masalah dan memperbaikinya. (Anda mungkin perlu menjalankan program beberapa kali dan mencoba nilai yang berbeda untuk melihat program yang gagal)

- Ganti dua pernyataan if dengan pernyataan if/else



import java.util.Scanner;

public class AgeCheck {

public static void CekKategoriUsia() {

Scanner input = new Scanner(System.in);

System.out.print("Masukkan usia Anda: ");

int age = input.nextInt();

// nambahin if else

if (age >= 18) {

System.out.println("Anda sudah dewasa.");

} else {

System.out.println("Anda belum dewasa.");

}

input.close();

}

}

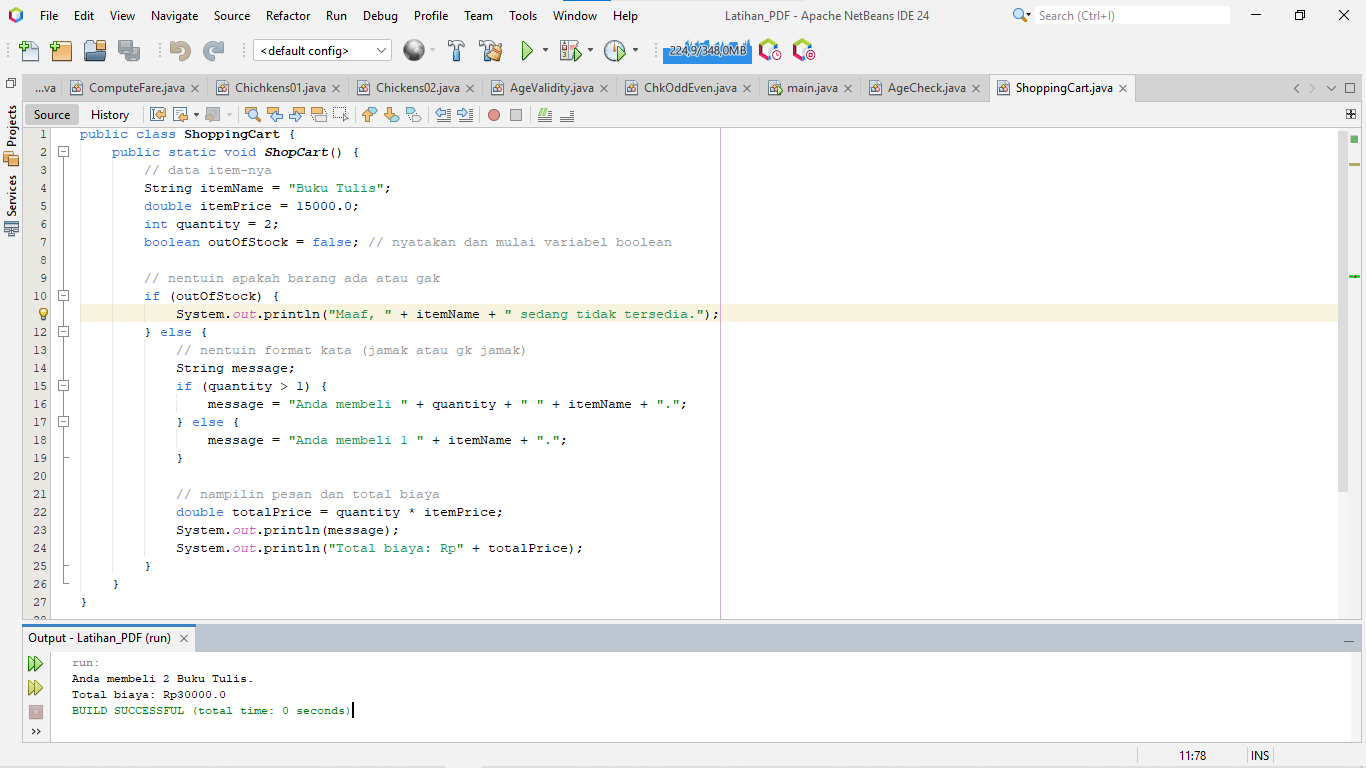
Latihan 4

* Tambahkan file ShoppingCart.java ke proyek yang Anda buat untuk latihan 1
* Teliti ShoppingCart.java
* Gunakan pernyataan if/else untuk menerapkan hal berikut:

- Nyatakan dan mulai variabel boolean, outOfStock

- Jika kuantitas > 1, ubah variabel pesan untuk menunjukkan plural

- Jika stok item habis, beri tahu pengguna bahwa item tersebut tidak tersedia, atau, cetak pesan dan total biaya



public class ShoppingCart {

public static void ShopCart() {

// data item-nya

String itemName = "Buku Tulis";

double itemPrice = 15000.0;

int quantity = 2;

boolean outOfStock = false; // nyatakan dan mulai variabel boolean

// nentuin apakah barang ada atau gak

if (outOfStock) {

System.out.println("Maaf, " + itemName + " sedang tidak tersedia.");

} else {

// nentuin format kata (jamak atau gk jamak)

String message;

if (quantity > 1) {

message = "Anda membeli " + quantity + " " + itemName + ".";

} else {

message = "Anda membeli 1 " + itemName + ".";

}

// nampilin pesan dan total biaya

double totalPrice = quantity \* itemPrice;

System.out.println(message);

System.out.println("Total biaya: Rp" + totalPrice);

}

}

}

Latihan 5

* Tambahkan file StringEquality.java ke proyek yang Anda buat untuk latihan 1
* Teliti StringEquality.java
* Gunakan pernyataan if dan if/else:

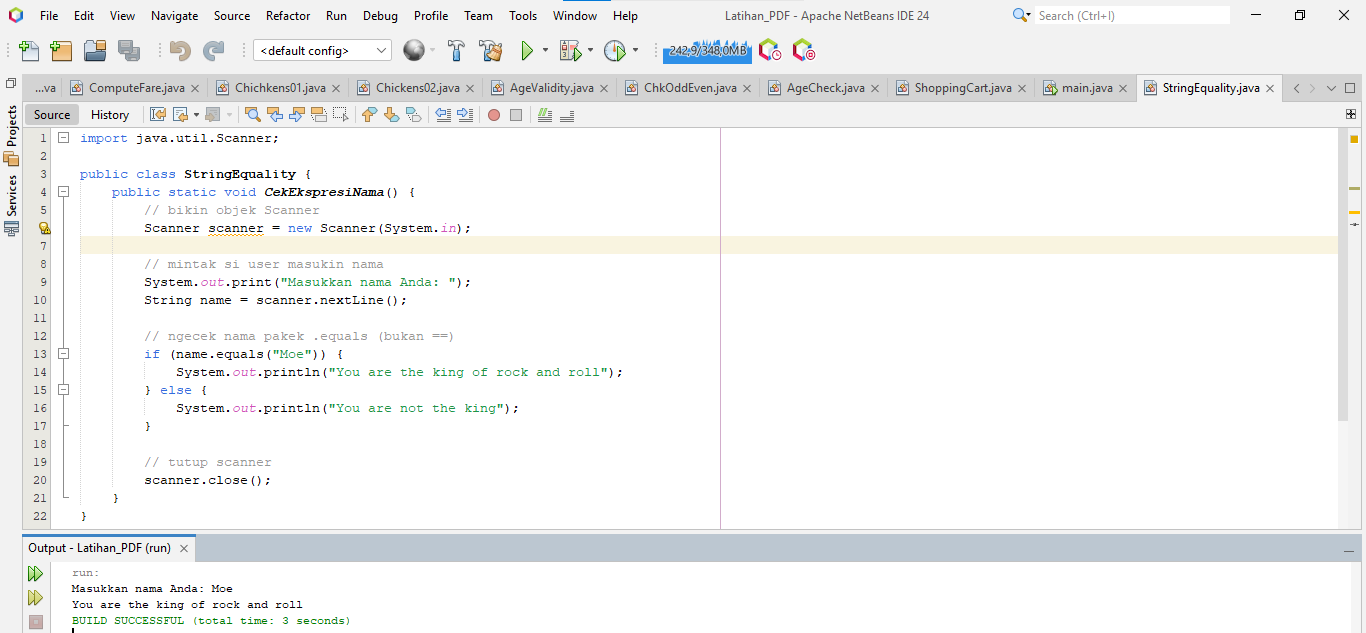
-Nyatakan nama variabel String

-Minta pengguna memasukkan nilai untuk nama

-Periksa apakah namanya adalah "Moe," lalu cetak "You are the king of rock and roll"

-Atau cetak "You are not the king"

-Jangan gunakan ==



import java.util.Scanner;

public class StringEquality {

public static void CekEkspresiNama() {

// bikin objek Scanner

Scanner scanner = new Scanner(System.in);

// mintak si user masukin nama

System.out.print("Masukkan nama Anda: ");

String name = scanner.nextLine();

// ngecek nama pakek .equals (bukan ==)

if (name.equals("Moe")) {

System.out.println("You are the king of rock and roll");

} else {

System.out.println("You are not the king");

}

// tutup scanner

scanner.close();

}

}

**Jfo 6\_1**

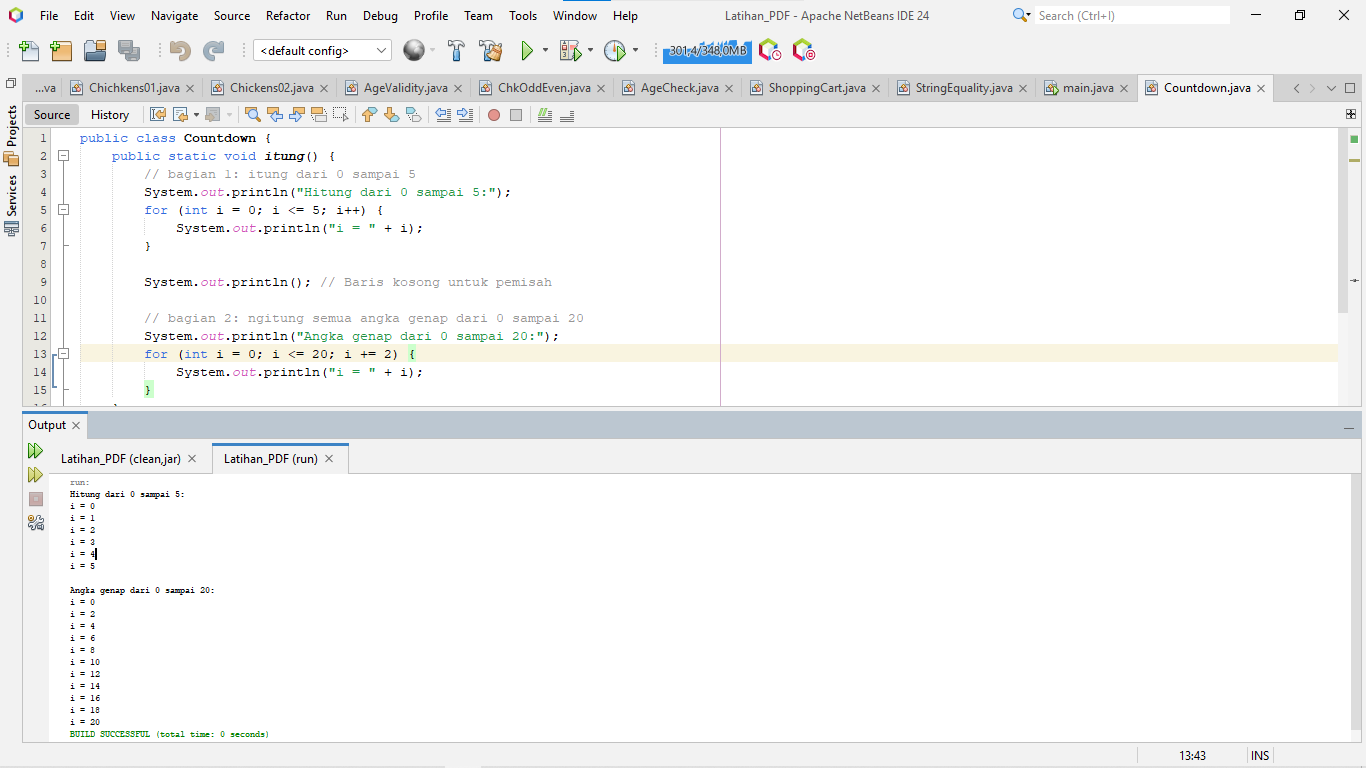
Latihan 1, Bagian 1 dan 2

* Buat proyek baru dan tambahkan file Countdown.java ke proyek
* Atur breakpoint in Countdown.java dan amati

- Bagaimana loop for memengaruhi eksekusi kode

- Bagaimana nilai i berubah

* Dapatkah Anda memodifikasi kode untuk menghitung mulai dari 0 hingga 5?
* Dapatkah Anda memodifikasi kode untuk menghitung semua angka genap mulai dari 0 hingga 20?



public class Countdown {

public static void itung() {

// bagian 1: itung dari 0 sampai 5

System.out.println("Hitung dari 0 sampai 5:");

for (int i = 0; i <= 5; i++) {

System.out.println("i = " + i);

}

System.out.println(); // Baris kosong untuk pemisah

// bagian 2: ngitung semua angka genap dari 0 sampai 20

System.out.println("Angka genap dari 0 sampai 20:");

for (int i = 0; i <= 20; i += 2) {

System.out.println("i = " + i);

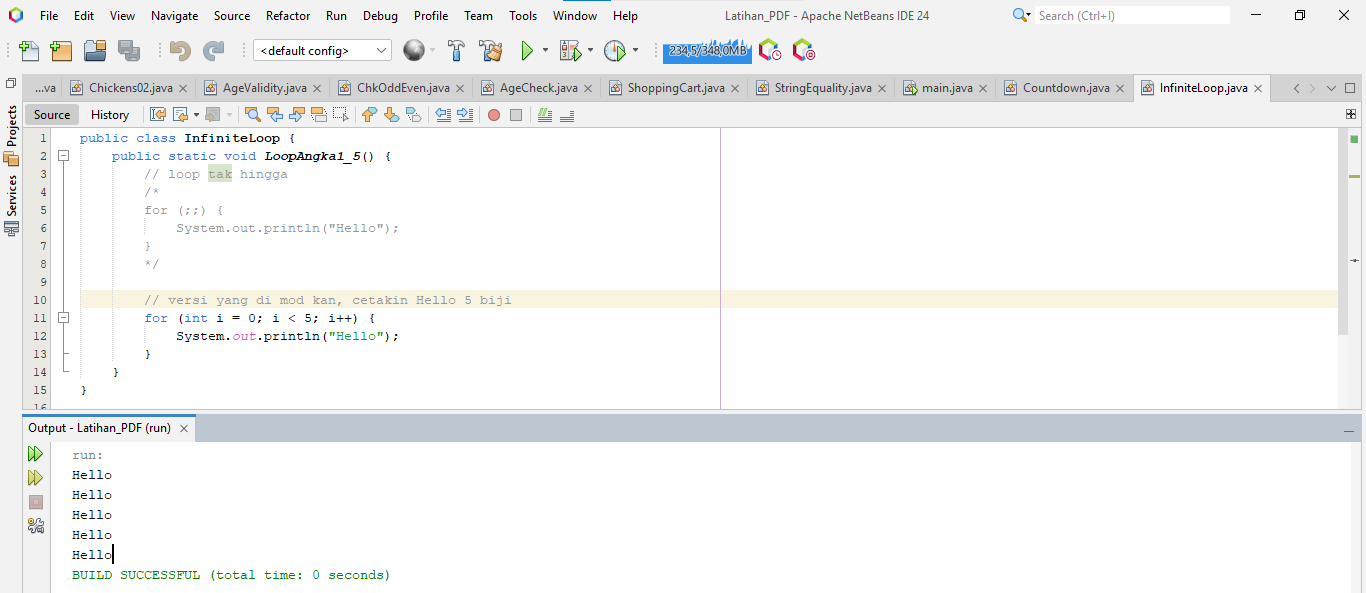
}

}

}

Latihan 2

* Tambahkan file InfiniteLoop.java ke proyek yang Anda buat untuk latihan 1
* Jalankan InfiniteLoop.java dan amati outputnya
* Modifikasi loop for dalam InfiniteLoop.java untuk mencetak "Hello" lima kali



public class InfiniteLoop {

public static void LoopAngka1\_5() {

// loop tak hingga

/\*

for (;;) {

System.out.println("Hello");

}

\*/

// versi yang di mod kan, cetakin Hello 5 biji

for (int i = 0; i < 5; i++) {

System.out.println("Hello");

}

}

}

Latihan 3

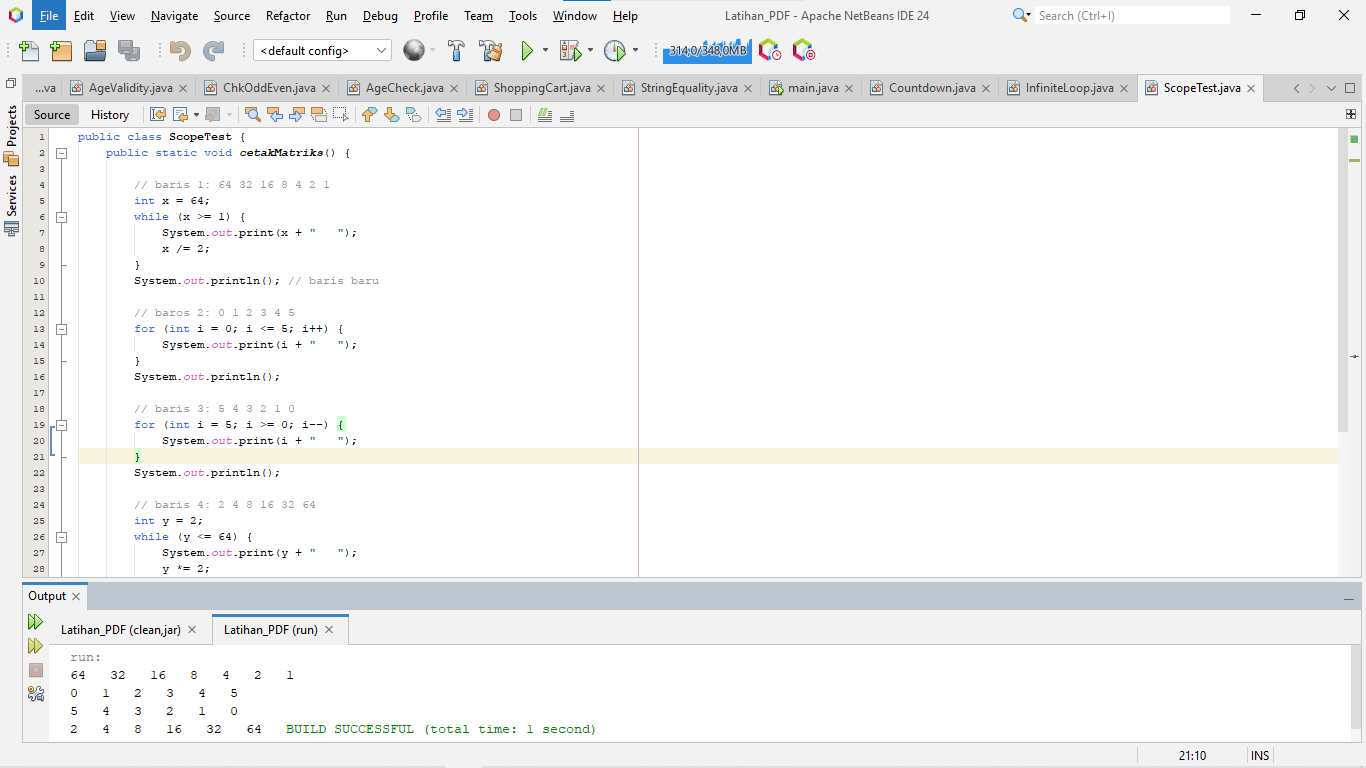
* Tambahkan file ScopeTest.java ke proyek yang Anda buat untuk latihan 1
* ScopeTest.java rusak
* Dapatkah Anda memperbaikinya?
* Anda seharusnya mendapatkan output berikut:

- 64 32 16 8 4 2 1

- 0 1 2 3 4 5

- 5 4 3 2 1 0

- 2 4 8 16 32 64



public class ScopeTest {

public static void cetakMatriks() {

// baris 1: 64 32 16 8 4 2 1

int x = 64;

while (x >= 1) {

System.out.print(x + " ");

x /= 2;

}

System.out.println(); // baris baru

// baros 2: 0 1 2 3 4 5

for (int i = 0; i <= 5; i++) {

System.out.print(i + " ");

}

System.out.println();

// baris 3: 5 4 3 2 1 0

for (int i = 5; i >= 0; i--) {

System.out.print(i + " ");

}

System.out.println();

// baris 4: 2 4 8 16 32 64

int y = 2;

while (y <= 64) {

System.out.print(y + " ");

y \*= 2;

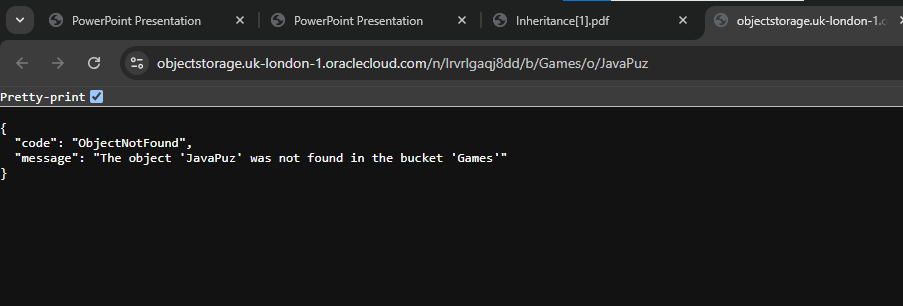
}

}

}

**Jfo 7\_1**

Latihan 1 ERROR LINK



* Buka   
  <https://objectstorage.uk-london-1.oraclecloud.com/n/Irvrlgaqj8dd/b/Games/o/JavaPuzzleBall/index.html>
* Mainkan Basic Puzzles 1 sampai 5
* -Sasaran Anda: Mendesain solusi yang membelokkan bola ke Duke
* Pertimbangkan yang berikut ini:
* -Objek apa yang terdapat dalam bidang permainan?
* -Apa yang terjadi bila Anda memasukkan dinding segitiga atau ikon dinding sederhana di roda biru?

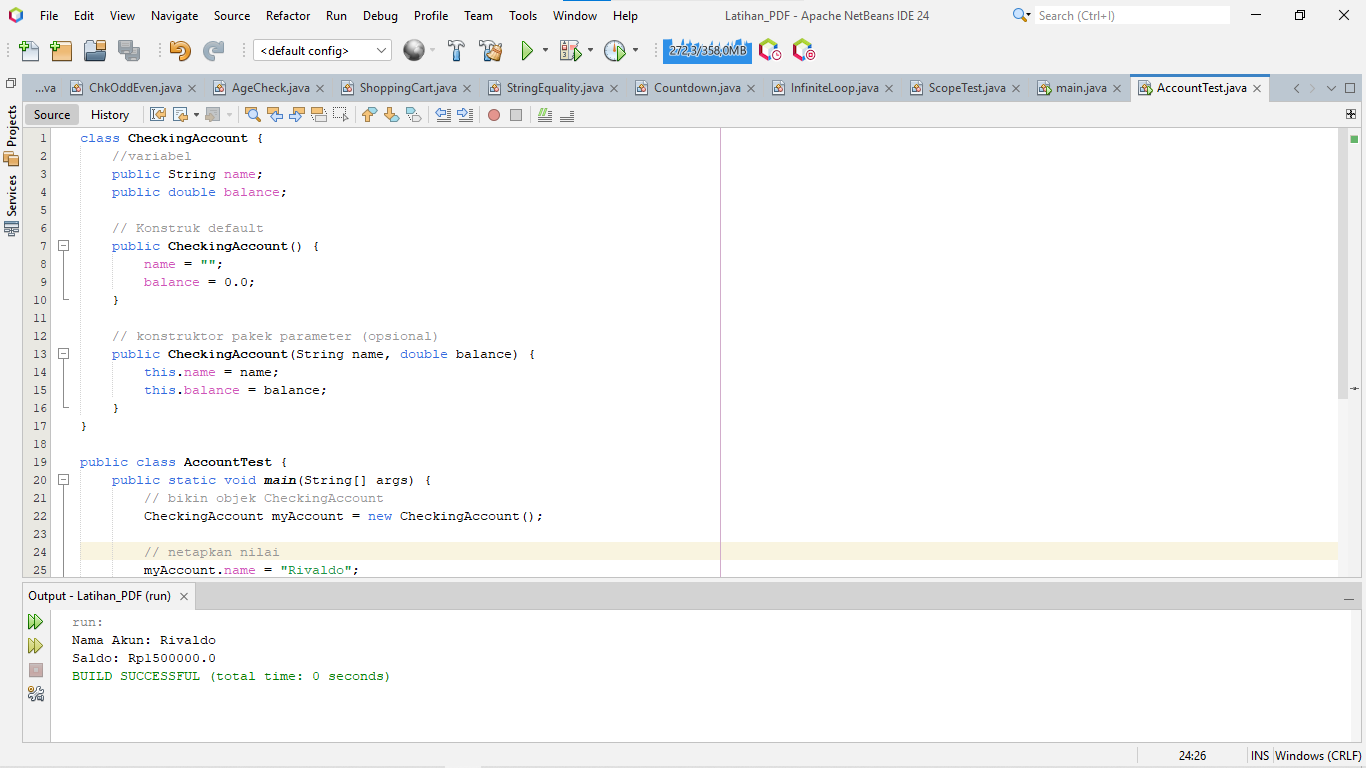
Latihan 2

* Buat proyek Java baru
* Buat class AccountTest dengan metode utama
* Buat class CheckingAccount

-Sertakan bidang untuk balance dan name

* Mulai objek CheckingAccount dari metode utama

- Tetapkan nilai untuk bidang balance dan name



class CheckingAccount {

//variabel

public String name;

public double balance;

// Konstruk default

public CheckingAccount() {

name = "";

balance = 0.0;

}

// konstruktor pakek parameter (opsional)

public CheckingAccount(String name, double balance) {

this.name = name;

this.balance = balance;

}

}

public class AccountTest {

public static void main(String[] args) {

// bikin objek CheckingAccount

CheckingAccount myAccount = new CheckingAccount();

// netapkan nilai

myAccount.name = "Rivaldo";

myAccount.balance = 1500000.0;

// nampilkan data akun

System.out.println("Nama Akun: " + myAccount.name);

System.out.println("Saldo: Rp" + myAccount.balance);

}

}

Latihan 3

* Lanjutkan mengedit proyek AccountTest
* Tulis metode withdraw() untuk memeriksa rekening yang ..

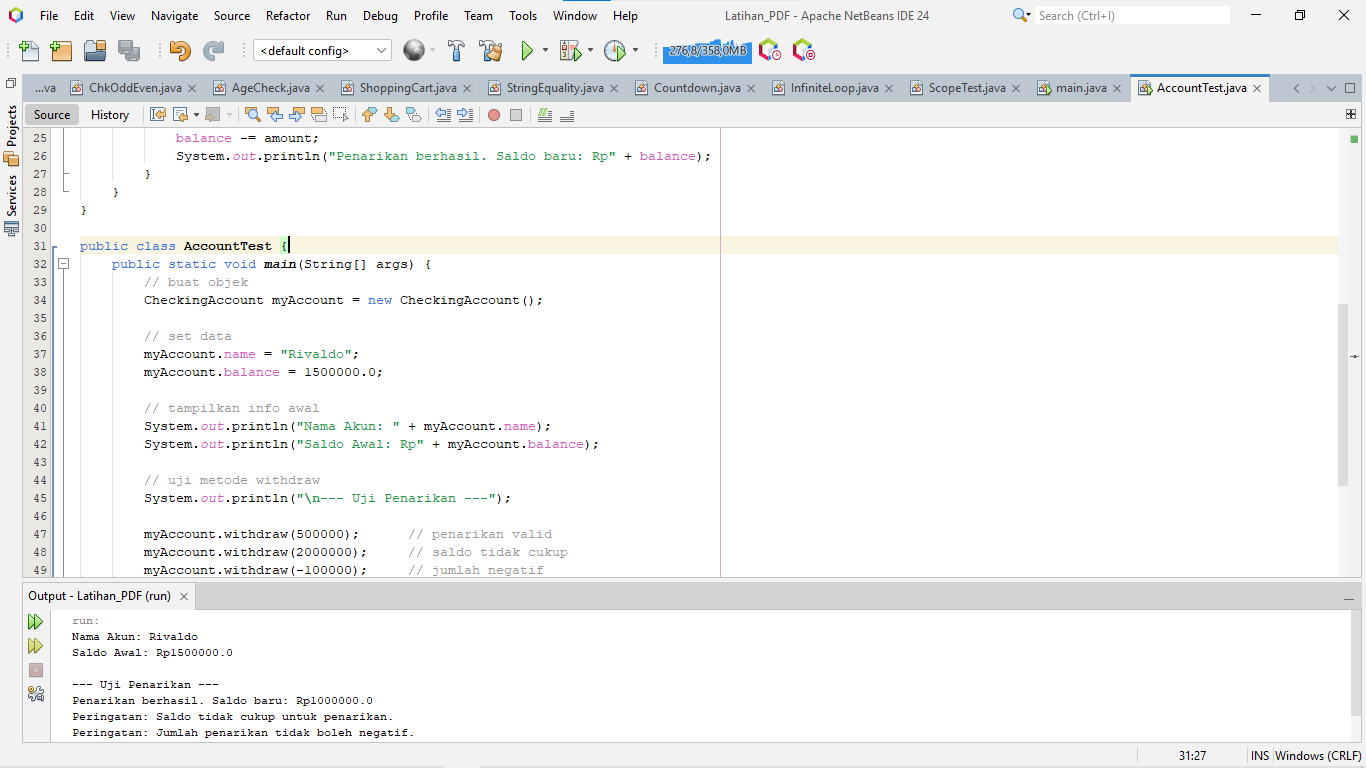
- Menerima argumen ganda untuk jumlah yang akan ditarik

- Mencetak peringatan jika saldo terlalu rendah untuk melakukan penarikan

- Mencetak peringatan jika argumen penarikan negatif

- Jika tidak ada peringatan, jumlah penarikan akan dikurangi dari saldo. Cetak saldo baru

* Uji metode ini dengan instance dari Latihan 2



class CheckingAccount {

// variabel

public String name;

public double balance;

// konstruktor default

public CheckingAccount() {

name = "";

balance = 0.0;

}

// konstruktor dengan parameter

public CheckingAccount(String name, double balance) {

this.name = name;

this.balance = balance;

}

// metode withdraw

public void withdraw(double amount) {

if (amount < 0) {

System.out.println("Peringatan: Jumlah penarikan tidak boleh negatif.");

} else if (amount > balance) {

System.out.println("Peringatan: Saldo tidak cukup untuk penarikan.");

} else {

balance -= amount;

System.out.println("Penarikan berhasil. Saldo baru: Rp" + balance);

}

}

}

public class AccountTest {

public static void main(String[] args) {

// buat objek

CheckingAccount myAccount = new CheckingAccount();

// set data

myAccount.name = "Rivaldo";

myAccount.balance = 1500000.0;

// tampilkan info awal

System.out.println("Nama Akun: " + myAccount.name);

System.out.println("Saldo Awal: Rp" + myAccount.balance);

// uji metode withdraw

System.out.println("\n--- Uji Penarikan ---");

myAccount.withdraw(500000); // penarikan valid

myAccount.withdraw(2000000); // saldo tidak cukup

myAccount.withdraw(-100000); // jumlah negatif

}

}

Latihan 4

* Lanjutkan mengedit proyek Account Test
* Buat class baru sesuai dengan deskripsi
* Pastikan untuk memulai class dan menguji metodenya

- Membuat Obligasi Tabungan

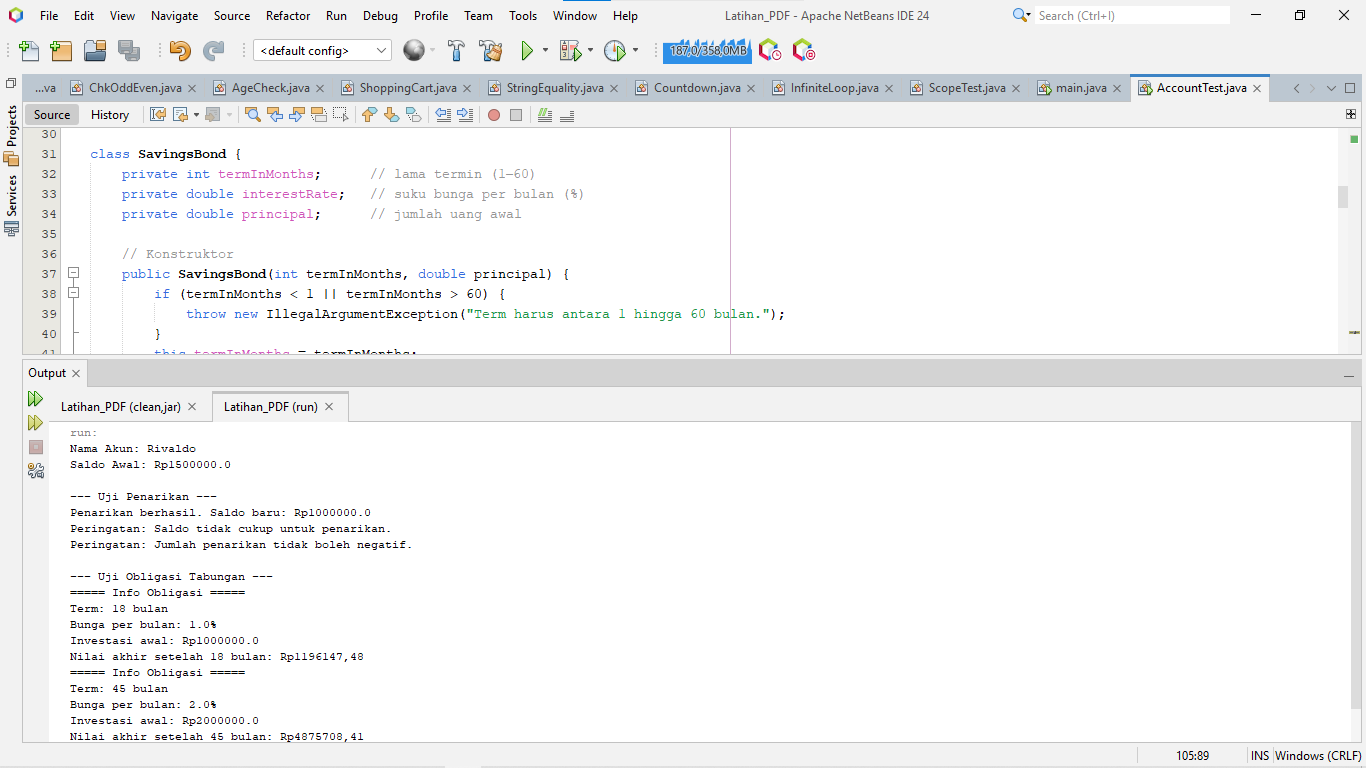
- Seseorang dapat membeli obligasi untuk termin apa pun antara 1 hingga 60 bulan

- Obligasi akan mendapatkan bunga setiap bulannya hingga jatuh tempo termin (sisa 0 bulan)

- Termin dan suku bunga diatur di saat yang bersamaan

- Suku bunga obligasi didasarkan pada terminnya, sesuai dengan sistem tier berikut:

* + 0-11 bulan: 0,5%
  + 12-23 bulan: 1,0%
  + 24-35 bulan: 1,5%
  + 36-47 bulan: 2,0%
  + 48-60 bulan: 2,5%



class CheckingAccount {

// variabel

public String name;

public double balance;

// konstruktor default

public CheckingAccount() {

name = "";

balance = 0.0;

}

// konstruktor dengan parameter

public CheckingAccount(String name, double balance) {

this.name = name;

this.balance = balance;

}

// metode withdraw

public void withdraw(double amount) {

if (amount < 0) {

System.out.println("Peringatan: Jumlah penarikan tidak boleh negatif.");

} else if (amount > balance) {

System.out.println("Peringatan: Saldo tidak cukup untuk penarikan.");

} else {

balance -= amount;

System.out.println("Penarikan berhasil. Saldo baru: Rp" + balance);

}

}

}

class SavingsBond {

private int termInMonths; // lama termin (1–60)

private double interestRate; // suku bunga per bulan (%)

private double principal; // jumlah uang awal

// Konstruktor

public SavingsBond(int termInMonths, double principal) {

if (termInMonths < 1 || termInMonths > 60) {

throw new IllegalArgumentException("Term harus antara 1 hingga 60 bulan.");

}

this.termInMonths = termInMonths;

this.principal = principal;

this.interestRate = determineInterestRate(termInMonths);

}

// nentuin bunga berdasarkan term

private double determineInterestRate(int term) {

if (term < 12) {

return 0.5;

} else if (term < 24) {

return 1.0;

} else if (term < 36) {

return 1.5;

} else if (term < 48) {

return 2.0;

} else {

return 2.5;

}

}

// ngitung total nilai setelah jatuh tempo

public void displayFinalValue() {

System.out.println("===== Info Obligasi =====");

System.out.println("Term: " + termInMonths + " bulan");

System.out.println("Bunga per bulan: " + interestRate + "%");

System.out.println("Investasi awal: Rp" + principal);

// itung akumulasi bunga

double total = principal;

for (int i = 0; i < termInMonths; i++) {

total += total \* (interestRate / 100);

}

System.out.printf("Nilai akhir setelah %d bulan: Rp%.2f\n", termInMonths, total);

}

}

public class AccountTest {

public static void main(String[] args) {

// bikinin objek

CheckingAccount myAccount = new CheckingAccount();

// set data

myAccount.name = "Rivaldo";

myAccount.balance = 1500000.0;

// tampilkan info awal

System.out.println("Nama Akun: " + myAccount.name);

System.out.println("Saldo Awal: Rp" + myAccount.balance);

// uji metode withdraw

System.out.println("\n--- Uji Penarikan ---");

myAccount.withdraw(500000); // penarikan valid

myAccount.withdraw(2000000); // saldo tidak cukup

myAccount.withdraw(-100000); // jumlah negatif

// ngujikan saving bodnya

System.out.println("\n--- Uji Obligasi Tabungan ---");

SavingsBond bond1 = new SavingsBond(18, 1000000); // Term 18 bulan, Rp1.000.000

bond1.displayFinalValue();

SavingsBond bond2 = new SavingsBond(45, 2000000); // Term 45 bulan, Rp2.000.000

bond2.displayFinalValue();

}

}

**Jfo 8\_1**

Latihan 1

* Apakah Anda dapat mengidentifikasi ketiga komponen deklarasi array untuk masing-masing array dari jenis data primitive?

-Jenis Data

- Nama

- Ukuran

* int[] myArray;
* myArray = new int[20];
* char[] sentence = new char[100];
* double[] teamPoints = new double[5];

Komponen Array:

Jenis Data → Tipe data elemen array (contoh: int, char, double)

Nama → Nama variabel array

Ukuran → Jumlah elemen dalam array (jumlah indeks)

1. int[] myArray;

Jenis Data: int

Nama: myArray

Ukuran: Belum ditentukan → ini hanya deklarasi (belum alokasi memori)

2. myArray = new int[20];

Jenis Data: int

Nama: myArray, Ukuran: 20 elemen   
(Gabungan dari dua baris di atas = deklarasi dan alokasi array integer dengan 20 elemen)

3. char[] sentence = new char[100];

Jenis Data: char

Nama: sentence

Ukuran: 100 elemen

4. double[] teamPoints = new double[5];

Jenis Data: double

Nama: teamPoints

Ukuran: 5 elemen

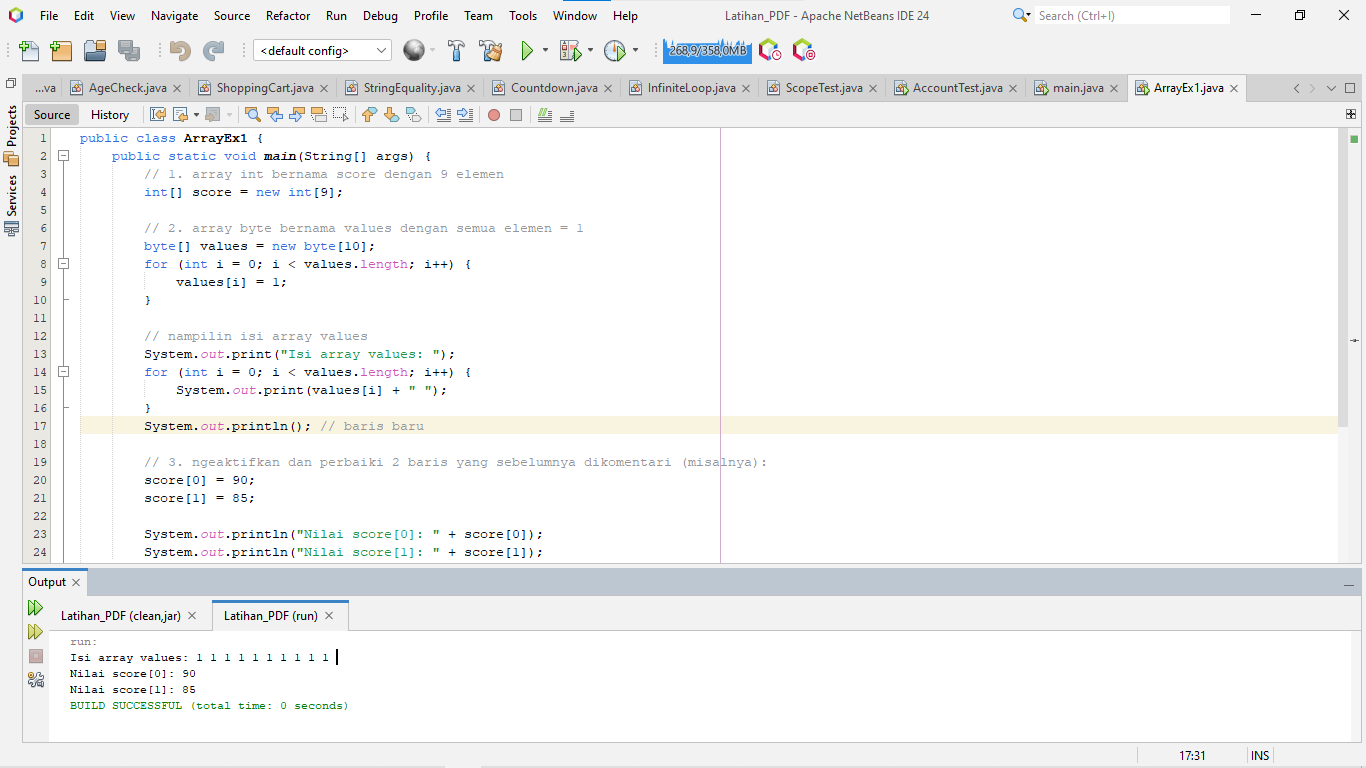
Latihan 2

* Buat proyek baru dan tambahkan file ArrayEx1.java ke proyek
* Teliti ArrayExl.java
* Modifikasi program untuk menerapkan

- Nyatakan array satu dimensi yang diberi nama score jenis int yang dapat menyimpan 9 nilai

- Nyatakan dan inisialisasi array bita satu dimensi yang diberi nama values berukuran 10 sehingga semua masukan berisi 1

- Batalkan komentar dua garis yang diberi komentar, lalu atasi kesalahan sintaks



public class ArrayEx1 {

public static void main(String[] args) {

// 1. array int bernama score dengan 9 elemen

int[] score = new int[9];

// 2. array byte bernama values dengan semua elemen = 1

byte[] values = new byte[10];

for (int i = 0; i < values.length; i++) {

values[i] = 1;

}

// nampilin isi array values

System.out.print("Isi array values: ");

for (int i = 0; i < values.length; i++) {

System.out.print(values[i] + " ");

}

System.out.println(); // baris baru

// 3. ngeaktifkan dan perbaiki 2 baris yang sebelumnya dikomentari (misalnya):

score[0] = 90;

score[1] = 85;

System.out.println("Nilai score[0]: " + score[0]);

System.out.println("Nilai score[1]: " + score[1]);

}

}

Latihan 3

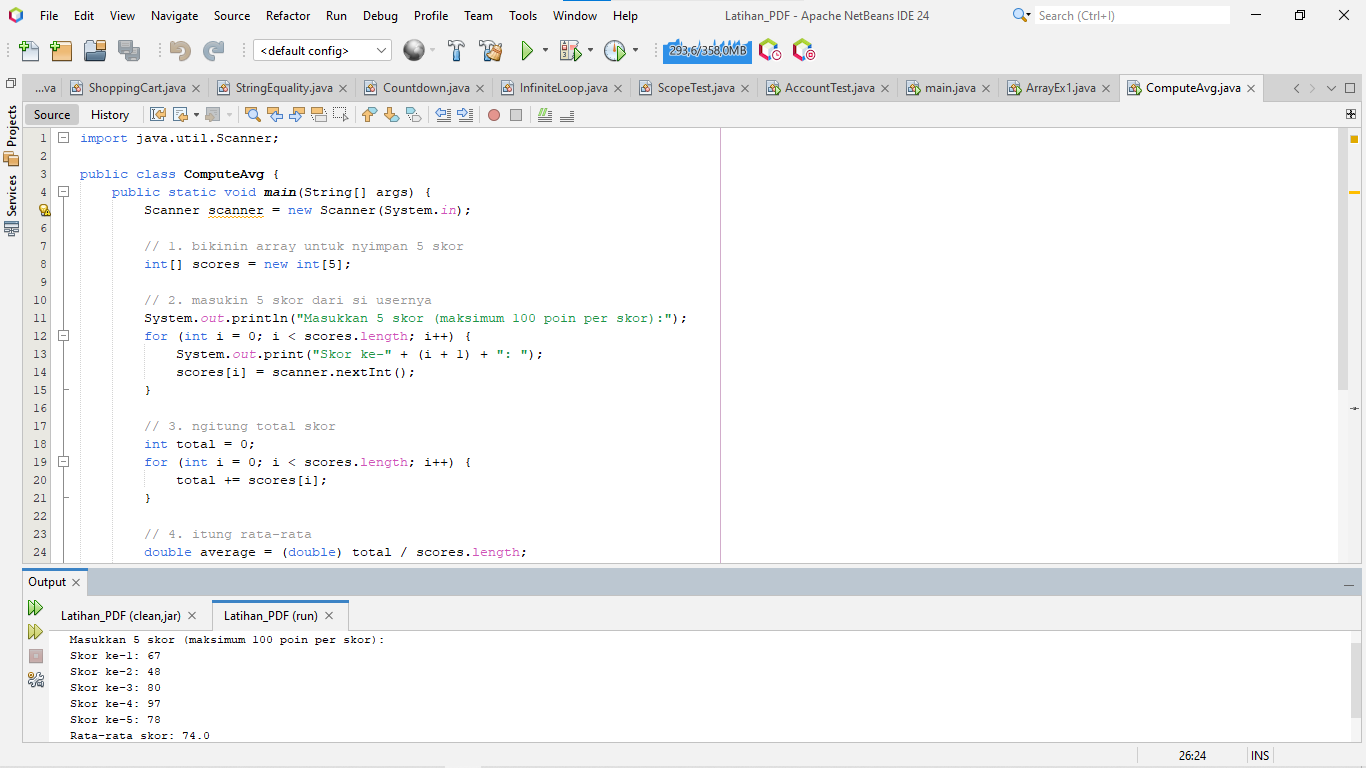
* Tambahkan file ComputeAvg.java ke proyek yang Anda buat untuk latihan 2
* Teliti ComputeAvg.java
* Modifikasi program untuk menerapkan

-Dalam class tertentu, terdapat lima set, yang masing-masing bernilai 100 poin

-Masukkan lima skor uji dari konsol

-Simpan skor uji di array

-Hitung rata-rata skor siswa



import java.util.Scanner;

public class ComputeAvg {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// 1. bikinin array untuk nyimpan 5 skor

int[] scores = new int[5];

// 2. masukin 5 skor dari si usernya

System.out.println("Masukkan 5 skor (maksimum 100 poin per skor):");

for (int i = 0; i < scores.length; i++) {

System.out.print("Skor ke-" + (i + 1) + ": ");

scores[i] = scanner.nextInt();

}

// 3. ngitung total skor

int total = 0;

for (int i = 0; i < scores.length; i++) {

total += scores[i];

}

// 4. itung rata-rata

double average = (double) total / scores.length;

// 5. nampilkan hasil

System.out.println("Rata-rata skor: " + average);

scanner.close();

}

}

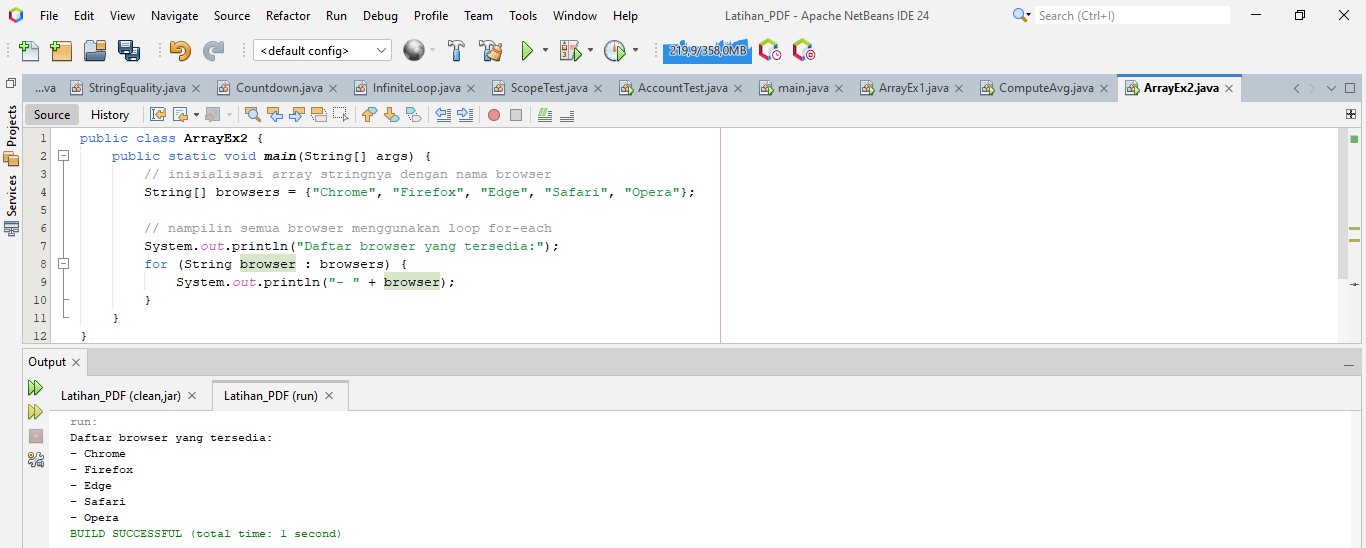
Latihan 4

* Tambahkan file ke ArrayEx2.java ke proyek yang Anda buat untuk latihan 2
* Teliti ArrayEx2.java
* Lakukan hal berikut:

-Jalankan program dan amati kesalahan

- Modifikasi program untuk mengatasi kesalahan

- Menggunakan loop for-each, tampilkan semua browser yang disimpan dalam array



public class ArrayEx2 {

public static void main(String[] args) {

// inisialisasi array stringnya dengan nama browser

String[] browsers = {"Chrome", "Firefox", "Edge", "Safari", "Opera"};

// nampilin semua browser menggunakan loop for-each

System.out.println("Daftar browser yang tersedia:");

for (String browser : browsers) {

System.out.println("- " + browser);

}

}

}