

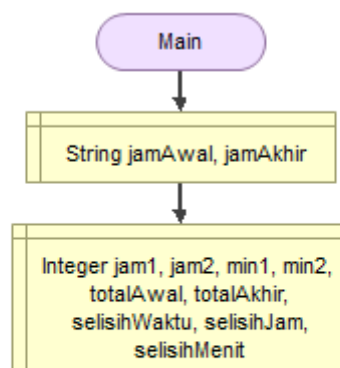
1. Soal 1

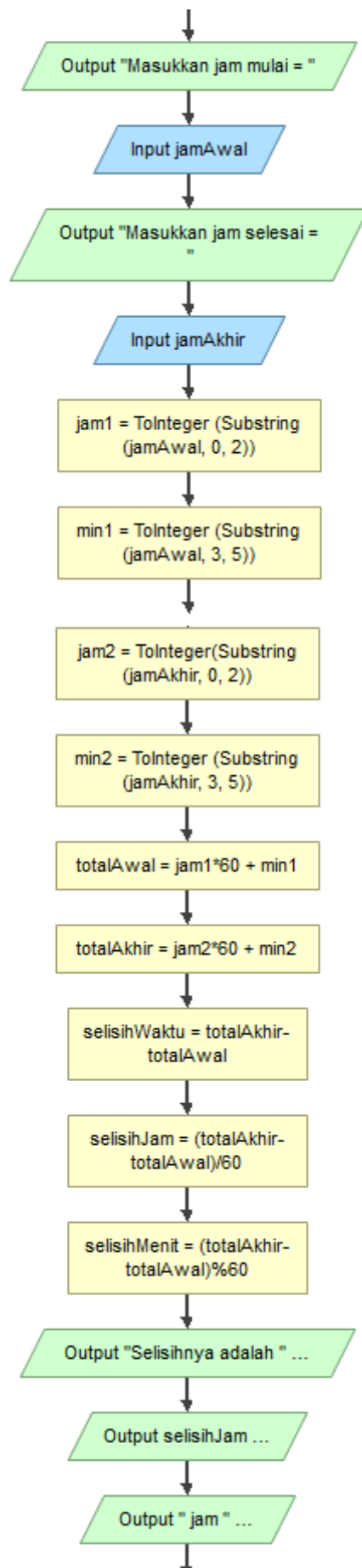
a. Pseudocode

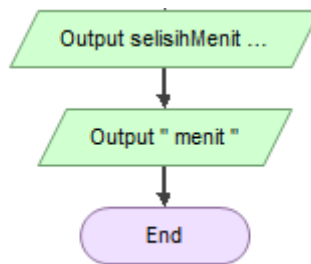
Algoritme untuk Menampilkan Input dan Output { Algoritme ini digunakan untuk menampilkan input dan output }	
1	DEKLARASI
2	jamAwal, jamAkhir : String
3	jam1, jam2, min1, min2, totalAwal, totalAkhir, selisihWaktu, selisihJam,
4	selisihMenit : integer
5	DESKRIPSI MAIN METHOD
6	Cetak "Masukkan jam mulai = "
7	Input jamAwal
8	Cetak "Masukkan jam selesai = "
9	Input jamAkhir
10	jam1 \leftarrow StringJadiInteger(Substring(jamAwal, 0, 2))
11	min1 \leftarrow StringJadiInteger(Substring(jamAwal, 3, 5))
12	jam2 \leftarrow StringJadiInteger(Substring(jamAkhir, 0, 2))
13	min2 \leftarrow StringJadiInteger(Substring(jamAkhir, 3, 5))
14	totalAwal \leftarrow jam1*60 + min1
15	totalAkhir \leftarrow jam2*60 + min2
16	selisihWaktu \leftarrow totalAkhir-totalAwal
17	selisihJam \leftarrow (totalAkhir-totalAwal)/60
18	selisihMenit \leftarrow (totalAkhir-totalAwal)%60
19	Cetak "Selisihnya adalah ", selisihJam, " jam ", selisihMenit, " menit "
20	DESKRIPSI StringJadiInteger(String text) METHOD
21	Menerima text : String
22	Deklarasi hasil : integer
23	Konversi text menjadi integer
24	Keluaran : hasil
25	DESKRIPSI Substring(String text, int start, int length)
26	Menerima text : String
27	Menerima start, length : integer
28	Deklarasi hasil : String
29	Ambil digit dari String dari start hingga sepanjang length
30	Keluaran : hasil

b. Flowchart

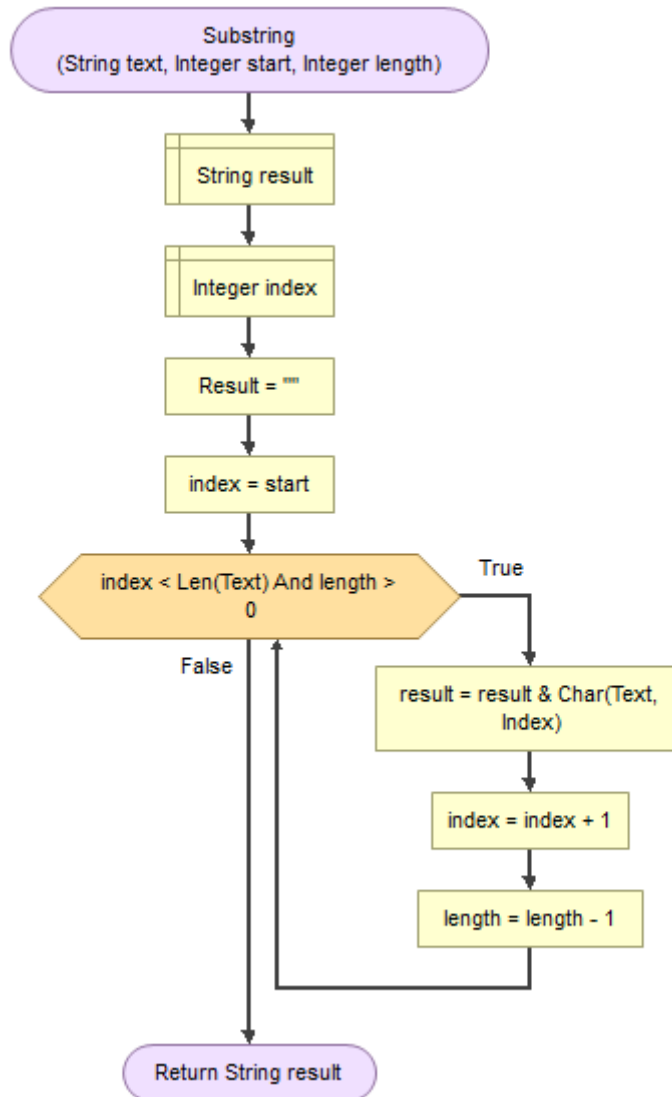
- Main Method







- ToInteger Method → Function bawaan di flowgorithm
- Substring Method



- c. Source Code (diktik dengan font Consolas/Courier New seperti contoh berikut)

Source Code Untuk Menampilkan Input dan Output	
1	import java.util.Scanner;
2	
3	/**

4	Nama 1 : Pieter Christy Yan Yudhistira 1 (235150201111068)
5	Nama 2 : Kaneysha Nadetta Julief 2 (235150200111067)
6	Nama 3 : Christopher Robin Tanugroho 3 (235150201111067)
7	
8	Tanggung jawab 1: mengerjakan soal nomor 1, membuat program Java yang tertera di source code, mendebug kesalahan argumen substring untuk mendapatkan jam dan menitnya, membuat substring function tersendiri di flowgorithm, mengecek kembali format pseudocode nomor 1, menyusun format pdf dan membuat pembahasan source code nomor 1 bersama tanggung jawab 2, mengrevisi
9	pembahasan source code nomor 2 dengan tanggung jawab 3. Tanggung jawab 2: mengerjakan soal nomor 1, membuat flowchart di flowgorithm, membuat pseudocode mengikuti flowchart, dan membuat pembahasan source code bersama
10	tanggung jawab 1. Tanggung jawab 3: mengerjakan soal nomor 2, membuat program Java yang tertera di source code, mengimplementasikan format output dengan NumberFormat dan LocalDate untuk output program dan membuat
11	pembahasan source code nomor 2.
12	
13	**/
14	
15	public class
16	T1_235150201111068_PieterChristyYanYudhistira_1 {
17	public static void main(String[] args) {
18	// 1) Deklarasi scanner dan variabel yang
19	dibutuhkan
20	Scanner scan = new Scanner(System.in);
21	String jamAwal, jamAkhir;
22	int jam1, min1, jam2, min2, totalAwal,
23	totalAkhir, selisihWaktu, selisihJam, selisihMenit;
24	// 2) Minta user untuk input jam mulai dan jam
25	selesai
26	System.out.print("Masukkan jam mulai = ");
27	jamAwal = scan.nextLine();
28	System.out.print("Masukkan jam selesai = ");
29	jamAkhir = scan.nextLine();
30	// 3) Mengubah jamAwal dan jamAkhir menjadi
31	masing-masing jam dan menit
32	jam1 =
33	Integer.parseInt(jamAwal.substring(0,2));
34	min1 =
35	Integer.parseInt(jamAwal.substring(3,5));
36	jam2 =
37	Integer.parseInt(jamAkhir.substring(0,2));

29	min2 =
	Integer.parseInt(jamAkhir.substring(3,5));
30	// 4) Ubah total waktu menjadi menit
31	totalAwal = jam1*60 + min1;
32	totalAkhir = jam2*60 + min2;
33	// 5) Cari selisih waktu dan cari jam selisih dan menit selisih
34	selisihWaktu = totalAkhir-totalAwal;
35	selisihJam = (selisihWaktu)/60;
36	selisihMenit = (selisihWaktu)%60;
37	// 6) Output selisih waktunya dalam jam dan menit
38	System.out.printf("Selisihnya adalah %d jam %d menit\n", selisihJam, selisihMenit);
39	scan.close();
40	}
41	}

d. Pembahasan/Penjelasan dari Source Code

Baris 1, import java.util.Scanner untuk mengambil data input dari pengguna/user. Baris 3 sampai 12, membuat komen untuk data mahasiswa dan tanggung jawab dalam tugas ini. Baris 14 dan 15, deklarasi public class sesuai dengan format file nama java yatu **public class T1_235150201111068_PieterChristyYanYudhistira_1** dan deklarasi metode main utama yaitu **public static void main(String[] args)**.

Baris 17, deklarasi variabel input sebagai scanner dengan **Scanner scan = new Scanner(System.in)**. Baris 18, deklarasi variabel jamAwal dan jamAkhir dengan tipe data string. Baris 19, deklarasi variabel jam1, min1, jam2, min2, totalAwal, totalAkhir, selisihWaktu, selisihJam, dan selisihMenit dengan tipe data integer.

Baris 21, menggunakan kelas System dengan metode print untuk menampilkan "Masukkan jam mulai = " agar baris dengan input-nya tidak berganti. Baris 22, variabel jamAwal diisi dengan scanner input string menggunakan nextLine() agar pengguna dapat menginput data dalam bentuk angka dan tanda baca berupa titik dua (:). Baris 23, menggunakan kelas System dengan metode print untuk menampilkan "Masukkan jam selesai = " agar baris input dan output print berada di satu baris. Baris 24, diisi variabel jamAkhir dengan scanner input string menggunakan nextLine() agar pengguna dapat menginput data dalam bentuk angka dan tanda baca berupa titik dua (:).

Baris 26, pertama mengambil dua karakter pertama jam mulai sebagai jam menggunakan substring() dengan index awal 0 dan index akhir 2, kemudian dikonversikan dari String menjadi integer melalui method parseInt() dalam tipe

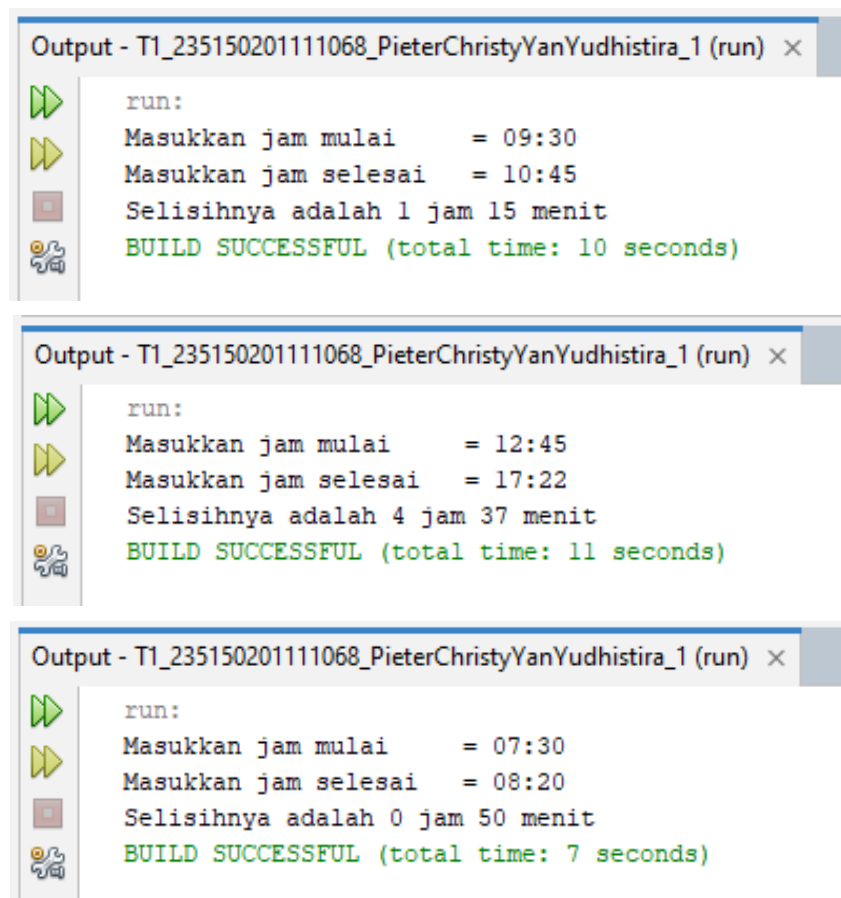
data Integer yaitu kodenya **jam1** = **Integer.parseInt(jamAwal.substring(0,2))**. Baris 27, pertama mengambil dua karakter terakhir jam mulai sebagai menit menggunakan substring dengan index awal 3 dan index akhir 5, kemudian dikonversikan dari String menjadi integer melalui method **parseInt()** dalam tipe data Integer yaitu kodenya **min1 = Integer.parseInt(jamAwal.substring(3,5))**. Baris 28, pertama mengambil dua karakter pertama jam selesai sebagai jam menggunakan **substring()** dengan index awal 0 dan index akhir 2, kemudian dikonversikan dari String menjadi integer melalui method **parseInt()** dalam tipe data Integer yaitu kodenya **jam2 = Integer.parseInt(jamAkhir.substring(0,2))**. Baris 29, pertama mengambil dua karakter terakhir jam selesai sebagai menit menggunakan substring dengan index awal 3 dan index akhir 5, kemudian dikonversikan dari String menjadi integer melalui method **parseInt()** dalam tipe data Integer yaitu kodenya **min2 = Integer.parseInt(jamAkhir.substring(3,5))**.

Baris 31, variabel **totalAwal** diisikan hasil kali variabel **jam1** dengan 60 lalu menambahkannya dengan **min1** untuk mengubah input pengguna menjadi menit. Baris 32, variabel **totalAkhir** diisikan hasil kali variabel **jam2** dengan 60 lalu menambahkannya dengan **min2** untuk mengubah input pengguna menjadi menit.

Baris 34, variabel **selisihWaktu** diisikan hasil kurang variabel **totalAkhir** dengan variabel **totalAwal** untuk menghitung jarak waktu dalam menit. Baris 35, variabel **selisihJam** diisikan hasil bagi variabel **selisihWaktu** dengan 60 untuk untuk mengambil hanya angka bulat sehingga kami mendapatkan jamnya saja. Baris 36, variabel **selisihMenit** hasil modulus variabel **selisihWaktu** dengan 60 untuk untuk mengambil hanya sisa dari pembagian bilangan bulat sehingga kami mendapatkan menitnya saja.

Baris 38, mencetak dengan kelas **System** dengan metode **printf** untuk menampilkan "Selisihnya adalah %d jam %d menit\n" dimana %d pertama mewakili hasil operasi dari variabel **selisihJam**, dan %d kedua mewakili hasil operasi dari **selisihMenit** dan \n untuk mencetak ke baris baru. Baris 39 menutup scanner dengan **scan.close()**. Baris 40, kurung kurawal untuk menutup kode metode **main** utama yang diawali di baris 15. Baris 41, kurung kurawal untuk menutup kode **public class** yang diawali di baris 14.

e. Screenshot



2. Soal 2

a. Source Code

Source Code Untuk Menampilkan Input dan Output	
1	import java.util.Scanner;
2	import java.time.LocalDate;
3	import java.util.Locale;
4	import java.text.NumberFormat;
5	
6	/**
7	Nama 1 : Pieter Christy Yan Yudhistira 1
	(235150201111068)
8	Nama 2 : Kaneysha Nadetta Julief 2
	(235150200111067)
9	Nama 3 : Christopher Robin Tanugroho 3
	(235150201111067)
10	Tanggung jawab 1: mengerjakan soal nomor 1, membuat program Java yang tertera di source code, mendebug kesalahan argumen substring untuk mendapatkan jam dan menitnya, membuat substring function tersendiri di flowgorithm, mengecek kembali format pseudocode nomor 1,

	dan membuat pembahasan source code bersama tanggung jawab 2.
11	Tanggung jawab 2: mengerjakan soal nomor 1, membuat flowchart di flowgorithm, membuat pseudocode mengikuti flowchart, dan membuat pembahasan source code bersama tanggung jawab 1.
12	Tanggung jawab 3:mengerjakan soal nomor 2, membuat program Java yang tertera di source code, mengimplementasikan format output dengan NumberFormat dan LocalDate untuk output program dan membuat pembahasan source code nomor 2..
13	
14	*/
15	public class
	Tl_235150201111068_PieterChristyYanYudhistira_2 {
16	public static void main(String[] args) {
17	Scanner input = new Scanner (System.in);
18	String ID, name;
19	char x = ' ';
20	double meal, beverage, dessert, payment, qty,
	qty1, qty2, sub, sub1, sub2;
21	double total, change;
22	System.out.print("Customer ID \t: ");
23	ID = input.nextLine();
24	System.out.print("Customer Name \t: ");
25	name = input.nextLine();
26	System.out.println("Order Price: ");
27	System.out.print("Meal, Qty \t: ");
28	meal = input.nextDouble();
29	qty = input.nextDouble();
30	System.out.print("Beverage, Qty \t: ");
31	beverage = input.nextDouble();
32	qty1 = input.nextDouble();
33	System.out.print("Dessert, Qty \t: ");
34	dessert = input.nextDouble();
35	qty2 = input.nextDouble();
36	System.out.print("Payment Money \t: ");
37	payment = input.nextDouble();
38	System.out.println("");
39	
40	sub = meal * qty;
41	sub1 = beverage * qty1;
42	sub2 = dessert * qty2;
43	total = sub + sub1 + sub2;
44	change = payment - total;
45	
46	NumberFormat a = NumberFormat.getNumberInstance
	(Locale.GERMANY);
47	NumberFormat b = NumberFormat.getNumberInstance
	(Locale.GERMANY);

48	NumberFormat c = NumberFormat.getNumberInstance (Locale.GERMANY);
49	NumberFormat d = NumberFormat.getNumberInstance (Locale.GERMANY);
50	NumberFormat e = NumberFormat.getNumberInstance (Locale.GERMANY);
51	NumberFormat f = NumberFormat.getNumberInstance (Locale.GERMANY);
52	NumberFormat g = NumberFormat.getNumberInstance (Locale.GERMANY);
53	NumberFormat h = NumberFormat.getNumberInstance (Locale.GERMANY);
54	NumberFormat i = NumberFormat.getNumberInstance (Locale.GERMANY);
55	
56	String aa = a.format(meal);
57	String bb = b.format(beverage);
58	String cc = c.format(dessert);
59	String dd = d.format(payment);
60	String ee = e.format(sub);
61	String ff = f.format(sub1);
62	String gg = g.format(sub2);
63	String hh = h.format(total);
64	String ii = i.format(change);
65	
66	LocalDate today = LocalDate.now();
67	System.out.println("<<Printing Invoice>>\n");
68	System.out.println("Customer ID \t: " + ID);
69	System.out.println("Customer Name \t: " + name);
70	System.out.println("Date \t\t: " + today);
71	System.out.println("");
72	System.out.printf("%-9s %c %-12s %c %-8s %c %s\n", "Menu", x, "Price", x, "Qty", x, "SubTotal");
73	
74	System.out.println("===== =====");
75	System.out.printf("%-9s %c %13s %c %7.0f %c %9s\n", "Meal", x, aa, x, qty, x, ee);
76	System.out.printf("%-9s %c %13s %c %7.0f %c %9s\n", "Beverage", x, bb, x, qty1, x, ff);
77	System.out.printf("%-9s %c %13s %c %7.0f %c %9s\n", "Dessert", x, cc, x, qty2, x, gg);
78	System.out.println("===== =====");
79	System.out.printf("%32s %4c %9s\n", "Total", x, hh);
80	System.out.printf("%34s %2c %9s\n", "Payment", x, dd);

81	System.out.println("=====
82	=====");
83	System.out.printf("%33s %3c %9s\n", "Change", x,
84	ii);
85	input.close();
	}
	}

b. Pembahasan/Penjelasan dari Source Code

Baris 1 sampai 4, import java.util.Scanner yang berfungsi mengambil data dari pengguna/user, java.time.LocalDate yang berfungsi untuk menampilkan waktu ketika proses output dilakukan, java.util.Locale untuk lokalisasi waktu dan Number Format dan java.text.NumberFormat untuk **Instance Number** (seperti penggunaan tanda titik pada ratusan ribu. Contoh : 100.000). Baris 6 sampai 15, membuat komen untuk data mahasiswa dan tanggung jawab dalam tugas ini. Baris 14 dan 15, deklarasi public class sesuai dengan format file nama java yaitu **public class T1_235150201111068_PieterChristyYanYudhistira_2** dan deklarasi metode main utama yaitu **public static void main(String[] args)**.

Baris 17, deklarasi variabel input sebagai scanner dengan **Scanner scan = new Scanner(System.in)**. Baris 18, deklarasi variabel ID dan name dengan tipe data String. Baris 19, deklarasi variabel x dan diisikan ']' dengan tipe data char. Baris 20 dan 21, deklarasikan variabel meal, beverage, dessert, payment, qty, qty1, qty2, sub, sub1, sub2, total, dan change dengan tipe data double.

Baris 22, menggunakan kelas System dengan metode print untuk menampilkan "Customer ID \t: ". Baris 23, variabel ID diisikan dengan scanner input menggunakan nextLine dikarenakan bertipe data String. Baris 24, menggunakan kelas System dengan metode print lagi untuk menampilkan "Customer name \t: ". Baris 25, variabel name diisikan dengan scanner input menggunakan nextLine dikarenakan bertipe data String.

Baris 26, menggunakan kelas System kembali dengan metode println agar otomatis new line untuk menampilkan "Order Price : ". Baris 27, menggunakan kelas System dengan metode print untuk menampilkan "Meal, Qty \t: ". Baris 28 dan 29, variabel meal dan qty diisikan dengan scanner input dengan nextDouble dikarenakan tipe data variabel itu double. Baris 30, menggunakan kelas System dengan metode print untuk menampilkan "Beverage, Qty /t: ". Baris 31 dan 32, variabel beverage dan qty1 diisikan dengan scanner input menggunakan nextDouble dikarenakan tipe data variabel itu double.

Baris 33, menggunakan kelas System dengan metode print untuk menampilkan "Dessert, Qty \t: ". Baris 34 dan 35 variabel dessert dan qty2 diisikan dengan scanner input menggunakan nextDouble dikarenakan tipe data variabel itu double. Baris 36, menggunakan kelas System dengan metode print untuk

menampilkan "Payment Money \t: ". Baris 37, variabel payment diisi dengan scanner input menggunakan nextDouble dikarenakan tipe data variabel itu double. Mengenai penggunaan "\t" pada kode adalah untuk menyamakan titik dua agar sejajar seperti pada contoh output yang diberikan.

Baris 40, variabel sub diisi dengan hasil perkalian harga meal * qty. Baris 41, variabel sub1 diisi dengan hasil perkalian harga beverage * qty1. Baris 42, variabel sub2 diisi dengan hasil perkalian harga dessert * qty2. Baris 43, variabel total diisi dengan hasil penjumlahan semua biaya yang dikeluarkan dengan format "s + sub1 + sub2". Baris 44, variabel change diisi dengan hasil pengurangan untuk menghitung kembalian dengan format "payment - total".

Baris 46 hingga 54, membuat variabel baru dari a sampai dengan i dengan "NumberFormat (variabel) = NumberFormat.getInstance(Locale.GERMANY)". Hal ini untuk menampilkan input yang diatas (contohnya 15000) menjadi 15.000 (terdapat titik sebagai pengidentifikasi bilangan ribu). Baris 56 hingga 64, membuat variabel aa untuk format variabel meal, bb untuk format variabel beverage, cc untuk format variabel dessert, dd untuk format variabel payment, ee untuk format variabel sub, ff untuk format variabel sub1, gg untuk format variabel sub2, hh untuk format variabel total, dan ii untuk format variabel change dengan tipe data String untuk menampilkan variabel a sampai dengan i yang sudah dideklarasikan dengan NumberFormat.getInstance.

Baris 66, variabel today diisi dengan menggunakan kelas LocalDate untuk menampilkan tanggal ketika proses penginputan terjadi. Baris 67, menggunakan kelas System dengan metode println untuk menampilkan "<<Printing Invoice>>\n". Baris 68, menggunakan kembali kelas System metode println untuk menampilkan "Customer ID \t: " + ID. Baris 69, menggunakan kembali kelas System metode println untuk menampilkan "Customer Name \t: " + name. Baris 70, menggunakan kembali kelas System metode println untuk menampilkan "Date \t\t: " + today. Baris 71, menggunakan kelas System.out metode println() agar ada new line yang memisahkan ID customer, nama customer dan hari customer.

Baris 72, menggunakan kelas System metode format printf untuk menampilkan **Menu | Price | Qty | SubTotal** dengan format "%-9s %c %-12s %c %-8s %c %s\n"%-9s (mewakili String "Menu" dengan 9 kolom rata kiri) %c (mewakili variabel x) %-12s (mewakili "Price", 12 kolom rata kiri) %c (mewakili variabel x) %-8s (mewakili "Qty", 8 kolom rata kiri) %c (mewakili variabel x) %s\n (mewakili "SubTotal"). Baris 74, menggunakan kelas System metode println untuk menampilkan "===== " sebagai pemisah atau divider, seperti pada contoh hasil output yang telah diberikan. Baris 75, menggunakan kelas System dengan metode format untuk menampilkan **Meal | (harga) | (kuantitas) | (total biaya)** dengan format %-9s (mewakili String "Meal" dengan 9 kolom rata kiri), %c (mewakili variabel x), %13s (mewakili variabel aa, 13 kolom rata kanan), %c (mewakili variabel x), %7.0f

(mewakili variabel qty, 7 kolom rata kanan), **%c** (mewakili variabel x), dan **%9s\n** (mewakili variabel ee, 9 kolom rata kanan).

Baris 76, menggunakan kelas System dengan metode format untuk menampilkan **Beverage | (harga) | (kuantitas) | (total biaya)** dengan format **%-9s** (mewakili String "Beverage" dengan 9 kolom rata kiri), **%c** (mewakili variabel x), **%13s** (mewakili variabel bb,13 kolom rata kanan), **%c** (mewakili variabel x), **%7.0f** (mewakili variabel qty1, 7 kolom rata kanan), **%c** (mewakili variabel x), dan **%9s\n** (mewakili variabel ff, 9 kolom rata kanan). Baris 77, menggunakan kelas System dengan metode format untuk menampilkan **Dessert | (harga) | (kuantitas) | (total biaya)** dengan format **%-9s** (mewakili String "Dessert" dengan 9 kolom rata kiri), **%c** (mewakili variabel x), **%13s** (mewakili variabel cc,13 kolom rata kanan), **%c** (mewakili variabel x), **%7.0f** (mewakili variabel qty2, 7 kolom rata kanan), **%c** (mewakili variabel x), dan **%9s\n** (mewakili variabel gg, 9 kolom rata kanan).

Baris 79, menggunakan kelas System metode println untuk menampilkan "===== " sebagai pemisah/divider, seperti pada contoh hasil output yang telah diberikan. Baris 80, menggunakan kelas System dengan metode format untuk menampilkan **Total | (total harga)** dengan format **%32s** (mewakili "Total", 32 kolom rata kanan), **%4c** (mewakili variabel x, 4 kolom rata kanan) dan **%9s\n** (mewakili variabel hh, 9 kolom rata kanan). Baris 80, menggunakan kelas System dengan metode format untuk menampilkan **Payment | (total harga)** dengan format **%34s** (mewakili "Payment", 34 kolom rata kanan), **%2c** (mewakili variabel x, 2 kolom rata kanan) dan **%9s\n** (mewakili variabel dd, 9 kolom rata kanan).

Baris 83, Setelah itu, menggunakan kelas System metode println untuk menampilkan "===== " sebagai pemisah/divider, seperti pada contoh hasil output yang telah diberikan. Baris 84, menggunakan kelas System dengan metode format untuk menampilkan **Change | (total harga)** dengan format **%33s** (mewakili "Change", 33 kolom rata kanan), **%3c** (mewakili variabel x, 3 kolom rata kanan) dan **%9s\n** (mewakili variabel ii, 9 kolom rata kanan). Penggunaan "\n" pada kode adalah untuk membuat baris baru agar hasil output baris sebelumnya tidak terletak pada baris yang sama.

c. Screenshot

run:	run:
Customer ID : 12345678901234	Customer ID : 235150201111068
Customer Name : Bill Gates	Customer Name : Pieter Christy Yan Yudhistira
Order Price:	Order Price:
Meal, Qty : 15000 1	Meal, Qty : 140000 5
Beverage, Qty : 20000 2	Beverage, Qty : 178000 12
Dessert, Qty : 10000 1	Dessert, Qty : 25000 50
Payment Money : 100000	Payment Money : 5000000
<<Printing Invoice>>	<<Printing Invoice>>
Customer ID : 12345678901234	Customer ID : 235150201111068
Customer Name : Bill Gates	Customer Name : Pieter Christy Yan Yudhistira
Date : 2023-09-27	Date : 2023-09-27
Menu Price Qty SubTotal	Menu Price Qty SubTotal
Meal 15.000 1 15.000	Meal 140.000 5 700.000
Beverage 20.000 2 40.000	Beverage 178.000 12 2.136.000
Dessert 10.000 1 10.000	Dessert 25.000 50 1.250.000
Total 65.000	Total 4.086.000
Payment 100.000	Payment 5.000.000
Change 35.000	Change 914.000
BUILD SUCCESSFUL (total time: 54 seconds)	BUILD SUCCESSFUL (total time: 41 seconds)

run:	run:
Customer ID : 235150201111067	Customer ID : 235150200111067
Customer Name : Christopher Robin Tanugroho	Customer Name : Kaneysha Nadetta Julief
Order Price:	Order Price:
Meal, Qty : 400 10	Meal, Qty : 75000 15
Beverage, Qty : 750 3	Beverage, Qty : 50000 20
Dessert, Qty : 100 30	Dessert, Qty : 25000 20
Payment Money : 30000	Payment Money : 5000000
<<Printing Invoice>>	<<Printing Invoice>>
Customer ID : 235150201111067	Customer ID : 235150200111067
Customer Name : Christopher Robin Tanugroho	Customer Name : Kaneysha Nadetta Julief
Date : 2023-09-27	Date : 2023-09-27
Menu Price Qty SubTotal	Menu Price Qty SubTotal
Meal 400 10 4.000	Meal 75.000 15 1.125.000
Beverage 750 3 2.250	Beverage 50.000 20 1.000.000
Dessert 100 30 3.000	Dessert 25.000 20 500.000
Total 9.250	Total 2.625.000
Payment 30.000	Payment 5.000.000
Change 20.750	Change 2.375.000
BUILD SUCCESSFUL (total time: 47 seconds)	BUILD SUCCESSFUL (total time: 41 seconds)