**LAPORAN PRAKTIKUM PROGRAM DASAR**

*Diajukan untuk memenuhi UAS mata kuliah Pemrograma Dasar Semester Ganjil*

*Tahun Ajaran 2021-2022*

Dosen Pengampu

**Rizdania S.T M.Kom**

**

Disusun oleh:

**AHMAD RUDIANTO (21157201114)**

**PROGRAM STUDI ILMU KOMPUTER 2021 B**

**FAKULTAS TEKNOLOGI DAN SAINS**

**SEKOLAH TINGGI KEGURUAN DAN ILMU PENDIDIKAN**

**UNIVERSITAS PGRI WIRANEGARA**

**2022**

**KATA PENGANTAR**

Puji syukur ke hadirat Tuhan Yang Maha Esa., Selawat serta salam semoga senantiasa dilimpahkan kepada Nabi muhammad SAW, beserta keluarganya, sahabatnya tabu’innya dan seluruh umatnya yang istiqomah mengikuti tuntunan dan teladan sampai akhir zaman.

Atas berkat Rahmat Tuhan Yang Maha Esa ,Kami dapat menyelesaikan tugas makalah yang berjudul "Laporan Praktikum Program Dasar " dengan tepat waktu.

Makalah disusun untuk memenuhi UAS Mata Kuliah Praktikum Pemrograman Dasar . saya selaku penulis mengucapkan terima kasih kepada Ibu Rizdania S.T M.Kom selaku guru Mata Kuliah Praktikum Pemrograman Dasar. Ucapan terima kasih juga disampaikan kepada semua pihak yang telah membantu diselesaikannya makalah ini.

Bagi saya sebagai penyusun merasa bahwa masih banyak kekurangan dalam penyusunan makalah ini karena keterbatasan pengetahuan dan pengalaman kami. Untuk itu kami sangat mengharapkan kritik dan saran yang membangun dari pembaca demi kesempurnaan.

Pasuruan,11 Januari 2022

# DAFTAR ISI

[DAFTAR ISI 3](#_Toc92913257)

[BAB 1 4](#_Toc92913258)

[1.1 Pengenalan java 4](#_Toc92913259)

[1.2 Instalasi JDK 4](#_Toc92913260)

[1.3 Editor Java 4](#_Toc92913261)

[BAB 2 6](#_Toc92913262)

[2.1 Membuat kelas Sederhana 6](#_Toc92913263)

[2.2 VARIABEL, DATA DAN OPERATOR 7](#_Toc92913264)

[2.3 Pemilihan 1 12](#_Toc92913265)

[2.3 Percabangan dalam percabangan (Nested if) 16](#_Toc92913266)

[2.4 ARRAY 1 20](#_Toc92913267)

[2.5 PERULANGAN 25](#_Toc92913268)

[2.6 Perulangan 2 30](#_Toc92913269)

[2.7 Array 2 33](#_Toc92913270)

# BAB 1

**PENGENALAN JAVA**

## 1.1 Pengenalan java

Java adalah salah satu bahasa pemrograman yang bersifat open source yang merupakan produk dari Sun Microsystem. Bahasa Java adalah bahasa modern yang telah diterima masyarakat komputasi dunia. Hampir semua perusahaan perangkat lunak dan komputer besar mendukung dan mengembangkan aplikasi sistem berbasis Java. Java adalah salah satu bahasa pemrogaman Multiplatform (Bisa berjalan di berbagai macam system operasi) karena pada dasarnya java mempunyai Jre (java runtime environment) atau dapat kita artikan sebagai mesin tersendiri untuk mengeksekusi binary code hasil dari compilasi program yang telah kita buat, bereda dengan bahasa pemrograman vb, c++ yang memanfaatkan komponen system dalam Windows untuk mengeksekusi binary code hasil kompilasi program.

Java adalah salah satu bahasa pemrogaman Multiplatform (Bisa berjalan di berbagai macam system operasi) karena pada dasarnya java mempunyai Jre (java runtime environment) atau dapat kita artikan sebagai mesin tersendiri untuk mengeksekusi binary code hasil dari compilasi program yang telah kita buat, bereda dengan bahasa pemrograman vb, c++ yang memanfaatkan komponen system dalam Windows untuk mengeksekusi binary code hasil kompilasi program. Pada Tahun 1996, Sun Microsystem secara resmi merilis versi awal Java yang kemudian terus berkembang hingga muncul JDK 1.1 (Java Development Kit versi 1.1). Perkembangan terus dilakukan hingga muncul versi baru yang disebut Java 2. Perubahan utama antara versi sebelumnya adalah adanya Swing yang merupakan teknologi Graphical User Interface (GUI) yang mampu menghasilkan aplikasi desktop yang benar-benar baik. Untuk mengembangkan aplikasi berbasis java, kita memerlukan Java Development Kit(JDK), saat ini java telah mencapai versi 2 yang kenal dengan java 2 platform. Java 2 platform, dibagi menjaditiga kategori yaitu:

1. Java 2 Standart Edition (J2SE)
2. Java 2 Enterprice Edition(J2EE)
3. Java 2 Micro Edition(J2ME)

## 1.2 Instalasi JDK

Sebelum anda membuat aplikasi dengan java maka anda harus menginstal software Bahasa pemrograman java. java dapat anda dapatkan dengan cara download di alamat link berikut http://www.oracle.com/technetwork/java/javase/downloads/index.html. Kemudian setelah anda mendownloadnya maka mulai dengan menginstal, pada dasarnya proses insalasi java sangat mudah yaitu dengan cara mengklik ganda pada filemaster javakemudian ikuti prosedur yang diperintahkan.

## 1.3 Editor Java

Ada banyak editor yang bisa kita gunakan untuk menuliskan soce code bahasa pemrograman java, bahkan semua editor dimungkinkan selama editor tersebut mampu menyimpan file dengan ekstensi .java yang merupakan ekstensi dari file untuk menyimpan source code bahasa pemrograman java.

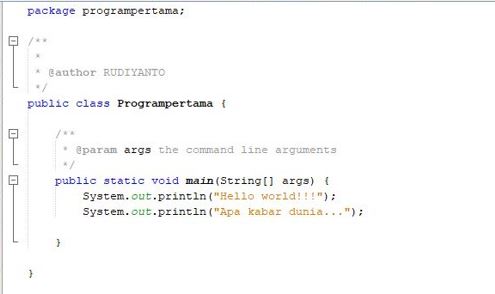
# BAB 2

**PEMBAHASAN**

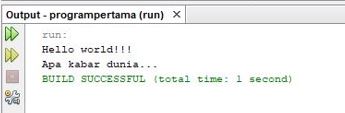
2.1 Membuat kelas Sederhana

* Kelas sederhana percoaan ke 1

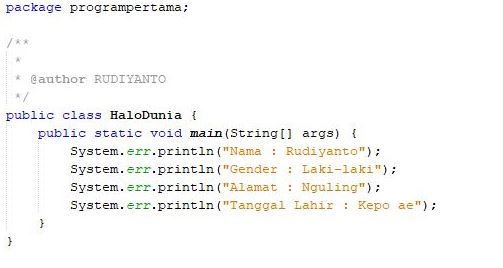
Membuat kelas sederhana pada pemrograman java, pada neatbeans klik file dan pilih new project, pilih java dan java applications kemudian next, isi projct name dengan PemrogramanPertama lalu klik finish.

Dibawah ini hasil program saya.

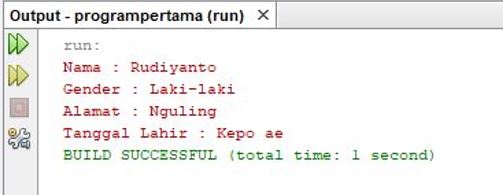
Untuk menjalankan program klik kanan lalu Run file, maka hasilnya ;



* kelas sederhana percobaan ke 2

Buat projek baru di netbeans dan sebuah class lalu ketik ulang listing program berikut ini!

Untuk menjalankan program klik kanan lalu Run file, maka hasilnya ;



2.2 VARIABEL, DATA DAN OPERATOR

* **Keyword dalam Java**

Keyword dalam Java Kata kunci adalah identifier yang telah dipesan untuk didefinisikan sebelumnya oleh Java untuk tujuan tertentu. Anda tidak dapat menggunakan keyword sebagai nama variabel, class, method.

* **Tipe Data Primitif**

Tipe Data Primitif Bahasa pemrograman Java mendefinisikan delapan tipe data primitif. Mereka diantaranya adalah boolean (untuk bentuk logika), char (untuk bentuk tekstual), byte, short, int, long (integral), double and float (floating point).

* **Variabel Variabel**

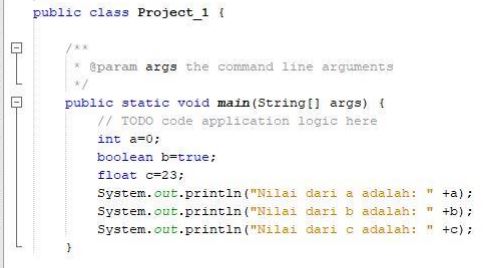
Variabel Variabel adalah item yang digunakan data untuk menyimpan pernyataan objek. Variabel memiliki tipe data dan nama. Tipe data menandakan tipe nilai yang dapat dibentuk oleh variabel itu sendiri. Nama variabel harus mengikuti aturan untuk identifier. Berikut Aturan penamaan variable :

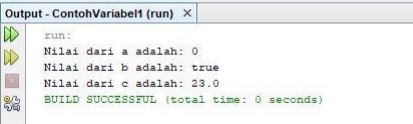
* Diawali dengan – huruf/abjad – karakter mata uang – underscore ( \_ )
* Terdiri dari huruf/abjad, angka dan underscore
* Tidak boleh mengandung karakter khusus atau spasi
* Tidak boleh diawali dengan angka
* **Deklarasi dan Inisialisasi Variabel**

Untuk deklarasi variabel adalah sebagai berikut, [=initial value]; Nilainya berada diantara <> adalah nilai yang disyaratkan, sementara nilai dalam tanda [] bersifat optional sesuai dengan tipe data yang dipakai untuk deklarasi variabel.

* **Menampilkan Data Variabel**

Untuk mengeluarkan nilai dari variabel yang diinginkan, kita dapat menggunakan perintah sebagai berikut: System.out.println() atau System.out.print().

* Contoh hasil praktikum saya dengan nama Contohvariabell :

 Jika di Run maka hasilnya :

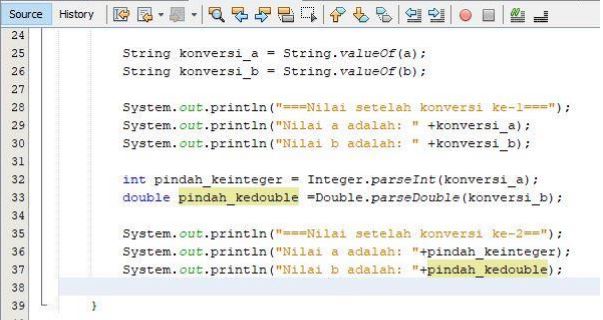
* **Teknik Konversi Data**

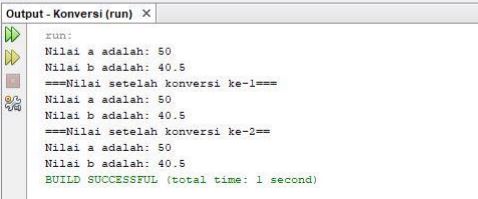
Konversi adalah cara untuk memberikan nilai suatu data dari satu tipe data ke tipe data yang lain. Yang perlu diketahui, bahwa konversi data dilakukan jika kita membutuhkan suatu tipe data yang sama untuk menjalankan sebuah proses, jika data tersebut sudah sama tipe datanya maka tidak perlu dilakukan proses konversi data. Untuk konversi data harus memenuhi dua criteria berikut ini :

1. Dua tipe data yang kompatibel

2. Tipe data awal lebih kecil dibandingkan dengan tipe data tujuan.

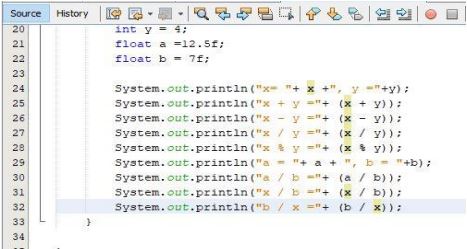
Jika dua syarat tersebut tidak terpenuhi maka bisa dilakukan teknik casting, yaitu melakukan perubahan tipe data dengan cara dipaksa. Casting ini bisa langsung dilakukan otomotis oleh java dengan cara menambahkan tipe data cating dikurung kemudian dilanjutkan dengan proses yang mau dijalankan. contoh : int a=4; int b=5; double c=(double)a/b;

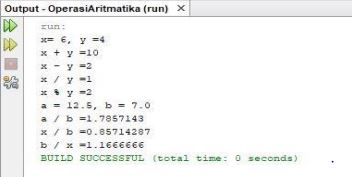
* Contoh hasil praktikum saya dengan nama konvesi

Jika di Run maka hasilnya :

* **Operator**

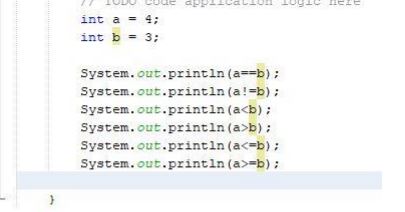
Operator Dalam Java, ada beberapa tipe operator. Ada operator aritmatika, operator relasi, operator logika, dan operator kondisi. Operator ini mengikuti bermacam-macam prioritas yang pasti sehingga compilernya akan tahu yang mana operator untuk dijalankan lebih dulu dalam kasus beberapa operator yang dipakai bersama-sama dalam satu pernyataan.

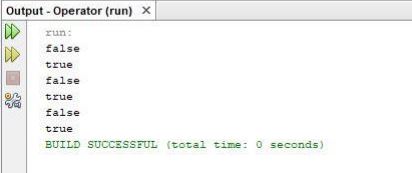
* Contoh hasil praktikum saya dengan nama OperasiAritmatika

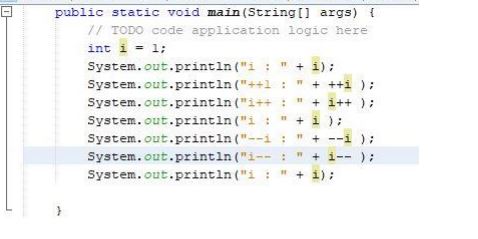
Jika di Run maka hasilnya :

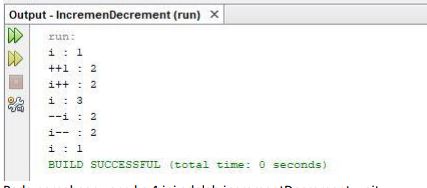
* **Operator Relasi**

Operator Relasi membandingkan dua nilai dan menentukan keterhubungan diantara nilai-nilai tersebut. Hasil keluarannya berupa nilai boolean yaitu true atau false.

* Contoh hasil praktikum saya dengan nama Operator :

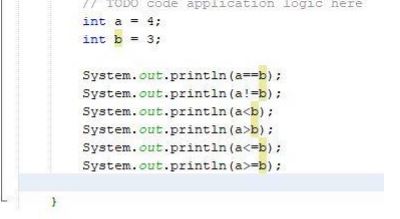
 Jika di Run maka hasilnya :

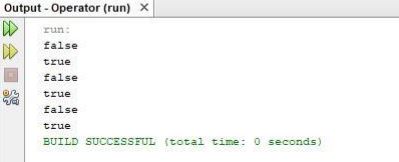
* **Operator Penjumlahan dan Pengurangan**
* Contoh hasil praktikum saya dengan nama IncremenDecrement.

Jika di Run maka hasilnya :

* **Operator Logika**

Operator logika memiliki satu atau lebih operand boolean yang menghasilkan nilai boolean. Terdapat enam operator logika yaitu: && (logika AND), & (boolean logika AND), || (logika OR), | (boolean logika inclusive OR), ^ (boolean logika exclusive OR), dan ! (logika NOT).

* Contoh hasil praktikum saya dengan nama Operatorlogika

 Jika di Run maka hasilnya :

2.3 Pemilihan 1

* **Fungsi kondisi “ if ”**

“if statement” adalah statement conditional yang paling sederhana dan sering digunakan. Jika kondisi pada ekspresi if menghasilkan nilai true, maka kode dalam blok akan dieksekusi, bila tidak program akan dilanjutkan ke baris kode setelah blok if.

* **Fungsi kondisi “ if else ”**

Variasi lain adalah “if else statement” yang menyajikan blok kode untuk dieksekusi bila kondisi bernilai false.

* **Kondisional Else if**

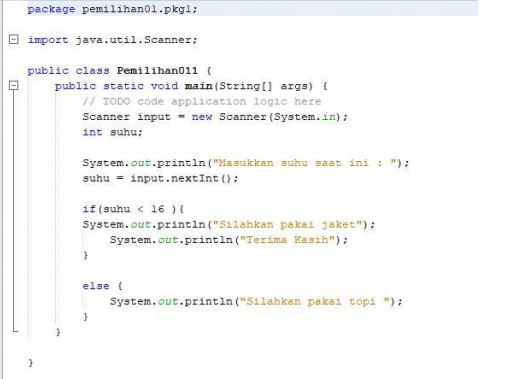
Statement “else if” dapat digunakan untuk memeriksa beberapa kondisi sekaligus.

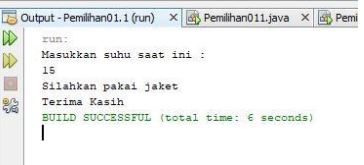
* **Switch Case**

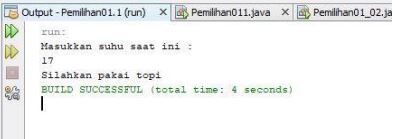
Kondisi SWITCH CASE adalah percabangan kode program dimana kita membandingkan isi sebuah variabel dengan beberapa nilai. Jika proses perbandingan tersebut menghasilkan true, maka block kode program akan di proses. Kondisi SWITCH CASE terdiri dari 2 bagian, yakni perintah SWITCH dimana terdapat nama variabel yang akan diperiksa, serta 1 atau lebih perintah CASE untuk setiap nilai yang akan diperiksa.

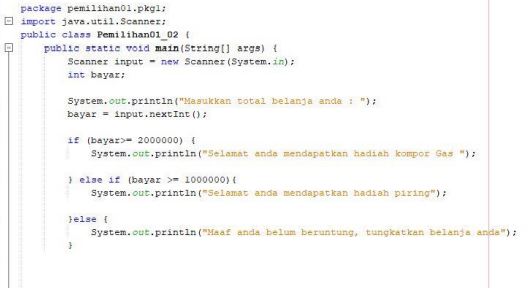
* **Ternary Operator**

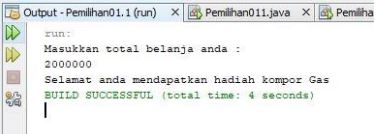
Operator ternary adalah salah satu operator yang digunakan di dalam bahasa pemrograman java. Mempunyai fungsi untuk menggantikan kondisi if else ,yang dimana jika suatu kondisi dapat bernilai true atau false, akan menampilkan output atau statement. Selain itu juga dengan menggunakan operator ternary , code akan terlihat lebih ringkas dan mudah di baca. Ciri-ciri jika baris code tersebut , menggunakan operator ternary. Adanya pernyataan atau kondisi yang di apit oleh simbol (?) dan juga (:).

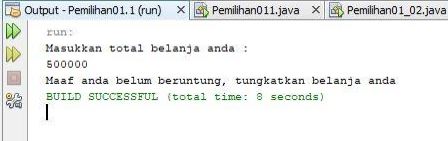
* Contoh hasil praktikum saya dengan nama Pemilihan01

Jika di Run maka hasilnya : Jika di inputkan angka < 16 misal,15 maka hasilnya “Silahkan pakai jaket”.

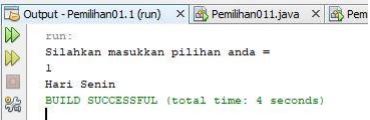
 Jika di inputkan angka 16 atau >16 misal,17 maka hasilnya “Silahkan pakai topi”.

* fungsi kondisional “ else if ”,
* Contoh hasil praktikum saya dengan nama pemilihan02

 Hasil running program; Jika di inputkan jumlah Rp 2000000 maka, hasilnya “Selamat anda mendapatkan hadiah kompor gas”

Jika di inputkan jumlah Rp 1000000 maka, hasilnya “Selamat anda mendapatkan hadiah piring”

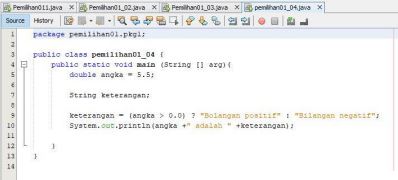
* fungsi “ Switch Case ”, buatlah project baru
* Contoh hasil praktikum saya dengan nama pemilihan01\_03

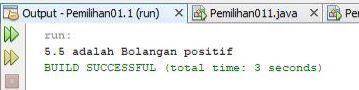
Jika di inputkan angka 1 maka, hasilnya “Hari senin

Dan jika di inputkan angka 2 maka, hasilnya “Hari selasa” Begitupun seterusnya sampai ke 7

Jika yang di inputkan angka selain 1 sampe 7 misal, angka 10 maka, hasilnya “Maaf angka yang anda masukkan salah”

* .fungsi “ Ternary Operator “

Contoh hasil praktikum saya dengan nama pemilihano1\_04

Hasil running

2.3 Percabangan dalam percabangan (Nested if)

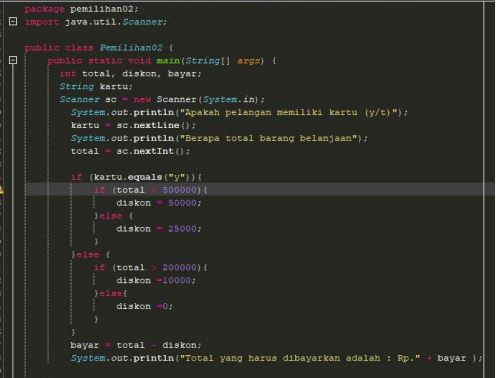
Kita telah mempelajari penggunaan pernyataan IF untuk memilih sebuah tidak, pernyataan IF-ELSE untuk memilih antara dua tindakan, serta pernyataan IF-ELSE IF-ELSE dan SWITCH-CASE untuk memilih beberapa tindakan (3 atau lebih). Terkadang kita membutuhkan pengambilan keputusan dalam bentuk level (bertingkat) sehingga di dalam suatu pernyataan IF (atau IF-ELSE) bisa saja terdapat pernyataan IF (atau IF-ELSE) yang lain. Jenis percabangan seperti ini disebut NESTED IF (percabangan bersarang).

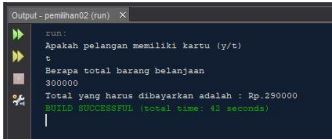
Pada bentuk penulisan pernyataan NESTED-IF tersebut, kondisi yang akan diseleksi pertama kali adalah kondisi IF yang berada di posisi terluar (kondisi 1).

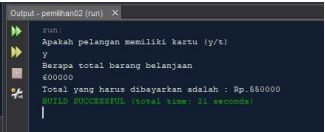
* Jika kondisi 1 bernilai salah, maka pernyataan ELSE terluar (pasangan dari IF yang bersangkutan) yang akan diproses. Namun, jika pernyataan ELSE (pasangan dari IF) tidak ditulis, maka penyeleksian kondisi akan dihentikan.
* Jika ternyata kondisi 1 bernilai benar, maka kondisi berikutnya yang lebih dalam (kondisi 2) akan diseleksi. Jika kondisi 2 bernilai salah, maka pernyataan ELSE (pasangan dari IF yang bersangkutan) yang akan diproses. Namun, jika pernyataan ELSE (pasangan dari IF) tidak ditulis, maka penyeleksian kondisi akan dihentikan.

Dengan cara yang sama, penyeleksian kondisi akan dilakukan sampai dengan kondisi n, jika kondisikondisi sebelumnya bernilai benar.

Berikut ini adalah contoh kode program untuk penggunaan NESTED IF pada contoh kasus pembayaran di kasir :

* **Percabangan dalam percabangan (Nested if)**

Jika di inputkan (t) sebagai pelanggan tidak memiliki kartu dan di inputkan total barang belanjaan 300000 maka akan di diskon 10000 menjadi =290000

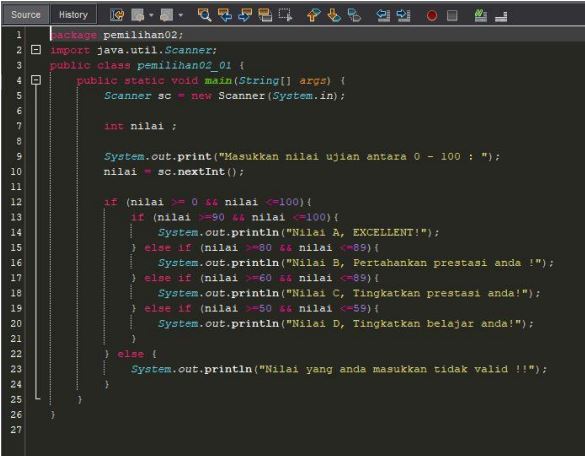
Jika di inputkan (y) sebagai pelanggan yang memiliki kartu dan di inputkan total barang belanjaan 600000 maka akan di diskon 50000 menjadi =550000

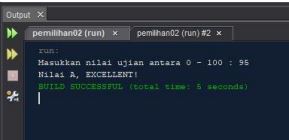
* **Penggunaan operator logika dalam percabangan Operator**

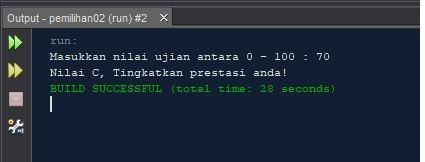
logika merupakan jenis operator yang akan membandingkan logika hasil dari Relational operator.

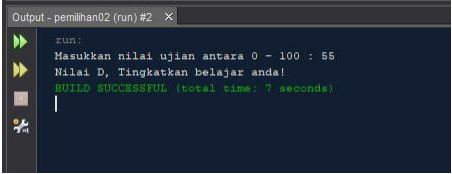
* && (AND) Jika semua operand bernilai benar
* (TRUE), maka kondisi tersebut dinyatakan bernilai benar.
* || (OR) Jika salah satu operand bernilai benar (TRUE), maka kondisi tersebut dinyatakan bernilai benar. !
* (NOT) Operator ini juga disebut sebagai invers atau negasi. Operator ini akan membalikan nilai boolean true menjadi false dan begitu juga sebaliknya.

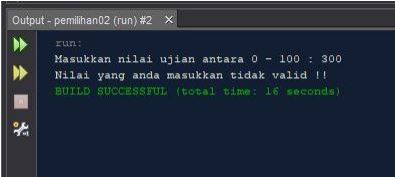
Contoh implementasi operator logika dalam percabangan: Operator logika digunakan untuk menggabungkan beberapa kondisi. Jika Anda ingin program yang menghasilkan output “Welcome!” hanya ketika usia variabel lebih besar dari 18 dan uang variabel lebih besar dari 500. Dalam kasus tersebut, Anda bisa menggunakan logika AND (&&) (deretan kode bagian atas)

Berikut ini adalah contoh kode program untuk penggunaan operator logika dalam percabangan Operator :

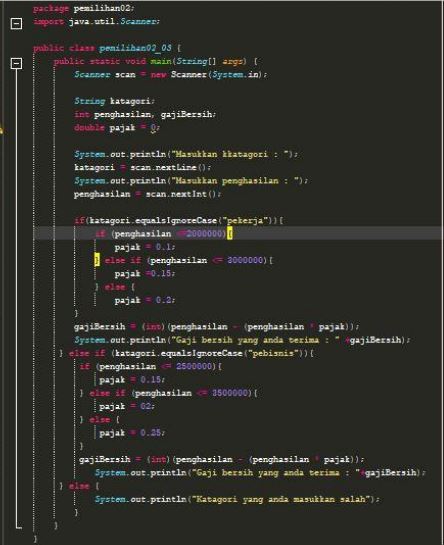
jika di inputkan nilai antara 90-100 misal 95 maka hasil nilainya ( nilai A, EXCELLENT! )

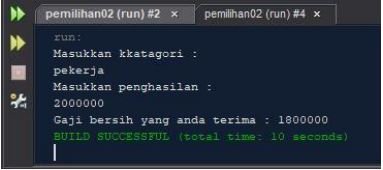
jika yang di inputkan nilai antara 60-79 misal 70 maka hasil nilainya ( nilai C,Tingkatkan prestasi anda!

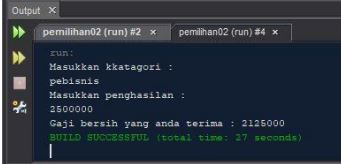
jika yang di inputkan nilai antara 50-59 misal 55 maka hasil nilainya ( nilai D, Tingkatkan belajar anda!)

jika yang di inputkan di luar progra atau ngasal misal 300 maka hasil nilainya (nilai yang anda masukkan tidak valid!!)

Pengecekan pertama digunakan untuk mengecek kategori (pekerja atau pebisnis). Selanjutnya dilakukan pengecekan kedua untuk menentukan besarnya pajak berdasarkan penghasilan yang telah dimasukkan. Kemudian tambahkan kode program untuk menghitung gaji bersih yang diterima setelah dipotong pajak.

* **Deklarasi variabel katagori,penghasilan gaji bersih dan pajak**

Running pebisnis

Running pekerja

2.4 ARRAY 1

Array adalah object yang digunakan untuk menyimpan banyak data dengan tipe yang sama. Tipe dari array bisa : tipe data primitif atau class. Terdapat 3 langkah untuk membuat array:

* Mendeklarasikan variabel array
* Memcreate array beserta ukurannya.
* Memberikan sebuah nilai padas etiap element array.

Membuat Array (Mendeklarasikan Variabel Array)

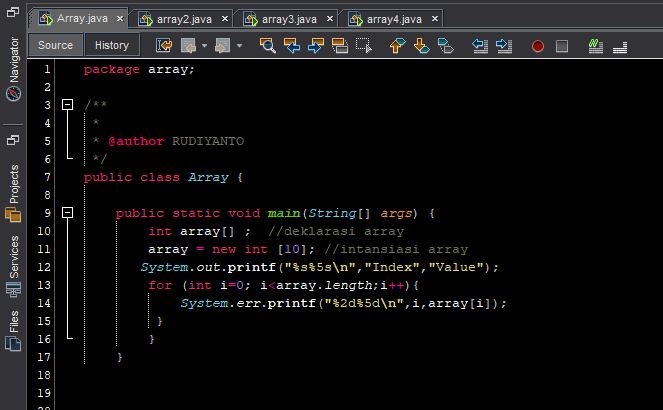
* int[] ints
* double[] dubs
* Dimension[] dims ;
* float[][] twoDee
* int [5] scores;

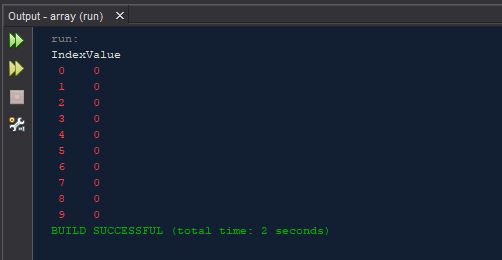
Membuat Array (Men-create Array Beserta Ukurannya) Karena array adalah sebuah object, maka create array dengan operator new. Besar array ditentukan pada saat runtime

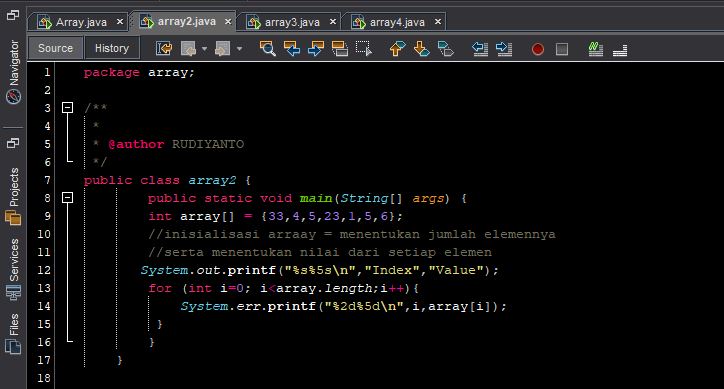
* int[] ints ; –
* ints = new int[25] ;

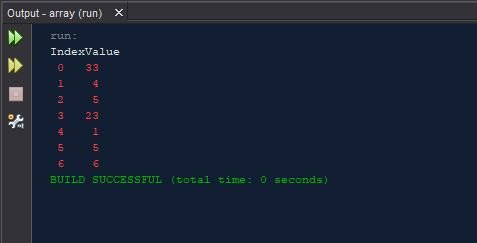
Pada saat array di buat, isi array diinisialisasi dengan default value.

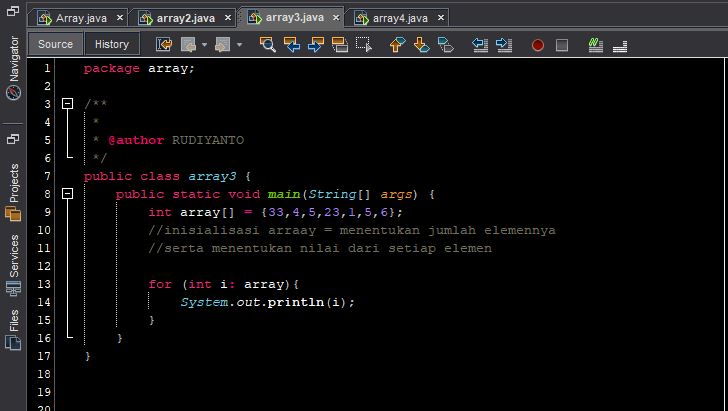
* Berikut ini adalah contoh kode program untuk penggunaan array

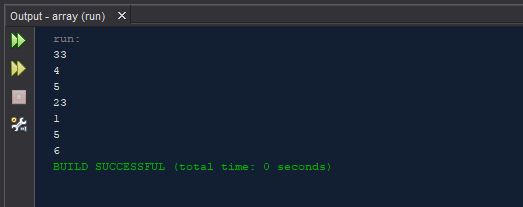
1. **- Percobaan array 1**

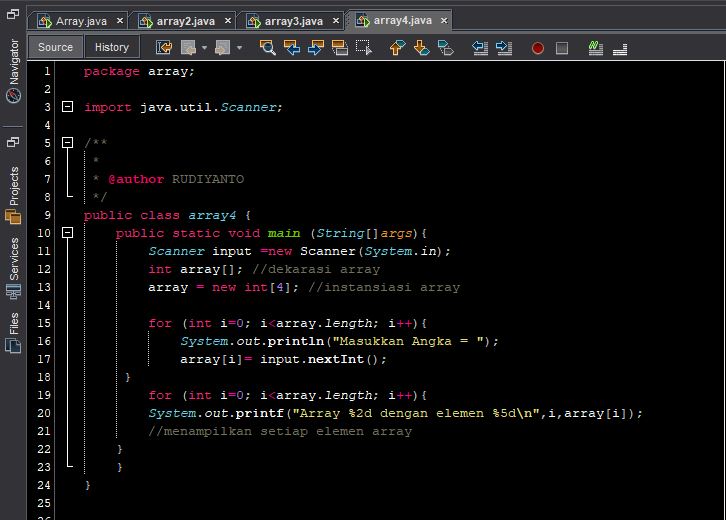
**-hasil Running**

**2 - Percobaan array2**

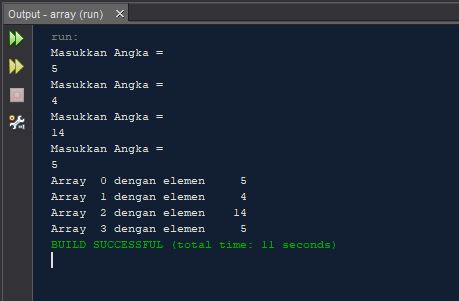
* **Hasil Runnuing**

**3 – Percobaan array3**

* **Hasil Running**

**4 – Percobaan array4**

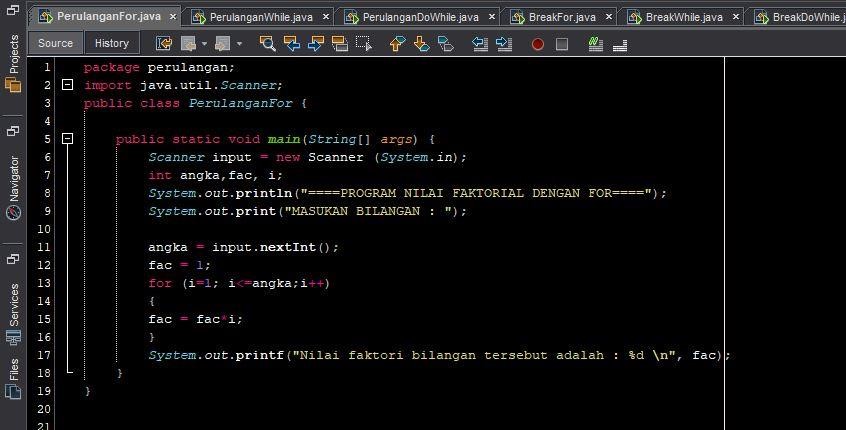
* **Hasil Running**



2.5 PERULANGAN

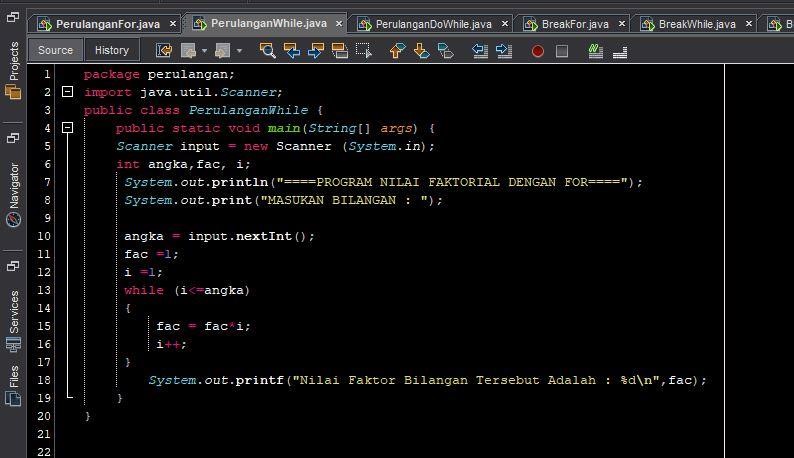
Perulangan adalah suatu blok atau kelompok instruksi yang dilaksanakan secara berulangulang. Perulangan akan membuat efisiensi proses dibandingkan jika dioperasikan secara manual. Kebanyakan aplikasi perangkat lunak melakukan pekerjaan berulang-ulang sampai sebuah kondisi yang diinginkan, oleh karena itu pengulangan merupakan bagian yang penting dalam pemrograman karena dengan adanya pengulangan pembuat program tidak perlu menulis kode program sebanyak pengulangan yang diinginkan. Perulangan yang dijelaskan pada jobsheet ini adalah :

* Perulangan dengan for
* Perulangan dengan while
* Perulangan dengan do-while For For adalah kode yang digunakan untuk menjalankan serangkaian kode secara berulang-ulang.
* **For**

For adalah kode yang digunakan untuk menjalankan serangkaian kode secara berulang-ulang. Pada kode for ini terdapat beberapa komponen yang dicantumkan, antara lain: (1) inisialisasi, (2) kondisi, (3) perubahan nilai, (4) statement yang diulang

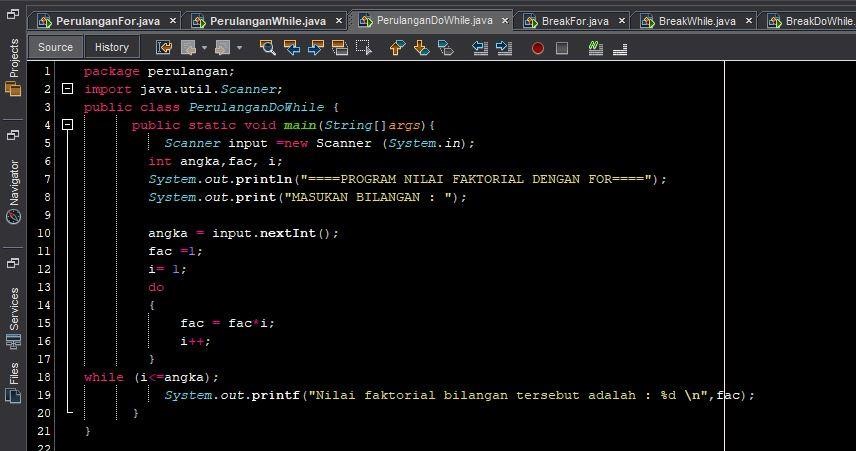
* **While**

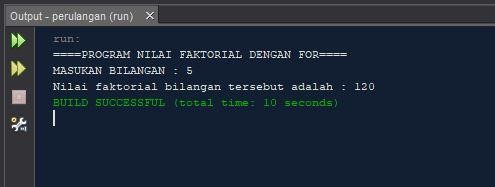
Kode while merupakan kode alternatif untuk melakukan perulangan selain for. Perulangan while biasanya digunakan jika jumlah perulangan tidak diketahui atau memiliki kemungkinan dapat dilakukan kurang dari batas perulangan yang telah ditentukan. Perulangan while hanya akan melakukan perulangan selama kondisi perulangan terpenuhi. Perintah-perintah akan dilaksanakan apabila ekspresi boolean dalam keadaan true. Di dalam loop ada nilai yang mengontrol loop dan nilainya harus berubah, sehingga pada akhir program akan keluar dari loop.



* **Do – While**

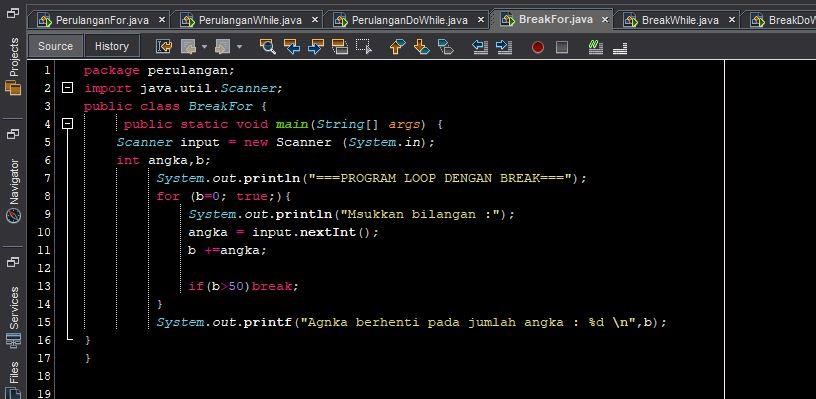
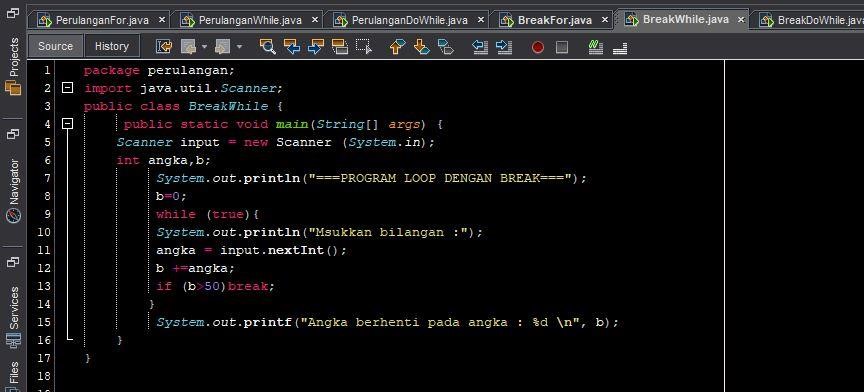
Kode do-while merupakan kode while-do dengan sintaks yang berbeda. Cara kerja do-while relatif sama dengan while. Perintah do-while() akan mengulang statement miliknya selama syarat pengulangannya terpenuhi. Hanya saja, perintah do-while() menjalankan statementnya terlebih dahulu, setelah itu baru memeriksa syaratnya. Sedangkan perintah while() memeriksa syarat terlebih dahulu. Oleh karena itu, perintah do-while() akan menjalankan statementnya paling tidak sebanyak satu kali, meskipun syarat pengulangan tidak terpenuhi.

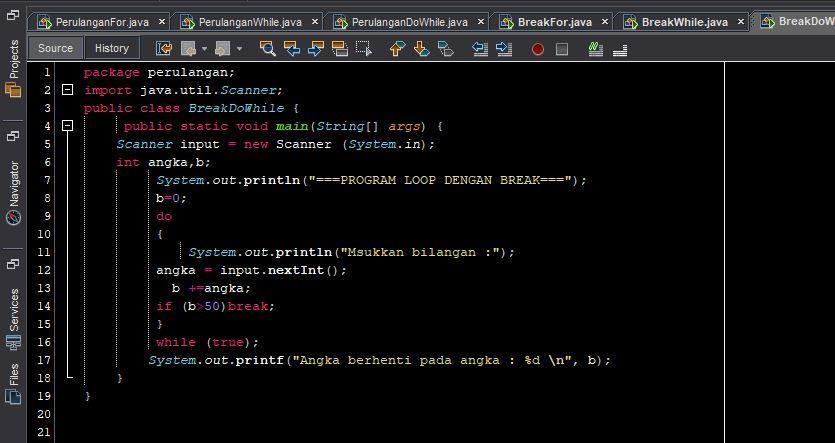


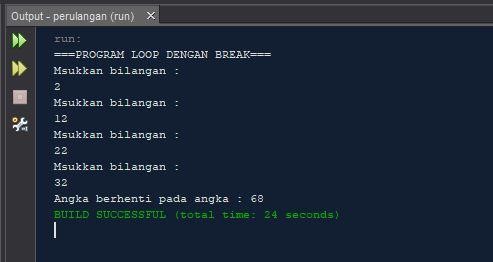


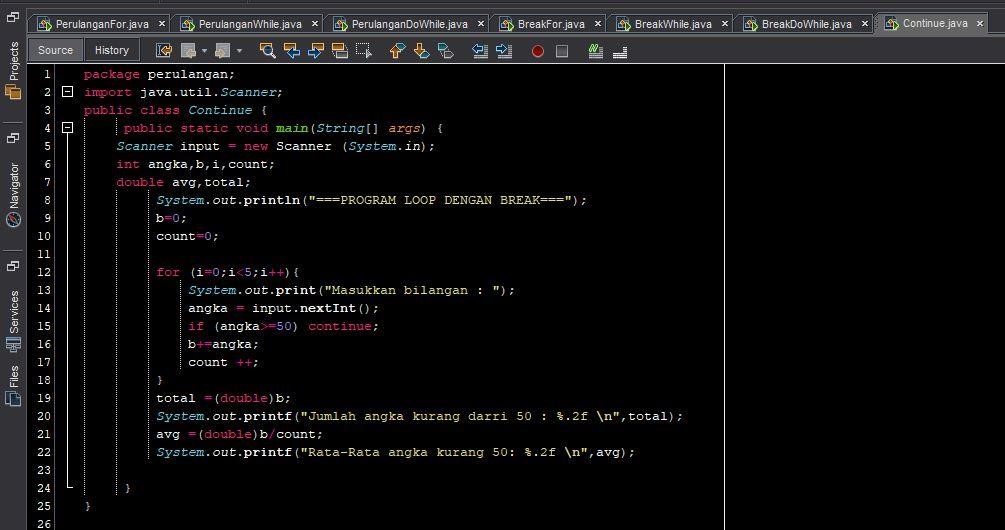
* **Break dan Continue**

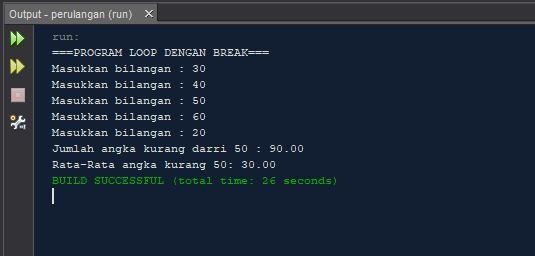
Break dan continue tergolong ke dalam keyword di bahasa pemrograman java, yang keduanya digunakan pada suatu kondisi tertentu , pada perulangan seperti while ,do while dan for. Jika fungsi break digunakan untuk menghentikan suatu pernyataan (statement), dan jika fungsi continue digunakan untuk mengabaikan ,lalu melanjutkan suatu pernyataan pada perulangan. Keyword break dan continue juga biasa digunakan, bersamaan dengan Control Flow seperti if else, dan switch case di dalam program java







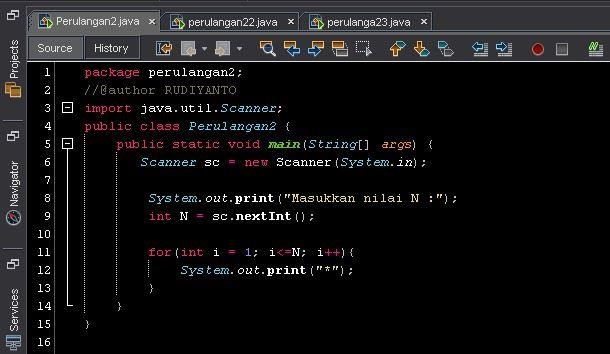


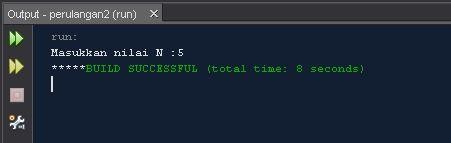


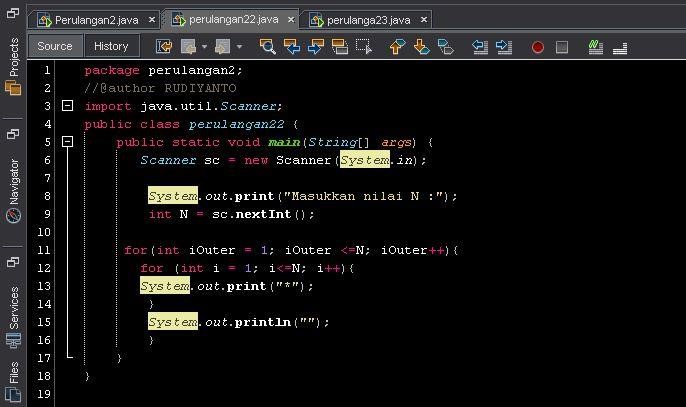
2.6 Perulangan 2

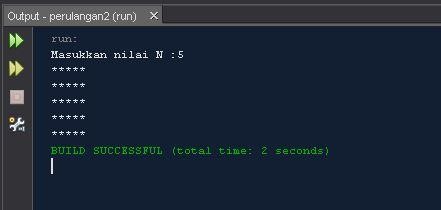
Pengertian Perulangan Bersarang (Nested Loop) Pada bahasan sebelumnya, telah dibahas tentang konsep dasar perulangan. Pada bahasan tersebut disebutkan bahwa logika perulangan digunakan untuk melakukan beberapa proses atau statement program secara berulang-ulang, dengan suatu pola tertentu. Pada perulangan, proses atau satetement akan terus dilakukan secara berulang-ulang, selama kondisi perulangan bernilai benar/true. Dan sebaliknya, perulangan akan berhenti dan proses atau statement tidak akan dieksekusi lagi ketika kondisi perulangan bernilai salah/false. Jadi, dalam logika perulangan, suatu kondisi perulangan diperlukan untuk menentukan apakah suatu perulangan masih akan akan berlangsung lagi atau harus berhenti. Perulangan bersarang (nested loop) adalah struktur perulangan yang berada di dalam perulangan lainnya. Pada umumnya, struktur perulangan yang berada di dalam perulangan lainnya tersebut memiliki hubungan yang saling terkait dalam menyesuaikan sebuah kasus. Pada dasarnya tidak ada batasan dalam jumlah perulangan bersarang. Tetapi sebaiknya tidak terlalu dalam, untuk menghindari kompleksitas yang tinggi serta alur program menjadi lebih sukar untuk dipahami.

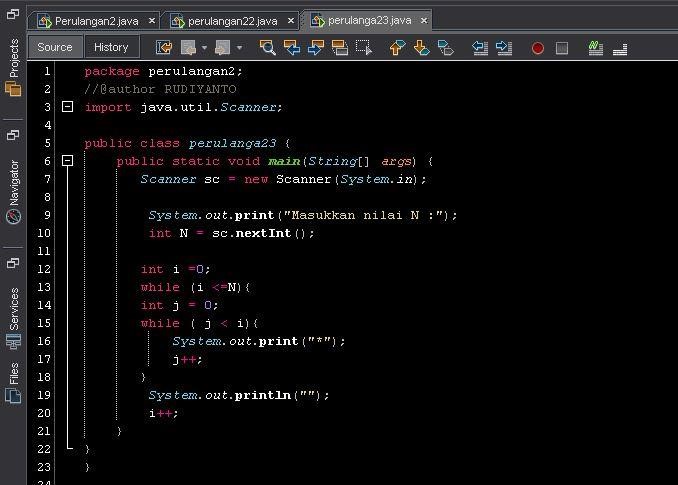
Berikut adalah kode program dan Hasil Runningnya :

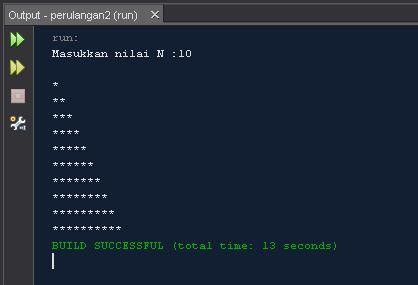










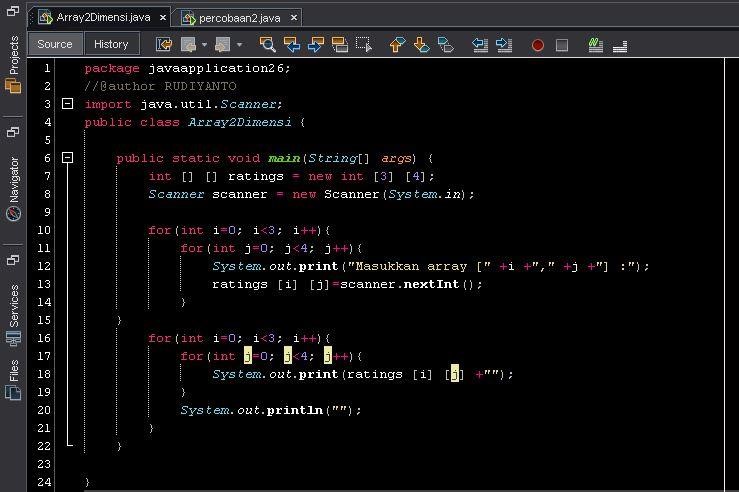
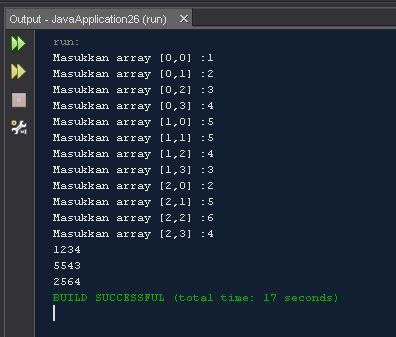


2.7 Array 2

Array yang telah kita pelajari sebelumnya adalah satu dimensi, hanya terdiri dari 1 baris elemen. Terkadang data perlu disajikan dalam bentuk tabel, misalkan spredsheet yang tentunya membutuhkan array 2 dimensi.

Ukuran Array 2

Dimensi Setiap array memiliki ukuran baik array 1D atau 2D, yaitu sudah tersedia attribute length. Contohnya adalah di bawah ini: – a.length menghasilkan 3, atau barisnya (dimensi yang pertama) – a[0].length menghasilkan 4, atau kolomnya (dimensi yang kedua) Ketika menggunakan attribute/variable length, keuntungannya adalah ketika ukuran array berubah kita tidak perlu mengubah kode untuk input/menampilkan array.

* Berikut ini adalah kode program dan hail Runningnya :
* 
* 
* 