**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**JOBSHEET 10 PERTEMUAN 13 DOULE LINKED LIST**

Dosen Pengampu : Triana Fatmawati, S.T., M.T.



Muhammad Afiq Firdaus

2341760189 / 21

SIB 1 E

**PRODI D-IV SISTEM INFORMASI BISNIS**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

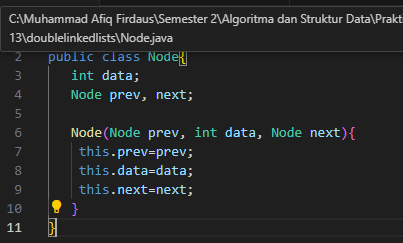
**2024**

**Praktikum 1**

Buat paket baru dengan nama doublelinkedlists

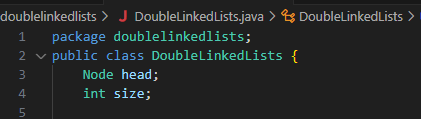
Buat class di dalam paket tersebut dengan nama Node

Serta isikan code sesuai jobsheet

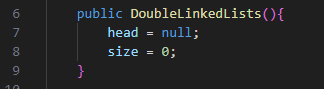


Buatlah sebuah class baru bernama DoubleLinkedLists pada package yang sama dengan node

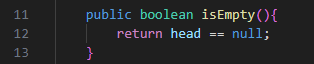
Pada class DoubleLinkedLists tersebut, deklarasikan atribut sesuai dengan diagram class di atas.



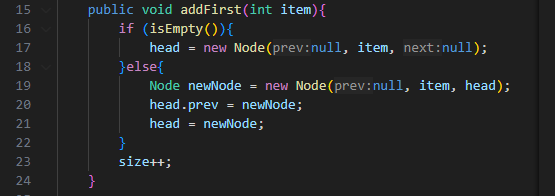
Selajuntnya, buat konstruktor pada class DoubleLinkedLists



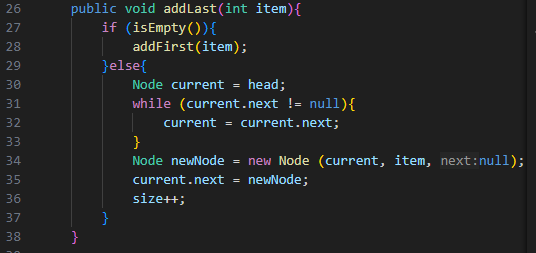
Buat method isEmpty(). Method ini digunakan untuk memastikan kondisi linked list kosong



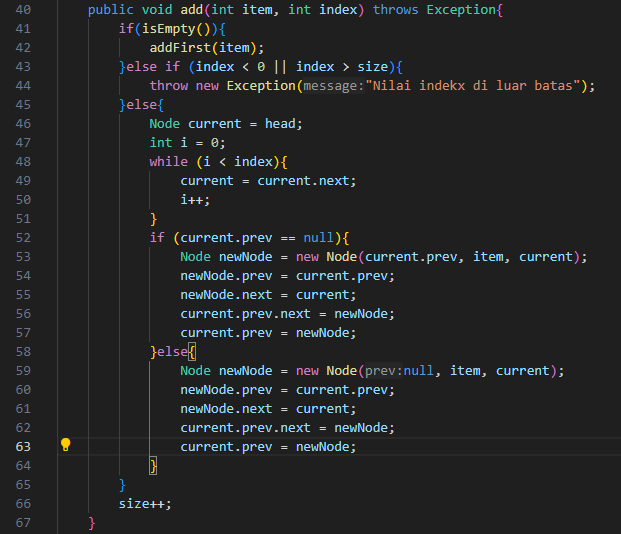
Kemudian, buat method addFirst(). Method ini akan menjalankan penambahan data di bagian depan linked list.



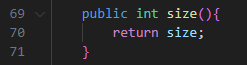
Selain itu pembuatan method addLast() akan menambahkan data pada bagian belakang linked list



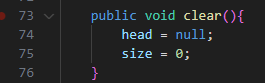
Untuk menambahkan data pada posisi yang telah ditentukan dengan indeks, dapat dibuat dengan method add(int item, int index)



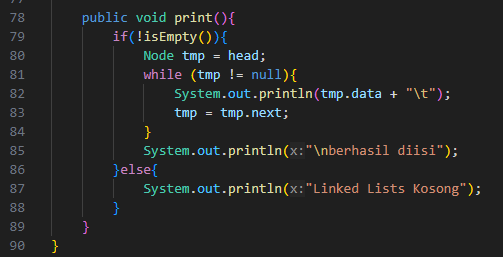
.Jumlah data yang ada di dalam linked lists akan diperbarui secara otomatis,sehingga dapat dibuat method size() untuk mendapatkan nilai dari size.



Selanjutnya dibuat method clear() untuk menghapus semua isi linked lists, sehingga linked lists dalam kondisi kosong.



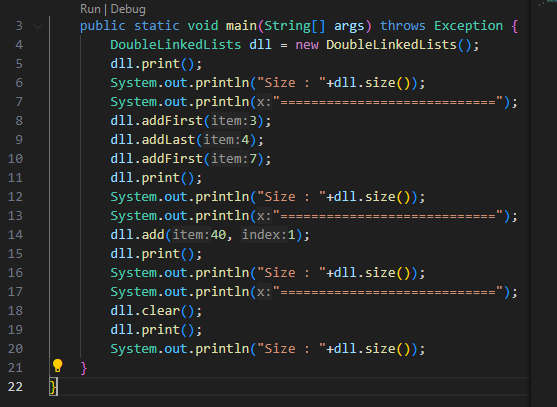
Untuk mencetak isi dari linked lists dibuat method print(). Method ini akan mencetak isi linked lists berapapun size-nya. Jika kosong akan dimunculkan suatu pemberitahuan bahwa linked lists dalam kondisi kosong.



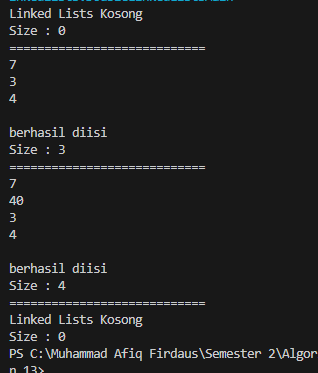
Selanjutya dibuat class Main DoubleLinkedListsMain untuk mengeksekusi semua method yang ada pada class DoubleLinkedLists.



Pada main class pada langkah 16 di atas buatlah object dari class DoubleLinkedLists kemudian eksekusi



**Verifikasi Hasil Percobaan**

****

**Pertanyaan Percobaan**

1. Jelaskan perbedaan antara single linked list dengan double linked lists!
2. Perhatikan class Node, di dalamnya terdapat atribut next dan prev. Untuk apakah atribut tersebut?
3. Perhatikan konstruktor pada class DoubleLinkedLists. Apa kegunaan inisialisasi atribut head dan size seperti pada gambar berikut ini?



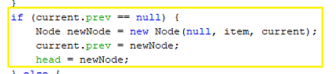
1. Pada method addFirst(), kenapa dalam pembuatan object dari konstruktor class Node prev dianggap sama dengan null?



1. Perhatikan pada method addFirst(). Apakah arti statement head.prev = newNode ?
2. Perhatikan isi method addLast(), apa arti dari pembuatan object Node dengan mengisikan parameter prev dengan current, dan next dengan null?



1. Pada method add(), terdapat potongan kode program sebagai berikut:



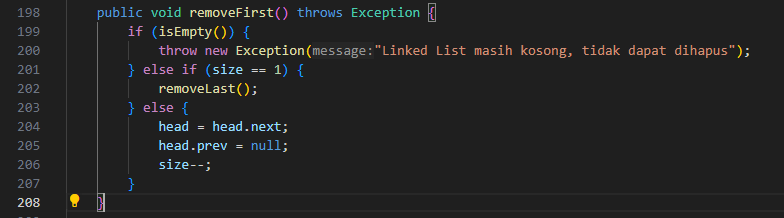
jelaskan maksud dari bagian yang ditandai dengan kotak kuning.

**Jawaban**

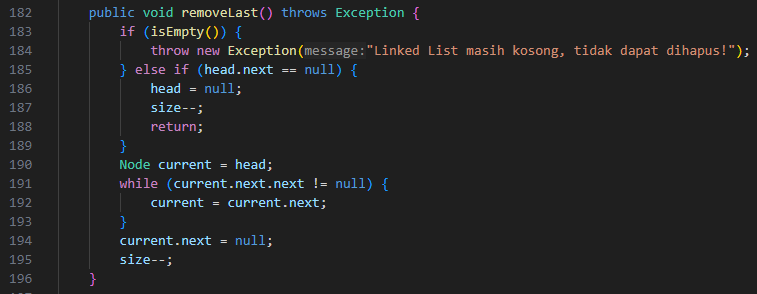
1. Perbedaan utama antara single linked list dan double linked list adalah bahwa single linked list hanya memiliki referensi ke node berikutnya, sedangkan double linked list memiliki referensi ke node berikutnya dan sebelumnya. Hal ini membuat double linked list lebih fleksibel dan efisien dalam beberapa operasi, tetapi dengan biaya penggunaan memori dan kompleksitas yang lebih tinggi.
2. Atribut next dan prev dalam kelas Node berfungsi untuk menghubungkan node satu sama lain dalam dua arah dalam struktur double linked list. next menghubungkan node saat ini ke node berikutnya, sementara prev menghubungkan node saat ini ke node sebelumnya.
3. Inisialisasi atribut head ke null dan size ke **0** dalam konstruktor kelas DoubleLinkedLists memastikan bahwa objek dimulai dalam keadaan yang konsisten dan siap untuk digunakan. Ini membantu menjaga integritas struktur data dan mempermudah pelaksanaan operasi-operasi pada double linked list.
4. Mengatur prev ke null saat membuat node baru dalam metode addFirst adalah langkah penting untuk menunjukkan bahwa node baru tersebut tidak memiliki node sebelumnya, karena node baru ini akan menjadi node pertama dalam double linked list. Ini memastikan bahwa struktur data tetap konsisten dan traversal dari head ke node lainnya berfungsi dengan benar.
5. Pernyataan head.prev = newNode menghubungkan node baru yang ditambahkan di awal daftar dengan node yang sudah ada sebelumnya. Ini memastikan bahwa node yang sebelumnya menjadi head sekarang memiliki referensi ke node baru sebagai node sebelumnya (prev). Ini merupakan langkah penting untuk memelihara integritas struktur double linked list saat menambahkan node baru di awal daftar.
6. Pernyataan Node newNode = new Node(current, item, null); dalam metode addLast() digunakan untuk membuat node baru yang akan ditambahkan di akhir double linked list. Parameter prev diisi dengan referensi ke node terakhir saat ini (current), dan next diisi dengan null untuk menunjukkan bahwa node baru adalah node terakhir dalam daftar.
7. Potongan kode dalam blok if pada metode add() digunakan untuk menyisipkan node baru ke dalam double linked list di posisi awal (indeks 0) dari daftar. Node baru dibuat dan diatur dengan benar, kemudian referensi prev dari node pertama diubah untuk menunjuk ke node baru, dan head diperbarui untuk menunjuk ke node baru (karena sekarang node baru adalah node pertama dalam daftar setelah penyisipan).

**Kegiatan Praktikum 2**

Buatlah method removeFirst() di dalam class DoubleLinkedLists.



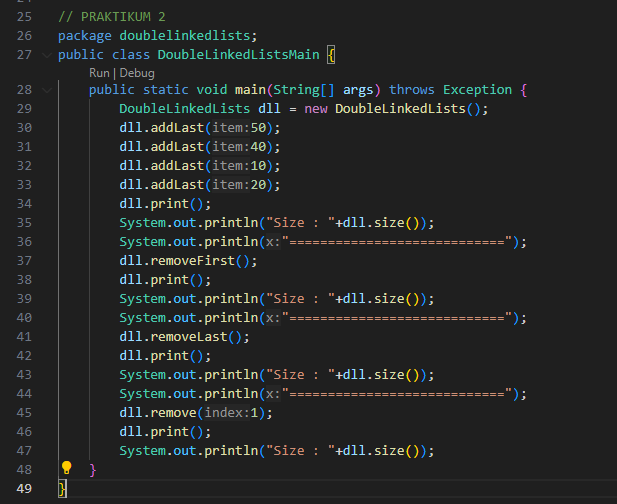
Tambahkan method removeLast() di dalam class DoubleLinkedLists.



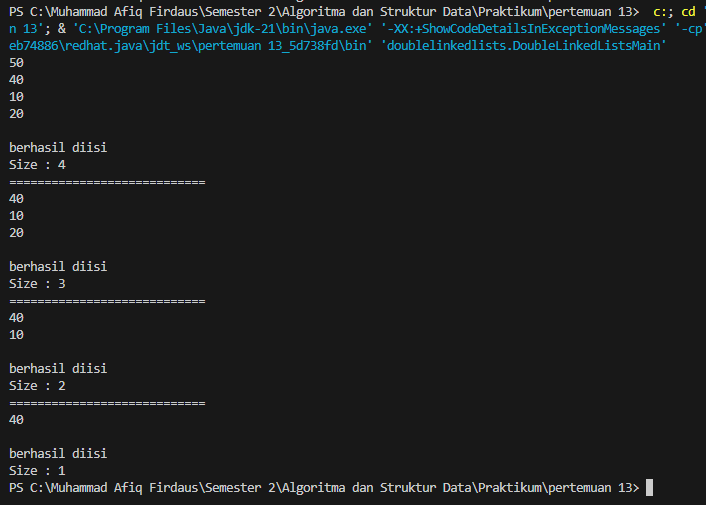
Tambahkan pula method remove(int index) pada class DoubleLinkedLists dan amati hasilnya.



Untuk mengeksekusi method yang baru saja dibuat, tambahkan potongan kode program berikut pada main class



**Verifikasi Hasil Percobaan**

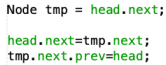
****

**Pertanyaan Percobaan**

1. Apakah maksud statement berikut pada method removeFirst()?

head = head.next; head.prev = null;

1. Bagaimana cara mendeteksi posisi data ada pada bagian akhir pada method removeLast()?
2. Jelaskan alasan potongan kode program di bawah ini tidak cocok untuk perintah remove!



1. Jelaskan fungsi kode program berikut ini pada fungsi remove!

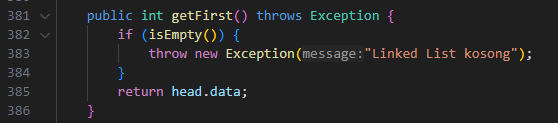


**Jawaban :**

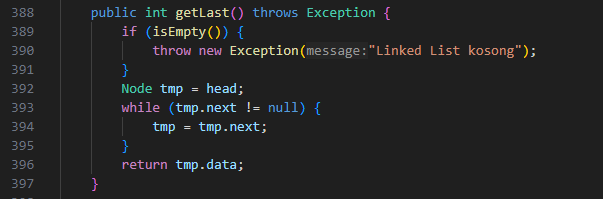
1. Pernyataan head = head.next; head.prev = null; dalam metode removeFirst() digunakan untuk menghapus node pertama (head) dari double linked list dengan menggeser head ke node berikutnya dalam daftar dan kemudian mengatur referensi prev dari node baru yang menjadi head menjadi null. Hal ini memastikan bahwa node baru tersebut menjadi node pertama dalam daftar dengan prev yang benar diatur ke null, dan daftar tetap konsisten setelah penghapusan node pertama.
2. memeriksa apakah daftar kosong sebelum memulai iterasi. Selanjutnya, iterasi menggunakan loop while untuk mencari node terakhir dalam daftar. Setelah menemukan node terakhir, operasi current.prev.next = null; menghapus node terakhir dengan menghapus referensi next-nya ke node sebelumnya. Terakhir, size-- digunakan untuk mengurangi ukuran daftar setelah penghapusan.
3. Potongan kode tersebut tidak benar-benar menghapus tmp dari daftar secara utuh. Meskipun node tmp (node kedua) tidak lagi memiliki referensi ke node ketiga, node tersebut masih memiliki referensi ke head (node pertama).
4. Potongan kode program current.prev.next = current.next; dan current.next.prev =current.prev; pada fungsi remove() digunakan untuk menghapus node current dari double linked list dengan memperbarui referensi prev dan next dari node tetangga sebelum dan sesudah current.

**Kegiatan Praktikum 3**

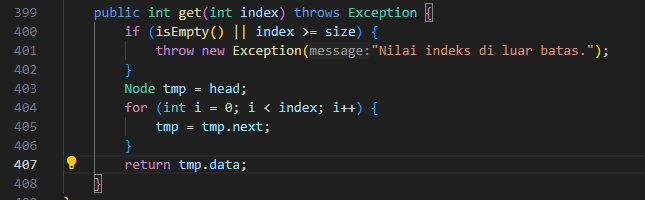
Buatlah method getFirst() di dalam class DoubleLinkedLists untuk mendapatkan data pada awal linked lists.



Selanjutnya, buatlah method getLast() untuk mendapat data pada akhir linked lists.



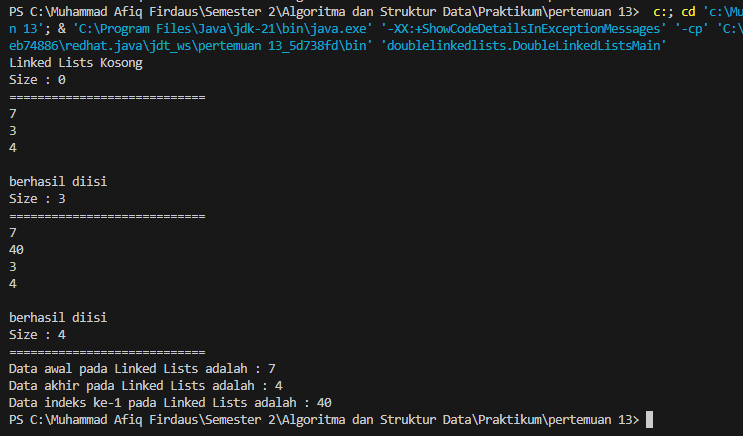
Method get(int index) dibuat untuk mendapatkan data pada indeks tertentu



Pada main class tambahkan potongan program berikut dan amati hasilnya!

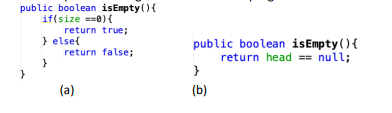


**Verifikasi Hasil Percobaan**

****

**Pertanyaan Percobaan**

1. Jelaskan method size() pada class DoubleLinkedLists!
2. Jelaskan cara mengatur indeks pada double linked lists supaya dapat dimulai dari indeks ke1!
3. Jelaskan perbedaan karakteristik fungsi Add pada Double Linked Lists dan Single Linked Lists!
4. Jelaskan perbedaan logika dari kedua kode program di bawah ini!



**Jawaban :**

1. Metode size() pada class DoubleLinkedLists bertujuan untuk mengembalikan jumlah elemen (node) yang ada dalam double linked list. Ini adalah nilai dari atribut size yang di-maintain dan diperbarui setiap kali ada penambahan atau penghapusan elemen dalam daftar.
2. **Tentukan Konvensi Indeks:**

membuat konvensi bahwa indeks pertama (1) mengacu pada elemen pertama dalam double linked list. Ini berbeda dengan indeks standar dalam bahasa pemrograman **seperti Java yang dimulai dari 0.**

Modifikasi Metode Penambahan dan Penghapusan:

perlu memodifikasi metode add(int item, int index) dan remove(int index) sehingga mereka menerima indeks yang dimulai dari 1 sebagai input, bukan indeks yang dimulai dari 0.

1. **Single Linked List:**

* Lebih sederhana dan menggunakan memori lebih sedikit.
* Cocok untuk aplikasi di mana operasi traversal hanya satu arah diperlukan.

**Double Linked List:**

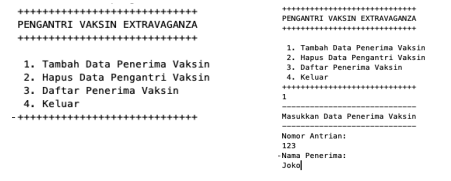
* Lebih kompleks dan menggunakan lebih banyak memori.
* Mendukung traversal dua arah yang lebih fleksibel dan efisien dalam beberapa operasi seperti penghapusan node tertentu.

1. Pemilihan metode tergantung pada konteks dan kebutuhan spesifik implementasi linked list Anda.

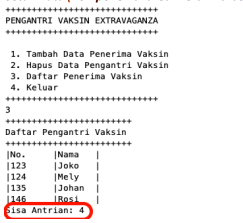
* Jika Anda sudah memelihara ukuran list dengan variabel size dan memastikan size selalu akurat, metode (a) bisa digunakan.
* Jika Anda tidak ingin menambah kompleksitas dengan variabel tambahan atau lebih suka logika yang lebih sederhana, metode (b) adalah pilihan yang baik.

**Tugas Praktikum**

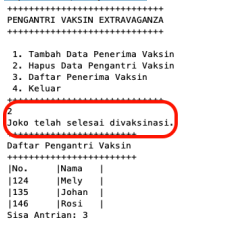
Buat program antrian vaksinasi menggunakan queue berbasis double linked list sesuai ilustrasi dan menu di bawah ini! (counter jumlah antrian tersisa di menu cetak(3) dan data orang yang telah divaksinasi di menu Hapus Data(2) harus ada) Ilustrasi Program Menu Awal dan Penambahan Data

****

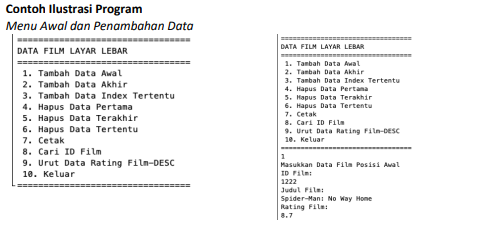
Cetak Data (Komponen di area merah harus ada)

****

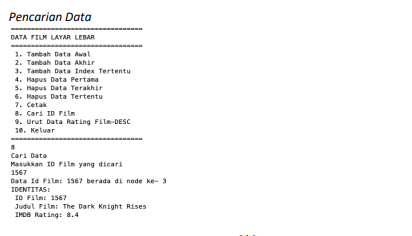
Hapus Data (Komponen di area merah harus ada)

****

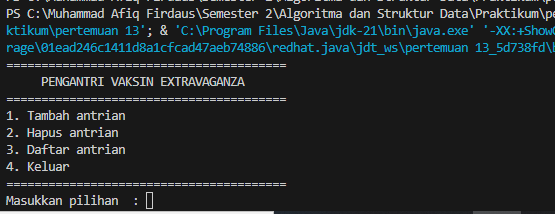
2. Buatlah program daftar film yang terdiri dari id, judul dan rating menggunakan double linked lists, bentuk program memiliki fitur pencarian melalui ID Film dan pengurutan Rating secara descending. Class Film wajib diimplementasikan dalam soal ini.

****

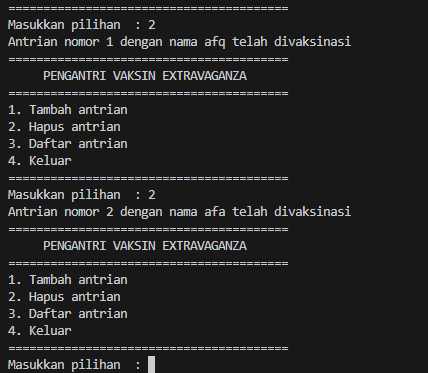
****

****

**Jawaban**

* + - 1. 





* + - 1. 