**Laporan Praktikum ASD**

**Pertemuan 12**

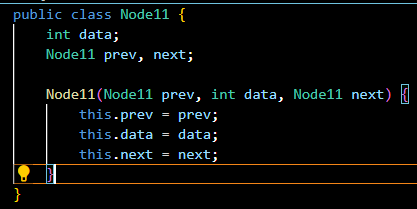
**Nama: Gegas Anugrah Derajat**

**Kelas: SIB-1F**

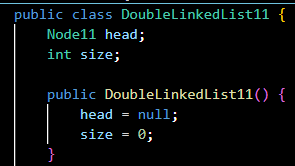
**NIM: 2341760140**

**Praktikum 1**

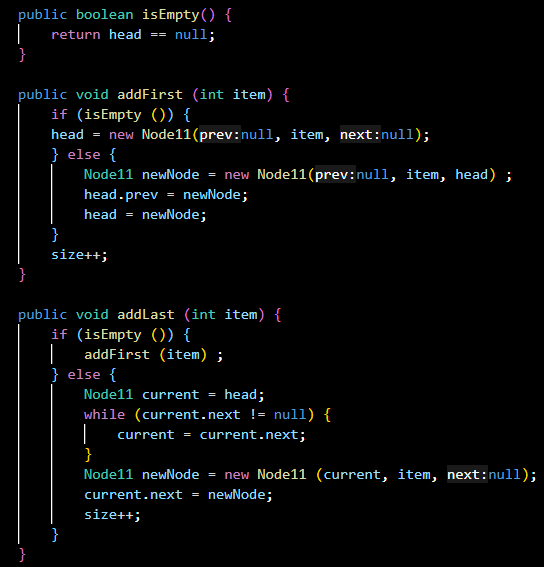
**Membuat class Node, mendeklarasikan atribut dan menambahkan konstruktor**

****

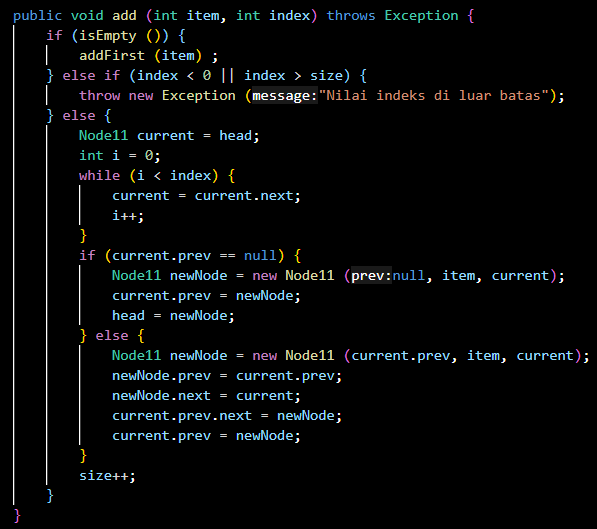
**Membuat class DoubleLinkedList**

****

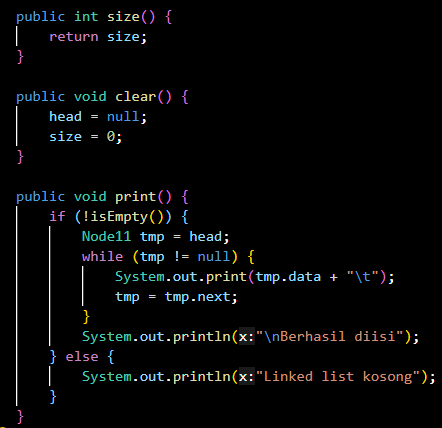
**Membuat method isEmpty, addFirst dan addLast**

****

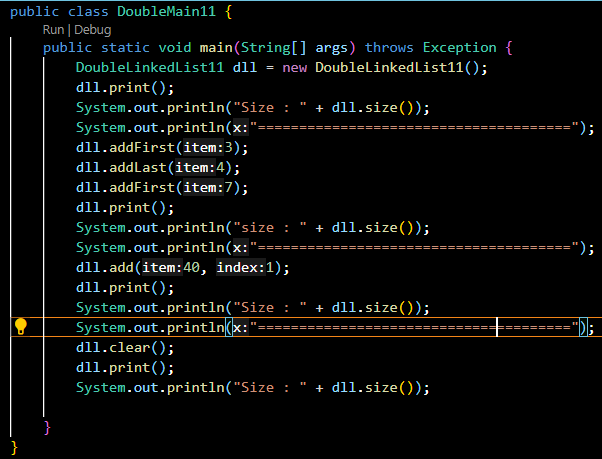
**Membuat method add, untuk menambahkan nilai sesuai dengan indeks yang diberikan**

****

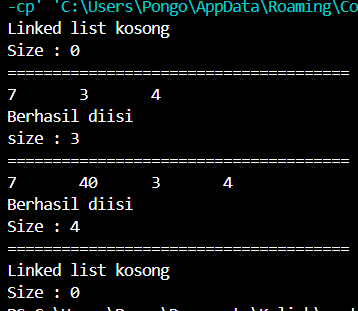
**Membuat method size, clear dan print**

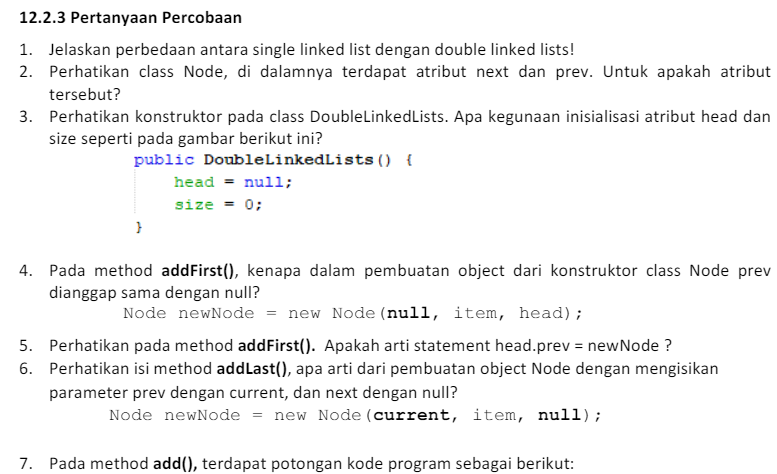
****

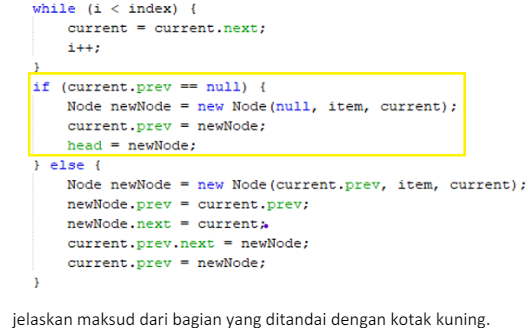
**Membuat class Main dan mengeksekusi semua method**

****

**Hasil**

****

****

****

**Jawaban:**

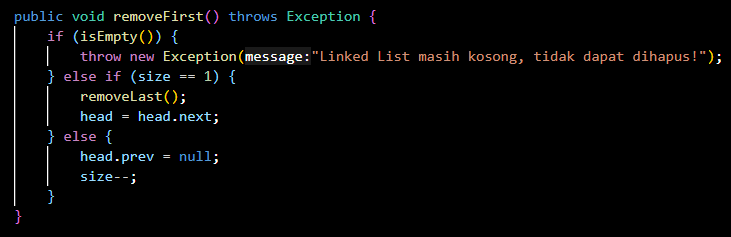
1. **Single linked list hanya memiliki satu arah yaitu node awal dan akhir, dan hanya memiliki satu node yaitu next.**

**Double linked list memiliki dua arah dan memiliki dua node yaitu prev dan next.**

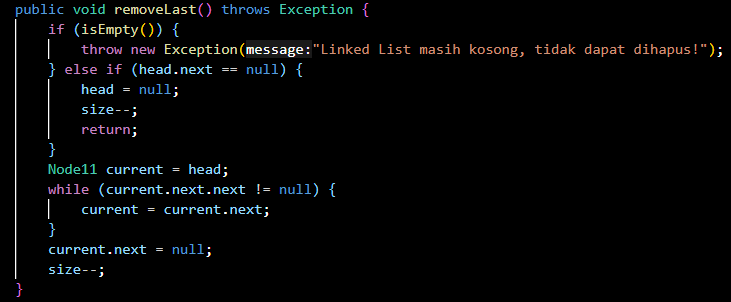
1. **atribut prev digunakan untuk menunjuk node sebelumnya, sedangkan next digunakan untuk menunjuk node berikutnya.**
2. **Head menunjuk node pertama dalam daftar, size menyimpan jumlah node dalam daftar.**
3. **AddFirst befungsi untuk menambahkan data pada saat pertama kali. Karena isi daftar node pada awal berisikan 0, tidak ada data. Jadi tidak ada yang bisa untuk dilakukan prev.**
4. **Mengatur node yang sebelumnya menjadi head untuk menunjuk ke node baru yang kita tambahkan newNode.**
5. **AddLast befungsi menambahkan data pada urutan terakhir. Karena akan di tempatkan pada posisi terakhir.**
6. **Mendeteksi apakah posisi sebelumnya null, jika ya maka posisi sebelumnya akan di isi oleh data baru, dan head akan ada pada posisi data yang baru saja ditambahkan**

**Praktikum 2**

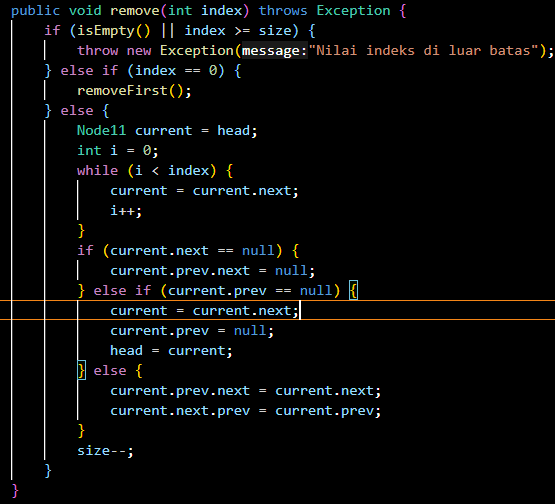
**Membuat method removeFirst pada class DoubleLinkedList**

****

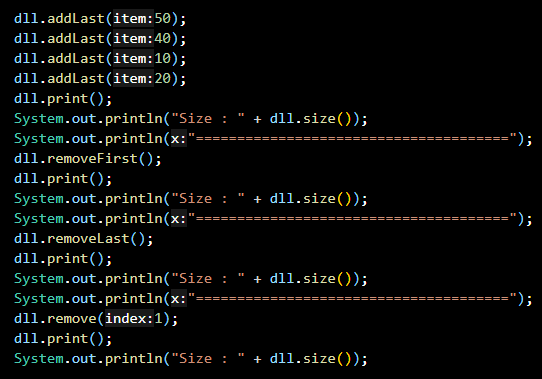
**Membuat method removeLast**

****

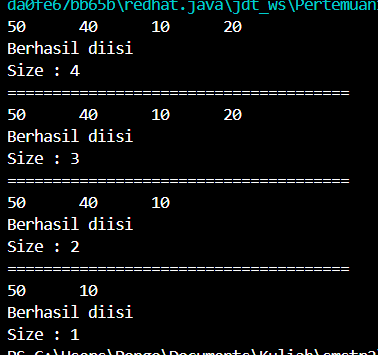
**Membuat method remove, untuk menghapus nilai pada indeks tertentu**

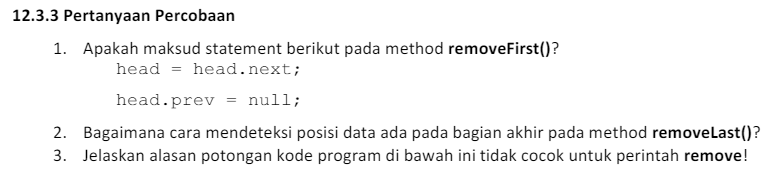
****

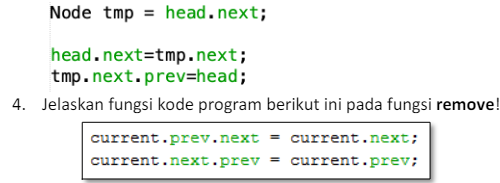
**Pada class Main menambahkan kode untuk eksekusi method yang baru dibuat**

****

**Hasil**

****

****

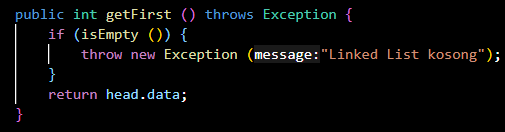
****

**Jawaban:**

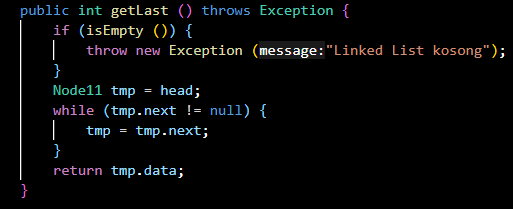
1. **head=head.next berfungsi untuk memperbarui pointer head yang menunjuk ke node berikutnya setelah node pertama yang dihapus. head.prev=null berfungsi untuk menghapus ponter prev karena head berasa pada daftar pertama.**
2. **Dengan menggunakan iterasi loop while, mencari current.next hingga ke daftar yang terakhir. Karena daftar terakhir tidak bisa melakukan .next, jadi data menunjukan data terakhir, dan ponter akan di simpan di variable current.**
3. **Karena kode program di atas tidak memeriksa kondisi spesifik dari daftar.**
4. **Kode program pertama berfungsi memperbarui pointer next dari node sebelumnya untuk menunjuk ke node berikutnya Kode program kedua berfungsi memperbarui ponter prev dari node berikutnya untuk menunjuk ke node sebelumnya.**

**Praktikum 3**

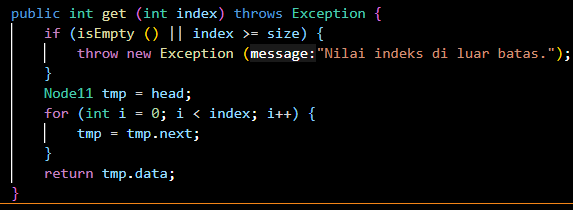
**Membuat method getFirst pada class DoubleLinkedList**

****

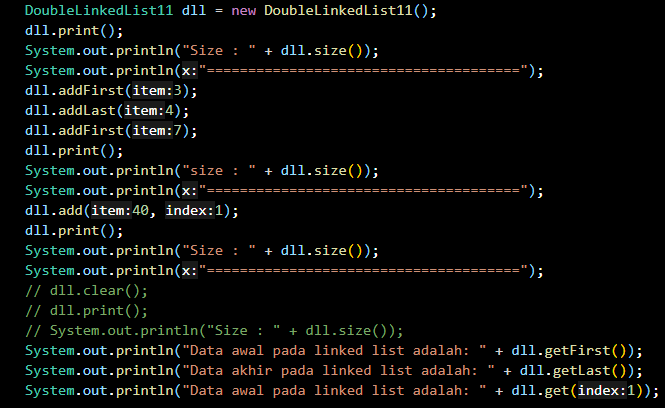
**Membuat method getLast**

****

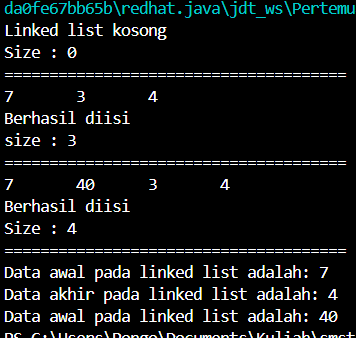
**Membuat method get**

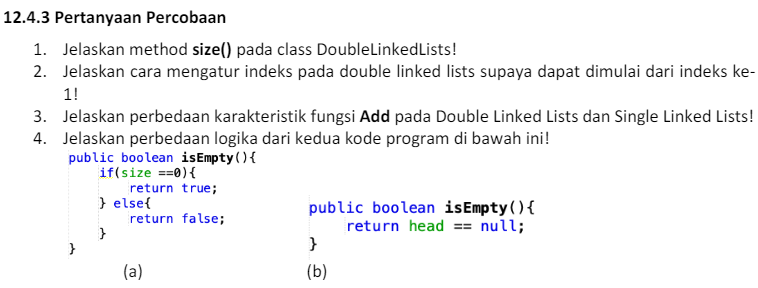
****

**Menambahkan kode pada class main untuk mengeksekusi method yang baru dibuat**

****

**hasil**

****

****

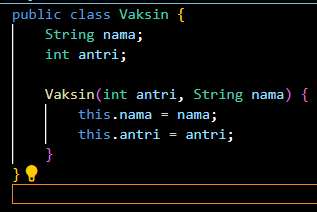
**Jawaban:**

1. **Size() mengembalikan nilai size yang telah di atur pada awal menjalankan program.**
2. **Secara default index umumnya dimulai dari 0. Namun dengan dengan sedikit mengubah pada method add dengan memberikan validasi untuk index seperti dibawah ini.**
3. **Penambahan pada single linked hanya satu arah dari depan ke belakang, tidak bisa menambahkan elemen di tengah daftar Penambahan pada double linked memiliki 2 arah, bisa dari depan dan belakang, bisa menyisipkan kode di mana saja.**
4. **pada kode A akan memeriksa jumlah elemen, jika jumlah elemen tersebut bernilai 0 maka akan mengembalikan true, namun jika tidak sama dengan 0 maka akan mengembalikan false Pada kode B akan memeriksa ponter head apakah berisikan null, jika iya maka akan mengembalikan true, jika tidak null maka kan mengembalikan false.**

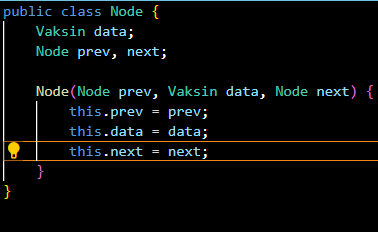
**Tugas**

1. 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)

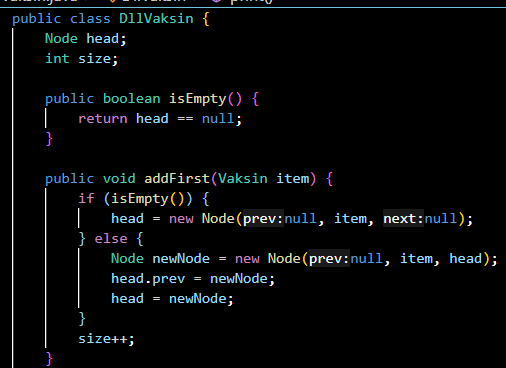
**Membuat class vaksin**

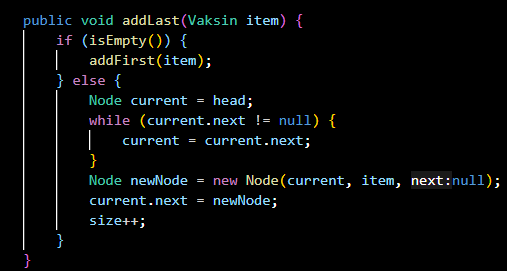
****

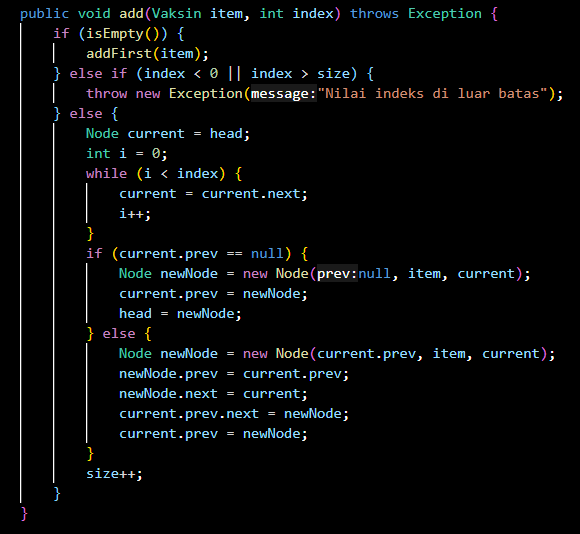
**Membuat Class Node**

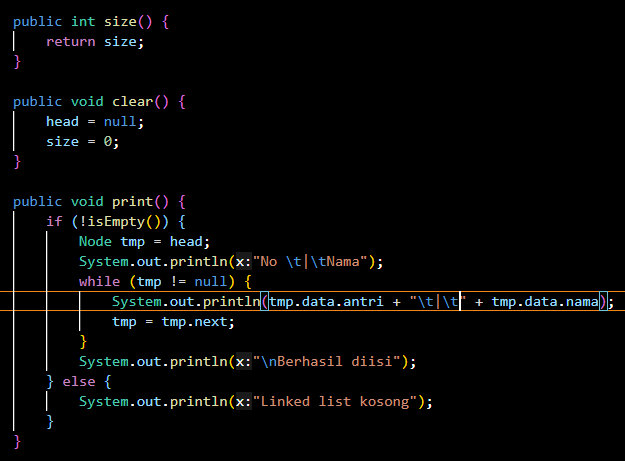
****

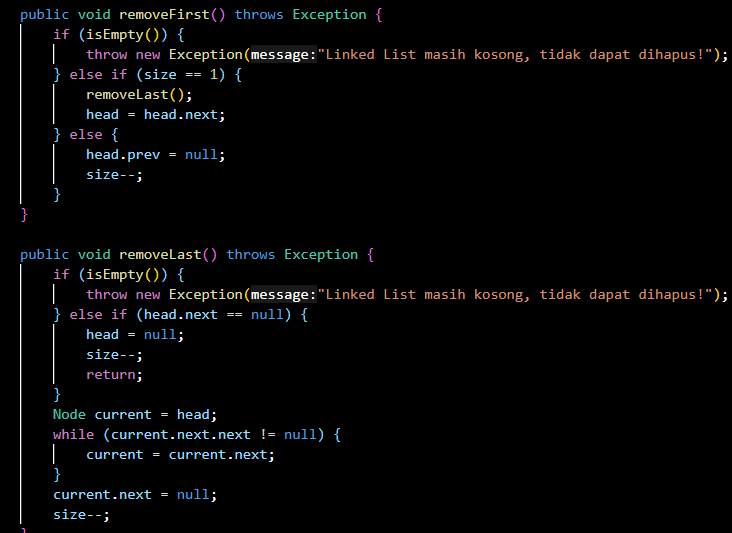
**Membuat class DllVakasin dan menambahkan methodnya**

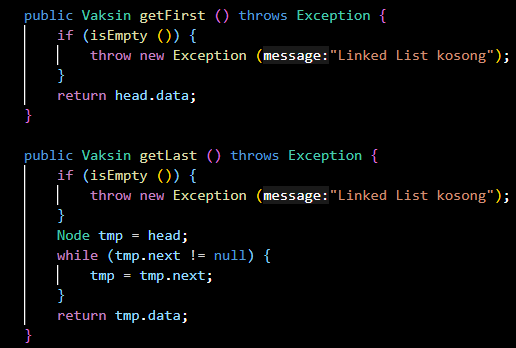
****

****

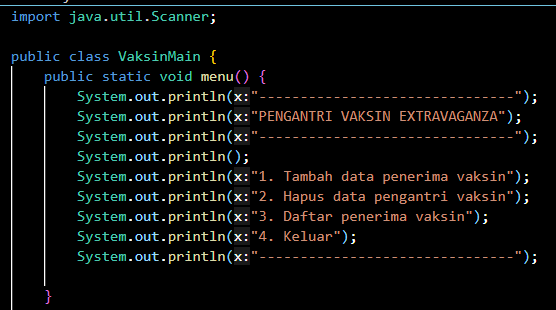
****

****

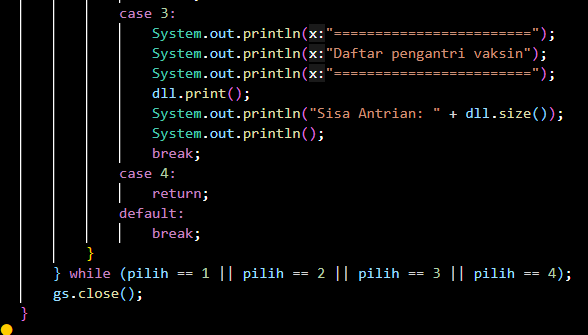
****

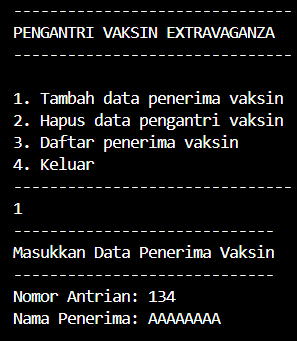
****

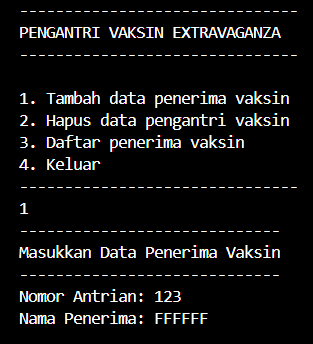
**Membuat class VaksiMain dan method menu serta mengeksekusi method yang telah dibuat**

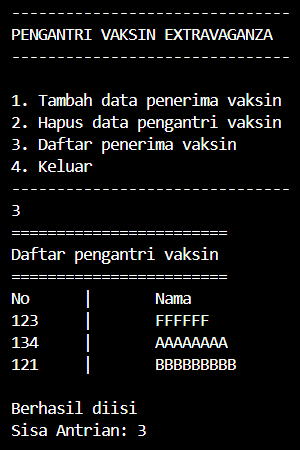
****

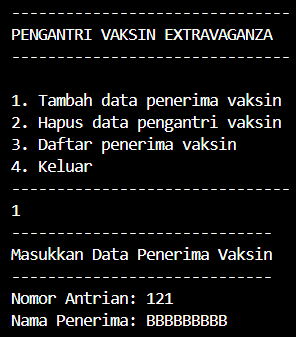
****

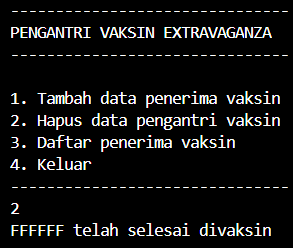
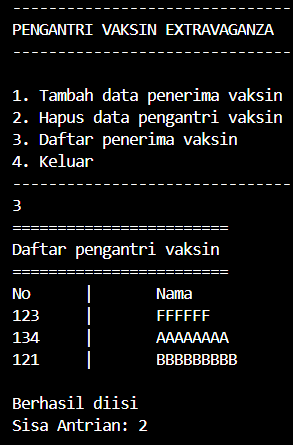
****

**Hasil**

****

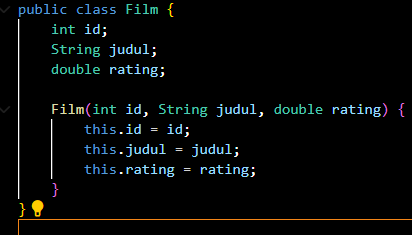
****

****

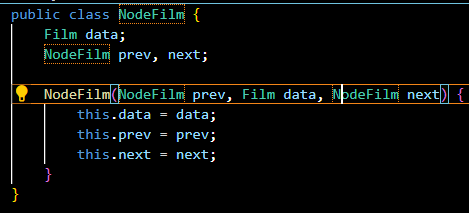
****

1. 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.

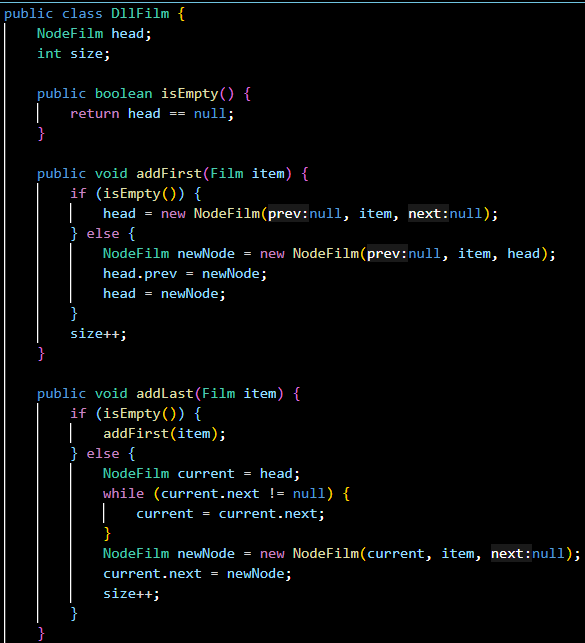
**Membuat class Film**

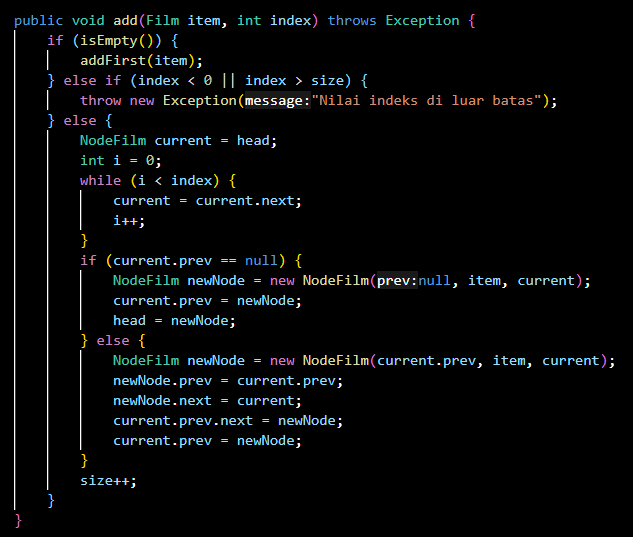
****

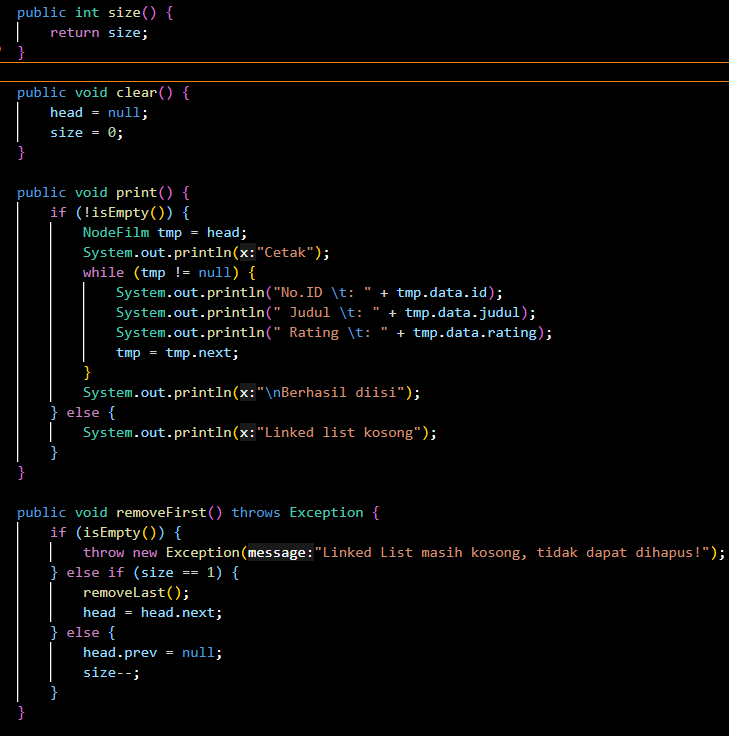
**Membuat class NodeFilm**

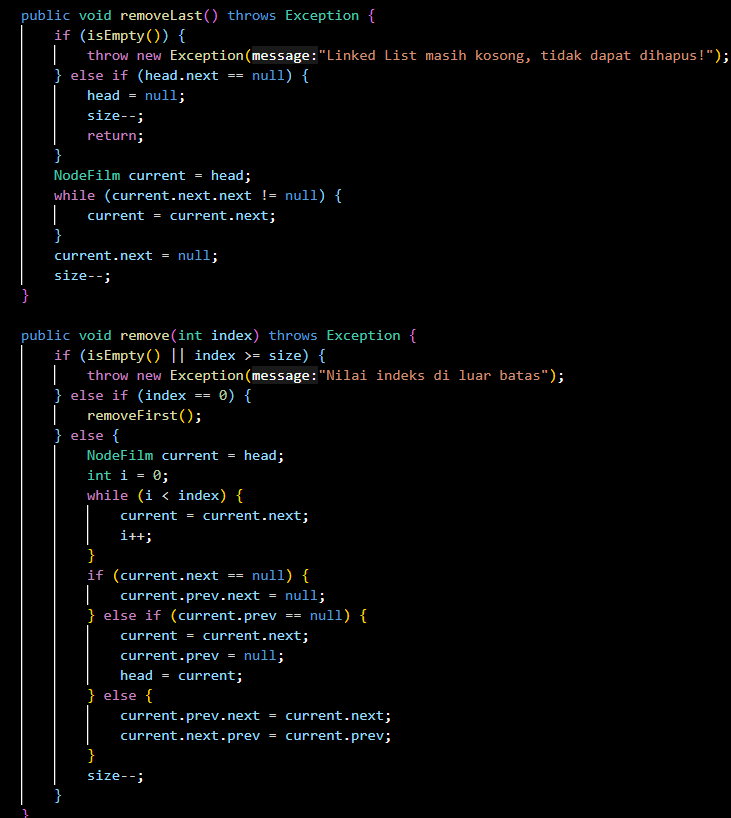
****

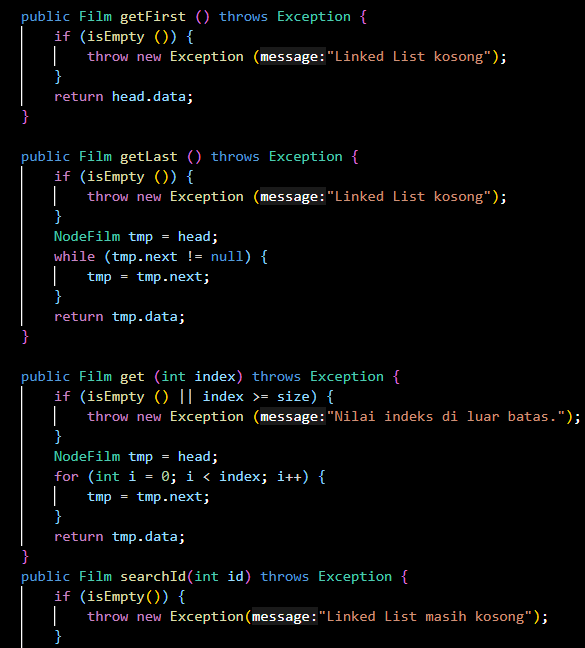
**Membuat class DllFilm**

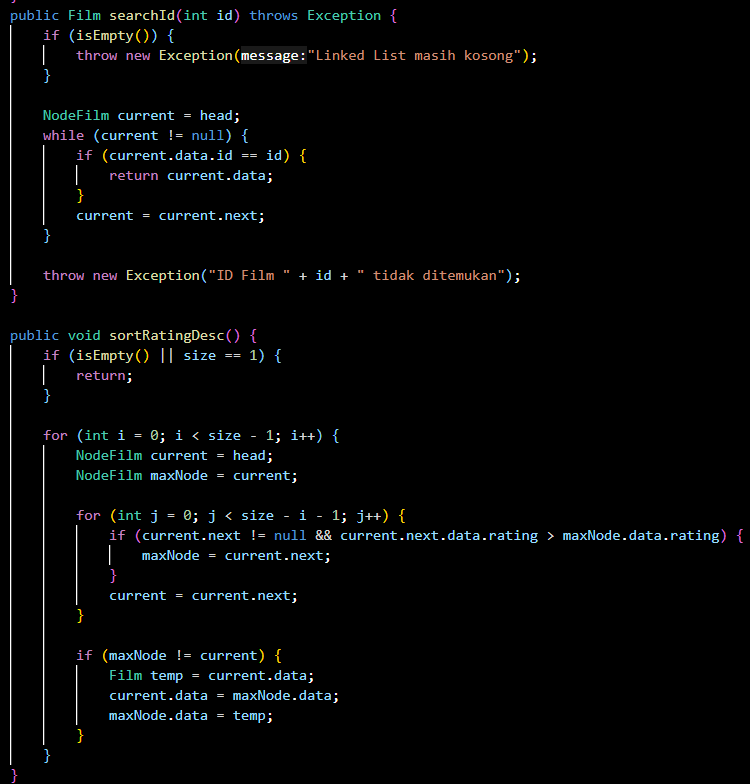
****

****

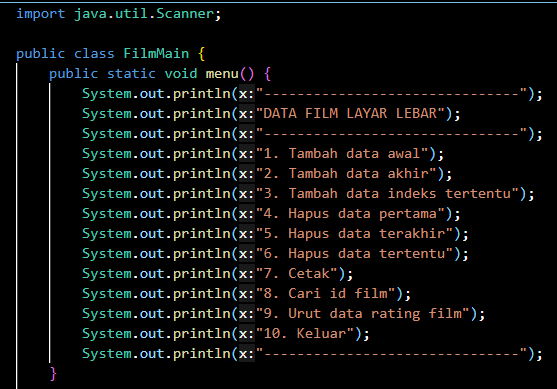
****

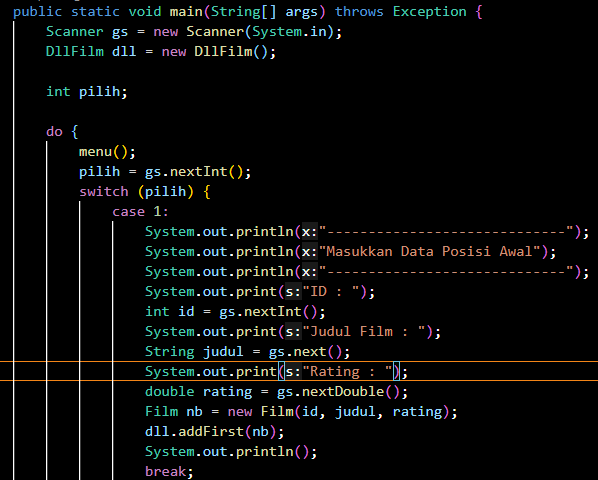
****

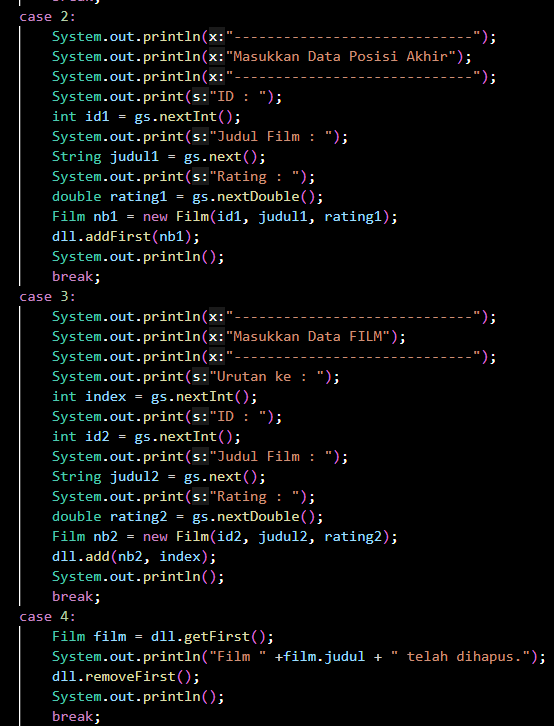
****

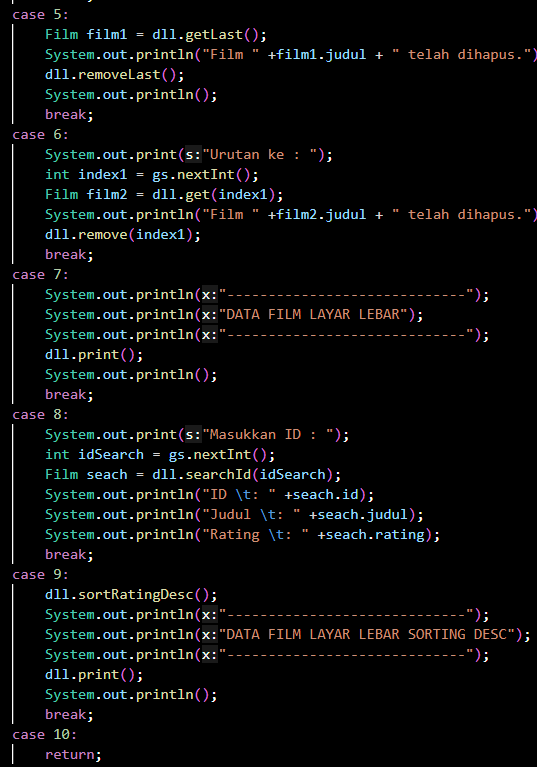
****

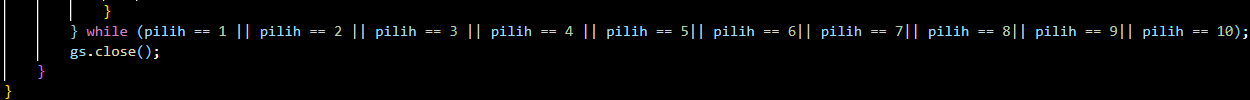
**Membuat class Main**

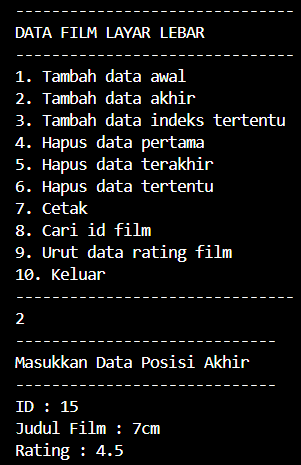
****

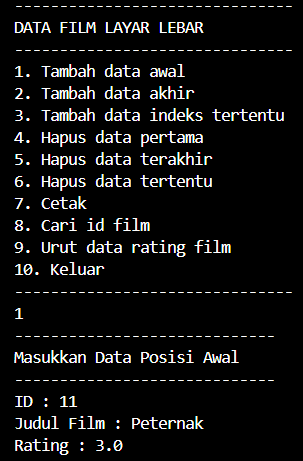
****

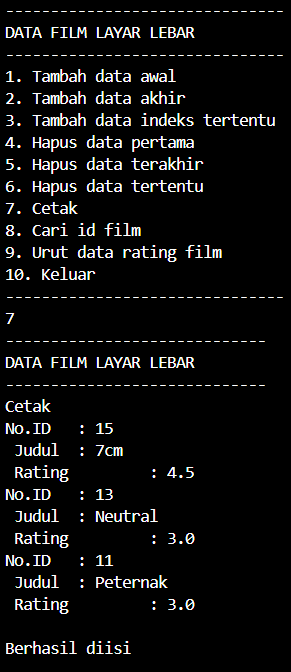
****

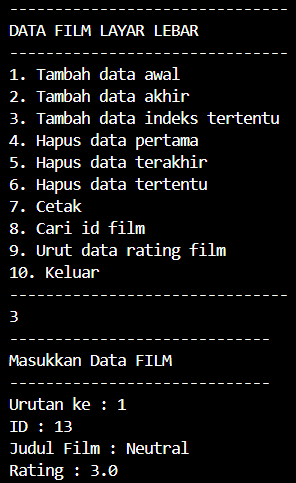
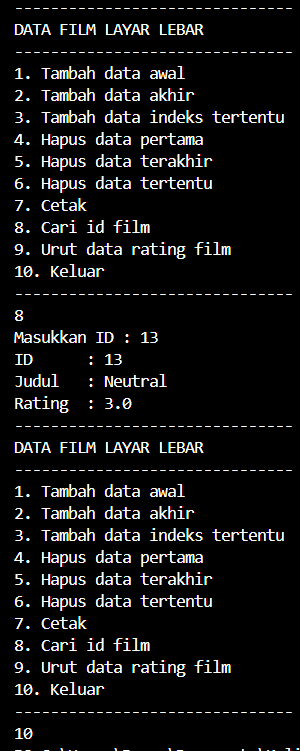
****

****

**Hasil**

****

****

****