**LAPORAN PRAKTIKUM STRUKTUR DATA**

****

**Oleh :**

**DEVIA PUJI ASTUTI**

**2311531005**

**Dosen Pengampu :**

**Dr. WAHYUDI, MT.**

**DEPARTEMEN INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ANDALAS**

**PADANG**

1. **TUJUAN**
2. Memahami cara penggunaan Linked List.
3. Memahami cara kerja Linked List pada data
4. Membuat program menggunakan Linked List.
5. **KAJIAN TEORI**

Linked list adalah salah satu struktur data linear yang digunakan untuk menyimpan data. Berbeda dengan array yang juga merupakan struktur data linear tapi bersifat statis, linked list bersifat dinamis. Artinya, ukuran linked list bisa bertambah atau berkurang seiring berjalannya program.

Linked list terdiri dari elemen-elemen yang disebut sebagai **simpul** (node). Setiap simpul memiliki dua bagian penting:

1. Data: Bagian ini berisi nilai aktual yang disimpan, misalnya angka, huruf, atau objek lain.
2. Penunjuk (pointer): Bagian ini menyimpan alamat memori simpul berikutnya di dalam linked list. Dengan penunjuk inilah terbentuk rangkaian antar simpul, sehingga urutan data bisa terjaga.

Analogi Kereta Api: Bayangkan linked list seperti rangkaian kereta api. Setiap gerbong kereta bisa dianalogikan sebagai simpul. Gerbong kereta memiliki tempat untuk memuat barang (data), dan di antara gerbong terdapat penghubung (penunjuk) yang memungkinkan kereta berjalan sebagai rangkaian.

**Kelebihan Linked List:**

* **Fleksibel:** Mudah untuk menambah atau menghapus data di tengah linked list tanpa perlu menggeser seluruh data yang lain, seperti yang terjadi pada array.
* **Memori efisien:** Tidak perlu mengalokasikan memori secara penuh di awal program, hanya dialokasikan sesuai kebutuhan saat data ditambahkan.

**Kekurangan Linked List:**

* **Akses data kurang efisien:** Untuk mengakses data tertentu, program harus menelusuri linked list mulai dari simpul awal sampai ke simpul yang diinginkan. Hal ini berbeda dengan array yang bisa diakses langsung berdasarkan indeks.
* **Membutuhkan memori tambahan:** Setiap simpul membutuhkan memori untuk menyimpan penunjuk ke simpul berikutnya.

**Jenis-jenis Linked List:**

* **Single Linked List:** Setiap simpul hanya memiliki satu penunjuk ke simpul berikutnya.
* **Doubly Linked List:** Setiap simpul memiliki dua penunjuk, yaitu ke simpul berikutnya dan ke simpul sebelumnya.

Secara ringkas Linked list adalah struktur data yang berguna untuk menyimpan data secara dinamis. Meskipun memiliki kelemahan dalam hal akses data, linked list tetap menjadi pilihan yang baik untuk program yang membutuhkan kemudahan dalam menambah dan menghapus data.

Dalam Java, konsep **Node** pada **Linked List** disebut sebagai class. Class ini berisikan representasi dari sebuah simpul (node) di dalam linked list. Umumnya, class Node memiliki atribut sebagai berikut:

1. **Data:** Variabel untuk menyimpan nilai aktual yang ada di dalam simpul. Tipe data variabel ini bisa bermacam-macam, tergantung kebutuhan program Anda. Misalnya, bisa berupa int, String, atau objek dari class lain.
2. **Next (atau pointer):** Variabel bertipe referensi (reference) yang menunjuk ke simpul berikutnya di dalam linked list. Dengan kata lain, variabel ini menyimpan alamat memori dari simpul selanjutnya.

Penjelasan code:

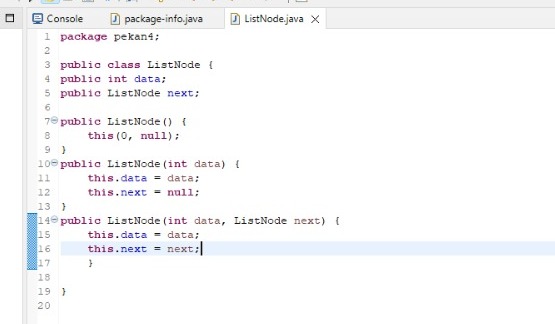
* Class Node dibuat bersifat generic (<T>) sehingga bisa menyimpan berbagai tipe data pada variabel data.
* Variabel data digunakan untuk menyimpan nilai aktual.
* Variabel next bertipe Node<T> yang merupakan referensi ke simpul berikutnya.
* Constructor Node(T data) berfungsi untuk membuat objek Node baru dengan nilai data tertentu.

**Catatan:**

* Implementasi class Node bisa berbeda tergantung pada jenis Linked List yang digunakan (Single Linked List atau Doubly Linked List). Pada Doubly Linked List, class Node biasanya memiliki tambahan variabel penunjuk ke simpul sebelumnya.
* Class Node biasanya digunakan secara internal oleh class LinkedList yang disediakan oleh Java. Anda tidak perlu secara langsung membuat objek Node untuk menggunakan Linked List.

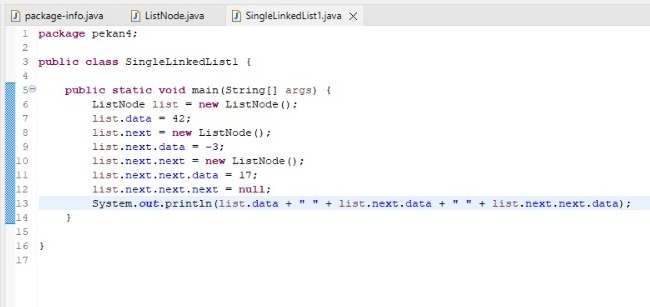
1. **LANGKAH PENGERJAAN** 
   1. List Node

Pada List Node kita buat codingan dalam bahasa java dengan nama class ”ListNode” lalu buat seperti dibawah ini.

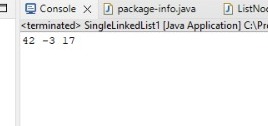


* 1. Single Linked List

Buat sebuah class dengan nama ”SingleLinkedList1” lalu mulai buat kodingan seperti dibawah ini dan pastikan tidak ada yang mengalami error.

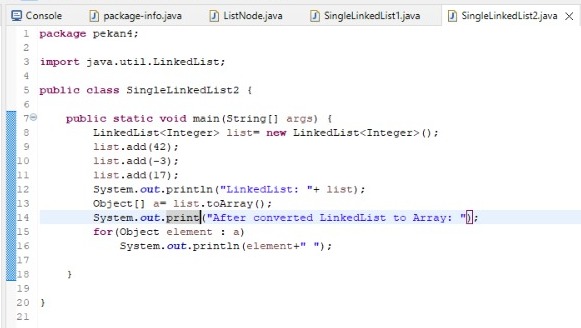


Bila tidak terjadi error maka output yang dihasilkan akan sama seperti dibawah ini.

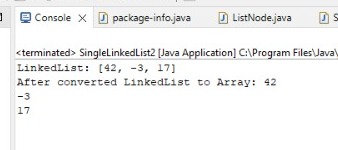


* 1. Single Linked List

Sama seperti sebelumnya kita menggunakan Single Linked List pada codingan ini, kita beri nama classnya dengan ”SingleLinkedList2”. Buat codingan seperti dibawah agar tidak terjadi error.

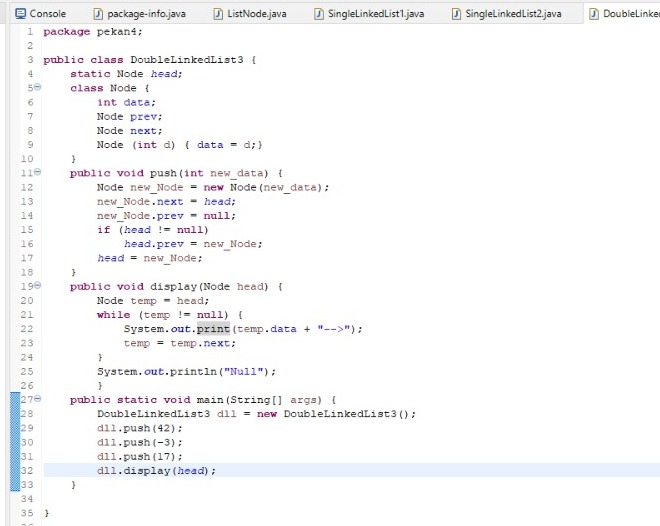


Bila tidak terjadi error maka output yang dihasilkan akan sama seperti dibawah ini.



* 1. Double Linked List

Sekarang kita gunakan jenis yang ke dua dari Linked List,yaitu Double LInked List. Beri nama class pada codingan berupa ”DoubleLinkedList” dan mulai program dengan mengikuti codingan dibawah agar tidak terjadi error.



Bila tidak terjadi error maka output yang dihasilkan akan sama seperti dibawah ini.

