**LAPORAN PRAKTIKUM**

**STRUKTUR DATA LINIER**

**Project Logistic**

Dosen Pengampu

JB. Budi Darmawan S.T., M.Sc.

****

DISUSUN OLEH

**ALDIAN OCTAVIUS SOESANTO** (225314023) 45%

**EMMANUEL KRISHNANDITO LAKSANA** (225314024) 10%

**FORTUNA RATNA DEWI** (225314025) 30%

**RAMA SANTA PRANATA** (225314026) 15%

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS SANATA DHARMA**

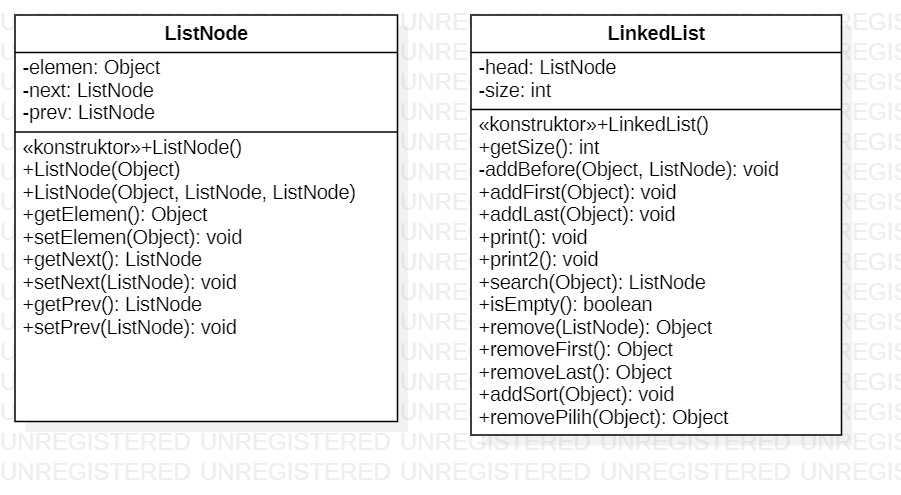
**YOGYAKARTA**

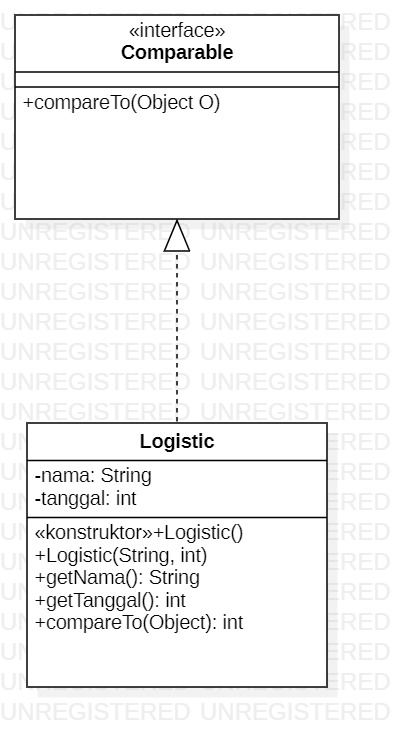
**2022**

1. **TUJUAN PRAKTIKUM**

Untuk mengurutkan barang Logistic berdasarkan tanggal kadaluarsa dalam sebulan

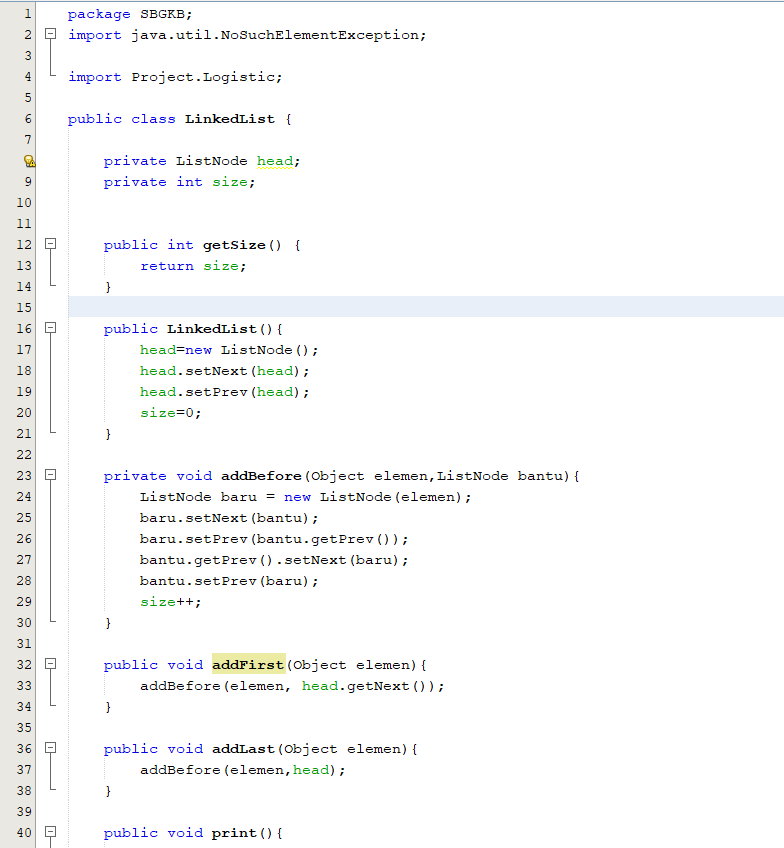
1. **DIAGRAM UML**



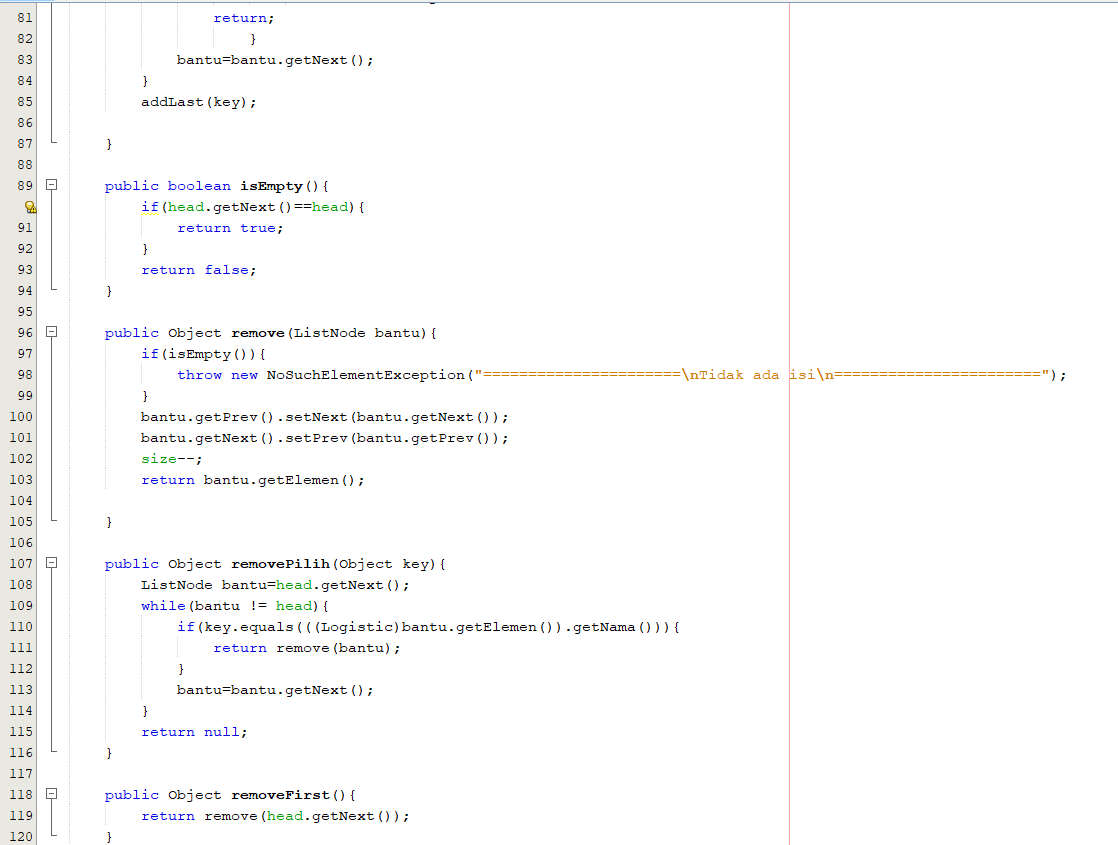


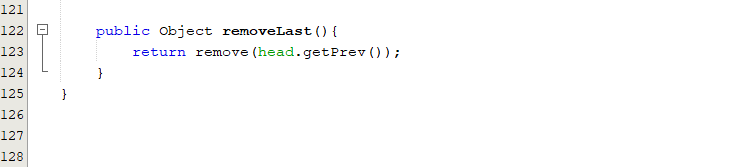
1. **SOURCE CODE**

Class LinkedList

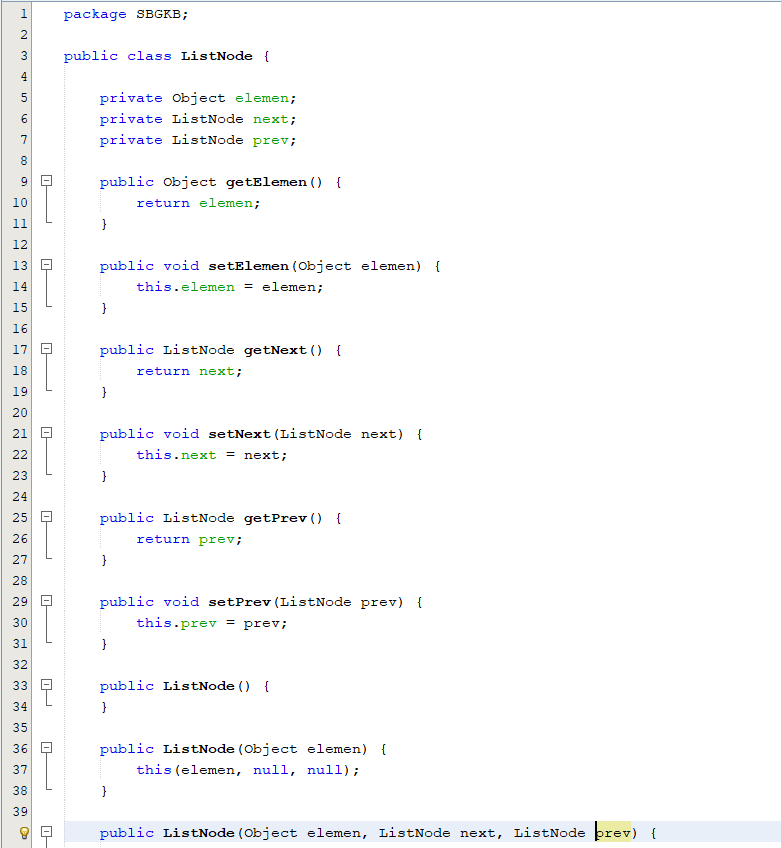


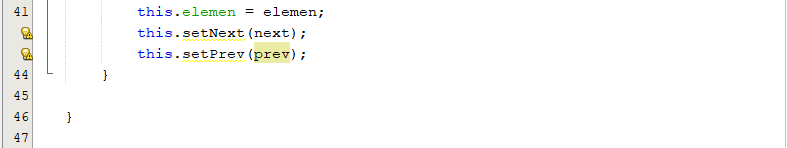
****

****

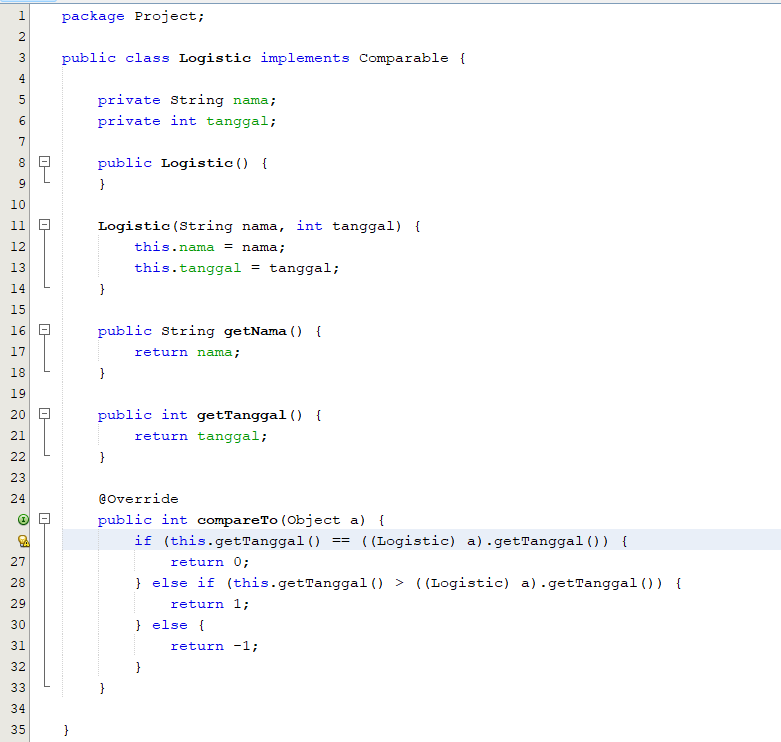
****

Class ListNode



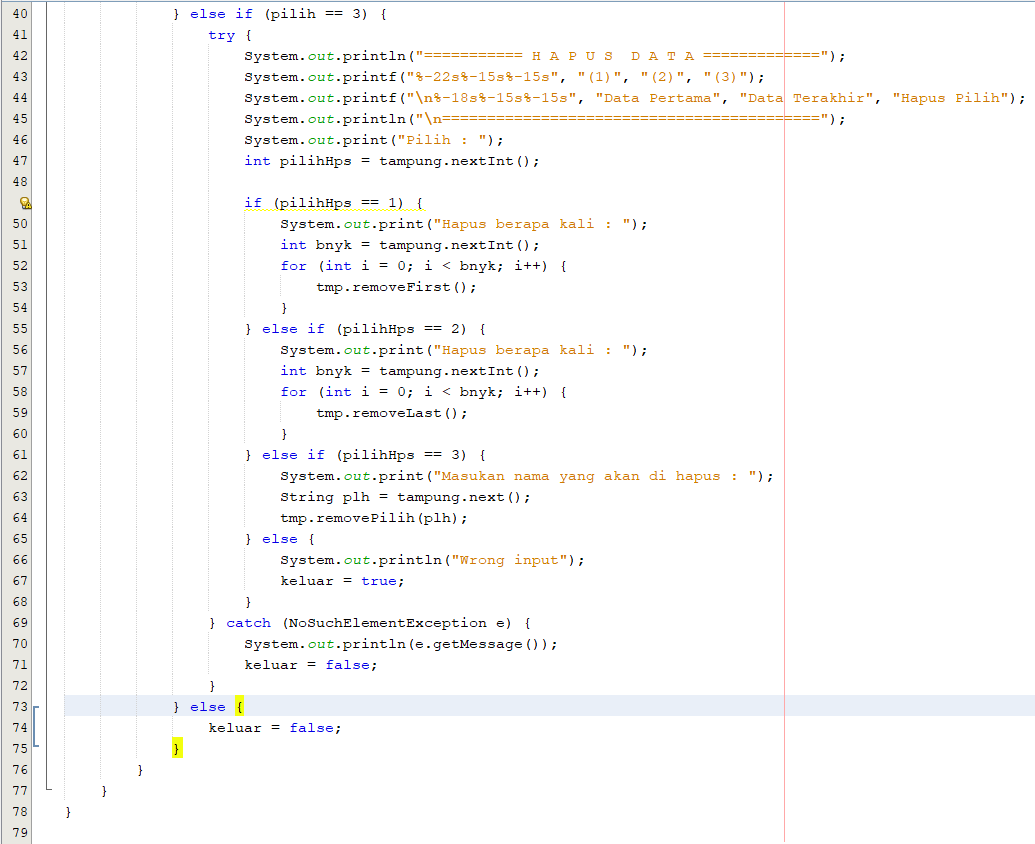


Class Logistic

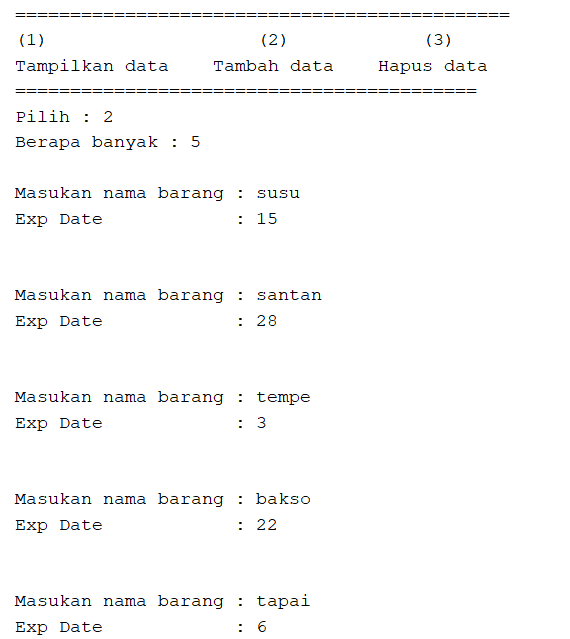


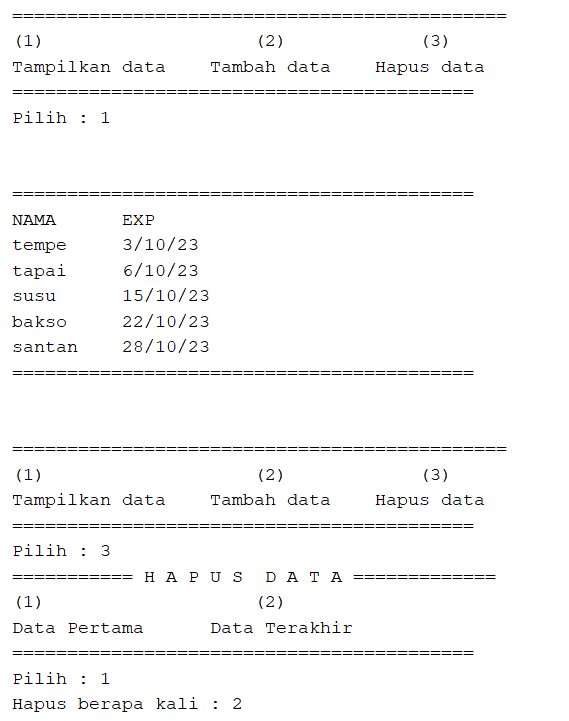
Class MainLogistic

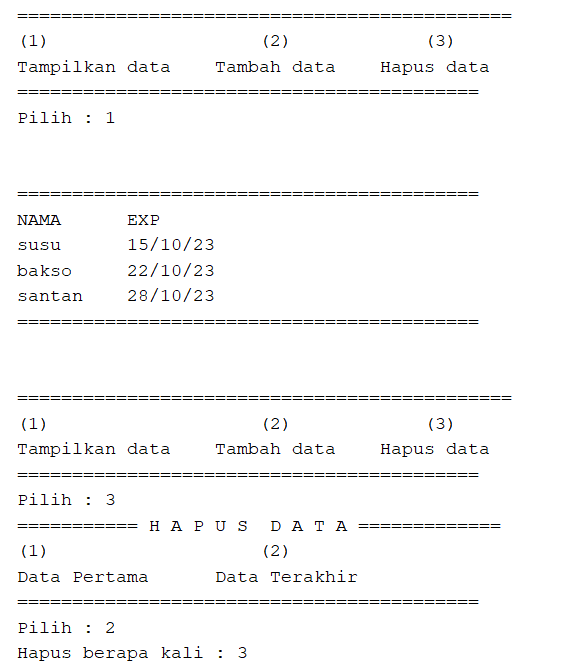


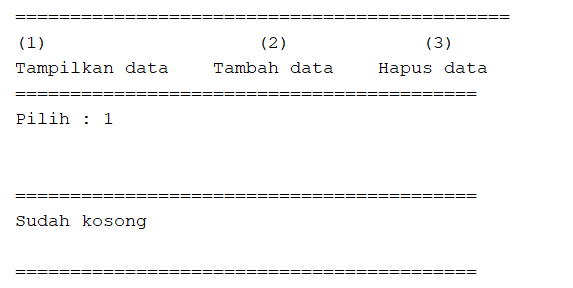


1. **OUTPUT**









1. **ANALISA**

**Method print2**

method ini berfungsi untuk menampilkan data – data yang ada berupa nama barang dan tanggal expired barang tersebut. Sebelum menampilkan data perlu mendeklarasikan variabel bantu bertipe ListNode dan diinisialisasi dengan elemen berikutnya setelah head. Kemudian melakukan pengecekan apakah linked list dalam keadaan kosng atau tidak. Jika bantu == head, menandakan bahwa linked list dalam keadaan kosong dan akan mencetak tulisan “Sudah kosong”. Jika linked list tidak dalam keadaan kosong, maka akan melakukan perulangan while dengan syarat perulangan akan berjalan terus selama bantu tidak sama dengan head. Kemudian melakukan casting terhadap tipe data bantu.getElemen() menjadi tipe Logistic dan melakukan pemanggilan untuk method getNama() dan getTanggal() dari objek yang disimpan di dalam bantu. Setelah itu memperbaharui bantu dengan elemen berikutnya sehingga perulangan di dalam linked list dapat berlanjut hingga semua elemen dicetak.

**Method remove**

Method remove ini untuk menghapus hubungan atau kaitan antar node dalam list. Pada method remove ini memiliki parameter yaitu node bantu yang ingin dihapus dalam sebuah list. Sebelum melakukan penghapusan node, maka perlu dilakukan pengecekan apakah list berada dalam keadaan kosong atau tidak. Jika list dalam keadaan kosong, maka akan di throw noSuchElementExpection. Sebaliknya, jika list tidak dalam keadaaan kosong, maka mengubah referensi dari next untuk menghubungkan elemen sebelum bantu dengan elemen setelah bantu. Kemudian mengubah referensi dari prev untuk menghubungkan elemen setelah bantu dengan elemen sebelum bantu. . Selanjutnya nilai dari zise akan dikurangi dan method ini akan mengembalikan elemen yang dihapus dari bantu dengan menggunakan metode getElemen().

**Method removeFirst**

Pada method ini, program akan memanggil method remove dengan parameter head.getNext() dengan tujuan untuk menghapus elemen setelah head yang ada di linked list dan mengembalikan objek yang dihapus.

**Method removeLast**

Pada method ini, program akan memanggil method remove dengan parameter head.getPrev() dengan tujuan untuk menghapus elemen sebelum head yang diberikan dari linked list dan mengembalikan objek yang dihapus.

**Method addSort**

Method ini untuk mengurutkan data dengan parameter yaitu key yang bertipe data Object. Sebelum mengurutkan, perlu mendeklarasikan bantu yang bertipe ListNode dan diinisalisasi dengan elemen setelah head. Kemudian melakukan perulangan

Dengan syarat perulangan akan terus berjalan selama bantu tidak sama dengan head. Perulangan ini berfungsi untuk menyisipkan elemen baru dalam linked list. Jika tanggal dari objek bantu yang bertipe Logistic lebih besar dari objek key yang bertipe Logistic, maka key harus disisipkan sebelum bantu dengan menggunakan pemanggilan method addBefore. Kemudian return untuk keluar dari metode addSort() setelah penambahan dilakukan dan memperbaharui bantu dengan elemen berikutnya sehingga perulangan akan terus berjalan hingga menemukan posisi yang tepat untuk menyisipkan key. Kemudian memanggil method addLast untuk menambahkan key sebagai elemen terakhir dikarenakan tidak ada posisi yang ditemukan untuk menyisipkan key.

**Method RemovePilih**

Pada method ini, program ini mengiterasi melalui linked list yang dimulai dari head dan mencari elemen dengan nama yang cocok dengan key. Jika ditemukan, elemen tersebut dihapus dari linked list. Jika tidak ada kecocokan ditemukan, program mengembalikan null.