

Nama : Dzakwan Dhiya Ulhaq

NIM : 6706210092

Kelas : 45-03

Penjelasan Jurnal 5 Soal 2

Robert lagi-lagi merasa bingung. Banyaknya tugas, kuis, belum ditambah dengan kesibukan Robert di kegiatan kemahasiswaan, telah membuat Robert seringkali melupakan hal-hal yang seharusnya Robert lakukan. Pagi ini saja Robert lupa membawa tugas ISD yang harus dikumpulkan karena salah mengira waktu pengumpulan. Bantulah Robert untuk mencatat dan mengingatkan pekerjaan apa saja yang harus dikerjakan dalam waktu dekat dengan membuat program to-do list sederhana. Fitur yang harus dimiliki aplikasi adalah sebagai berikut:

- Menambah nama pekerjaan yang harus dilakukan.
- Menampilkan nama pekerjaan yang harus diselesaikan terlebih dahulu. Pekerjaan yang dimasukkan pertama harus diselesaikan lebih dahulu, kemudian pekerjaan kedua, ketiga, dan seterusnya. Contoh, jika yang dimasukkan pertama kali adalah Jurnal 05 ISD, kemudian Kuis Sistem Basis Data, lalu Tugas Besar, maka urutan tugas yang harus diselesaikan adalah Jurnal 05 ISD – Kuis Sistem Basis Data – Tugas Besar.
- Menghapus pekerjaan yang sudah diselesaikan. Dari contoh pada bagian b, jika pekerjaan Jurnal 05 ISD sudah selesai dikerjakan, maka tugas tersebut harus dihapus dari list to-do.
- Menampilkan seluruh isi to-do list. Tuliskan penjelasan program dan kumpulan beserta project untuk Soal 2. Jangan lupa menuliskan NIM dan Nama Anda pada file penjelasan

Penjelasan:

#Class ToDoList/Queue

1. Membuat Class ToDoList/Queue

Pertama kita kita buat class queue yang isinya merupakan method-method untuk mengelola data. Disini saya beri nama Class nya yaitu ToDoList. lalu kita masukkan import java.util.* untuk mencakup semua masukkan seperti LinkedList dan Iterator.

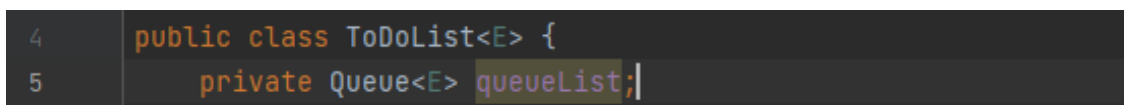


```
1 package com.company;
2 import java.util.*;
3
```

Gambar 1. 1 Membuat Class ToDoList

2. Atribut class ToDoList/Queue

lalu kita buat variable Queue<E> dengan nama variable queueList;



```
4 public class ToDoList<E> {
5     private Queue<E> queueList;
```

Gambar 1. 2 Atribut Class ToDoList

3. Membuat method ToDoList

selanjutnya kita buat method ToDoList yang isi nya merupakan inisialilasi queueList sama dengan LinkedList baru.

```
6      public ToDoList(){
7          queueList = new LinkedList<E>();
8      }
```

Gambar 1. 3 Method ToDoList

4. Membuat method void enqueue

Lalu kita buat method void enqueue dengan parameter E yang variabelnya diberi nama object. Fungsi method ini yaitu untuk menambah objek baru kedalam queueList dengan perintah .add yang masukan adalah variable object.

```
9      public void enqueue(E object){
10         queueList.add(object);
11     }
```

Gambar 1. 4 Method Void enqueue

5. Membuat method E dequeue

Selanjutnya kita buat Method E dequeue yang fungsi nya untuk menghapus data awal pada queueList atau list to-do. Misalkan, jika di dalam sudah ada list Jurnal 05 ISD – Kuis Sistem Basis Data – Tugas Besar, maka Jurnal 05 ISD akan dihapus dari queueList. Caranya dengan menambahkan .remove pada queueList.

```
12     public E dequeue() throws NoSuchElementException{
13         return queueList.remove();
14     }
```

Gambar 1. 5 Method E dequeue

6. Membuat method boolean isEmpty

Lalu kita buat method boolean dengan nama isEmpty yang difungsikan untuk mengecek apakah queue kosong atau tidak. Cara kita tambahkan .isEmpty pada queueList.

```
15     public boolean isEmpty(){
16         return queueList.isEmpty();
17     }
```

Gambar 1. 6 Method Boolean isEmpty

7. Membuat method print.

Terakhir di class ToDoList (Queue) kita buat method print untuk mencetak/menampilkan isi To-do List yang masih ada atau yang belum dikerjakan. Pertama kita buat variable Iterator<E> baru dengan nama iterator, lalu kita buat variable iterator sama dengan queueList.iterator(). Karena saya ingin membuat style print out nya seperti inline dengan strip seperti Jurnal-05-ISD – Kuis Sistem-Basis-Data – Tugas-Besar. Maka kita harus membuat nested loop for. For pertama untuk

row dan for kedua untuk colom. Dimana colom pertama dan selanjut akan di beri strip(-) setelah data berhasil dicetak dan colom terakhir hanya mencetak data terakhir saja tanpa strip setelahnya. Di dalam perulangan for kedua/for colom juga kita buat variable Object baru dengan nama value yang nantinya akan di print sebagai isi data pada queueList. Variable value sama dengan iterator.next().

```
18     public void print(){
19         Iterator<E> iterator = queueList.iterator();
20         for(int i = 0; i < queueList.size(); i++){
21             for(int j = 0; j < queueList.size(); j++){
22                 Object value = iterator.next();
23
24                 if(j == queueList.size() - 1){
25                     System.out.print(value);
26                     break;
27                 }
28                 else{
29                     System.out.print(value + " - ");
30                 }
31             }
32
33             System.out.println();
34             break;
35         }
36     }
37 }
```

Gambar 1. 7 Method Void Print

#Class Main

1. Membuat class Main

Pertama kita masukkan import java.util.* untuk untuk mencakup semua masukkan termasuk Scanner.

```
1 package com.company;
2 import java.util.*;
3
```

Gambar 2. 1 Membuat Class Main

2. Atribut class main

Pertama kita buat list masukkan ToDoList baru bertipe data String dengan nama variable nya yaitu queue, Lalu kita buat Scanner baru dengan nama input, dan kita buat variable integer dengan nama option untuk memilih opsi menu yang akan dipilih.

```

4 ▶ public class Main {
5
6 ▶   public static void main(String[] args) {
7       ToDoList<String> queue = new ToDoList<>();
8       Scanner input = new Scanner(System.in);
9       int option = 0;

```

Gambar 2. 2 Atribut Class Main

3. Membuat menu pengelolaan data to-do list

Disini saya akan membuat 4 opsi yaitu menambah data baru, menghapus to-do, mencetak isi to-to list, dan close program. Kita buat perulangan while dengan syarat perulangan jika input pada option bukan sama dengan 4.

```

12   while(option != 4){
13       System.out.println("1. Input to-do baru");
14       System.out.println("2. Hapus to-do");
15       System.out.println("3. Print to-do List");
16       System.out.println("4. Keluar");
17       option = input.nextInt();

```

Gambar 2. 3 Membuat Menu

4. Membuat opsi 1 : Input data to-do list

Jika kita memilih opsi 1 maka kita akan membuat input data to-do baru. Pertama kita buat input String baru dengan nama variable toDo untuk menginput toDo yang baru. Lalu kita masukkan toDo yang baru dinput kan dengan menambah kan method .enqueue() pada variable list queue dengan masukkan data yaitu toDo.

```

18   if(option == 1){
19       System.out.println("Masukkan to-do:");
20       String toDo = input.next();
21       queue.enqueue(toDo);
22       System.out.println();
23   }

```

Gambar 2. 4 Membuat Opsi 1: Input Data

5. Membuat opsi 2 : Menghapus data to-do

Jika memilih opsi 2 atau menginput option=2 maka kita akan menghapus data. Data yang akan dihapus adalah data yang pertama pada list atau data yang akan dihapus disini adalah data to-do yang sudah dikerjakan oleh Robert. Caranya dengan menambah method .dequeue() pada variable list queue. Lalu setelah dihapus. Kita buat opsi lagi, yaitu opsi mencetak data to-do terkini setelah data atau pekerjaan pada list telah dihapus. Disini kita buat (y) untuk mencetak to-do list terkini dan (n) untuk kembali ke menu utama. Jika kita memilih (y) maka kita akan mencetak isi to-do list, caranya dengan menambahkan method .print() pada variable list queue.

```

24     else if(option == 2){
25         queue.dequeue();
26         System.out.println("To-do Pertama berhasil dihapus");
27         System.out.println("Cetak to-do terkini ? (y/n):");
28         String option2 = input.next();
29         while (!option2.equals("y") || !option2.equals("n")){
30             if(option2.equals("y")){
31                 System.out.println("List to-do yang belum diselesaikan:");
32                 queue.print();
33                 System.out.println();
34                 break;
35             }
36             else if(option2.equals("n")){
37                 System.out.println();
38                 break;
39             }
40         }
41     }
42 }

```

Gambar 2. 5 Membuat Opsi 2 : Hapus Data

6. Membuat opsi 3 : Mencetak to-do list

Terakhir jika kita memilih opsi 3 atau memasukkan option=3 maka kita akan mencetak isi dari to-do list dengan cara menambahkan method .print() pada variable list queue.

```

43     else if(option == 3){
44         System.out.println("List to-do yang belum diselesaikan:");
45         queue.print();
46         System.out.println();
47     }
48 }
49 }

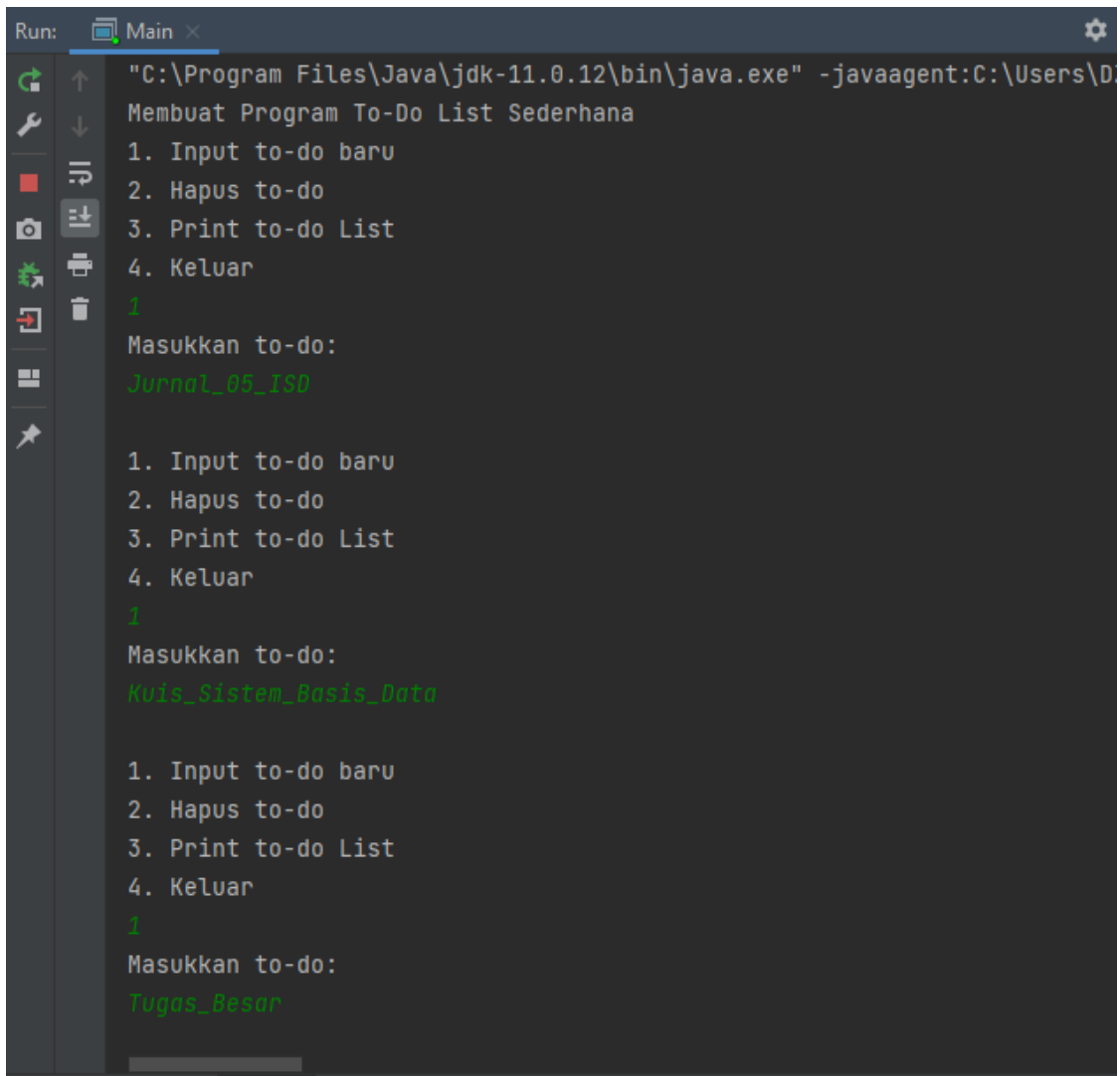
```

Gambar 2. 6 Membuat Opsi 3 : Cetak to-do list

#Mencoba Program

1. Mencoba opsi 1 : menambahkan data

Pertama kita akan mencoba memasukkan 3 data pekerjaan kedalam to-do list pada percobaan ini yaitu Jurnal-05-ISD – Kuis-Sistem-Basis-Data – Tugas-Besar. Carany dengan memilih/memasukkan input option=1.



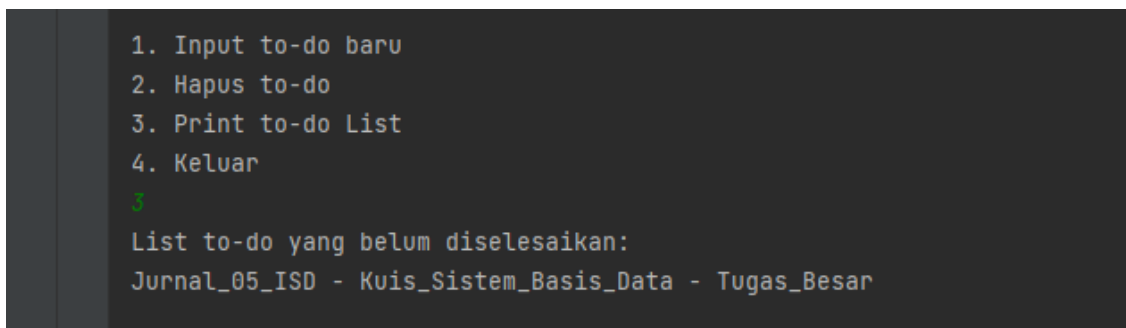
```
Run: Main x
"C:\Program Files\Java\jdk-11.0.12\bin\java.exe" -javaagent:C:\Users\D
Membuat Program To-Do List Sederhana
1. Input to-do baru
2. Hapus to-do
3. Print to-do List
4. Keluar
1
Masukkan to-do:
Jurnal_05_ISD

1. Input to-do baru
2. Hapus to-do
3. Print to-do List
4. Keluar
1
Masukkan to-do:
Kuis_Sistem_Basis_Data

1. Input to-do baru
2. Hapus to-do
3. Print to-do List
4. Keluar
1
Masukkan to-do:
Tugas_Besar
```

Gambar 3. 1 Mencoba Opsi 1 : Menambah Data

2. Mencoba opsi 3 : mencetak data to-do list terkini
sekarang kita akan mencoba mencetak data to-do list terkini menggunakan menu utama. Caranya dengan memasukkan option=3.

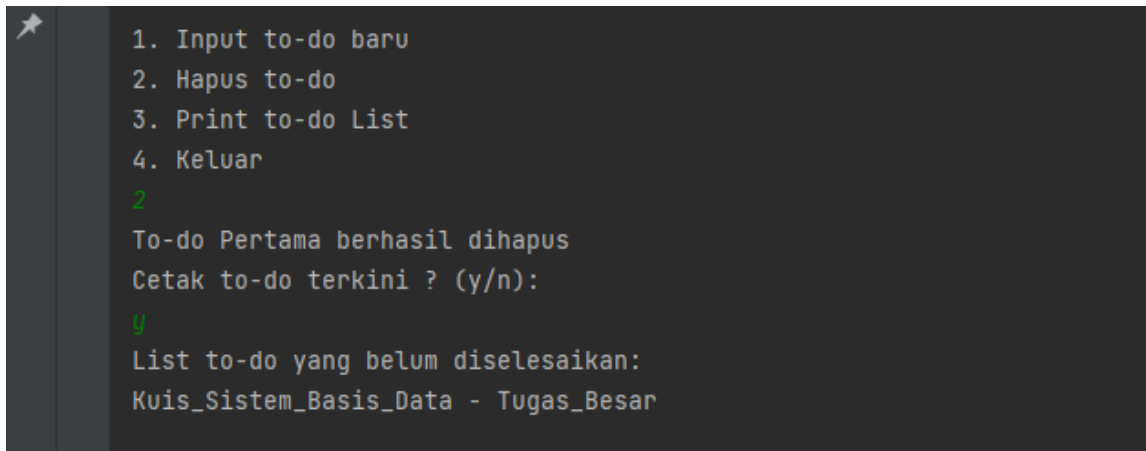


```
1. Input to-do baru
2. Hapus to-do
3. Print to-do List
4. Keluar
3
List to-do yang belum diselesaikan:
Jurnal_05_ISD - Kuis_Sistem_Basis_Data - Tugas_Besar
```

Gambar 3. 2 Mencoba Opsi 3 : Mencetak Data

3. Mencoba opsi 2 : menghapus data dan mencetak data to-do terkini setelah dihapus
sekarang kita akan mencoba menghapus data yang sudah dikerjakan atau data pertama pada to-do list lalu Setelah itu kita akan langsung mencoba mencetak hasil data to-do list terkini. Cara nya

dengan memilih/memasukkan option=2 untuk menghapus dan memilih/memasukkan opsi (y) setelahnya untuk mencetak data terkini.



```
1. Input to-do baru
2. Hapus to-do
3. Print to-do List
4. Keluar
2
To-do Pertama berhasil dihapus
Cetak to-do terkini ? (y/n):
y
List to-do yang belum diselesaikan:
Kuis_Sistem_Basis_Data - Tugas_Besar
```

Gambar 3. 3 Mencoba Opsi 2 : Menghapus Data