**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**PERTEMUAN KE 10 JOBSHEET 8 QUEUE**

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



Muhammad Afiq Firdaus

2341760189 / 21

SIB1E

**PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

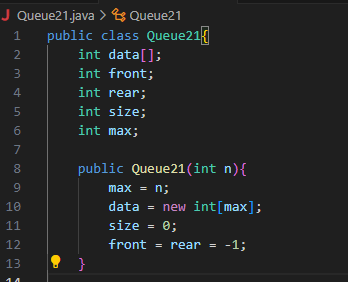
**2024**

**8.2 Praktikum 1**

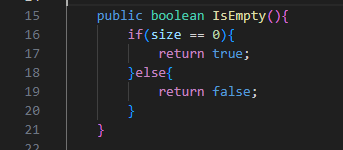
buat class baru dengan nama Queue.



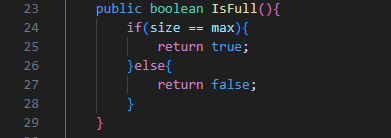
Tambahkan atribut-atribut Queue sesuai diagram class, kemudian tambahkan pula konstruktornya



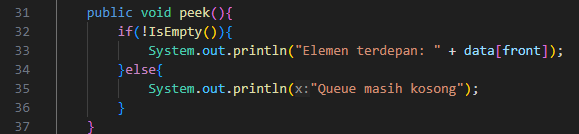
Buat method IsEmpty bertipe boolean yang digunakan untuk mengecek apakah queue kosong.



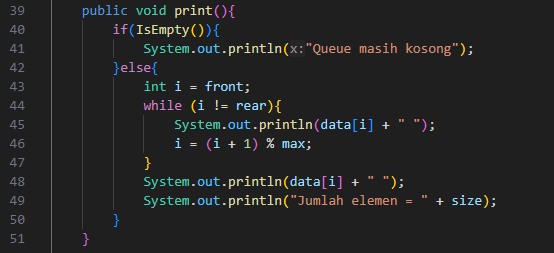
Buat method IsFull bertipe boolean yang digunakan untuk mengecek apakah queue sudah penuh.



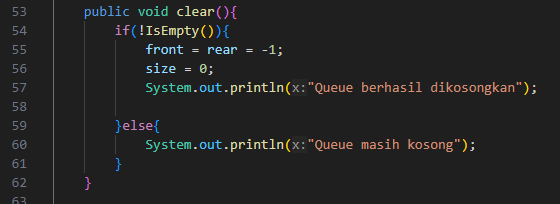
Buat method peek bertipe void untuk menampilkan elemen queue pada posisi paling depan.



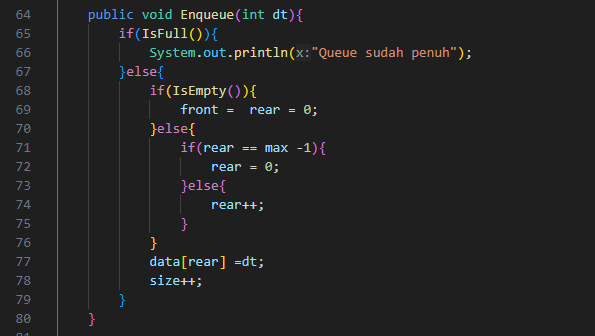
Buat method print bertipe void untuk menampilkan seluruh elemen pada queue mulai dari posisi front sampai dengan posisi rear.



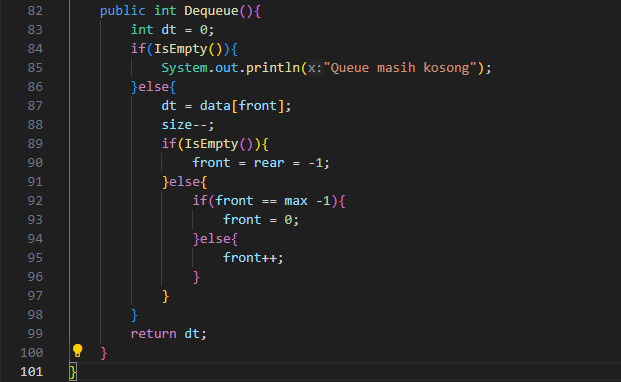
Buat method clear bertipe void untuk menghapus semua elemen pada queue



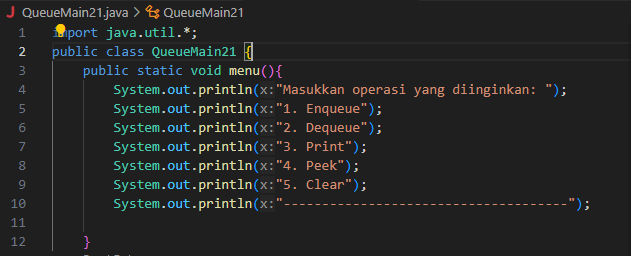
Buat method Enqueue bertipe void untuk menambahkan isi queue dengan parameter dt yang bertipe integer

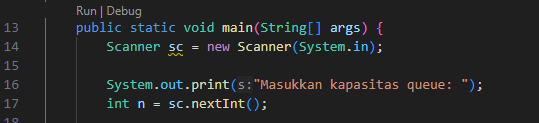


Buat method Dequeue bertipe int untuk mengeluarkan data pada queue di posisi paling depan



Selanjutnya, buat class baru dengan nama QueueMain. Buat method menu bertipe void untuk memilih menu program pada saat dijalankan.

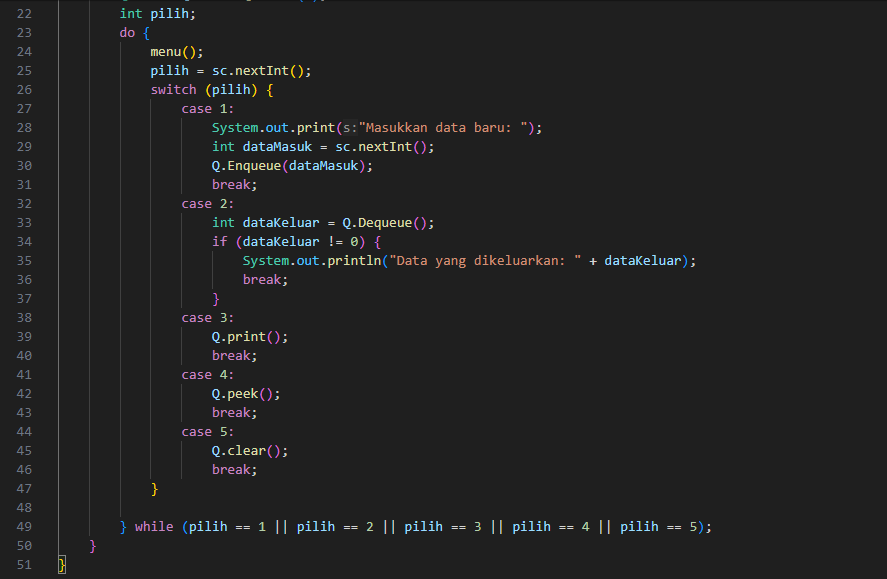
  
Buat fungsi main, kemudian deklarasikan Scanner dengan nama sc. Buat variabel n untuk menampung masukan berupa jumlah maksimal elemen yang dapat disimpan pada queue.



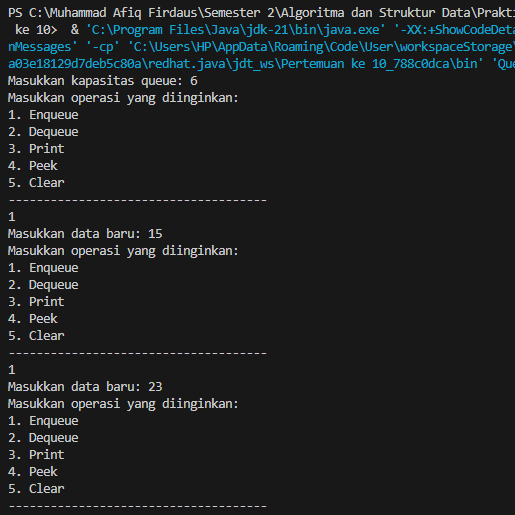
Lakukan instansiasi objek Queue dengan nama Q dengan mengirimkan parameter n sebagai kapasitas elemen queue.

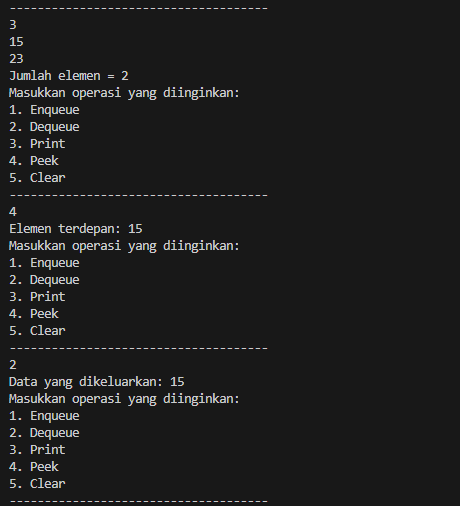


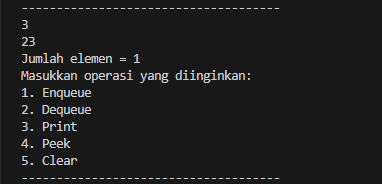
Deklarasikan variabel dengan nama pilih bertipe integer untuk menampung pilih menu dari pengguna. Lakukan perulangan menggunakan do-while untuk menjalankan program secara terus menerus sesuai masukan yang diberikan. Di dalam perulangan tersebut, terdapat pemilihan kondisi menggunakan switch-case untuk menjalankan operasi queue sesuai dengan masukan pengguna.



Compile dan jalankan class QueueMain, kemudian amati hasilnya.







**8.2.3 Pertanyaan**

1. Pada konstruktor, mengapa nilai awal atribut front dan rear bernilai -1, sementara atribut size bernilai 0?
2. Pada method Enqueue, jelaskan maksud dan kegunaan dari potongan kode berikut!



1. Pada method Dequeue, jelaskan maksud dan kegunaan dari potongan kode berikut!



1. Pada method print, mengapa pada proses perulangan variabel i tidak dimulai dari 0 (int i=0), melainkan int i=front?
2. Perhatikan kembali method print, jelaskan maksud dari potongan kode berikut!

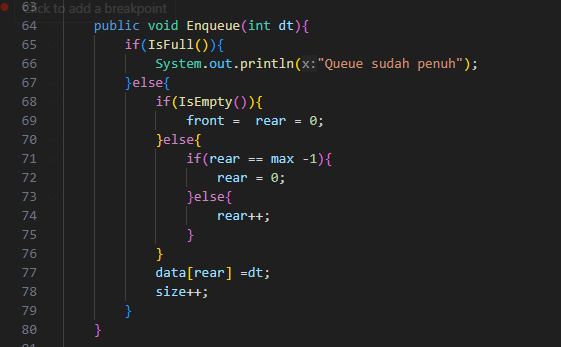


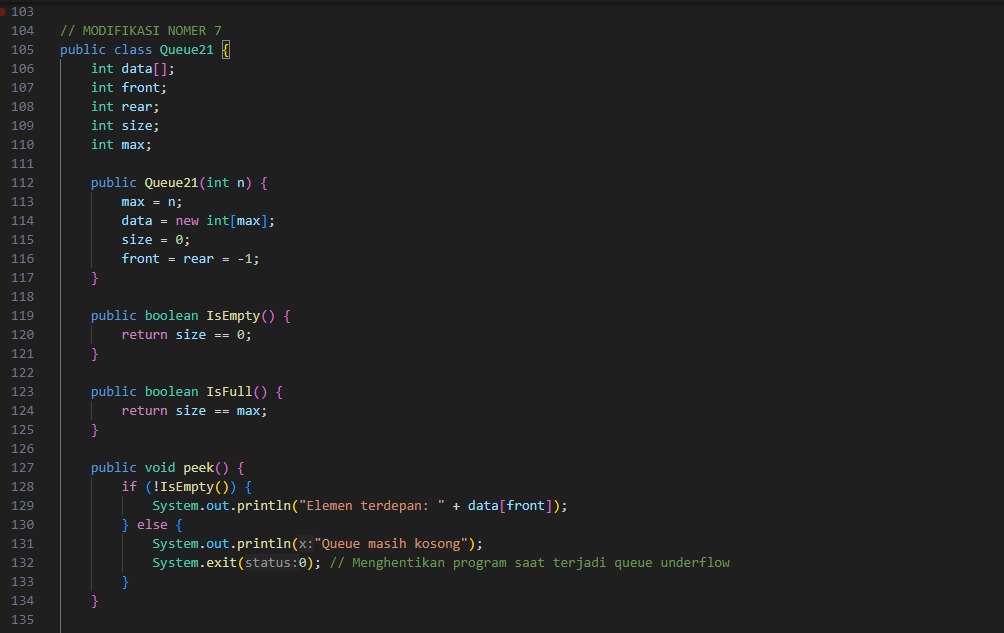
1. Tunjukkan potongan kode program yang merupakan queue overflow!
2. Pada saat terjadi queue overflow dan queue underflow, program tersebut tetap dapat berjalan dan hanya menampilkan teks informasi. Lakukan modifikasi program sehingga pada saat terjadi queue overflow dan queue underflow, program dihentikan!

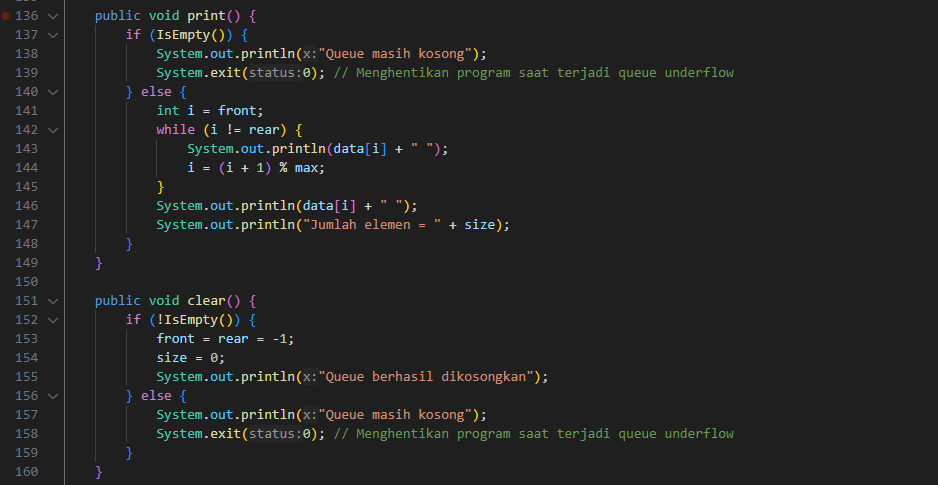
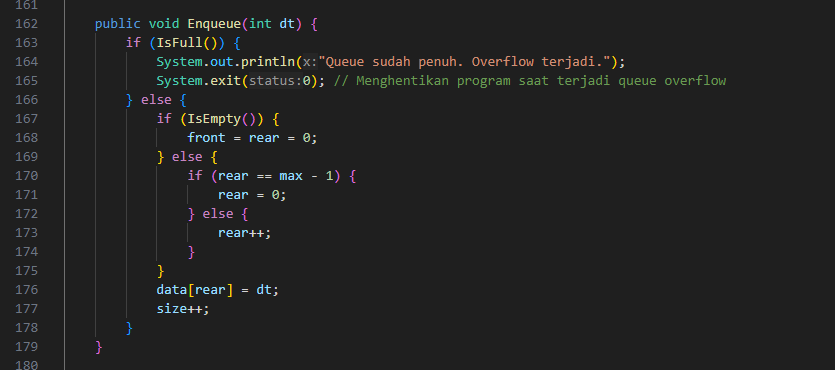
**Jawaban**

1. nilai awal atribut front dan rear diatur sebagai -1 karena -1 menandakan bahwa antrian (queue) saat itu kosong. Ini mengindikasikan bahwa tidak ada elemen di dalam antrian pada awalnya.
2. Maksud dan kegunaan dari code tersebut adalah bagian dari operasi enqueue pada struktur data antrian (queue). Dalam kode tersebut, kondisi rear == max - 1 memeriksa apakah indeks rear sudah mencapai batas maksimum yang ditentukan (max - 1).ini berarti antrian sudah penuh.
3. kode tersebut merupakan bagian dari operasi dequeue pada struktur data antrian (queue). Dalam kode tersebut, kondisi front == max - 1 memeriksa apakah indeks front sudah mencapai batas maksimum yang ditentukan (max - 1). ini berarti kita sudah sampai di akhir antrian.
4. Pada method print dalam implementasi kelas Queue21, proses perulangan variabel i dimulai dari front bukan dari 0 karena kita ingin mencetak elemen-elemen yang ada dalam antrian (queue) sesuai dengan urutan dari front ke rear. ini dilakukan untuk memastikan bahwa kita mencetak elemen secara berurutan, sesuai dengan cara penyimpanan data dalam array circular yang digunakan untuk mewakili queue.
5. Potongan kode i = (i + 1) % max; dalam method print pada implementasi kelas Queue21 memiliki tujuan untuk melanjutkan indeks **i** ke elemen berikutnya dalam array secara circular.
6. Queue overflow terjadi ketika kita mencoba menambahkan elemen ke dalam antrian yang sudah penuh (mencapai kapasitas maksimumnya). Dalam implementasi queue menggunakan array seperti yang disediakan di kelas Queue21, queue overflow dapat terjadi saat mencoba melakukan operasi enqueue (Enqueue(int dt)) pada antrian yang sudah mencapai kapasitas maksimumnya (max).

Berikut adalah code yang menunjukkan Queue overflow



1. 

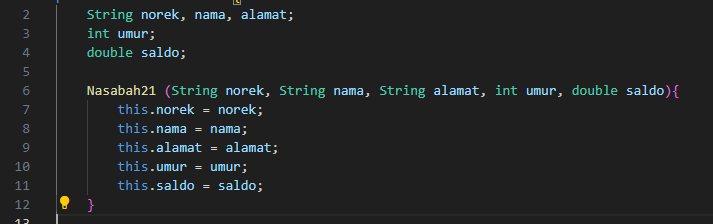


**Praktikum 2**

buat class baru dengan nama Nasabah.

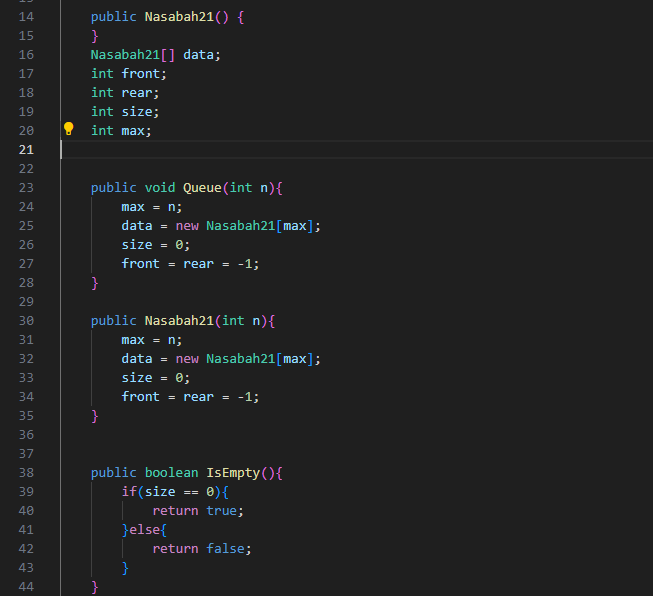


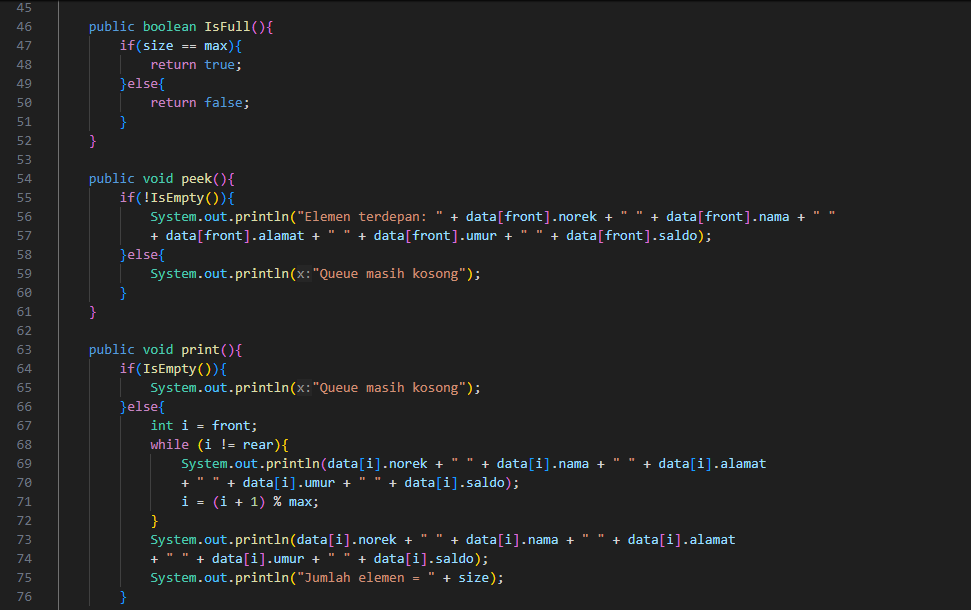
Tambahkan atribut-atribut Nasabah seperti pada Class Diagram, kemudian tambahkan pula konstruktornya



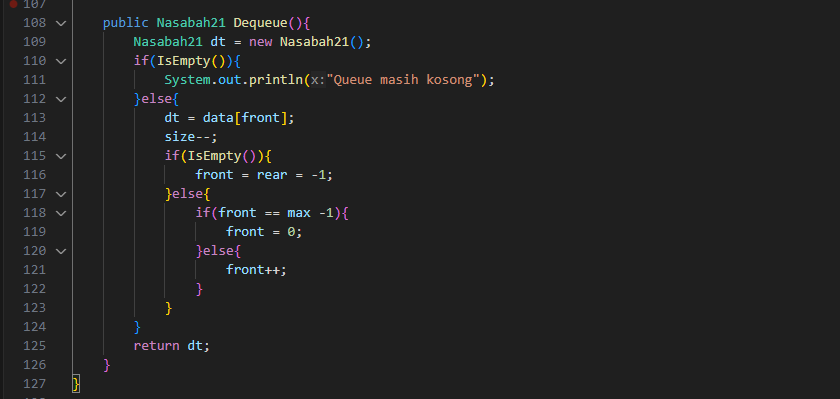
Salin kode program class Queue pada Praktikum 1 untuk digunakan kembali pada Praktikum 2 ini. Karena pada Praktikum 1, data yang disimpan pada queue hanya berupa array bertipe integer, sedangkan pada Praktikum 2 data yang digunakan adalah object, maka perlu dilakukan modifikasi pada class Queue tersebut.

modifikasi pada class Queue dengan mengubah tipe int[] data menjadi Nasabah[] data karena pada kasus ini data yang akan disimpan pada queue berupa object Nasabah. Modifikasi perlu dilakukan pada atribut, method Enqueue, dan method Dequeue.









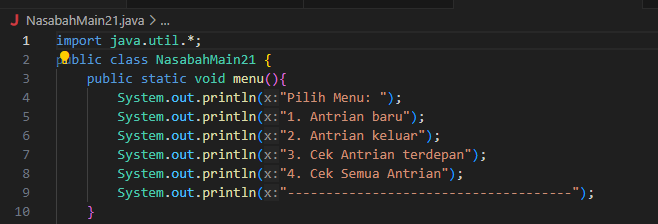
Baris program Nasabah dt = new Nasabah(); akan ditandai sebagai error, untuk mengatasinya, tambahkan konstruktor default di dalam class Nasabah.



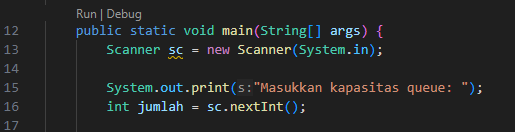
Karena satu elemen queue terdiri dari beberapa informasi (norek, nama, alamat, umur, dan saldo), maka ketika mencetak data juga perlu ditampilkan semua informasi tersebut, sehingga meodifikasi perlu dilakukan pada method peek dan method print.



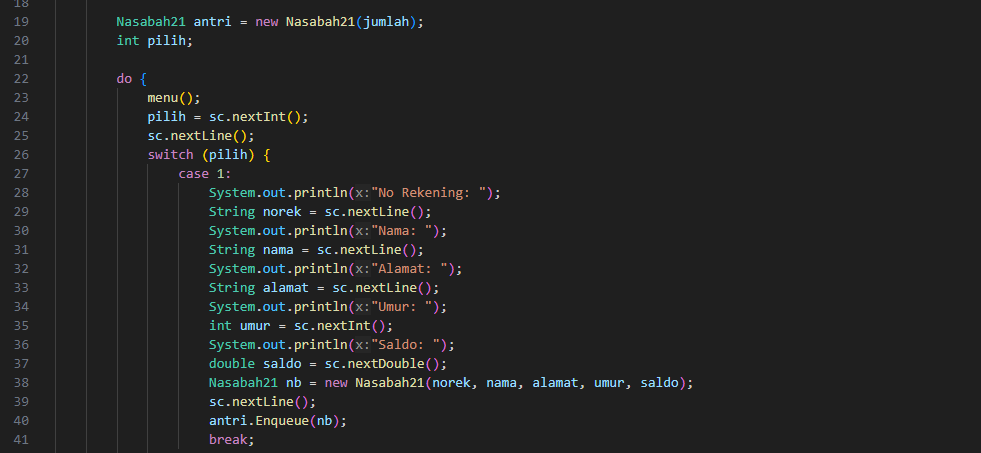
buat class baru dengan nama QueueMain. Buat method menu untuk mengakomodasi pilihan menu dari masukan pengguna



Buat fungsi main, deklarasikan Scanner dengan nama sc. Buat variabel max untuk menampung kapasitas elemen pada queue. Kemudian lakukan instansiasi objek queue dengan nama antri dan nilai parameternya adalah variabel jumlah.

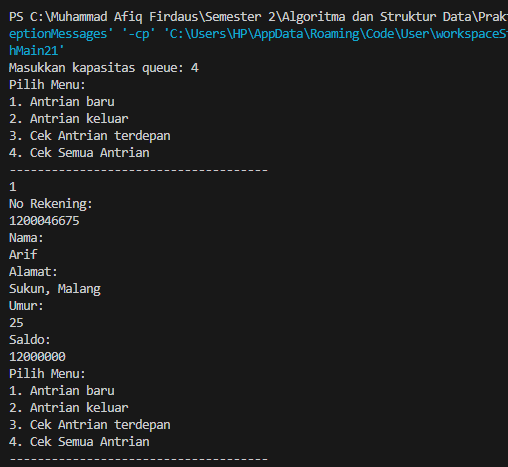


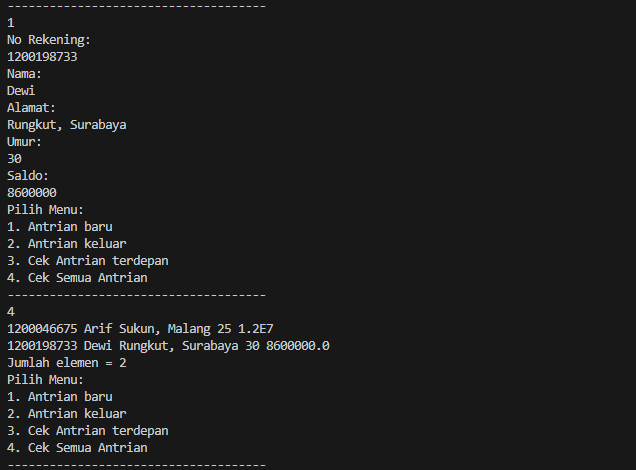
Deklarasikan variabel dengan nama pilih bertipe integer untuk menampung pilih menu dari pengguna. Tambahkan kode berikut untuk melakukan perulangan menu sesuai dengan masukan yang diberikan oleh pengguna.

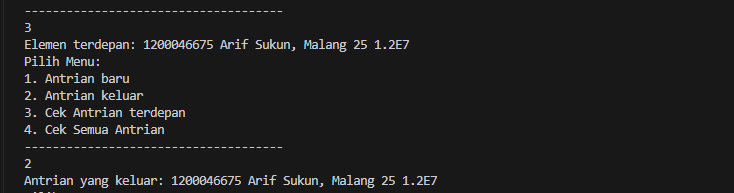




Compile dan jalankan class QueueMain, kemudian amati hasilnya

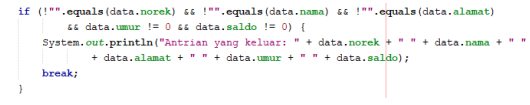






**8.3.3 Pertanyaan**

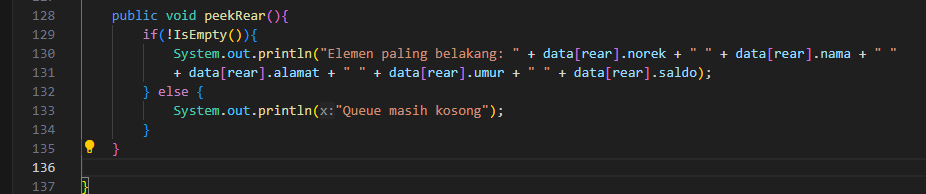
1. Pada class QueueMain, jelaskan fungsi IF pada potongan kode program berikut!

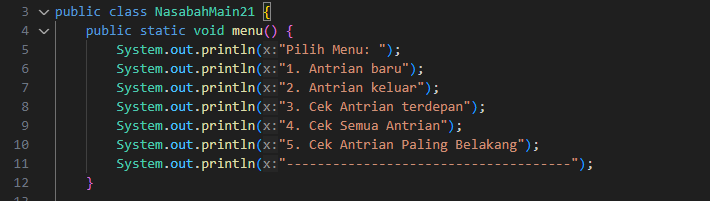


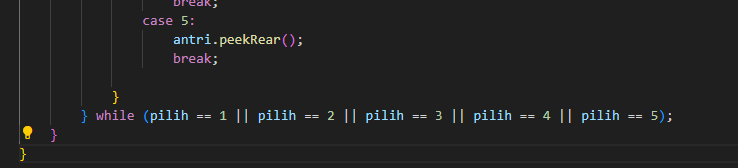
1. Lakukan modifikasi program dengan menambahkan method baru bernama peekRear pada class Queue yang digunakan untuk mengecek antrian yang berada di posisi belakang! Tambahkan pula daftar menu 5. Cek Antrian paling belakang pada class QueueMain sehingga method peekRear dapat dipanggil!

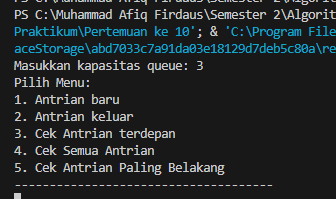
**Jawaban**

1. Fungsi dari kondisi if ini adalah untuk memeriksa apakah data yang dikeluarkan dari antrian (data) valid atau tidak sebelum melakukan operasi cetak (print).



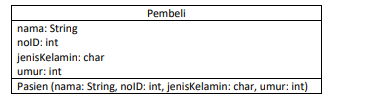




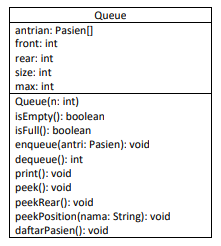


**8.4 Tugas**

1. Buatlah program antrian untuk mengilustrasikan antrian pasien di sebuah klinik. Ketika seorang pasien akan mengantri, maka dia harus mendaftarkan nama, nomor identitas, jenis kelamin dan umur seperti yang digambarkan pada Class diagram berikut:



Class diagram Queue digambarkan sebagai berikut:



Keterangan method:

• Method create(), isEmpty(), isFull(), enqueue(), dequeue() dan print(), kegunaannya sama seperti yang telah dibuat pada Praktikum

• Method peek(): digunakan untuk menampilkan data Pasien yang berada di posisi antrian paling depan

• Method peekRear(): digunakan untuk menampilkan data Pasien yang berada di posisi antrian paling belakang

• Method peekPosition(): digunakan untuk menampilkan seorang pasien (berdasarkan nama) posisi antrian ke berapa

• Method daftarPasien(): digunakan untuk menampilkan data seluruh pasien

Berikut adalah hasil dari tugas diatas

