Laporan Hasil Praktikum

Algoritma Dan Struktur Data

Jobsheet 9



Angel Chelssa Leoniy Eka Permatasari

244107020202

1E

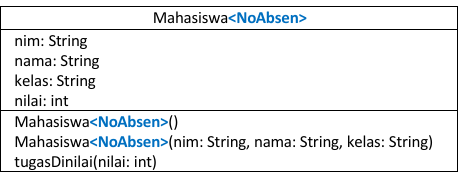
Program Studi Teknologi Informasi

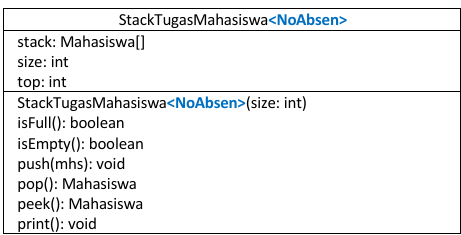
Jurusan Teknik Informatika

POLINEMA

2025

**Percobaan 1**



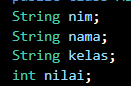


Langkah-langkah Percobaan

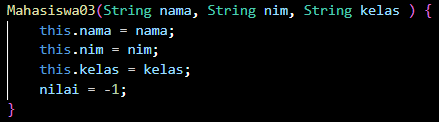
1. **Class Mahasiswa**
2. Buat folder baru bernama Jobsheet9 di dalam repository Praktikum ASD. Buat file baru, beri nama Mahasiswa<NoAbsen>.java



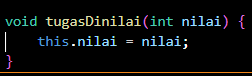
1. Lengkapi class Mahasiswa dengan atribut yang telah digambarkan di dalam class diagram Mahasiswa, yang terdiri dari atribut nama, nim, kelas, dan nilai



1. Tambahkan konstruktor berparameter pada class Mahasiswa sesuai dengan class diagram Mahasiswa. Berikan nilai default nilai = -1 sebagai nilai awal ketika tugas belum dinilai



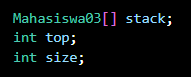
1. Tambahkan method tugasDinilai() yang digunakan untuk mengeset nilai ketika dilakukan penilaian tugas mahasiswa



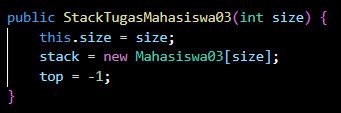
1. **Class StackTugasMahasiswa**
2. Setelah membuat class Mahasiswa, selanjutnya perlu dibuat class StackTugasMahasiswa<NoAbsen>.java sebagai tempat untuk mengelola tumpukan tugas. Class StackTugasMahasiswa merupakan penerapan dari struktur data Stack



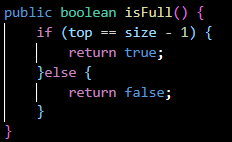
1. Lengkapi class StackTugasMahasiswa dengan atribut yang telah digambarkan di dalam class diagram StackTugasMahasiswa, yang terdiri dari atribut stack, size, dan top



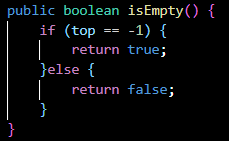
1. Tambahkan konstruktor berparameter pada class StackTugasMahasiswa untuk melakukan inisialisasi kapasitas maksimum data tugas mahasiswa yang dapat disimpan di dalam Stack, serta mengeset indeks awal dari pointer top



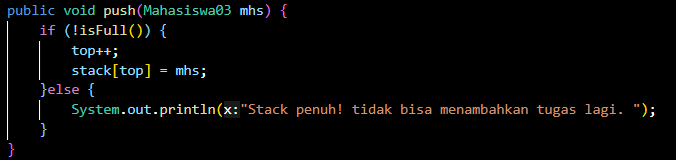
1. Selanjutnya, buat method isFull bertipe boolean untuk mengecek apakah tumpukan tugas mahasiswa sudah terisi penuh sesuai kapasitas



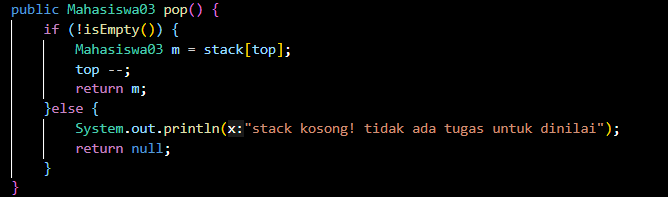
1. Pada class StackTugasMahasiswa, buat method isEmpty bertipe boolean untuk mengecek apakah tumpukan tugas masih kosong



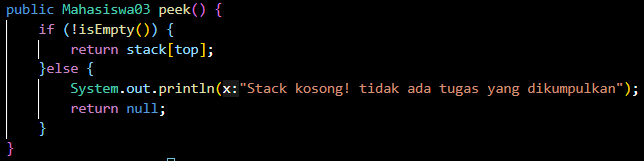
1. Untuk dapat menambahkan berkas tugas ke dalam tumpukan Stack, maka buat method push. Method ini menerima parameter mhs yang berupa object dari class Mahasiswa



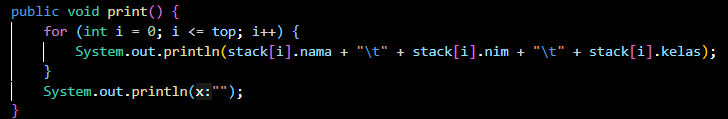
1. Penilaian tugas mahasiswa yang dilakukan oleh dosen dilakukan dengan menggunakan method pop untuk mengeluarkan tugas yang akan dinilai. Method ini tidak menerima parameter apapun namun mempunyai nilai kembalian berupa object dari class Mahasiswa



1. Buat method peek untuk dapat mengecek tumpukan tugas mahasiswa yang berada di posisi paling atas



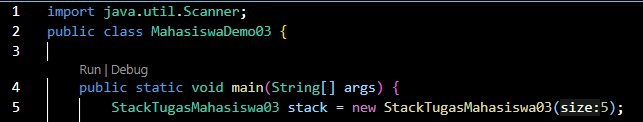
1. Tambahkan method print untuk dapat menampilkan semua daftar tugas mahasiswa pada Stack



1. **Class Utama**
2. Buat file baru, beri nama MahasiswaDemo<NoAbsen>.java



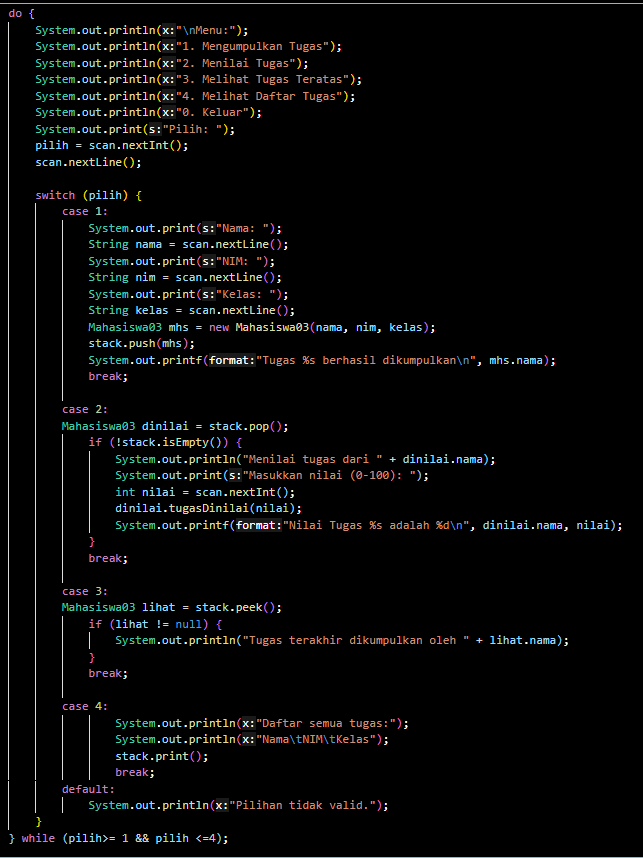
1. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi main
2. Di dalam fungsi main, lakukan instansiasi object StackTugasMahasiswa bernama stack dengan nilai parameternya adalah 5.



1. Deklarasikan Scanner dengan nama variabel scan dan variabel pilih bertipe int



1. Tambahkan menu untuk memfasilitasi pengguna dalam memilih operasi Stack dalam mengelola data tugas mahasiswa menggunakan struktur perulangan do-while



**Kode program**

**Class Mahasiswa03.java**

|  |
| --- |
| public class Mahasiswa03 {  String nim;  String nama;  String kelas;  int nilai;      Mahasiswa03(String nama, String nim, String kelas ) {          this.nama = nama;          this.nim = nim;          this.kelas = kelas;          nilai = -1;      }      void tugasDinilai(int nilai) {          this.nilai = nilai;      }  } |

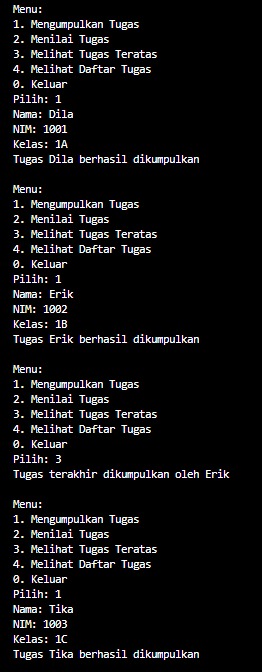
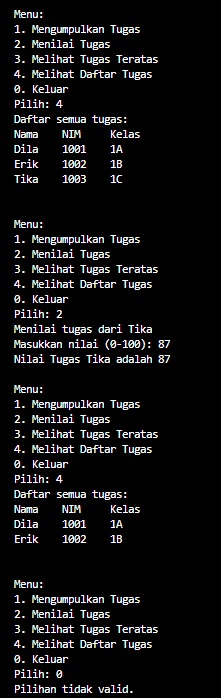
**Class StuckTugasMahasiswa03.java**

|  |
| --- |
| public class StackTugasMahasiswa03 {      Mahasiswa03[] stack;      int top;      int size;      public StackTugasMahasiswa03(int size) {          this.size = size;          stack = new Mahasiswa03[size];          top = -1;      }      public boolean isFull() {          if (top == size - 1) {              return true;          }else {              return false;          }      }      public boolean isEmpty() {          if (top == -1) {              return true;          }else {              return false;          }      }      public void push(Mahasiswa03 mhs) {          if (!isFull()) {              top++;              stack[top] = mhs;          }else {              System.out.println("Stack penuh! tidak bisa menambahkan tugas lagi. ");          }      }      public Mahasiswa03 pop() {          if (!isEmpty()) {              Mahasiswa03 m = stack[top];              top --;              return m;          }else {              System.out.println("stack kosong! tidak ada tugas untuk dinilai");              return null;          }      }      public Mahasiswa03 peek() {          if (!isEmpty()) {              return stack[top];          }else {              System.out.println("Stack kosong! tidak ada tugas yang dikumpulkan");              return null;          }      }        public void print() {          for (int i = 0; i <= top; i++) {              System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);          }          System.out.println("");      }  } |

**Class MahasiswaDemo03.java**

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaDemo03 {      public static void main(String[] args) {          StackTugasMahasiswa03 stack = new StackTugasMahasiswa03(5);          Scanner scan = new Scanner(System.in);          int pilih;          do {              System.out.println("\nMenu:");              System.out.println("1. Mengumpulkan Tugas");              System.out.println("2. Menilai Tugas");              System.out.println("3. Melihat Tugas Teratas");              System.out.println("4. Melihat Daftar Tugas");              System.out.println("0. Keluar");              System.out.print("Pilih: ");              pilih = scan.nextInt();              scan.nextLine();              switch (pilih) {                  case 1:                      System.out.print("Nama: ");                      String nama = scan.nextLine();                      System.out.print("NIM: ");                      String nim = scan.nextLine();                      System.out.print("Kelas: ");                      String kelas = scan.nextLine();                      Mahasiswa03 mhs = new Mahasiswa03(nama, nim, kelas);                      stack.push(mhs);                      System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);                      break;                  case 2:                  Mahasiswa03 dinilai = stack.pop();                      if (!stack.isEmpty()) {                          System.out.println("Menilai tugas dari " + dinilai.nama);                          System.out.print("Masukkan nilai (0-100): ");                          int nilai = scan.nextInt();                          dinilai.tugasDinilai(nilai);                          System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);                      }                      break;                  case 3:                  Mahasiswa03 lihat = stack.peek();                      if (lihat != null) {                          System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);                      }                      break;                  case 4:                          System.out.println("Daftar semua tugas:");                          System.out.println("Nama\tNIM\tKelas");                          stack.print();                          break;                  default:                      System.out.println("Pilihan tidak valid.");              }          } while (pilih>= 1 && pilih <=4);      }  } |

**Hasil kode**

**Pertanyaan**

1. Lakukan perbaikan pada kode program, sehingga keluaran yang dihasilkan sama dengan verifikasi hasil percobaan! Bagian mana yang perlu diperbaiki?

Modifikasi kode

**Kode program class Mahasiswa03.java**

|  |
| --- |
| public class Mahasiswa03 {      String nim, nama, kelas;      int nilai;      Mahasiswa03() {          this.nim = "";          this.nama = "";          this.kelas = "";          this.nilai = -1;      }      Mahasiswa03(String nim, String nama, String kelas) {          this.nim = nim;          this.nama = nama;          this.kelas = kelas;          this.nilai = -1;      }      void tugasDinilai(int nilai) {          this.nilai = nilai;      }  } |

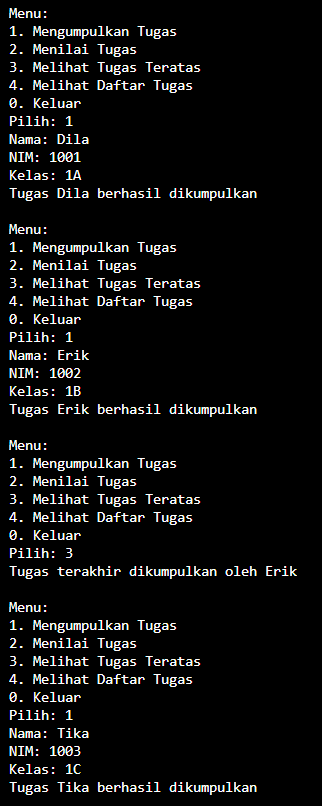
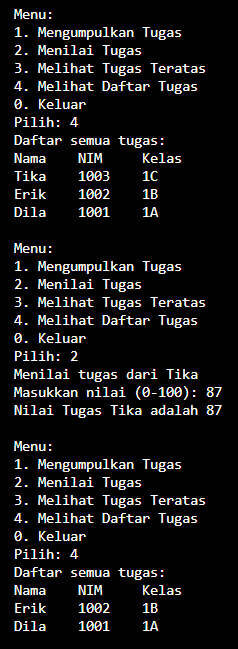
**Kode program class StuckTugasMahasiswa**

|  |
| --- |
| public class StackTugasMahasiswa03 {      Mahasiswa03[] stack;      int size, top;      StackTugasMahasiswa03(int size) {          this.size = size;          this.stack = new Mahasiswa03[size];          this.top = -1;      }      boolean isFull() {          return top == size - 1;      }      boolean isEmpty() {          return top == -1;      }      void push(Mahasiswa03 mhs) {          if (!isFull()) {              stack[++top] = mhs;          } else {              System.out.println("Stack penuh. Tidak dapat menambahkan tugas.");          }      }      Mahasiswa03 pop() {          if (!isEmpty()) {              return stack[top--];          } else {              System.out.println("Stack kosong. Tidak ada tugas yang bisa dinilai.");              return null;          }      }      Mahasiswa03 peek() {          if (!isEmpty()) {              return stack[top];          } else {              System.out.println("Stack kosong.");              return null;          }      }      void print() {          if (!isEmpty()) {              for (int i = top; i >= 0; i--) {                  System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);              }          } else {              System.out.println("Stack kosong.");          }      }  } |

**Kode program class MahasiswaDemo03.java**

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaDemo03 {      public static void main(String[] args) {          StackTugasMahasiswa03 stack = new StackTugasMahasiswa03(5);          Scanner scan = new Scanner(System.in);          int pilih;          do {              System.out.println("\nMenu:");              System.out.println("1. Mengumpulkan Tugas");              System.out.println("2. Menilai Tugas");              System.out.println("3. Melihat Tugas Teratas");              System.out.println("4. Melihat Daftar Tugas");              System.out.println("0. Keluar");              System.out.print("Pilih: ");              pilih = scan.nextInt();              scan.nextLine();              switch (pilih) {                  case 1:                      System.out.print("Nama: ");                      String nama = scan.nextLine();                      System.out.print("NIM: ");                      String nim = scan.nextLine();                      System.out.print("Kelas: ");                      String kelas = scan.nextLine();                      Mahasiswa03 mhs = new Mahasiswa03(nim, nama, kelas);                      stack.push(mhs);                      System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);                      break;                  case 2:                      Mahasiswa03 dinilai = stack.pop();                      if (dinilai != null) {                          System.out.println("Menilai tugas dari " + dinilai.nama);                          System.out.print("Masukkan nilai (0-100): ");                          int nilai = scan.nextInt();                          dinilai.tugasDinilai(nilai);                          System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);                      }                      break;                  case 3:                      Mahasiswa03 lihat = stack.peek();                      if (lihat != null) {                          System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);                      }                      break;                  case 4:                      System.out.println("Daftar semua tugas:");                      System.out.println("Nama\tNIM\tKelas");                      stack.print();                      break;                  case 0:                      System.out.println("Terima kasih.");                      break;                  default:                      System.out.println("Pilihan tidak valid.");              }          } while (pilih != 0);      }  } |

**Hasil kode yang sudah diubah**

1. Berapa banyak data tugas mahasiswa yang dapat ditampung di dalam Stack? Tunjukkan potongan kode programnya!

**Jawaban :** 5 data



1. Mengapa perlu pengecekan kondisi **!isFull()** pada method **push**? Kalau kondisi if-else tersebut dihapus, apa dampaknya?

**Jawaban** : Karena kalau Stack udah penuh, terus kamu masih coba masukin data, program bisa error (kayak nabrak tembok).

Dengan cek !isFull(), kita pastiin dulu masih ada tempat buat masukin data. Kalau kamu hapus pengecekan itu , nanti bisa muncul eror.

ArrayIndexOutOfBoundsException, alias kamu maksa taruh data di tempat yang nggak ada.

1. Modifikasi kode program pada class MahasiswaDemo dan StackTugasMahasiswa sehingga pengguna juga dapat melihat mahasiswa yang pertama kali mengumpulkan tugas melalui operasi lihat tugas terbawah!

**Jawaban:**

**Modifikasi kode**

**Kode program class Mahasiswa03.java**

|  |
| --- |
| public class Mahasiswa03 {      String nim, nama, kelas;      int nilai;      Mahasiswa03() {          this.nim = "";          this.nama = "";          this.kelas = "";          this.nilai = -1;      }      Mahasiswa03(String nim, String nama, String kelas) {          this.nim = nim;          this.nama = nama;          this.kelas = kelas;          this.nilai = -1;      }      void tugasDinilai(int nilai) {          this.nilai = nilai;      }  } |

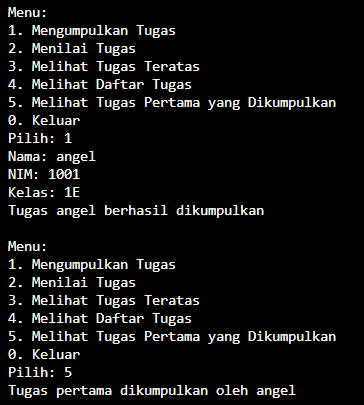
**Kode program class StackTugasMahasiswa03.java**

|  |
| --- |
| public class StackTugasMahasiswa03 {      Mahasiswa03[] stack;      int size, top;      StackTugasMahasiswa03(int size) {          this.size = size;          this.stack = new Mahasiswa03[size];          this.top = -1;      }      boolean isFull() {          return top == size - 1;      }      boolean isEmpty() {          return top == -1;      }      void push(Mahasiswa03 mhs) {          if (!isFull()) {              stack[++top] = mhs;          } else {              System.out.println("Stack penuh. Tidak dapat menambahkan tugas.");          }      }      Mahasiswa03 pop() {          if (!isEmpty()) {              return stack[top--];          } else {              System.out.println("Stack kosong. Tidak ada tugas yang bisa dinilai.");              return null;          }      }      Mahasiswa03 peek() {          if (!isEmpty()) {              return stack[top];          } else {              System.out.println("Stack kosong.");              return null;          }      }      void print() {          if (!isEmpty()) {              for (int i = top; i >= 0; i--) {                  System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);              }          } else {              System.out.println("Stack kosong.");          }      }      Mahasiswa03 bottom() {          if (!isEmpty()) {              return stack[0];          } else {              System.out.println("Stack kosong.");              return null;          }      }    } |

**Kode program class MahasiswaDemo03.java**

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaDemo03 {      public static void main(String[] args) {          StackTugasMahasiswa03 stack = new StackTugasMahasiswa03(5);          Scanner scan = new Scanner(System.in);          int pilih;          do {              System.out.println("\nMenu:");              System.out.println("1. Mengumpulkan Tugas");              System.out.println("2. Menilai Tugas");              System.out.println("3. Melihat Tugas Teratas");              System.out.println("4. Melihat Daftar Tugas");              System.out.println("5. Melihat Tugas Pertama yang Dikumpulkan");              System.out.println("0. Keluar");              System.out.print("Pilih: ");              pilih = scan.nextInt();              scan.nextLine();              switch (pilih) {                  case 1:                      System.out.print("Nama: ");                      String nama = scan.nextLine();                      System.out.print("NIM: ");                      String nim = scan.nextLine();                      System.out.print("Kelas: ");                      String kelas = scan.nextLine();                      Mahasiswa03 mhs = new Mahasiswa03(nim, nama, kelas);                      stack.push(mhs);                      System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);                      break;                  case 2:                      Mahasiswa03 dinilai = stack.pop();                      if (dinilai != null) {                          System.out.println("Menilai tugas dari " + dinilai.nama);                          System.out.print("Masukkan nilai (0-100): ");                          int nilai = scan.nextInt();                          dinilai.tugasDinilai(nilai);                          System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);                      }                      break;                  case 3:                      Mahasiswa03 lihat = stack.peek();                      if (lihat != null) {                          System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);                      }                      break;                  case 4:                      System.out.println("Daftar semua tugas:");                      System.out.println("Nama\tNIM\tKelas");                      stack.print();                      break;                    case 5:                      Mahasiswa03 terbawah = stack.bottom();                      if (terbawah != null) {                          System.out.println("Tugas pertama dikumpulkan oleh " + terbawah.nama);                      }                      break;                    case 0:                      System.out.println("Terima kasih.");                      break;                  default:                      System.out.println("Pilihan tidak valid.");              }          } while (pilih != 0);      }  } |

**Hasil kode**

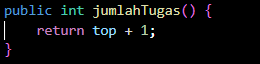


1. Tambahkan method untuk dapat menghitung berapa banyak tugas yang sudah dikumpulkan saat ini, serta tambahkan operasi menunya!

**Jawaban :**

**modifikasi kode**

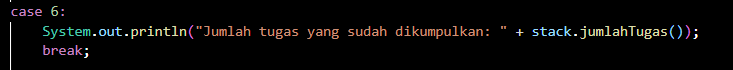
Tambahkan method jumlahTugas pada class StackTugasMahasiswa03.java



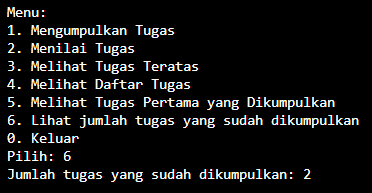
Tambahkan menu operasi di class MahasiswaDemo03.java



Tambahkan class handling di dalam blok switch

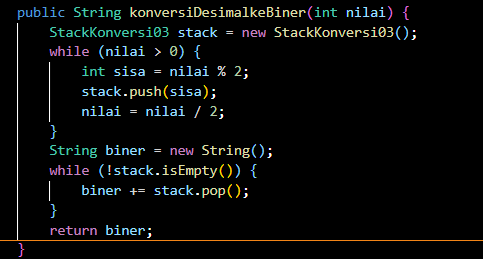


**Hasil kode**

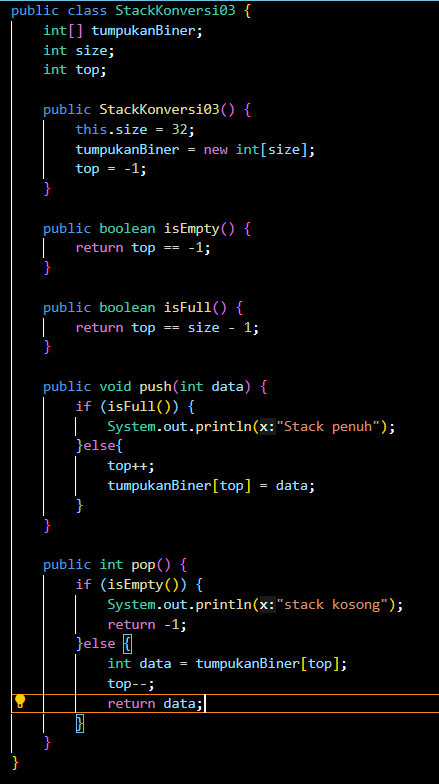


**Percobaan 2**

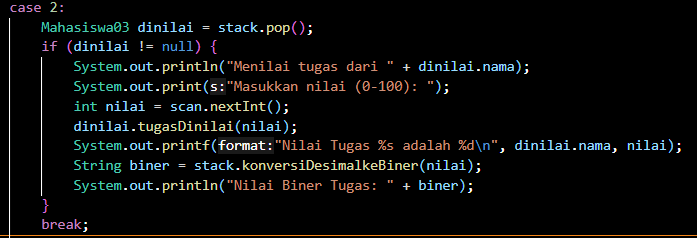
1. Buka kembali file StackTugasMahasiwa03.java
2. Tambahkan method konversiDesimalKeBiner dengan menerima parameter kode bertipe int



1. Tambahkan empat method yaitu **isEmpty**, **isFull**, **push**, dan **pull** sebagai operasi utama Stack pada class **StackKonversi**

****

1. Agar nilai tugas mahasiswa dikonversi ke dalam bentuk biner setelah dilakukan penilaian, maka tambahkan baris kode program pada method pop di class MahasiswaDemo

****

**Kode program**

**Class Mahasiswa03.java**

|  |
| --- |
| public class Mahasiswa03 {      String nim, nama, kelas;      int nilai;      Mahasiswa03() {          this.nim = "";          this.nama = "";          this.kelas = "";          this.nilai = -1;      }      Mahasiswa03(String nim, String nama, String kelas) {          this.nim = nim;          this.nama = nama;          this.kelas = kelas;          this.nilai = -1;      }      void tugasDinilai(int nilai) {          this.nilai = nilai;      }  } |

**Class StackTugasMahasiswa03.java**

|  |
| --- |
| public class StackTugasMahasiswa03 {      Mahasiswa03[] stack;      int size, top;      StackTugasMahasiswa03(int size) {          this.size = size;          this.stack = new Mahasiswa03[size];          this.top = -1;      }      boolean isFull() {          return top == size - 1;      }      boolean isEmpty() {          return top == -1;      }      void push(Mahasiswa03 mhs) {          if (!isFull()) {              stack[++top] = mhs;          } else {              System.out.println("Stack penuh. Tidak dapat menambahkan tugas.");          }      }      Mahasiswa03 pop() {          if (!isEmpty()) {              return stack[top--];          } else {              System.out.println("Stack kosong. Tidak ada tugas yang bisa dinilai.");              return null;          }      }      Mahasiswa03 peek() {          if (!isEmpty()) {              return stack[top];          } else {              System.out.println("Stack kosong.");              return null;          }      }      void print() {          if (!isEmpty()) {              for (int i = top; i >= 0; i--) {                  System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);              }          } else {              System.out.println("Stack kosong.");          }      }      Mahasiswa03 bottom() {          if (!isEmpty()) {              return stack[0];          } else {              System.out.println("Stack kosong.");              return null;          }      }      public int jumlahTugas() {          return top + 1;      }          public String konversiDesimalkeBiner(int nilai) {          StackKonversi03 stack = new StackKonversi03();          while (nilai > 0) {              int sisa = nilai % 2;              stack.push(sisa);              nilai = nilai / 2;          }          String biner = new String();          while (!stack.isEmpty()) {              biner += stack.pop();          }          return biner;      }  } |

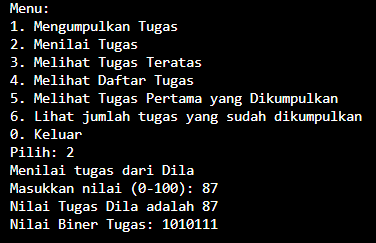
**Class StackKonversi03.java**

|  |
| --- |
| public class StackKonversi03 {      int[] tumpukanBiner;      int size;      int top;      public StackKonversi03() {          this.size = 32;          tumpukanBiner = new int[size];          top = -1;      }      public boolean isEmpty() {          return top == -1;      }      public boolean isFull() {          return top == size - 1;      }      public void push(int data) {          if (isFull()) {              System.out.println("Stack penuh");          }else{              top++;              tumpukanBiner[top] = data;          }      }      public int pop() {          if (isEmpty()) {              System.out.println("stack kosong");              return -1;          }else {              int data = tumpukanBiner[top];              top--;              return data;          }      }  } |

**Class MahasiswaDemo03.java**

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaDemo03 {      public static void main(String[] args) {          StackTugasMahasiswa03 stack = new StackTugasMahasiswa03(5);          Scanner scan = new Scanner(System.in);          int pilih;          do {              System.out.println("\nMenu:");              System.out.println("1. Mengumpulkan Tugas");              System.out.println("2. Menilai Tugas");              System.out.println("3. Melihat Tugas Teratas");              System.out.println("4. Melihat Daftar Tugas");              System.out.println("5. Melihat Tugas Pertama yang Dikumpulkan");              System.out.println("6. Lihat jumlah tugas yang sudah dikumpulkan");              System.out.println("0. Keluar");              System.out.print("Pilih: ");              pilih = scan.nextInt();              scan.nextLine();              switch (pilih) {                  case 1:                      System.out.print("Nama: ");                      String nama = scan.nextLine();                      System.out.print("NIM: ");                      String nim = scan.nextLine();                      System.out.print("Kelas: ");                      String kelas = scan.nextLine();                      Mahasiswa03 mhs = new Mahasiswa03(nim, nama, kelas);                      stack.push(mhs);                      System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);                      break;                  case 2:                      Mahasiswa03 dinilai = stack.pop();                      if (dinilai != null) {                          System.out.println("Menilai tugas dari " + dinilai.nama);                          System.out.print("Masukkan nilai (0-100): ");                          int nilai = scan.nextInt();                          dinilai.tugasDinilai(nilai);                          System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);                          String biner = stack.konversiDesimalkeBiner(nilai);                          System.out.println("Nilai Biner Tugas: " + biner);                      }                      break;                  case 3:                      Mahasiswa03 lihat = stack.peek();                      if (lihat != null) {                          System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);                      }                      break;                  case 4:                      System.out.println("Daftar semua tugas:");                      System.out.println("Nama\tNIM\tKelas");                      stack.print();                      break;                    case 5:                      Mahasiswa03 terbawah = stack.bottom();                      if (terbawah != null) {                          System.out.println("Tugas pertama dikumpulkan oleh " + terbawah.nama);                      }                      break;                    case 6:                      System.out.println("Jumlah tugas yang sudah dikumpulkan: " + stack.jumlahTugas());                      break;                    case 0:                      System.out.println("Terima kasih.");                      break;                  default:                      System.out.println("Pilihan tidak valid.");              }          } while (pilih != 0);      }  } |

**Hasil kode program**



**Pertanyaan**

1. Jelaskan alur kerja dari method konversiDesimalKeBiner!

**Jawaban:**

* 1. Nilai desimal dimasukkan ke method.
  2. Selama nilai desimal lebih dari 0:
     1. Nilai dibagi 2, sisa pembagiannya (0 atau 1) dimasukkan ke dalam stack.
     2. Nilai dibagi 2 (integer division).
  3. Setelah semua sisa dimasukkan, angka biner dicetak dengan mengambil (pop) elemen dari stack (sehingga urutannya benar).
  4. Hasil akhirnya adalah representasi biner dari nilai desimal yang dimasukkan.

1. Pada method konversiDesimalKeBiner, ubah kondisi perulangan menjadi while (kode != 0), bagaimana hasilnya? Jelaskan alasannya!

**Jawaban :**

Hasilnya: Tidak ada perubahan dalam output, program tetap berfungsi dengan baik.

**Alasannya:**  
Kondisi kode > 0 dan kode != 0 menghasilkan perilaku yang sama selama nilai kode selalu positif, karena:

1. Untuk bilangan bulat positif: kode > 0 ≡ kode != 0
2. Perbedaannya baru terasa jika ada kemungkinan nilai negatif (yang dalam konteks konversi bilangan biner tidak relevan, karena nilai tugas selalu positif).

**Sehingga, hasil program tidak berubah.**

**Kode program lengkap**

**Class Mahasiswa03.java**

|  |
| --- |
| public class Mahasiswa03 {      String nim, nama, kelas;      int nilai;      Mahasiswa03() {          this.nim = "";          this.nama = "";          this.kelas = "";          this.nilai = -1;      }      Mahasiswa03(String nim, String nama, String kelas) {          this.nim = nim;          this.nama = nama;          this.kelas = kelas;          this.nilai = -1;      }      void tugasDinilai(int nilai) {          this.nilai = nilai;      }  } |

**Class StackTugasMahasiswa03.java**

|  |
| --- |
| public class StackTugasMahasiswa03 {      Mahasiswa03[] stack;      int size, top;      StackTugasMahasiswa03(int size) {          this.size = size;          this.stack = new Mahasiswa03[size];          this.top = -1;      }      boolean isFull() {          return top == size - 1;      }      boolean isEmpty() {          return top == -1;      }      void push(Mahasiswa03 mhs) {          if (!isFull()) {              stack[++top] = mhs;          } else {              System.out.println("Stack penuh. Tidak dapat menambahkan tugas.");          }      }      Mahasiswa03 pop() {          if (!isEmpty()) {              return stack[top--];          } else {              System.out.println("Stack kosong. Tidak ada tugas yang bisa dinilai.");              return null;          }      }      Mahasiswa03 peek() {          if (!isEmpty()) {              return stack[top];          } else {              System.out.println("Stack kosong.");              return null;          }      }      void print() {          if (!isEmpty()) {              for (int i = top; i >= 0; i--) {                  System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);              }          } else {              System.out.println("Stack kosong.");          }      }      Mahasiswa03 bottom() {          if (!isEmpty()) {              return stack[0];          } else {              System.out.println("Stack kosong.");              return null;          }      }      public int jumlahTugas() {          return top + 1;      }          public String konversiDesimalkeBiner(int nilai) {          StackKonversi03 stack = new StackKonversi03();          while (nilai > 0) {              int sisa = nilai % 2;              stack.push(sisa);              nilai = nilai / 2;          }          String biner = new String();          while (!stack.isEmpty()) {              biner += stack.pop();          }          return biner;      }  } |

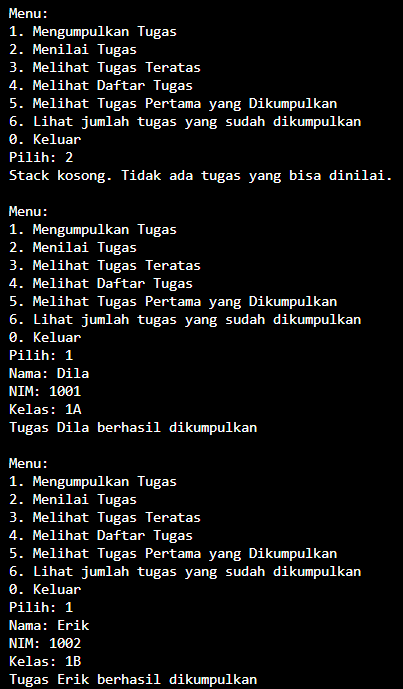
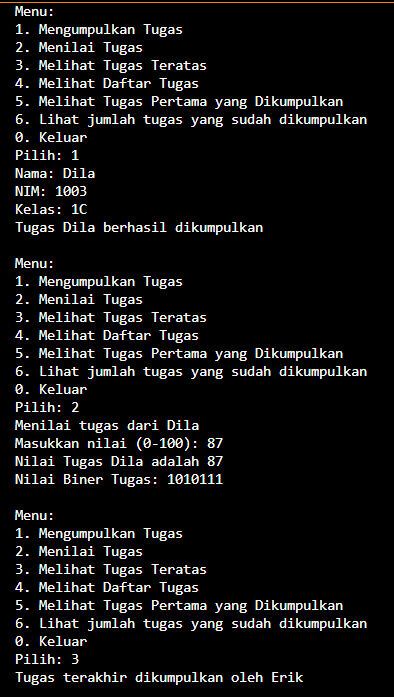
**Class StackKonversi03.java**

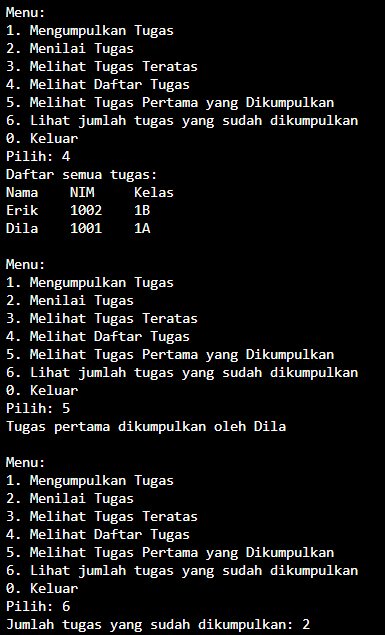
|  |
| --- |
| public class StackKonversi03 {      int[] tumpukanBiner;      int size;      int top;      public StackKonversi03() {          this.size = 32;          tumpukanBiner = new int[size];          top = -1;      }      public boolean isEmpty() {          return top == -1;      }      public boolean isFull() {          return top == size - 1;      }      public void push(int data) {          if (isFull()) {              System.out.println("Stack penuh");          }else{              top++;              tumpukanBiner[top] = data;          }      }      public int pop() {          if (isEmpty()) {              System.out.println("stack kosong");              return -1;          }else {              int data = tumpukanBiner[top];              top--;              return data;          }      }  } |

**Class MahasiswaDemo03.java**

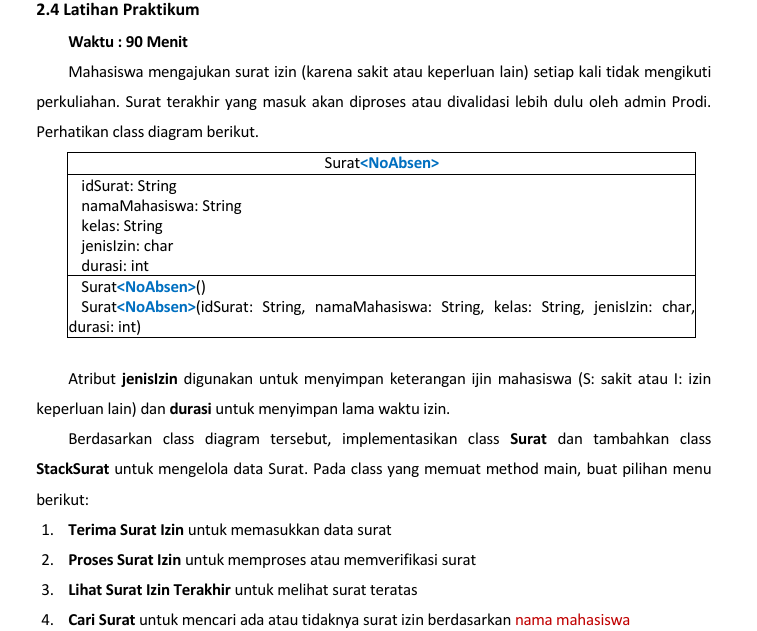
|  |
| --- |
| import java.util.Scanner;  public class MahasiswaDemo03 {      public static void main(String[] args) {          StackTugasMahasiswa03 stack = new StackTugasMahasiswa03(5);          Scanner scan = new Scanner(System.in);          int pilih;          do {              System.out.println("\nMenu:");              System.out.println("1. Mengumpulkan Tugas");              System.out.println("2. Menilai Tugas");              System.out.println("3. Melihat Tugas Teratas");              System.out.println("4. Melihat Daftar Tugas");              System.out.println("5. Melihat Tugas Pertama yang Dikumpulkan");              System.out.println("6. Lihat jumlah tugas yang sudah dikumpulkan");              System.out.println("0. Keluar");              System.out.print("Pilih: ");              pilih = scan.nextInt();              scan.nextLine();              switch (pilih) {                  case 1:                      System.out.print("Nama: ");                      String nama = scan.nextLine();                      System.out.print("NIM: ");                      String nim = scan.nextLine();                      System.out.print("Kelas: ");                      String kelas = scan.nextLine();                      Mahasiswa03 mhs = new Mahasiswa03(nim, nama, kelas);                      stack.push(mhs);                      System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);                      break;                  case 2:                      Mahasiswa03 dinilai = stack.pop();                      if (dinilai != null) {                          System.out.println("Menilai tugas dari " + dinilai.nama);                          System.out.print("Masukkan nilai (0-100): ");                          int nilai = scan.nextInt();                          dinilai.tugasDinilai(nilai);                          System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);                          String biner = stack.konversiDesimalkeBiner(nilai);                          System.out.println("Nilai Biner Tugas: " + biner);                      }                      break;                  case 3:                      Mahasiswa03 lihat = stack.peek();                      if (lihat != null) {                          System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);                      }                      break;                  case 4:                      System.out.println("Daftar semua tugas:");                      System.out.println("Nama\tNIM\tKelas");                      stack.print();                      break;                    case 5:                      Mahasiswa03 terbawah = stack.bottom();                      if (terbawah != null) {                          System.out.println("Tugas pertama dikumpulkan oleh " + terbawah.nama);                      }                      break;                    case 6:                      System.out.println("Jumlah tugas yang sudah dikumpulkan: " + stack.jumlahTugas());                      break;                    case 0:                      System.out.println("Terima kasih.");                      break;                  default:                      System.out.println("Pilihan tidak valid.");              }          } while (pilih != 0);      }  } |

**Hasil kode program lengkap**

**** 

****

**TUGAS**

****

**Kode program class Surat03.java**

|  |
| --- |
| pnublic class Surat03 {      String idSurat;      String namaMahasiswa;      String kelas;      char jenisIzin;      int durasi;      public Surat03() {}      public Surat03(String idSurat, String namaMahasiswa, String kelas, char jenisIzin, int durasi) {          this.idSurat = idSurat;          this.namaMahasiswa = namaMahasiswa;          this.kelas = kelas;          this.jenisIzin = jenisIzin;          this.durasi = durasi;      }      public String toString() {          String jenis = (jenisIzin == 'S' || jenisIzin == 's') ? "Sakit" : "Izin Lain";          return idSurat + " - " + namaMahasiswa + " - " + kelas + " - " + jenis + " - Durasi: " + durasi + " hari";      }  } |

**Kode program class StackSurat03.java**

|  |
| --- |
| public class StackSurat03 {      Surat03[] stack;      int size;      int top;      public StackSurat03(int size) {          this.size = size;          stack = new Surat03[size];          top = -1;      }      public boolean isFull() {          return top == size - 1;      }      public boolean isEmpty() {          return top == -1;      }      public void push(Surat03 surat) {          if (!isFull()) {              stack[++top] = surat;          } else {              System.out.println("Stack surat penuh!");          }      }      public Surat03 pop() {          if (!isEmpty()) {              return stack[top--];          } else {              System.out.println("Tidak ada surat yang diproses.");              return null;          }      }      public Surat03 peek() {          if (!isEmpty()) {              return stack[top];          } else {              System.out.println("Stack kosong.");              return null;          }      }      public void print() {          if (!isEmpty()) {              System.out.println("Daftar Surat Izin:");              for (int i = top; i >= 0; i--) {                  System.out.println(stack[i]);              }          } else {              System.out.println("Belum ada surat.");          }      }      public void searchByName(String nama) {          boolean found = false;          for (int i = top; i >= 0; i--) {              if (stack[i].namaMahasiswa.equalsIgnoreCase(nama)) {                  System.out.println("Surat ditemukan: " + stack[i]);                  found = true;              }          }          if (!found) {              System.out.println("Surat atas nama " + nama + " tidak ditemukan.");          }      }  } |

**Kode program class SuratDemo03.java**

|  |
| --- |
| import java.util.Scanner;  public class SuratDemo03 {      public static void main(String[] args) {          Scanner sc = new Scanner(System.in);          StackSurat03 stack = new StackSurat03(10);          int pilih;          do {              System.out.println("\n===== MENU SURAT IZIN =====");              System.out.println("1. Terima Surat Izin");              System.out.println("2. Proses Surat Izin");              System.out.println("3. Lihat Surat Terakhir");              System.out.println("4. Cari Surat berdasarkan Nama");              System.out.println("0. Keluar");              System.out.print("Pilih menu: ");              pilih = sc.nextInt(); sc.nextLine();              switch (pilih) {                  case 1:                      System.out.print("ID Surat         : ");                      String id = sc.nextLine();                      System.out.print("Nama Mahasiswa   : ");                      String nama = sc.nextLine();                      System.out.print("Kelas            : ");                      String kelas = sc.nextLine();                      System.out.print("Jenis Izin (S/I) : ");                      char jenis = sc.next().charAt(0);                      System.out.print("Durasi (hari)    : ");                      int durasi = sc.nextInt(); sc.nextLine();                      Surat03 surat = new Surat03(id, nama, kelas, jenis, durasi);                      stack.push(surat);                      break;                  case 2:                      Surat03 s = stack.pop();                      if (s != null) {                          System.out.println("Surat yang diproses:");                          System.out.println(s);                      }                      break;                  case 3:                      Surat03 terakhir = stack.peek();                      if (terakhir != null) {                          System.out.println("Surat Terakhir:");                          System.out.println(terakhir);                      }                      break;                  case 4:                      System.out.print("Masukkan nama mahasiswa: ");                      String cariNama = sc.nextLine();                      stack.searchByName(cariNama);                      break;                  case 0:                      System.out.println("Program selesai.");                      break;                  default:                      System.out.println("Pilihan tidak valid.");              }          } while (pilih != 0);      }  } |

**Hasil kode**

