

UAS ALGORITMA DAN STRUKTUR DATA



Oleh:
DZULFIKAR MUHAMMAD AL GHIFARI
NIM. 2341760071
SIB-1F / 08

D-IV SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

Link : <https://github.com/dzulfikar05/PraktikumAlgoritma/tree/main/uas>
Class Kendaraan

```
public class Kendaraan08 {
    String noTNKB, nama, jenis;
    int cc, tahun, bulanHarusBayar;

    Kendaraan08(String noTNKB, String nama, String jenis, int cc, int tahun, int
    bulanHarusBayar) {
        this.noTNKB = noTNKB;
        this.nama = nama;
        this.jenis = jenis;
        this.cc = cc;
        this.tahun = tahun;
        this.bulanHarusBayar = bulanHarusBayar;
    }
}
```

Class Transaksi Pajak

```
public class TransaksiPajak08 {
    int kode, bulanBayar;
    Long nominalBayar, denda;
    Kendaraan08 kendaraan08;

    TransaksiPajak08(int kode, Long nominalBayar, Long denda, int bulanBayar,
    Kendaraan08 kendaraan08) {
        this.kode = kode;
        this.nominalBayar = nominalBayar;
        this.denda = denda;
        this.bulanBayar = bulanBayar;
        this.kendaraan08 = kendaraan08;
    }
}
```

Class Pajak

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;
import java.util.Scanner;

Codeium: Refactor | Explain
public class Pajak08 {
    List<Kendaraan08> kendaraan08 = new ArrayList<>();
    List<TransaksiPajak08> transaksiPajak08 = new ArrayList<>();

    Codeium: Refactor | Explain | Generate Javadoc | X
    void tambahKendaraan(String noTNKB, String nama, String jenis, int cc, int
    tahun, int bulanHarusBayar) {
        Kendaraan08 kendaraan = new Kendaraan08( noTNKB, nama, jenis, cc, tahun,
        bulanHarusBayar);
        kendaraan08.add(kendaraan);
    }

    Codeium: Refactor | Explain | Generate Javadoc | X
    void tampilKendaraan() {
        System.out.println(x:"=====");
        System.out.println(x:"          DAFTAR Kendaraan          ");
        System.out.println(x:"=====");
        System.out.println(x:"Nomor TNKB |Nama Pemilik | Jenis\t | CC Kendaraan |
        Tahun | Bulan Harus Bayar");
        for (Kendaraan08 kendaraan : kendaraan08) {
            System.out.println(kendaraan.noTNKB+" \t"+kendaraan.nama+"\t\t"
            +kendaraan.jenis+"\t\t"+kendaraan.cc+"\t\t"+kendaraan.tahun+"\t\t"
            +kendaraan.bulanHarusBayar+"\t\t");
        }
        System.out.println(x:"-----");
    }
}
```

```

void bayarPajak() {
    Scanner input = new Scanner(System.in);
    System.out.println(x:"=====");
    System.out.println(x:"          BAYAR PAJAK KENDARAAN          ");
    System.out.println(x:"=====");

    System.out.print(s:"Masukkan Nomor TNKB: ");
    String notNKB = input.nextLine();

    System.out.print(s:"Masukkan Bulan Bayar: ");
    int bulanBayar = input.nextInt();
    input.nextLine();

    Kendaraan08 kendaraan = cariKendaraan(notNKB);
    if (kendaraan != null) {
        Long nominalBayar = hitungPajak(kendaraan, bulanBayar);
        Long denda = hitungDenda(bulanBayar, kendaraan.bulanHarusBayar);

        System.out.println(x:"=====");
        System.out.println(x:"          RINCIAN PEMBAYARAN          ");
        System.out.println(x:"=====");
        System.out.println("Nomor TNKB          : " + kendaraan.notNKB);
        System.out.println("Nama Pemilik          : " + kendaraan.nama);
        System.out.println("Jenis Kendaraan       : " + kendaraan.jenis);
        System.out.println("CC Kendaraan          : " + kendaraan.cc);
        System.out.println("Bulan Harus Bayar     : " + kendaraan.bulanHarusBayar);
        System.out.println("Bulan Bayar           : " + bulanBayar);
        System.out.println("Nominal Pajak         : Rp" + nominalBayar);
        System.out.println("Denda                 : Rp" + denda);
        System.out.println("Total Bayar           : Rp" + (nominalBayar + denda));
        System.out.println(x:"-----");

        System.out.print(s:"Konfirmasi pembayaran? (Y/N): ");
        String konfirmasi = input.nextLine().toUpperCase();

        if (konfirmasi.equals(anObject:"Y")) {
            TransaksiPajak08 transaksiPajak = new TransaksiPajak08(kode:1,
            nominalBayar, denda, bulanBayar, kendaraan);
            transaksiPajak08.add(transaksiPajak);
            System.out.println(x:"Pembayaran pajak berhasil.");
        } else {
            System.out.println(x:"Pembayaran pajak dibatalkan.");
        }
    } else {
        System.out.println("Kendaraan dengan nomor TNKB " + notNKB + " tidak
        ditemukan.");
    }
}

```

```

private Kendaraan08 cariKendaraan(String notNKB) {
    for (Kendaraan08 kendaraan : kendaraan08) {
        if (kendaraan.notNKB.equals(notNKB)) {
            return kendaraan;
        }
    }
    return null;
}

Codeium: Refactor | Explain | Generate Javadoc | X
Long hitungPajak(Kendaraan08 kendaraan, int bulanBayar) {
    int tarifPajak = 0;

    if (kendaraan.jenis.equalsIgnoreCase(anotherString:"Motor")) {
        if (kendaraan.cc < 100) {
            tarifPajak = 100000;
        } else if (kendaraan.cc <= 250) {
            tarifPajak = 250000;
        } else {
            tarifPajak = 500000;
        }
    } else if (kendaraan.jenis.equalsIgnoreCase(anotherString:"Mobil")) {
        if (kendaraan.cc < 1000) {
            tarifPajak = 750000;
        } else if (kendaraan.cc <= 2500) {
            tarifPajak = 1000000;
        } else {
            tarifPajak = 1500000;
        }
    }
    return tarifPajak;
}

Codeium: Refactor | Explain | Generate Javadoc | X
public Long hitungDenda(int bulanBayar, int bulanHarusBayar) {
    int denda = 0;

    if (bulanBayar > bulanHarusBayar) {
        int telatBulan = bulanBayar - bulanHarusBayar;

        if (telatBulan <= 3) {
            denda = 50000 * telatBulan;
        } else {
            denda = 50000 * 3;
            denda += (telatBulan - 3) * 50000;
        }
    }

    return denda;
}

```

```

public void tampilSemuaTransaksi() {
    if (transaksiPajak08.isEmpty()) {
        System.out.println(x:"Belum ada data transaksi pajak.");
    } else {
        System.out.println
        (x:"=====");
        System.out.println(x:"          DAFTAR TRANSAKSI
PAJAK          ");
        System.out.println
        (x:"=====");
        System.out.println(x:"Kode | TNKB | Nama Pemilik | Nominal | Denda
");
        for (TransaksiPajak08 transaksiPajak : transaksiPajak08) {
            System.out.println(transaksiPajak.kode + " \t " + transaksiPajak.
kendaraan08.notTNKB + " \t " + transaksiPajak.kendaraan08.nama + "\t
\t " + transaksiPajak.nominalBayar + " \t " + transaksiPajak.denda);
        }
        System.out.println
        (x:"=====");
    }
}

Codeium: Refactor | Explain | Generate Javadoc | X
public void tampilTransaksiPemiliki() {
    if (transaksiPajak08.isEmpty()) {
        System.out.println(x:"Belum ada data transaksi pajak.");
    } else {
        Collections.sort(transaksiPajak08, new Comparator<TransaksiPajak08>() {
            Codeium: Refactor | Explain | Generate Javadoc | X
            @Override
            public int compare(TransaksiPajak08 transaksi1, TransaksiPajak08
transaksi2) {
                return transaksi1.kendaraan08.nama.compareToIgnoreCase
(transaksi2.kendaraan08.nama);
            }
        });

        System.out.println
        (x:"=====");
        System.out.println(x:" DAFTAR TRANSAKSI PEMBAYARAN PAJAK ");
        System.out.println
        (x:"=====");
        System.out.println(x:"Kode | TNKB | Nama Pemilik | Nominal Bayar |
Denda");
        for (TransaksiPajak08 transaksiPajak : transaksiPajak08) {
            System.out.println(transaksiPajak.kode + " \t " + transaksiPajak.
kendaraan08.notTNKB + " \t " + transaksiPajak.kendaraan08.nama + "\t
\t " + transaksiPajak.nominalBayar + " \t " + transaksiPajak.denda);
        }
        System.out.println
        (x:"=====");
    }
}

```


Class Main

```
import java.util.Scanner;

Codeium: Refactor | Explain
public class Main {
    Run | Debug | Codeium: Refactor | Explain | Generate Javadoc | X
    public static void main(String[] args) {
        Pajak008 pajak = new Pajak008();
        Scanner input = new Scanner(System.in);
        int pilihan;

        // insert data mahasiswa
        pajak.tambahKendaraan(notTNKB:"N 001 C", nama:"abdur", jenis:"mobil",
            cc:2010, tahun:2001, bulanHarusBayar:4);
        pajak.tambahKendaraan(notTNKB:"B 002 C", nama:"kode", jenis:"motor", cc:100,
            tahun:2009, bulanHarusBayar:5);
        pajak.tambahKendaraan(notTNKB:"S 003 C", nama:"khodir", jenis:"mobil",
            cc:2500, tahun:2001, bulanHarusBayar:8);
        pajak.tambahKendaraan(notTNKB:"D 004 C", nama:"dudung", jenis:"motor",
            cc:200, tahun:2012, bulanHarusBayar:1);
        pajak.tambahKendaraan(notTNKB:"D 005 C", nama:"dono", jenis:"mobil", cc:1900,
            tahun:2005, bulanHarusBayar:3);

        do {
            System.out.println(x:"=====");
            System.out.println(x:" MENU [dzulfikar / 1FSIB - 08] ");
            System.out.println(x:"=====");
            System.out.println(x:"1. Daftar Kendaraan");
            System.out.println(x:"2. Bayar Pajak");
            System.out.println(x:"3. Tampil seluruh transaksi");
            System.out.println(x:"4. Urutkan transaksi berdasar nama pemilik");
            System.out.println(x:"5. Keluar");
            System.out.print(s:"Pilih: ");
            pilihan = input.nextInt();
            input.nextLine();

            switch (pilihan) {
                case 1:
                    pajak.tampilKendaraan();
                    break;
                case 2:
                    pajak.bayarPajak();
                    break;
                case 3:
                    pajak.tampilSemuaTransaksi();
                    break;
                case 4:
                    pajak.tampilTransaksiPemiliki();
                    break;
                case 5:
                    return;
            }
        } while (pilihan != 5);
    }
}
```

```
default:
    System.out.println(x:"Pilih antara 1-5.");
} while (pilihan != 5);
}
```

Hasil Compile

```
=====
MENU [dzulfikar / 1FSIB - 08]
=====
1. Daftar Kendaraan
2. Bayar Pajak
3. Tampil seluruh transaksi
4. Urutkan transaksi berdasar nama pemilik
5. Keluar
Pilih: 1

=====
Pilih: 1
=====
          DAFTAR Kendaraan
=====
Nomor TNKB | Nama Pemilik | Jenis      | CC Kendaraan | Tahun | Bulan Harus Bayar
N 001 C    | abdur          | mobil     | 2010         | 2001  | 4
B 002 C    | kode           | motor     | 100          | 2009  | 5
S 003 C    | khodir         | mobil     | 2500         | 2001  | 8
D 004 C    | dudung         | motor     | 200          | 2012  | 1
D 005 C    | dono           | mobil     | 1900         | 2005  | 3
=====
```

```
=====
Pilih: 2
=====
          BAYAR PAJAK KENDARAAN
=====
Masukkan Nomor TNKB: N 001 C
Masukkan Bulan Bayar: 5
=====
          RINCIAN PEMBAYARAN
=====
Nomor TNKB      : N 001 C
Nama Pemilik    : abdur
Jenis Kendaraan : mobil
CC Kendaraan    : 2010
Bulan Harus Bayar : 4
Bulan Bayar     : 5
Nominal Pajak   : Rp1000000
Denda           : Rp500000
Total Bayar     : Rp1050000
=====
Konfirmasi pembayaran? (Y/N): y
Pembayaran pajak berhasil.
=====
```

```
=====
Pilih: 3
=====
          DAFTAR TRANSAKSI PAJAK
=====
Kode | TNKB | Nama Pemilik | Nominal | Denda
1    | N 001 C | abdur        | 1000000 | 50000
2    | D 005 C | dono         | 1000000 | 0
3    | B 002 C | kode         | 250000  | 250000
=====
```

```
=====
Pilih: 4
=====
          DAFTAR TRANSAKSI PEMBAYARAN PAJAK
=====
Kode | TNKB | Nama Pemilik | Nominal Bayar | Denda
1    | N 001 C | abdur        | 1000000        | 50000
2    | D 005 C | dono         | 1000000        | 0
3    | B 002 C | kode         | 250000         | 250000
=====
```