**PEMROGRAMAN BERBASIS WEB**

**(Linked List Program Parkir)**

****

**Disusun Oleh:**

Amir Salim – 140810210015

Andre Nathaniel Adipraja – 140810200042

Prames Ray Lapian – 140810210059

Ibrahim Dafi Iskandar – 140810210039

**PROGRAM STUDI S-1 TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PADJADJARAN**

**JATINANGOR**

**2022**

1. Java Language
   1. Source Code:
      1. Class Waktu:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : ElementList.java*

*\* Deskripsi     : Soal parkir - class Waktu*

*\*/*

package Waktu;

import java.util.Scanner;

*public* class Waktu {

*private* int jam, menit, detik;

*public* Waktu(int *jam*, int *menit*, int *detik*){

        this.jam = *jam*;

        this.menit = *menit*;

        this.detik = *detik*;

    }

*public* Waktu(){

        this.jam = 0;

        this.menit = 0;

        this.detik = 0;

    }

*//Input*

*public* void setJam(int *jam*){

        this.jam = *jam*;

    }

*public* void setMenit(int *menit*){

        this.menit = *menit*;

    }

*public* void setDetik(int *detik*){

        this.detik = *detik*;

    }

*public* void inputJam(){

        Scanner input = *new* Scanner(System.in);

        System.out.print("Masukkan jam   : ");

        this.jam = input.nextInt();

        System.out.print("Masukkan menit : ");

        this.menit = input.nextInt();

        System.out.print("Masukkan detik : ");

        this.detik = input.nextInt();

    }

*//Output*

*public* int getJam(){

*return* this.jam;

    }

*public* int getMenit(){

*return* this.menit;

    }

*public* int getDetik(){

*return* this.detik;

    }

*public* String getWaktu(){

        String nolJam ="", nolMenit="", nolDetik="";

*if*(this.jam < 10){

            nolJam = "0";

        }

*if*(this.menit < 10){

            nolMenit = "0";

        }

*if*(this.detik < 10){

            nolDetik = "0";

        }

*return* nolJam + this.jam + ":" + nolMenit + this.menit + ":" + nolDetik + this.detik;

    }

*//Proses*

*public* int convertToSecond(){

        int hasil = this.detik + this.menit\*60 + this.jam\*3600;

*return* hasil;

    }

*public* void secondToClock(int *second*){

        this.menit = *second* / 60;

        this.detik = *second* % 60;

        this.jam = this.menit / 60;

        this.menit=this.menit % 60;

    }

*public* Waktu cariDurasi(Waktu *akhir*){

        Waktu temp =  *new* Waktu();

        int detikAwal = this.convertToSecond();

        int detikAkhir = *akhir*.convertToSecond();

*if*(detikAkhir < detikAwal){

            detikAkhir += 86400;

        }

        int detikHasil = detikAkhir - detikAwal;

        temp.secondToClock(detikHasil);

*return* temp;

    }

}

* + 1. Class Kendaraan:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : Kendaraan.java*

*\* Deskripsi     : Soal parkir - class Kendaraan*

*\*/*

package Kendaraan;

import Waktu.Waktu;

import java.util.Scanner;

*public* *abstract* class Kendaraan {

*protected* String no;

*protected* String jenis;

*protected* Waktu datang = *new* Waktu();

*protected* Waktu pulang = *new* Waktu();

*public* Kendaraan(){

            this.no = " ";

            this.jenis= " ";

        }

*//Input*

*public* void setNoKendaraan(String *no*){

            this.no=*no*;

        }

*public* void setJenis(String *jenis*){

            this.jenis=*jenis*;

        }

*public* void setWaktudatang(Waktu *datang*){

            this.datang=*datang*;

        }

*public* void setWaktuPulang(Waktu *pulang*){

            this.pulang=*pulang*;

        }

*public* void inputKendaraan(){

            Scanner input=*new* Scanner(System.in);

            System.out.println("\n--- INPUT KENDARAAN ---");

            System.out.print("No Kendaraan : ");

            this.no = input.nextLine();

            System.out.println("\n-- Jam Masuk Kendaraan --");

            this.datang.inputJam();

            System.out.println("\n-- Jam Keluar Kendaraan --");

            pulang.inputJam();

        }

*//Output*

*public* String getNoKendaraan(){

*return* this.no;

        }

*public* String getJenis(){

*return* this.jenis;

        }

*public* Waktu getWaktudatang(){

*return* this.datang;

        }

*public* Waktu getWaktuPulang(){

*return* this.pulang;

        }

*//Proses*

*public* Waktu getLamaParkir(){

*return* this.datang.cariDurasi(this.pulang);

        }

*public* int getLamaJam(){

            int hasil = 0;

*if*(this.getLamaParkir().getMenit()>=10 || this.getLamaParkir().getJam()>=1){

                hasil = this.getLamaParkir().getJam();

*if*( this.getLamaParkir().getMenit()>0 || this.getLamaParkir().getDetik()>0 ){

                    hasil +=1;

                }

            }

*return* hasil;

        }

*public* *abstract* int getBiayaParkir();

*// hasil = getLamaJam() \* 2000;*

*public* void getKendaraan(){

            System.out.println("No kendaraan = " + this.no);

            System.out.println("Jenis = " + this.jenis);

            System.out.println("Jam Masuk = " + this.getWaktudatang().getWaktu());

            System.out.println("Jam Pulang = " + this.getWaktuPulang().getWaktu());

            System.out.println("Lama Parkir = " + this.getLamaParkir().getWaktu());

            System.out.println("Lama jam = " + this.getLamaJam());

            System.out.println("Biaya = " + this.getBiayaParkir());

        }

}

* + 1. Class Mobil:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : Mobil.java*

*\* Deskripsi     : Soal parkir - class mobil list*

*\*/*

package Kendaraan;

*public* class Mobil extends Kendaraan {

*public* Mobil(){

        super();

        this.jenis = "Mobil";

    }

    @Override

*public* int getBiayaParkir() {

*return* getLamaJam() \* 3000;

    }

}

* + 1. Class Motor:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : Motor.java*

*\* Deskripsi     : Soal parkir - class Motor*

*\*/*

package Kendaraan;

*public* class Motor extends Kendaraan {

*public* Motor(){

        super();

        this.jenis="Motor";

    }

    @Override

*public* int getBiayaParkir() {

*return* getLamaJam() \* 2000;

    }

}

* + 1. Class Truck:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : ElementList.java*

*\* Deskripsi     : Soal parkir - class Truck*

*\*/*

package Kendaraan;

*public* class Truck extends Kendaraan {

*public* Truck(){

        super();

        this.jenis="Truck";

    }

    @Override

*public* int getBiayaParkir() {

*return* getLamaJam() \* 10000;

    }

}

* + 1. Class ElementList:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : ElementList.java*

*\* Deskripsi     : Soal parkir - class element list*

*\*/*

package DataStructure;

import Kendaraan.Kendaraan;

*public* class ElementList {

*private* Kendaraan data;

    ElementList next;

*public* ElementList(){

        this.data = null;

        this.next = null;

    }

*public* ElementList(Kendaraan *data*){

        this.data = *data*;

        this.next = null;

    }

*public* void setData(Kendaraan *data*){

        this.data = *data*;

    }

*public* Kendaraan getData(){

*return* this.data;

    }

}

* + 1. Class LinkedList:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : LinkedList.java*

*\* Deskripsi     : Soal parkir - class Linked List*

*\*/*

package DataStructure;

import java.util.Scanner;

import Kendaraan.\*;

*public* class LinkedList {

*private* ElementList first;

*public* LinkedList(){

        this.first = null;

    }

*public* void createList(){

        this.first = null;

    }

*public* ElementList createElmnt(){

        Scanner input = *new* Scanner(System.in);

        ElementList baru = *new* ElementList();

        int num;

        System.out.println("Jenis Kendaraan : ");

        System.out.println("1 . Mobil");

        System.out.println("2 . Motor");

        System.out.println("3 . Truck");

        System.out.print("Input...");

        num = input.nextInt();

*switch*(num){

*case* 1:

               baru.setData(*new* Mobil());

*break*;

*case* 2:

                baru.setData(*new* Motor());

*break*;

*case* 3:

                baru.setData(*new* Truck());

*break*;

        }

        baru.getData().inputKendaraan();

        baru.next = null;

*return* baru;

    }

*//Insert*

*public* void insertFirst(ElementList *baru*){

*if*(this.first == null){

            this.first = *baru*;

        }

*else*{

*baru*.next = this.first;

            this.first = *baru*;

        }

    }

*public* void insertLast(ElementList *baru*){

*if*(this.first == null){

            this.first = *baru*;

        }

*else*{

            ElementList last = first;

*while*(last.next !=null){

                last = last.next;

            }

            last.next = *baru*;

        }

    }

*//Searching*

*public* ElementList search(String *no*){

        ElementList hasilCari = this.first;

        int found = 0;

*while*(hasilCari!=null && found == 0){

*if*(hasilCari.getData().getNoKendaraan().compareTo(*no*) == 0){

                found = 1;

            }

*else*{

                hasilCari = hasilCari.next;

            }

        }

*return* hasilCari;

    }

*//Delete*

*public* void deleteFirst(){

        ElementList hapus;

*if*(this.first.next==null){

            hapus = this.first;

            this.first =null;

        }

*else* *if*(this.first ==null){

            System.out.println("Tidak ada yang dihapus ! ");

        }

*else*{

            hapus = this.first;

            this.first = this.first.next;

        }

    }

*public* void deleteLast(){

        ElementList hapus;

*if*(this.first.next==null){

            hapus = this.first;

            this.first =null;

        }

*else* *if*(this.first ==null){

            System.out.println("Tidak ada yang dihapus ! ");

        }

*else*{

            ElementList b4last = first;

*while*(b4last.next.next !=null){

                b4last = b4last.next;

            }

            hapus = b4last.next.next;

            b4last.next = null;

        }

    }

*//Traversal*

*public* void printData(){

        System.out.println("\t\t\t\tRekapitulasi Biaya parkir PT Parkir Jaya");

*if*(this.first == null){

            System.out.println("List Kosong ! ");

        }

*else*{

            int no = 1;

            System.out.println("================================================================================================================================");

            System.out.println("No\tNo Kendaraan\t\tJenis\tJam Masuk\tJam Pulang\tLama Parkir\tLama jam\tBiaya");

            System.out.println("================================================================================================================================");

            ElementList bantu = this.first;

*while*(bantu!=null){

                    System.out.println(

                        no + "\t" +

                        bantu.getData().getNoKendaraan() + "\t\t" +

                        bantu.getData().getJenis()+ "\t" +

                        bantu.getData().getWaktudatang().getWaktu() + "\t" +

                        bantu.getData().getWaktuPulang().getWaktu() + "\t" +

                        bantu.getData().getLamaParkir().getWaktu() + "\t " +

                        bantu.getData().getLamaJam() + "\t\t" +

                        bantu.getData().getBiayaParkir() + "\t  "

                         );

                    no++;

                    bantu = bantu.next;

            }

            System.out.println("================================================================================================================================");

        }

    }

*public* int totalBiaya(){

        int total = 0;

        ElementList bantu = this.first;

*while*(bantu!=null){

            total += bantu.getData().getBiayaParkir();

            bantu = bantu.next;

        }

*return* total;

    }

}

* + 1. Class Main:

*/\**

*\* Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas         : A*

*\* Tanggal       : 16 November 2022*

*\* Nama Program  : App.java*

*\* Deskripsi     : Soal parkir - class main (program parkir dengan singly linked list)*

*\*/*

import DataStructure.\*;

import java.util.Scanner;

*public* class App {

*public* *static* void main(String[] *args*) {

        Scanner input = *new* Scanner(System.in);

        String no;

        LinkedList listKendaraan = *new* LinkedList();

        ElementList baru = *new* ElementList();

        ElementList cari = *new* ElementList();

        String cont = "Y";

        int pil;

*try*{

*do*

        {

            menu();

            pil = input.nextInt();

            clearScreen();

*switch*(pil){

*case* 1:

                    System.out.println("=== Metode Penambahan ===");

                    System.out.println("1 . Insert First");

                    System.out.println("2 . Insert Last");

                    System.out.print("Masukkan Pilihan....");

                    pil = input.nextInt();

                    clearScreen();

*switch*(pil){

*case* 1:

                        System.out.println("Data ditambah dengan insert first...");

                        baru = listKendaraan.createElmnt();

                        listKendaraan.insertFirst(baru);

*break*;

*case* 2:

                        System.out.println("Data ditambah dengan insert last...");

                        baru = listKendaraan.createElmnt();

                        listKendaraan.insertLast(baru);

*break*;

*default*:

                        System.out.println("Bukan Termasuk pilihan ! ");

*break*;

                    }

*break*;

*case* 2:

                    System.out.println("=== Metode Penghapusan ===");

                    System.out.println("1 . Delete First");

                    System.out.println("2 . Delete Last");

                    System.out.print("Masukkan Pilihan....");

                    pil = input.nextInt();

                    clearScreen();

*switch*(pil){

*case* 1:

                        System.out.println("Data dihapus dengan delete first...");

                        listKendaraan.deleteFirst();

*break*;

*case* 2:

                        System.out.println("Data dihapus dengan delete last...");

                        listKendaraan.deleteLast();

*break*;

*default*:

                        System.out.println("Bukan Termasuk pilihan ! ");

*break*;

                    }

*break*;

*case* 3:

                    System.out.print("Masukkan Plat nomor yang ingin dicari : ");

                    input.nextLine();

                    no = input.nextLine();

                    cari = listKendaraan.search(no);

*if*(cari == null){

                        System.out.println("Data Tidak ditemukan ! ");

                    }

*else*{

                        System.out.println("Data Ditemukan , Detail : ");

                        cari.getData().getKendaraan();

                    }

*break*;

*case* 4:

                    listKendaraan.printData();

                    System.out.println( "Total Biaya parkir = " + listKendaraan.totalBiaya());

*break*;

*case* 5:

                    cont ="N";

*break*;

*default*:

                    System.out.println("Bukan termasuk Pilihan ! ");

*break*;

            }

*if*(pil!=5){

            System.out.println("\n\nApakah program masih ingin dilanjutkan ? (Y/N)");

            cont = input.next();

            clearScreen();

            }

        }*while*(cont.compareTo("Y") == 0);

        System.out.println("=== TERIMA KASIH ===");

    }

*catch*(Exception e){

        System.out.println("Terdapat suatu error pada input...program diakhiri");

    }

*finally*{

        input.close();

    }

    }

*static* void clearScreen(){

        System.out.print("\033[H\033[2J");

        System.out.flush();

    }

*static* void menu(){

        System.out.println("===== MENU =====");

        System.out.println("1 . Tambah Data");

        System.out.println("2 . Hapus Data");

        System.out.println("3 . Cari Data");

        System.out.println("4 . Tampilkan Data");

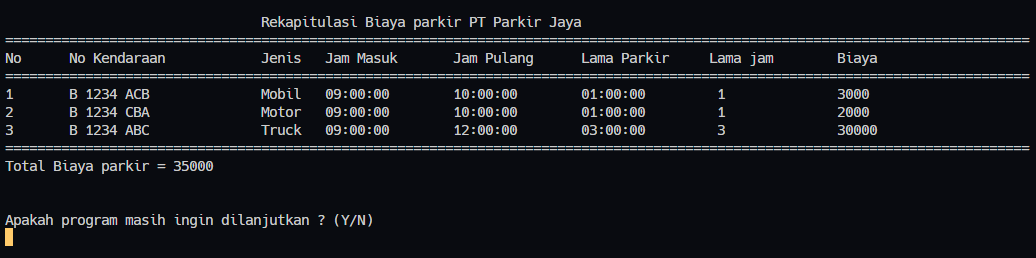
        System.out.println("5 . Keluar");

        System.out.print("Masukkan Pilihan....");

    }

}

* 1. Screenshot:



1. C++ Language
   1. Source Code:
      1. Class Waktu:
         1. Header:

*#ifndef* WAKTU\_H

*#define* WAKTU\_H

*#include*<iostream>

class Waktu{

    private:

        int jam,menit,detik;

     public:

        Waktu(int *jam*,int *menit*,int *detik*);

        Waktu();

        void setJam(int *jam*);

        void setMenit(int *menit*);

        void setDetik(int *detik*);

        void inputWaktu();

        int getJam();

        int getMenit();

        int getDetik();

        std::string getWaktu();

        int convertToSecond();

        void secondToClock(int *second*);

        Waktu cariDurasi(Waktu *akhir*);

};

*#endif*

* + - 1. Implement:

*#include* "waktu.h"

        Waktu::Waktu(int *jam*,int *menit*,int *detik*){

            this->jam = *jam*;

            this->menit = *menit*;

            this->detik = *detik*;

        }

        Waktu::Waktu(){

            this->jam = 0;

            this->menit =0;

            this->detik=0;

        }

        void Waktu::setJam(int *jam*){

            this->jam = *jam*;

        }

        void Waktu::setMenit(int *menit*){

            this->menit = *menit*;

        }

        void Waktu::setDetik(int *detik*){

            this->detik = *detik*;

        }

        void Waktu::inputWaktu(){

            std::cout<<"Masukkan jam : ";

            std::cin>>this->jam;

            std::cout<<"Masukkan menit : ";

            std::cin>>this->menit;

            std::cout<<"Masukkan detik : ";

            std::cin>>this->detik;

        }

        int Waktu::getJam(){

*return* this->jam;

        }

        int Waktu::getMenit(){

*return* this->menit;

        }

        int Waktu::getDetik(){

*return* this->detik;

        }

        std::string Waktu::getWaktu(){

            std::string nolJam ="";

            std::string nolMenit="";

            std::string nolDetik="";

*if*(this->jam<10){

                nolJam="0";

            }

*if*(this->menit<10){

                nolMenit="0";

            }

*if*(this->detik<10){

                nolDetik="0";

            }

*return* nolJam + std::to\_string(this->jam) + ":" + nolMenit+ std::to\_string(this->menit) + ":" +nolDetik+ std::to\_string(this->detik);

        }

        int Waktu::convertToSecond(){

            int hasil = this->detik + this->menit\*60 + this->jam\*3600;

*return* hasil;

        }

        void Waktu::secondToClock(int *second*){

            this->menit=*second*/60;

            this->detik=*second*%60;

            this->jam=this->menit/60;

            this->menit=this->menit%60;

        }

        Waktu Waktu::cariDurasi(Waktu *akhir*){

             Waktu temp;

            int detikAwal = this->convertToSecond();

            int detikAkhir = *akhir*.convertToSecond();

*if*(detikAkhir<detikAwal){

                detikAkhir+=86400;

            }

            int detikHasil = detikAkhir - detikAwal;

            temp.secondToClock(detikHasil);

*return* temp;

        }

* + 1. Class Kendaraan:
       1. Header:

*#ifndef* KENDARAAN\_H

*#define* KENDARAAN\_H

*#include*<iostream>

*#include* "waktu.h"

class Kendaraan{

    protected:

        std::string no;

        std::string jenis;

        Waktu datang;

        Waktu pulang;

    public:

        Kendaraan();

*//Input*

        void setNoKendaraan(std::string *no*);

        void setJenis(std::string *jenis*);

        void setWaktuDatang(Waktu *datang*);

        void setWaktuPulang(Waktu *pulang*);

        void inputKendaraan();

*//Output*

        std::string getNoKendaraan();

        std::string getJenis();

        Waktu getWaktudatang();

        Waktu getWaktuPulang();

        void getKendaraan();

*//Proses*

        Waktu getLamaParkir();

        int getLamaJam();

        virtual int getBiayaParkir();

};

*#endif*

* + - 1. Implement:

*#include* "Kendaraan.h"

        Kendaraan::Kendaraan(){

            this->no = " ";

        }

*//Input*

        void Kendaraan::setNoKendaraan(std::string *no*){

            this->no = no;

        }

        void Kendaraan::setJenis(std::string *jenis*){

            this->jenis = jenis;

        }

        void Kendaraan::setWaktuDatang(Waktu *datang*){

            this->datang = datang;

        }

        void Kendaraan::setWaktuPulang(Waktu *pulang*){

            this->pulang=pulang;

        }

        void Kendaraan::inputKendaraan(){

            std::cout<<"\n--- INPUT KENDARAAN ---\n";

            std::cout<<"No Kendaraan : ";

            std::cin.ignore();

            std::getline(std::cin,this->no);

            std::cout<<"\n-- Jam Masuk Kendaraan --\n";

            this->datang.inputWaktu();

            std::cout<<"\n-- Jam Keluar Kendaraan --\n";

            this->pulang.inputWaktu();

        }

*//Output*

        std::string Kendaraan::getNoKendaraan(){

*return* this->no;

        }

        std::string Kendaraan::getJenis(){

*return* this->jenis;

        }

        Waktu Kendaraan::getWaktudatang(){

*return* this->datang;

        }

        Waktu Kendaraan::getWaktuPulang(){

*return* this->pulang;

        }

*//Proses*

        Waktu Kendaraan::getLamaParkir(){

*return* this->datang.cariDurasi(this->pulang);

        }

        int Kendaraan::getLamaJam(){

        int hasil = 0;

*if*(this->getLamaParkir().getMenit()>=10 || this->getLamaParkir().getJam()>=1){

                hasil = this->getLamaParkir().getJam();

*if*( this->getLamaParkir().getMenit()>0 || this->getLamaParkir().getDetik()>0 ){

                    hasil +=1;

                }

            }

*return* hasil;

        }

        void Kendaraan::getKendaraan(){

            std::cout<<"No kendaraan = " << this->no<<"\n";

            std::cout<<"Jenis = " << this->jenis<<"\n";

            std::cout<<"Jam Masuk = " << this->getWaktudatang().getWaktu()<<"\n";

            std::cout<<"Jam Pulang = " << this->getWaktuPulang().getWaktu()<<"\n";

            std::cout<<"Lama Parkir = "<<this->getLamaParkir().getWaktu()<<"\n";

            std::cout<<"Lama jam = " << this->getLamaJam()<<"\n";

            std::cout<<"Biaya = " << this->getBiayaParkir();

        }

        int Kendaraan::getBiayaParkir(){return 0;}

* + 1. Class Mobil:
       1. Header:

*#ifndef* MOBIL\_H

*#define* MOBIL\_H

*#include* "Kendaraan.h"

class Mobil : public Kendaraan{

    public:

        Mobil():Kendaraan(){this->jenis="Mobil";};

        int getBiayaParkir();

};

*#endif*

* + - 1. Implement:

*#include* "Mobil.h"

int Mobil::getBiayaParkir(){

*return* this->getLamaJam()\*3000;

}

* + 1. Class Motor:
       1. Header:

*#ifndef* MOTOR\_H

*#define* MOTOR\_H

*#include* "Kendaraan.h"

class Motor : public Kendaraan{

    public:

        Motor():Kendaraan(){

            this->jenis="Motor";

        }

        int getBiayaParkir();

};

*#endif*

* + - 1. Implement:

*#include* "Motor.h"

int Motor::getBiayaParkir(){

*return* this->getLamaJam()\*2000;

 }

* + 1. Class Truck:
       1. Header:

*#ifndef* TRUCK\_H

*#define* TRUCK\_H

*#include* "Kendaraan.h"

class Truck : public Kendaraan{

    public:

        Truck():Kendaraan(){

            this->jenis="Truck";

        }

         int getBiayaParkir();

};

*#endif*

* + - 1. Implement:

*#include* "Truck.h"

 int Truck::getBiayaParkir(){

*return* this->getLamaJam()\*10000;

}

* + 1. Class ElementList:
       1. Header:

*#ifndef* ELEMENTLIST\_H

*#define* ELEMENTLIST\_H

*#include* "Kendaraan.h"

class ElementList {

    private:

        Kendaraan \*data;

    public:

        ElementList \*next;

        ElementList();

        ElementList(Kendaraan \**data*);

        void setData(Kendaraan \**data*);

        Kendaraan\* getData();

};

*#endif*

* + - 1. Implement:

*#include* "ElementList.h"

 ElementList::ElementList(){

        this->next = nullptr;

    }

ElementList::ElementList(Kendaraan \**data*){

        this->data = *data*;

        this->next = nullptr;

    }

     void  ElementList::setData(Kendaraan \**data*){

        this->data = *data*;

    }

    Kendaraan\* ElementList::getData(){

*return* this->data;

    }

* + 1. Class LinkedList:
       1. Header:

#ifndef LINKEDLIST\_H

#define LINKEDLIST\_H

#include "ElementList.h"

class LinkedList {

    private:

     ElementList \*first;

    public:

        LinkedList();

        void createList();

        ElementList \*createElmnt();

    //Insert

        void insertFirst(ElementList \*baru);

        void insertLast(ElementList \*baru);

    //Searching

        ElementList\* search(std::string no);

    //Delete

        void deleteFirst();

        void deleteLast();

    //Traversal

        void printData();

        int totalBiaya();

};

#endif

* + - 1. Implement:

*#include* "LinkedList.h"

*#include* "Motor.h"

*#include* "Mobil.h"

*#include* "Motor.h"

*#include* "Truck.h"

  LinkedList::LinkedList(){

        this->first = nullptr;

    }

    void LinkedList::createList(){

        this->first = nullptr;

    }

    ElementList\* LinkedList::createElmnt(){

        ElementList \*baru = new ElementList();

        int num;

       std::cout<<"Jenis Kendaraan : \n";

       std::cout<<"1 . Mobil\n";

       std::cout<<"2 . Motor\n";

       std::cout<<"3 . Truck\n";

       std::cout<<"Input...";

       std::cin>>num;

*switch*(num){

*case* 1:

               baru->setData(new Mobil());

*break*;

*case* 2:

                baru->setData(new Motor());

*break*;

*case* 3:

                baru->setData(new Truck());

*break*;

        }

        baru->getData()->inputKendaraan();

        baru->next = nullptr;

*return* baru;

    }

*//Insert*

    void LinkedList::insertFirst(ElementList \**baru*){

*if*(this->first == nullptr){

            this->first = baru;

        }

*else*{

            baru->next = this->first;

            this->first = baru;

        }

    }

   void LinkedList::insertLast(ElementList \**baru*){

*if*(this->first == nullptr){

            this->first = baru;

        }

*else*{

            ElementList\* last = first;

*while*(last->next !=nullptr){

                last = last->next;

            }

            last->next = baru;

        }

    }

*//Searching*

    ElementList\* LinkedList::search(std::string *no*){

        ElementList \*hasilCari = this->first;

        int found = 0;

*while*(hasilCari!=nullptr && found == 0){

*if*(hasilCari->getData()->getNoKendaraan() == no){

                found = 1;

            }

*else*{

                hasilCari = hasilCari->next;

            }

        }

*return* hasilCari;

    }

*//Delete*

   void LinkedList::deleteFirst(){

        ElementList \*hapus;

*if*(this->first->next==nullptr){

            hapus = this->first;

            this->first =nullptr;

        }

*else* *if*(this->first ==nullptr){

           std::cout<<"Tidak ada yang dihapus ! \n";

        }

*else*{

            hapus = this->first;

            this->first = this->first->next;

        }

    }

    void LinkedList::deleteLast(){

        ElementList\* hapus;

*if*(this->first->next==nullptr){

            hapus = this->first;

            this->first =nullptr;

        }

*else* *if*(this->first ==nullptr){

           std::cout<<"Tidak ada yang dihapus ! ";

        }

*else*{

            ElementList \*b4last = this->first;

*while*(b4last->next->next !=nullptr){

                b4last = b4last->next;

            }

            hapus = b4last->next->next;

            b4last->next = nullptr;

        }

    }

*//Traversal*

    void LinkedList::printData(){

       std::cout<<"\t\t\t\tRekapitulasi Biaya parkir PT Parkir Jaya\n";

*if*(this->first == nullptr){

           std::cout<<"List Kosong ! ";

        }

*else*{

            int no = 1;

           std::cout<<"================================================================================================================================\n";

           std::cout<<"No\tNo Kendaraan\t\tJenis\tJam Masuk\tJam Pulang\tLama Parkir\tLama jam\tBiaya\n";

           std::cout<<"================================================================================================================================\n";

            ElementList \*bantu = this->first;

*while*(bantu!=nullptr){

                   std::cout<<

                        no << "\t" <<

                        bantu->getData()->getNoKendaraan() << "\t\t" <<

                        bantu->getData()->getJenis()<< "\t" <<

                        bantu->getData()->getWaktudatang().getWaktu() << "\t" <<

                        bantu->getData()->getWaktuPulang().getWaktu() << "\t" <<

                        bantu->getData()->getLamaParkir().getWaktu() << "\t " <<

                        bantu->getData()->getLamaJam() << "\t\t" <<

                        bantu->getData()->getBiayaParkir() << "\t  \n"

                         ;

                    no++;

                    bantu = bantu->next;

            }

           std::cout<<"================================================================================================================================\n";

        }

    }

    int LinkedList::totalBiaya(){

        int hasil = 0;

          ElementList \*bantu = this->first;

            while(bantu!=nullptr){

                hasil +=bantu->getData()->getBiayaParkir();

                bantu = bantu->next;

            }

        return hasil;

    }

* + 1. Class Main:

*/\**

*\* Nama    : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*\* NPM     : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*\* Kelas : A*

*\* Tanggal : 11 November 2022*

*\* Nama Program : Main.cpp*

*\* Deskripsi :  Program Main cpp*

*\*/*

*#include* "waktu.h"

*#include* "waktu.cpp"

*#include* "Kendaraan.h"

*#include* "Kendaraan.cpp"

*#include* "LinkedList.h"

*#include* "LinkedList.cpp"

*#include* "ElementList.h"

*#include* "ElementList.cpp"

*#include* "Motor.h"

*#include* "Motor.cpp"

*#include* "Mobil.h"

*#include* "Mobil.cpp"

*#include* "Truck.h"

*#include* "Truck.cpp"

*#include* <iostream>

 void menu(){

        std::cout<<"===== MENU =====\n";

        std::cout<<"1 . Tambah Data\n";

        std::cout<<"2 . Hapus Data\n";

        std::cout<<"3 . Cari Data\n";

        std::cout<<"4 . Tampilkan Data\n";

        std::cout<<"5 . Keluar\n";

        std::cout<<"Masukkan Pilihan....\n";

    }

int main()

{

    ElementList \*baru = new ElementList();

    ElementList \*cari = new ElementList();

    LinkedList listKendaraan;

    listKendaraan.createList();

    int pil;

    std::string no;

    std::string cont ="Y";

*try*{

*do*

        {

            menu();

            std::cin>>pil;

            system("cls");

*switch*(pil){

*case* 1:

                   std::cout<<"=== Metode Penambahan ===\n";

                   std::cout<<"1 . Insert First\n";

                   std::cout<<"2 . Insert Last\n";

                    std::cout<<"Masukkan Pilihan....";

                    std::cin>>pil;

                    system("cls");

*switch*(pil){

*case* 1:

                       std::cout<<"Data ditambah dengan insert first...\n";

                        baru = listKendaraan.createElmnt();

                        listKendaraan.insertFirst(baru);

*break*;

*case* 2:

                       std::cout<<"Data ditambah dengan insert last...\n";

                        baru = listKendaraan.createElmnt();

                        listKendaraan.insertLast(baru);

*break*;

*default*:

                       std::cout<<"Bukan Termasuk pilihan ! \n";

*break*;

                    }

*break*;

*case* 2:

                   std::cout<<"=== Metode Penghapusan ===\n";

                   std::cout<<"1 . Delete First\n";

                   std::cout<<"2 . Delete Last\n";

                    std::cout<<"Masukkan Pilihan....";

                    std::cin>>pil;

                    system("cls");

*switch*(pil){

*case* 1:

                       std::cout<<"Data dihapus dengan delete first...\n";

                        listKendaraan.deleteFirst();

*break*;

*case* 2:

                       std::cout<<"Data dihapus dengan delete last...\n";

                        listKendaraan.deleteLast();

*break*;

*default*:

                       std::cout<<"Bukan Termasuk pilihan ! \n";

*break*;

                    }

*break*;

*case* 3:

                    std::cout<<"Masukkan Plat nomor yang ingin dicari : ";

                   std::cin.ignore();

                    std::getline(std::cin,no);

                    cari = listKendaraan.search(no);

*if*(cari == nullptr){

                       std::cout<<"Data Tidak ditemukan ! \n";

                    }

*else*{

                       std::cout<<"Data Ditemukan , Detail : \n";

                        cari->getData()->getKendaraan();

                    }

*break*;

*case* 4:

                    listKendaraan.printData();

                    std::cout<<"Total Biaya Parkir = "<<listKendaraan.totalBiaya()<<"\n";

*break*;

*case* 5:

                    cont ="N";

*break*;

*default*:

                   std::cout<<"Bukan termasuk Pilihan ! \n";

*break*;

            }

*if*(pil!=5){

           std::cout<<"\n\nApakah program masih ingin dilanjutkan ? (Y/N)\n";

            std::cin>>cont;

            system("cls");

            }

        }*while*(cont == "Y");

}

*catch*(...){

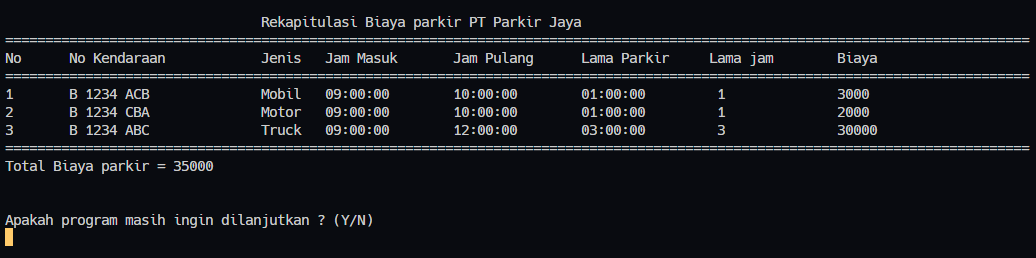
    std::cout<<"Terdapat error...";

}

    std::cout<<"=== TERIMA KASIH ===";

}

* 1. Screenshot:



1. Python Language
   1. Source Code:
      1. Class Waktu:

class Waktu:

*#Attribute*

    \_\_jam=0

    \_\_menit=0

    \_\_detik=0

*#Constructor*

    def \_\_init\_\_(*self*, \**args*):

*if* (len(args) == 3):

            self.\_\_jam = int(args[0])

            self.\_\_menit = int(args[1])

            self.\_\_detik = int(args[2])

*elif*(len(args)==0):

            self.\_\_jam = int(0)

            self.\_\_menit = int(0)

            self.\_\_detik = int(0)

*else*:

            print("False number of argument in constructor")

*#Input Method*

    def setJam(*self*,*jam*):

        self.\_\_jam = int(jam)

    def setMenit(*self*,*menit*):

        self.\_\_menit = int(menit)

    def setDetik(*self*,*detik*):

        self.\_\_detik = int(detik)

    def inputWaktu(*self*):

        self.\_\_jam = int(input("Masukkan jam : "))

        self.\_\_menit = int(input("Masukkan menit : "))

        self.\_\_detik = int(input("Masukkan detik : "))

*#Output Method*

    def getJam(*self*):

*return* self.\_\_jam

    def getMenit(*self*):

*return* self.\_\_menit

    def getDetik(*self*):

*return* self.\_\_detik

    def getWaktu(*self*):

        nolJam =""

        nolMenit=""

        nolDetik=""

*if*(self.\_\_jam<10):

            nolJam="0"

*if*(self.\_\_menit<10):

            nolMenit="0"

*if*(self.\_\_detik<10):

            nolDetik="0"

*return* nolJam + str(self.\_\_jam) + ":" + str(nolMenit)+ str(self.\_\_menit) + ":" +nolDetik+ str(self.\_\_detik)

*#Proses*

    def convertToSecond(*self*):

        hasil = self.\_\_detik + (int(60) \* self.\_\_menit) + (int(3600) \* self.\_\_jam)

*return* hasil

    def secondToClock(*self*,*second*:int):

        self.\_\_menit = int(second/60)

        self.\_\_detik = int(second%60)

        self.\_\_jam = int(self.\_\_menit/60)

        self.\_\_menit = int(self.\_\_menit%60)

    def cariDurasi(*self*,*akhir*):

        temp = Waktu()

        detikAwal = self.convertToSecond()

        detikAkhir = akhir.convertToSecond()

*if*(detikAkhir<detikAwal):

            detikAkhir+=86400

        detikHasil = detikAkhir - detikAwal

        temp.secondToClock(detikHasil)

*return* temp

* + 1. Class Kendaraan:

*from* Waktu *import* Waktu

class Kendaraan:

    \_no = " "

    \_jenis= " "

    \_datang = Waktu()

    \_pulang = Waktu()

    #Constructor

    def \_\_init\_\_(self):

        self.\_no = " "

        self.\_jenis = " "

        self.\_datang = Waktu(0,0,0)

        self.\_pulang = Waktu(0,0,0)

    #Input

    def setNoKendaraan(self,no):

        self.\_no = no

    def setJenis(self,jenis):

        self.\_jenis=jenis

    def setWaktuDatang(self,datang):

        self.\_datang = datang

    def setWaktuPulang(self,pulang):

        self.\_pulang = pulang

    def inputKendaraan(self):

        print("\n--- INPUT KENDARAAN---")

        self.\_no = input("No Kendaraan : ")

        print("\n--- Jam Masuk Kendaraan ---")

        self.\_datang.inputWaktu()

        print("\n--- Jam Keluar Kendaraan ---")

        self.\_pulang.inputWaktu()

    #Output

    def getNoKendaraan(self):

        return self.\_no

    def getJenis(self):

        return self.\_jenis

    def getWaktuDatang(self):

        return self.\_datang

    def getWaktuPulang(self):

        return self.\_pulang

    def getLamaParkir(self):

        return self.\_datang.cariDurasi(self.\_pulang)

    def getLamaJam(self):

        hasil = int(0)

        if(self.getLamaParkir().getMenit()>=10 or self.getLamaParkir().getJam()>=1):

            hasil = self.getLamaParkir().getJam()

            if(self.getLamaParkir().getMenit()>0 or self.getLamaParkir().getDetik()>0):

                hasil = hasil + 1

        return hasil

    def getKendaraan(self):

        print("Plat Nomor kendaraan = ",self.\_no)

        print("Jenis Kendaraan = " , self.\_jenis)

        print("Waktu masuk = ",self.\_datang.getWaktu())

        print("Waktu keluar = ",self.\_pulang.getWaktu())

        print("Lama Parkrir = ",self.getLamaParkir().getWaktu())

        print("Lama Jam = ",self.getLamaJam())

        print("Biaya = ",self.getBiayaParkir())

    def getBiayaParkir(self):

        ...

* + 1. Class Mobil:

from Kendaraan import Kendaraan

class Mobil(Kendaraan):

    def \_\_init\_\_(self):

        super().\_\_init\_\_()

        self.\_jenis = "Mobil"

    def getBiayaParkir(self):

        return self.getLamaJam() \* 3000

* + 1. Class Motor:

*from* Kendaraan *import* Kendaraan

class Motor(Kendaraan):

    def \_\_init\_\_(*self*):

        super().\_\_init\_\_()

*self*.\_jenis = "Motor"

    def getBiayaParkir(*self*):

        return *self*.getLamaJam() \* 2000

* + 1. Class Truck:

*from* Kendaraan *import* Kendaraan

class Truck(Kendaraan):

    def \_\_init\_\_(*self*):

        super().\_\_init\_\_()

*self*.\_jenis = "Truck"

    def getBiayaParkir(*self*):

*return* *self*.getLamaJam() \* 10000

* + 1. Class ElementList:

class ElementList:

*#Constructor*

    def \_\_init\_\_(*self*):

*self*.next = None

*self*.\_\_info = None

    def setData(*self*,*info*):

*self*.\_\_info = *info*

    def getData(*self*):

*return* *self*.\_\_info

* + 1. Class LinkedList:

*from* ElementList *import* ElementList

*from* Mobil *import* Mobil

*from* Motor *import* Motor

*from* Truck *import* Truck

class LinkedList:

    \_\_first  = ElementList()

    def \_\_init\_\_(*self*):

*self*.\_\_first = None

    def createList(*self*):

*self*.\_\_first = None

    def createElement(*self*):

        baru = ElementList()

        num = 0

        print("Jenis Kendaraan : ")

        print("1 . Mobil")

        print("2 . Motor")

        print("3 . Truck")

        num = int(input("Input..."))

*match* num:

*case* 1:

                baru.setData(Mobil())

*case* 2:

                baru.setData(Motor())

*case* 3:

                baru.setData(Truck())

        baru.getData().inputKendaraan()

*return* baru

*#Insert*

    def inserFirst(*self*,*baru*):

*if*(*self*.\_\_first == None):

*self*.\_\_first = *baru*

*else*:

*baru*.next = *self*.\_\_first

*self*.\_\_first = *baru*

    def insertLast(*self*,*baru*):

*if*(*self*.\_\_first == None):

*self*.\_\_first = *baru*

*else*:

            bantu = *self*.\_\_first

*while*(bantu.next != None):

                bantu = bantu.next

            bantu.next = *baru*

*#Delete*

    def deleteFirst(*self*):

        hapus = ElementList()

*if*(*self*.\_\_first.next==None):

            hapus = *self*.\_\_first

*self*.\_\_first =None

*elif*(*self*.\_\_first == None):

            print("Tidak ada yang Dihapus")

*else*:

            hapus = *self*.\_\_first

*self*.\_\_first = *self*.\_\_first.next

    def deleteLast(*self*):

        hapus = ElementList()

*if*(*self*.\_\_first== None):

            hapus = *self*.\_\_first

*self*.\_\_first = None

*elif*(*self*.\_\_first == None):

            print("Tidak ada yang dihapus")

*else*:

            b4last = *self*.\_\_first

*while*(b4last.next.next !=None):

                b4last = b4last.next

            hapus = b4last.next.next

            b4last.next = None

*#Search*

    def search(*self*,*no*):

        hasilCari = *self*.\_\_first

        found = 0

*while*(hasilCari!= None and found == 0):

*if*(hasilCari.getData().getNoKendaraan() == *no*):

                found = 1

*else*:

                hasilCari = hasilCari.next

*return* hasilCari

*#Print data*

    def cetakTabelParkir(*self*):

        print("\t\t\t\tRekapitulasi Biaya parkir PT Parkir Jaya\n")

*if*(*self*.\_\_first == None):

            print("List Kosong ! ")

*else*:

            no = int(1)

            bantu = *self*.\_\_first

            print("================================================================================================================================")

            print("No\tNo Kendaraan\t\tJenis\tJam Masuk\tJam Pulang\tLama Parkir\tLama jam\tBiaya");

            print("================================================================================================================================")

*while*(bantu != None):

                print(

                    no,"\t",

                    bantu.getData().getNoKendaraan(),"\t\t",

                    bantu.getData().getJenis(),"\t",

                    bantu.getData().getWaktuDatang().getWaktu(),"\t",

                    bantu.getData().getWaktuPulang().getWaktu(),"\t",

                    bantu.getData().getLamaParkir().getWaktu(),"\t ",

                    bantu.getData().getLamaJam(),"\t\t",

                    bantu.getData().getBiayaParkir(),"\t  "

                         )

                no = no+1

                bantu = bantu.next

            print("================================================================================================================================")

    def totalBiaya(*self*):

        total = 0

        bantu = *self*.\_\_first

*while*(bantu != None):

            total = total + bantu.getData().getBiayaParkir()

            bantu = bantu.next

*return* total

* + 1. Class Main:

1. *from* LinkedList *import* LinkedList
2. *from* ElementList *import* ElementList
3. *from* os *import* system
4. def menu():
5. print("===== MENU =====");
6. print("1 . Tambah Data");
7. print("2 . Hapus Data");
8. print("3 . Cari Data");
9. print("4 . Tampilkan Data");
10. print("5 . Keluar");
11. print("Masukkan Pilihan...." ,*end*="");
12. baru = ElementList()
13. listKendaraan = LinkedList()
14. cari = ElementList()
15. cont = "Y"
16. pil = 0
17. *try*:
18. *while*(True):
19. menu()
20. pil = int(input())
21. system('cls')
22. *match* pil:
23. *case* 1:
24. print("=== METODE INSERT ===")
25. print("1 . Insert First")
26. print("2 . Insert Last")
27. pil = int(input("Masukkan Pilihan..."))
28. system('cls')
29. *match* pil:
30. *case* 1:
31. print("Insert First...")
32. baru = listKendaraan.createElement()
33. listKendaraan.inserFirst(baru)
35. *case* 2:
36. print("Insert Last...")
37. baru = listKendaraan.createElement()
38. listKendaraan.insertLast(baru)
40. *case* \_:
41. print("Bukan termasuk pilihan ! ")
43. *case* 2:
44. print("=== HAPUS DATA ===")
45. print("1 . Delete First")
46. print("2 . Delete Last")
47. pil = int(input("Masukkan Pilihan..."))
48. system('cls')
49. *match* pil:
50. *case* 1:
51. print("Delete First...")
52. listKendaraan.deleteFirst()
54. *case* 2:
55. print("Delete Last...")
56. listKendaraan.deleteLast()
58. *case* \_:
59. print("Bukan termasuk pilihan ! ")
61. *case* 3:
62. no = input("Masukkan plat nomor kendaraan yang ingin dicari...")
63. cari = listKendaraan.search(no)
64. *if*(cari == None):
65. print("Data tidak ditemukan")
67. *else*:
68. print("Data Ditemukan , detail...")
69. cari.getData().getKendaraan()
71. *case* 4:
72. listKendaraan.cetakTabelParkir()
73. print("Total Biaya Parkir = ",listKendaraan.totalBiaya())
75. *case* 5:
76. cont = "N"
78. *case* \_:
79. print("Bukan termasuk pilihan ! ")
80. cont = input("Apakah program masih ingin dilanjutkan ? (Y/N)")
81. system('cls')
82. *if*(cont!="Y"):
83. *break*
84. print("=== TERIMA KASIH ===")
85. *except*:
86. print("Terdapat error...")
87. 1. Screenshot:

