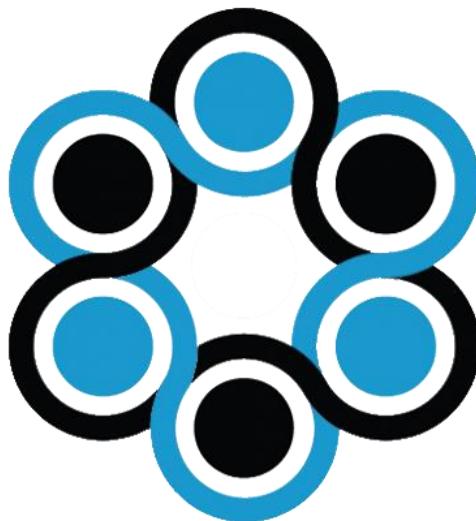


LAPORAN AKHIR
PRAKTIKUM PEMROGRAMAN
BERORIENTASI OBJEK
Periode VI



Disusun Oleh:

Aindya Berlinani Santoso [06.2019.1.07162]
Eric Wahyu Amiruddin

**FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI**
INSTITUT TEKNOLOGI ADHI TAMA SURABAYA
SURABAYA
2021

LEMBAR PENGESAHAN
PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK
Periode VI

Di susun oleh :

Nama : Anindya Berlinani Santoso

NPM : 06.2019.1.07162

Tanggal 15 Januari 2021

Dosen Pembimbing

M.Kurniawan, S.Kom., S.Kom
NIP 153045

Asisten Pembimbing

Eric Wahyu Amiruddin



RESMI

MODUL 1

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



Tugas Resmi

LAPORAN RESMI



**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
PERIODE VI - Tahun 2020/2021**

NAMA : Anindya Berlinani S
NPM : 063019107162
MODUL : 1



Tugas Resmi

Soal Resmi

Sistem Informasi Menu Makanan

Langkah Pertama

Buatlah file baru dengan nama PublicClassAnindya07162. Pastikan file yang telah terbuat nama kelas dan nama file harus sama. Buatlah sebuah void main di dalam class yang sudah dibuat tersebut. Buat class anindyaEntity yang digunakan untuk menyimpan data menu makanan dengan kode sebagai berikut:



Tugas Resmi

Source Code

```
package pkgpublic.pkgclass.anindya.pkg07162;

public class anindyaEntity {
    String anindya07162_nama;
    String anindya07162_pesan;
    String anindya07162_makanan;
    String anindya07162_harga;

    public anindyaEntity(String anindya07162_nama,
String anindya07162_pesan, String
anindya07162_makanan, String anindya07162_harga)
    {
        this.anindya07162_nama=anindya07162_nama;
        this.anindya07162_pesan=anindya07162_pesan;
        this.anindya07162_makanan=anindya07162_makanan;
        this.anindya07162_harga=anindya07162_harga;

    }
    public String getanindya07162_nama(){
        return anindya07162_nama;
    }
    public String getanindya07162_pesan(){
        return anindya07162_pesan;
    }
    public String getanindya07162_makanan(){
        return anindya07162_makanan;
    }
    public String getanindya07162_harga(){
        return anindya07162_harga;
    }
}
```



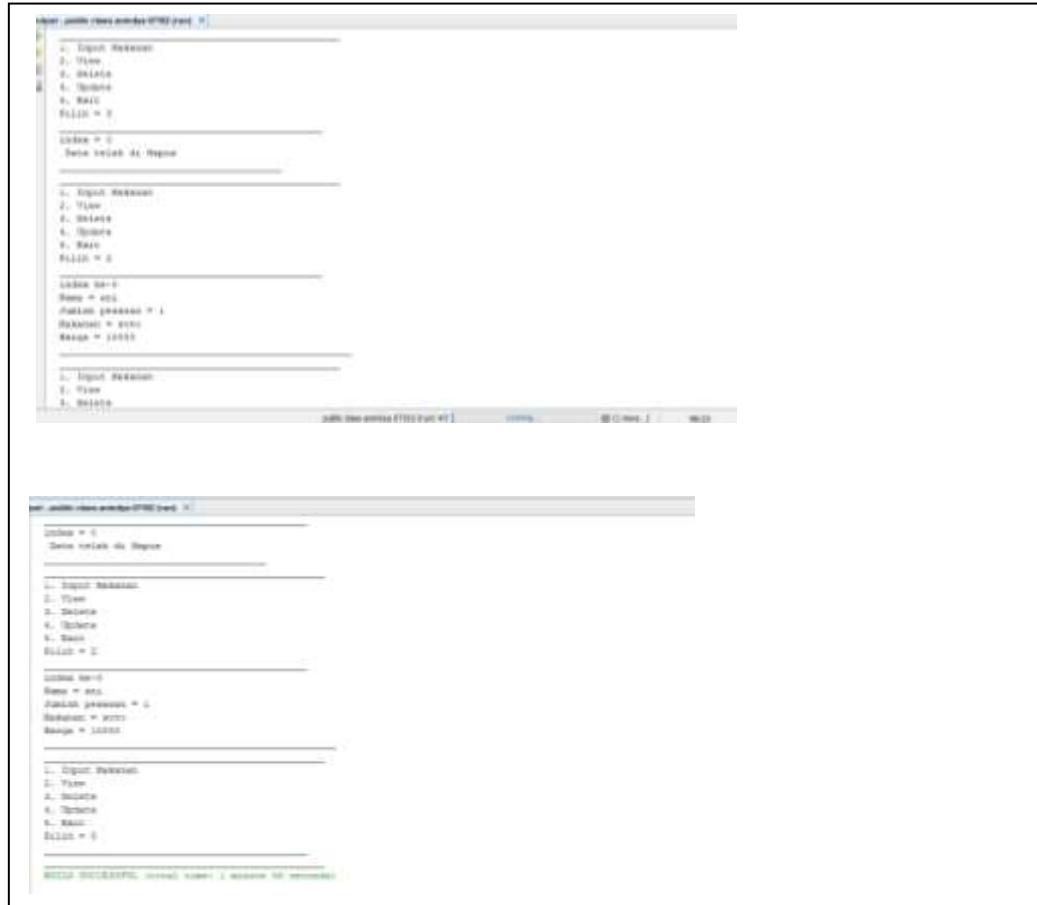
Tugas Resmi

Screenshot



Tugas Resmi

Screenshot



Analisa

Pada class `anindyaEntity` didalamnya terdapat attribute: `String anindya07162_nama, String anindya07162_pesanan, String anindya07162_makanan, String anindya07162_harga.` Lalu terdapat Constructor dengan parameternya: `public anindyaEntity(String anindya07162_nama, String anindya07162_pesanan, String anindya07162_makanan, String anindya07162_harga).` Juga terdapat get:

```
public String getanindya07162_nama(){
    return anindya07162_nama;
}
public String getanindya07162_pesanan(){
    return anindya07162_pesanan;
}
public String getanindya07162_makanan(){
    return anindya07162_makanan;
}
public String getanindya07162_harga(){
    return anindya07162_harga
}
```



Tugas Resmi

Langkah Kedua

Pada main class yang sudah di buat sebelumnya tulis kode seperti berikut :

Source Code

```
package pkgpublic.pkgclass.anindya.pkg07162;
import java.util.Scanner;
import java.util.ArrayList;
public class PublicClassAnindya07162 {
    static Scanner input = new Scanner (System.in);
    static ArrayList<anindyaEntity>datamakanan = new
ArrayList();

    static void anindya07162_makanan()
    {
        System.out.print("Inputkan Nama = ");
        String anindya07162_nama = input.next();
        System.out.print("Inputkan jumlah pesanan = ");
        String anindya07162_pesan = input.next();
        System.out.print("Inputkan makanan = ");
        String anindya07162_makanan = input.next();
        System.out.print("Inputkan harga = ");
        String anindya07162_harga = input.next();
        datamakanan.add(new
anindyaEntity(anindya07162_nama,anindya07162_pesan,anind
ya07162_makanan,anindya07162_harga));
    }
    static void anindya07162_view()
    {
        for(int i=0;i<datamakanan.size();i++)
        {
            System.out.println("index ke-"+(i));
            System.out.println("Nama =
"+datamakanan.get(i).getanindya07162_nama());
            System.out.println("Jumlah pesanan =
"+datamakanan.get(i).getanindya07162_pesan());
            System.out.println("Makanan =
"+datamakanan.get(i).getanindya07162_makanan());
            System.out.println("Harga =
"+datamakanan.get(i).getanindya07162_harga());

            System.out.println("_____");
        }
    }
}
```



Tugas Resmi

Source Code

```
static int carimakanan(String anindya07162_nama)
{
    int indexmakanan = -1;
    for(int i=0;i<datamakanan.size();i++)
    {

        if(anindya07162_nama.equals(datamakanan.get(i).getanindya07162_nama()))
        {
            indexmakanan = i;
        }
    }
    return indexmakanan;
}

static void anindya07162_hapus()
{
    System.out.print("index = ");
    int index = input.nextInt();
    datamakanan.remove(index);
    System.out.println(" Data telah di Hapus ");

    System.out.println("_____");
}

static void anindya07162_update()
{
    System.out.print("index = ");
    int index = input.nextInt();
    for(int i=0;i<datamakanan.size();i++)
    {
        if(index == i)
        {
            System.out.print("Inputkan nama = ");
            String anindya07162_nama = input.next();
            System.out.print("Inputkan jumlah
pesanan = ");
            String anindya07162_pesan =
            input.next();
            System.out.print("Inputkan menu makanan
= ");
            String anindya07162_makanan =
            input.next();
            System.out.print("Inputkan Harga = ");
            String anindya07162_harga =
        }
    }
}
```



Tugas Resmi

Source Code

```
System.out.print("Inputkan Harga = ");
String anindya07162_harga =
input.nextInt();

datamakanan.get(i).anindya07162_nama=anindya07162_nama;

datamakanan.get(i).anindya07162_pesanan=anindya07162_pesanan;

datamakanan.get(i).anindya07162_makanan=anindya07162_makanan;

datamakanan.get(i).anindya07162_harga=anindya07162_harga;
;
System.out.println("Data Telah di
Update");
}
}
}

public static void main(String[] args) {
System.out.println(" SELAMAT DATANG !!");

System.out.println("_____");
int pil;
do{
System.out.println("1. Input Makanan");
System.out.println("2. View");
System.out.println("3. Delete");
System.out.println("4. Update");
System.out.println("5. Exit");
System.out.print("Pilih = ");
pil = input.nextInt();

System.out.println("_____");
_____);
switch(pil){
case 1 : anindya07162_makanan();
break;
case 2 : anindya07162_view();
break;
case 3 : anindya07162_hapus();
break;
case 4 : anindya07162_update();
break;
}
```



Tugas Resmi

Source Code

```
}

System.out.println("_____
_____
}while(pil!=5);
}
}
```

Screenshot

```
Output - public class sevelida STT2021 (main) 1
1. Daftar Mahasiswa
2. Tambah
3. Delone
4. Update
5. Exit
Pilih = 1

Inputan Nama = sevelida
Inputan jumlah mahasiswa = 2
Inputan mahasiswa = sevelida
Inputan biaya = 10000

1. Input Mahasiswa
2. Tambah
3. Delone
4. Update
5. Exit
Pilih = 1

Inputan Nama = sevi
Inputan jumlah mahasiswa = 2
Inputan mahasiswa = sevi
Inputan biaya = 10000

1. Input Mahasiswa
2. Tambah

Output - public class sevelida STT2021 (main) 1
1. Daftar Mahasiswa
2. Tambah
3. Delone
4. Update
5. Exit
Pilih = 2

Output - public class sevelida STT2021 (main) 1
1. Daftar Mahasiswa
2. Tambah
3. Delone
4. Update
5. Exit
Pilih = 2

Output - public class sevelida STT2021 (main) 1
1. Daftar Mahasiswa
2. Tambah
3. Delone
4. Update
5. Exit
Pilih = 2

Output - public class sevelida STT2021 (main) 1
1. Daftar Mahasiswa
2. Tambah
3. Delone
4. Update
5. Exit
Pilih = 2
```



Tugas Resmi

Screenshot





Tugas Resmi

Analisa

Pada class main terdapat: `java.util.ArrayList` yang digunakan untuk menyimpan data berupa array. `Java.util.Scanner` yang digunakan untuk menginputkan data. Scanner dengan sistem scanner digunakan untuk sistem inputan int. `ArrayList` dari kelas `anindyaEntity` dengan nama datamakanan dan disimpan dengan new `ArrayList`.

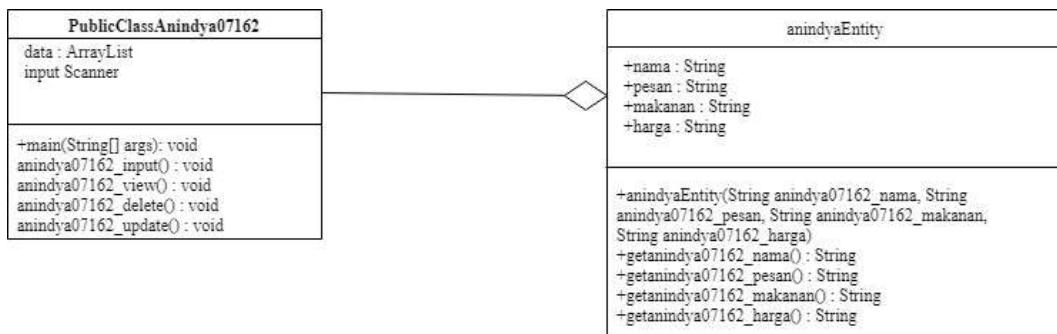
```
public static void main(String[] args) {  
    System.out.println(" SELAMAT DATANG !!");  
    System.out.println(" _____ ");  
    int pil;  
    do{  
        System.out.println("1. Input Makanan");  
        System.out.println("2. View");  
        System.out.println("3. Delete");  
        System.out.println("4. Update");  
        System.out.println("5. Exit");  
        System.out.print("Pilih = ");  
        pil = input.nextInt();  
  
        System.out.println(" _____ ");  
        switch(pil){  
            case 1 : anindya07162_makanan();  
            break;  
            case 2 : anindya07162_view();  
            break;  
            case 3 : anindya07162_hapus();  
            break;  
            case 4 : anindya07162_update();  
            break;  
        }  
  
        System.out.println(" _____ ");  
    }  
    }while(pil!=5);  
}
```

Source code diatas untuk masuk ke menu utama. Di menu utama terdapat 5 pilihan. Pilihan pertama akan darahkan ke method `anindya07162_makanan`. Di method `anindya07162_makanan` kita menginputkan data berupa nama, jumlah pesanan, makanan, harga. Jika sudah akan disimpan di datamakanan dengan new class `anindyaEntity`. Jika kita memilih pilihan 2 maka akan ditampilkan semua data yang sudah diinputkan tadi. Jika memilih pilihan 3 maka di uji dulu jika ada maka menginputkan index dari view jika inputannya sesuai maka `equalsnya` berarti dia berada di index 0 atau benar dan jika diuji `equalsnya` tidak sesuai maka akan muncul tulisan data telah dihapus. Lalu kita memilih pilihan 4, disini untuk mengupdate data lama menjadi data baru.



Tugas Resmi

Class Diagram :

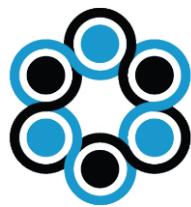




RESMI

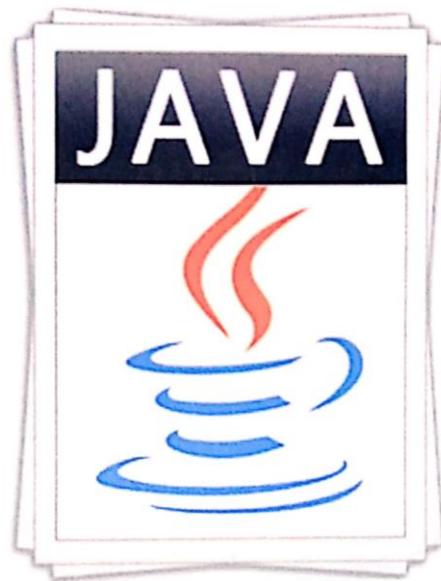
MODUL 2

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



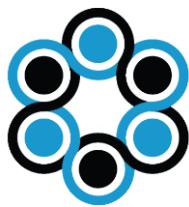
Tugas Resmi

LAPORAN RESMI



**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
PERIODE VI - Tahun 2020/2021**

NAMA : Anindya Berlinani
NPM : 06.2019.1.07162
MODUL : 2



Tugas Resmi

Soal Praktikum

Sistem Informasi Menu Makanan

Langkah Pertama

membuat class abstract

Source Code

```
package Entity07162;

public abstract class AbstractEntity07162 {
    public String nama,password;

    public AbstractEntity07162(String nama,String password){
        this.nama = nama;
        this.password = password;
    }
    public AbstractEntity07162 (){
    }

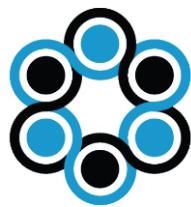
    public abstract String getNama();

    public void setNama(String nama) {
        this.nama = nama;
    }

    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}
```

Analisa :

Class AbstarctEntity digunakan untuk menginputkan login dengan menginputkan nama dan password.



Tugas Resmi

Langkah Kedua

Membuat class AdminEntity

Source Code

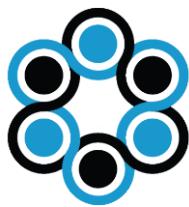
```
package Entity07162;

public class AdminEntity07162 extends
AbstractEntity07162 {

    public AdminEntity07162(String nama,String
password) {
        super (nama,password);
    }
    @Override
    public String getNama(){
        return nama;
    }
}
```

Analisa :

Class AdminEntity digunakan untuk menginputkan nama dan password pada saat login admin



Tugas Resmi

Source Code

```
package Entity07162;

public class MakananEntity07162 {
    public String namamakanan,jumlah_makanan,harga;

    public MakananEntity07162 (String
namamakanan,String jumlah_makanan,String harga){
        this.namamakanan = namamakanan;
        this.jumlah_makanan = jumlah_makanan;
        this.harga = harga;
    }

    public String getNamaMakanan() {
        return namamakanan;
    }
    public String getJumlahMakanan() {
        return jumlah_makanan;
    }

    public String getHarga() {
        return harga;
    }
}
```

Analisa :

Class MakananEntity berisi constructor String namamakanan, String jumlah_makanan, String harga



Tugas Resmi

Source Code

```
package Entity07162;

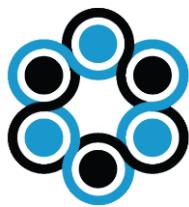
public class PembeliEntity07162 extends
AbstractEntity07162 {
    String alamat;
    String no_telp;

    public PembeliEntity07162(String
nama,String password,String alamat, String
no_telp){
        super (nama,password);
        this.alamat = alamat;
        this.no_telp = no_telp;
    }

    @Override
    public String getNama(){
        return nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public String getNoTelp() {
        return no_telp;
    }
}
```

Analisa :

Class AdminEntity digunakan untuk menginputkan nama dan password pada saat login pembeli

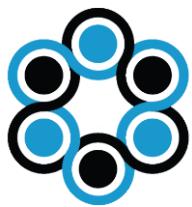


Tugas Resmi

Source Code

```
package Model07162;
import Entity07162.AdminEntity07162;
import java.util.ArrayList;
public class AdminModel07162 implements
ModelInterface07162
{
    private ArrayList<AdminEntity07162>
adminEntityArrayList;

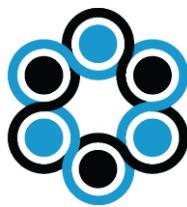
    public AdminModel07162()
    {
        adminEntityArrayList = new
ArrayList<AdminEntity07162>();
    }
    public void insertAdmin(AdminEntity07162
admin)
    {
        adminEntityArrayList.add(admin);
    }
    @Override
    public void view()
    {
        for (AdminEntity07162 admin :
adminEntityArrayList)
        {
            System.out.print(admin.getNama());
            System.out.print(admin.getPassword());
            System.out.println();
        }
    }
    public int cekData(String nama, String password){
        int loop = 0;
        for(AdminEntity07162 admin :
adminEntityArrayList){
            if(admin.getNama().equals(nama) &&
admin.getPassword().equals(password)){
                break;
            }else{
                loop++;
            }
        }
        return loop;
    }
}
```



Tugas Resmi

Analisa :

Pada class **AdminModel** berisikan data yang ada pada class
AdminEntity



Tugas Resmi

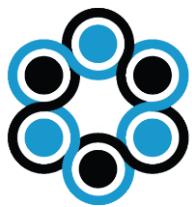
Source Code

```
package Model07162;
import Entity07162.PembeliEntity07162;
import java.util.ArrayList;
public class PembeliModel07162 implements
ModelInterface07162 {
    private ArrayList<PembeliEntity07162>
pembeliEntityArrayList;

    public PembeliModel07162()
    {
        pembeliEntityArrayList = new
ArrayList<PembeliEntity07162>();
    }
    public void insertPembeli(PembeliEntity07162
pembeli)
    {
        pembeliEntityArrayList.add(pembeli);
    }

    @Override
    public void view(){
        for(PembeliEntity07162 pembeli :
pembeliEntityArrayList)
        {
            System.out.print(pembeli.getNama());
            System.out.print(pembeli.getPassword());
            System.out.print(pembeli.getAlamat());
            System.out.print(pembeli.getNoTelp());
            System.out.println();
        }
    }

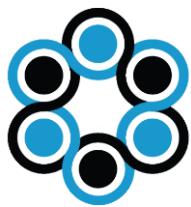
    public int cekData(String nama, String password){
        int loop = 0;
        while
(!pembeliEntityArrayList.get(loop).getNama().equals(nama
) &&
!pembeliEntityArrayList.get(loop).getPassword().equals(p
assword)){
            loop++;
        }
        return loop;
    }
}
```



Tugas Resmi

Analisa :

Pada class **PembeliModel** berisikan data yang ada pada class **PembeliEntity**



Tugas Resmi

Source Code

```
package Model07162;

public interface ModelInterface07162 {
    public void view();
    public int cekData (String nama, String
password);
}
```

Analisa :

Pada class Interface terdapat cekData untuk mengecek nama dan password
Serta terdapat View untuk melihat data.



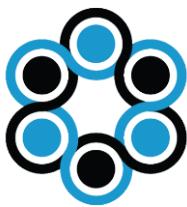
Tugas Resmi

Source Code

```
package modul2_07162;
import Entity07162.*;
import Model07162.*;
import java.util.ArrayList;
import java.util.Scanner;
public class Modul2_07162 {
private static AdminModel07162 adminModel = new
AdminModel07162();
private static PembeliModel07162 pembeliModel = new
PembeliModel07162();
static public ArrayList <MakananEntity07162>
datamakanan= new ArrayList();
static int cekAdmin;
static int cekPembeli;
static Scanner input = new Scanner(System.in);
    public static void main(String[] args) {
        System.out.println(" SELAMAT DATANG DI RUMAH
MAKAN !!");

System.out.println("=====");
=====
        int pilih;
        do{
            System.out.println("1. Daftar");
            System.out.println("2. Login");
            System.out.println("3. Exit");
            System.out.print("Pilih = ");
            pilih = input.nextInt();

System.out.println("=====");
=====
            switch(pilih){
                case 1:
                    if(pilih == 1){
                        daftarpembeli();
                    }
                    break;
                case 2 : if(pilih == 2){
                    int pilLogin = 0;
                    System.out.print("1. Login Admin" + "\n2.
Login pembeli" + "\n Pilih : ");
                    pilLogin = input.nextInt();
                    if(pilLogin == 1){
                        loginAdmin();
                    }
                }
            }
        }
    }
}
```



Tugas Resmi

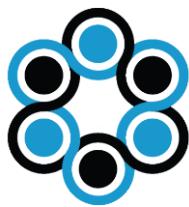
Source Code

```
        }else{
            loginPembeli();
        }
    }
    break;

}System.out.println("=====
=====");
}while (pilih!=3);
}
static void daftarpembeli()
{
    System.out.print("Input Nama = ");
    String nama = input.next();
    System.out.print("Input Password = ");
    String password = input.next();
    System.out.print("Alamat = ");
    String alamat = input.next();
    System.out.print("Input notelp = ");
    String no_telp = input.next();
    pembeliModel.insertPembeli(new
PembeliEntity07162(nama,password,alamat,no_telp));
}

static void dataAdmin()
{
    String nama[] = {"anindya","anin"};
    String password[] = {"123","123"};
    for(int i=0;i<nama.length;i++)
    {
        adminModel.insertAdmin(new
AdminEntity07162(nama[i],password[i]));
    }
}
static void makanan(){
    System.out.print("INPUT NAMA MAKANAN = ");
    String namamakanan = input.next();
    System.out.print(" INPUT JUMLAH MAKANAN = ");
    String jumlah_makanan = input.next();
    System.out.print("INPUT HARGA = ");
    String harga = input.next();

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



Tugas Resmi

Source Code

```
datamakanan.add(new MakananEntity07162
(namamakanan,jumlah_makanan,harga));
}
static void view(){
    System.out.println(" MAKANAN YANG TERSEDIA ");

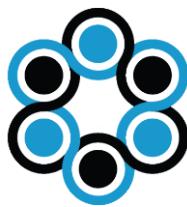
System.out.println("=====");
for(int i=0;i<datamakanan.size();i++)
{
    System.out.println("Nama Makanan =
"+datamakanan.get(i).getNamaMakanan());
    System.out.println("Jumlah Makanan =
"+datamakanan.get(i).getJumlahMakanan());
    System.out.println("Harga =
"+datamakanan.get(i).getHarga());

System.out.println("=====");
}
}
static void hapus(String namamakanan)
{
    for(int i=0;i<datamakanan.size();i++)
{

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
    datamakanan.remove(i);
    System.out.println("Data Telah Di
Hapus");

System.out.println("=====");
}
else
{
    System.out.println("Makanan Tidak
Tersedia");

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



Tugas Resmi

Source Code

```
}

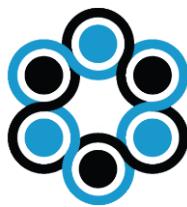
    static void update(String namamakanan, String jumlah_makanan, String newharga)
    {
        for(int i=0;i<datamakanan.size();i++)
        {

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
            datamakanan.get(i).jumlah_makanan =
jumlah_makanan;
            datamakanan.get(i).harga = newharga;
            System.out.println("Data Telah DI
Update");

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

static void loginAdmin(){
    int pil;
    System.out.print("Masukkan Nama Admin : ");
    String nama = input.next();
    System.out.print("Password Admin : ");
    String password = input.next();
    cekAdmin = adminModel.cekData(nama,password);
    do{
        System.out.println("1. Input Data
Makanan");
        System.out.println("2. View Data Makanan");
        System.out.println("3. Hapus Data
Makanan");
        System.out.println("4. Update Data
Makanan");
        System.out.println("5. Exit");
        System.out.print("Pilih = ");
        pil = input.nextInt();

System.out.println("=====
=====");
switch(pil){
    case 1 : makanan();
    break;
    .....
```

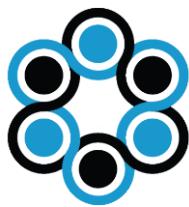


Tugas Resmi

Source Code

```
case 2 : view();
           break;
case 3 : System.out.print("Inputkan Nama
Makanan = ");
           String namamakanan =
input.next();
           hapus(namamakanan);
           break;
case 4 :
           System.out.print("Inputkan Nama
Makanan = ");
           namamakanan = input.next();
           for(int
i=0;i<datamakanan.size();i++)
{
if(namamakanan.equals(datamakanan.get(i).getNamaMakanan(
)))
{
           System.out.print("Inputkan
Harga Baru = ");
           String newharga =
input.next();
           System.out.print("Inputkan
jumlah makanan baru = ");
           String jumlah_makanan =
input.next();

System.out.println("=====
=====");
update(namamakanan,jumlah_makanan,newharga);
}
else
{
           System.out.println("Makanan
Tidak Tersedia");
System.out.println("=====
=====");
}
}
break;
}
```

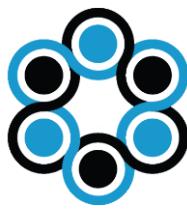


Tugas Resmi

Source Code

```
        }
            while(pil!=5);
        }
    static void loginPembeli()
    {
        int pilihan;
        System.out.print("Nama Pelanggan : ");
        String nama = input.next();
        System.out.print("Password : ");
        String password = input.next();
        cekPembeli = pembeliModel.cekData(nama,
password);
        do
        {
            System.out.println("1. Data Makanan
Yang Tersedia");
            System.out.println("2. Exit");
            System.out.print("Pilih = ");
            pilihan = input.nextInt();

            System.out.println("=====
=====");
            switch(pilihan)
            {
                case 1: view();
                ;
                break;
            }
        }
        while (pilihan!=2);
    }
}
```



Tugas Resmi

Screenshot

```
SELAMAT DATANG DI RUMAH MAKAN !!
1. Daftar
2. Login
3. Exit
Pilih = 1

Input Name = anindya
Input Password = 123
Input Email = anindya@gmail.com
Input nohp = 088005577083

1. Daftar
2. Login
3. Exit
Pilih = 2

1. Login Admin
2. Login pembeli
Pilih = 1
Masukkan Name Admin : anin
Password Admin : 123
1. Input Data Makanan
2. View Data Makanan
3. Update Data Makanan
4. Delete Data Makanan
5. Exit
Pilih = 1

INPUT NAMA MAKANAN = baks
INPUT JUMLAH MAKANAN = 15000
INPUT HARGA = 10000

1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 1

MAKANAN YANG TERSEDIA
Nama Makanan = baks
Jumlah Makanan = 15000
Harga = 10000

1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 4

Inputkan Nama Makanan = batso
Inputkan Harga Baru = 10000
Inputkan jumlah makanan baru = 2
Data Telah Di Update

1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 2

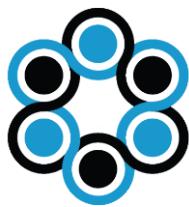
MAKANAN YANG TERSEDIA
Nama Makanan = baks
Jumlah Makanan = 2

MAKANAN YANG TERSEDIA
Nama Makanan = batso
Jumlah Makanan = 2

1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 5

1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 5

1. Daftar
2. Login
3. Exit
Pilih = 3
```



Tugas Resmi

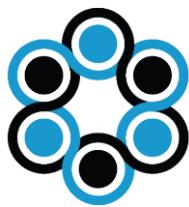
Analisa :

Pada class main terdapat: `java.util.ArrayList` yang digunakan untuk menyimpan data berupa array. `Java.util.Scanner` yang digunakan untuk menginputkan data. Scanner dengan sistem scanner digunakan untuk sistem inputan int. `ArrayList` dari kelas `MakananEntity07162` dengan nama datamakanan dan disimpan dengan new `ArrayList`.

```
public static void main(String[] args) {
    System.out.println(" SELAMAT DATANG DI RUMAH MAKAN !!!");

    System.out.println("=====");
    int pilih;
    do{
        System.out.println("1. Daftar");
        System.out.println("2. Login");
        System.out.println("3. Exit");
        System.out.print("Pilih = ");
        pilih = input.nextInt();

        System.out.println("=====");
        switch(pilih){
            case 1:
                if(pilih == 1){
                    daftarpembeli();
                }
                break;
            case 2 : if(pilih == 2){
                int pilLogin = 0;
                System.out.print("1. Login Admin" + "\n2. Login pembeli" + "\n
Pilih : ");
                pilLogin = input.nextInt();
                if(pilLogin == 1){
                    loginAdmin();
                }else{
                    loginPembeli();
                }
            }
            break;
        }
    }while (pilih!=3);
}
```



Tugas Resmi

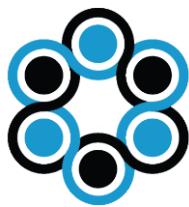
Pada source code diatas terdapat 3 pilihan yaitu Daftar untuk mendaftar nama pembeli, Login untuk login admin dan pembeli , serta exit untuk keluar.

```
static void loginAdmin(){
    int pil;
    System.out.print("Masukkan Nama Admin : ");
    String nama = input.next();
    System.out.print("Password Admin : ");
    String password = input.next();
    cekAdmin = adminModel.cekData(nama,password);
    do{
        System.out.println("1. Input Data Makanan");
        System.out.println("2. View Data Makanan");
        System.out.println("3. Hapus Data Makanan");
        System.out.println("4. Update Data Makanan");
        System.out.println("5. Exit");
        System.out.print("Pilih = ");
        pil = input.nextInt();

        System.out.println("=====");
        switch(pil){
            case 1 : makanan();
            break;
            case 2 : view();
            break;
            case 3 : System.out.print("Inputkan Nama Makanan = ");
                String namamakanan = input.next();
                hapus(namamakanan);
            break;
            case 4 :
                System.out.print("Inputkan Nama Makanan = ");
                namamakanan = input.next();
                for(int i=0;i<datamakanan.size();i++)
                {

                    if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
                    {
                        System.out.print("Inputkan Harga Baru = ");
                        String newharga = input.next();
                        System.out.print("Inputkan jumlah makanan baru = ");
                        String jumlah_makanan = input.next();

                        System.out.println("=====");
                        update(namamakanan,jumlah_makanan,newharga);
                    }
                }
            break;
        }
    }
}
```

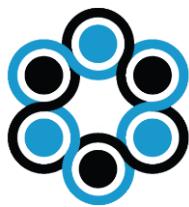


Tugas Resmi

```
        }
    else
    {
        System.out.println("Barang Tidak Tersedia");

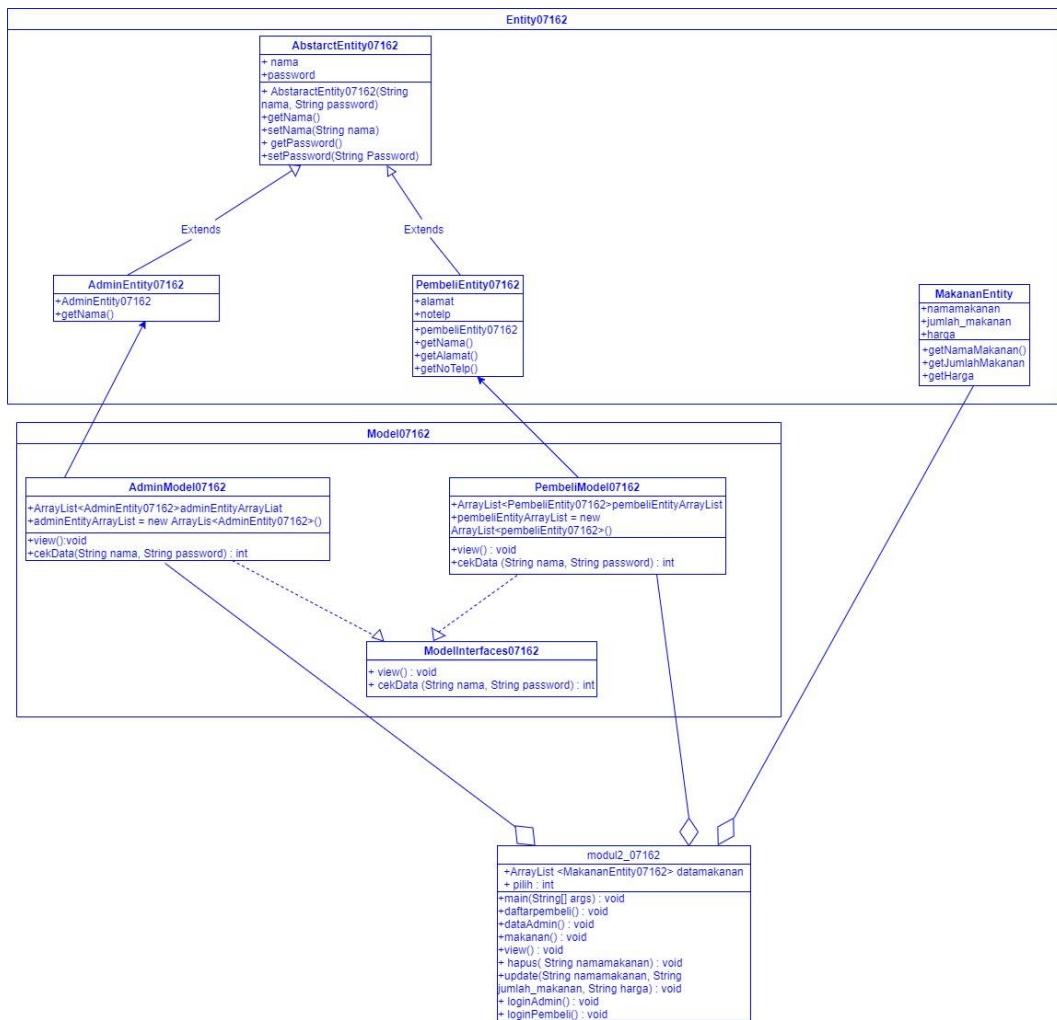
System.out.println("=====");
    }
}
break;
}
}
while(pil!=5);
}
```

Source code diatas untuk masuk ke menu utama. Di menu utama terdapat 5 pilihan. Pilihan pertama akan darahkan ke method_makanan. Di method makanan kita menginputkan data berupa nama, jumlah pesanan, harga. Jika sudah akan disimpan di datamakanan. Jika kita memilih pilihan 2 maka akan ditampilkan semua data yang sudah diinputkan tadi. Jika memilih pilihan 3 maka di uji dulu jika ada maka menginputkan index dari view.jika inputannya sesuai maka equallsnya berarti dia berada di index 0 atau benar. Lalu kita memilih pilihan 4, disini untuk mengupdate data lama menjadi data baru.



Tugas Resmi

Class diagram :

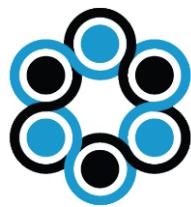




RESMI

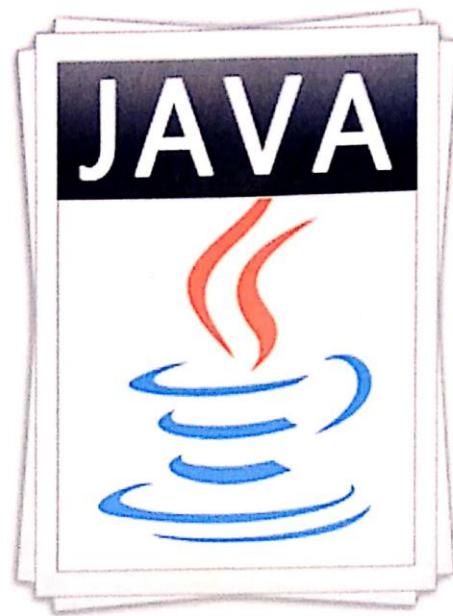
MODUL 3

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



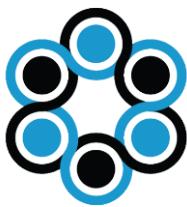
Tugas Resmi

LAPORAN RESMI



**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
PERIODE VI - Tahun 2020/2021**

NAMA : Anindya Berlinani
NPM : 06.2019.1.071b2
MODUL : 3



Tugas Resmi

Soal Praktikum

Sistem Informasi menu Makanan

Langkah Pertama

Membuat package Controller07162r, lalu membuat class dengan nama AllObjectModel07162

Source Code

```
package Controller07162;
import Model07162.*;
public class AllObjectModel07162{
    public static AdminModel07162 adminModel
= new AdminModel07162();
    public static PembeliModel07162
pembeliModel = new PembeliModel07162();
    public static MakananModel07162 makanan =
new MakananModel07162();
}
```

Analisa

Di class AllObjectModel07162 berisi import an dari package Model07162

Langkah Kedua

Membuat class dari package Controller07162 dengan nama ControllerInterface07162

Source Code

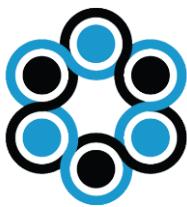
```
package Controller07162;
import java.util.Date;
public interface ControllerInterface07162{
    public void login(String nama, String
password);
}
```

Analisa

Di class ControllerInterface07162 terdapat void login yang berisi nama dan password untuk masuk sebagai admin/pembeli.

Langkah Ketiga

Membuat class dari package Controller07162 dengan nama MakananController07162



Tugas Resmi

Source Code

```
package Controller07162;
import Entity07162.MakananEntity07162;
public class MakananController07162 {
    int index1 = 0;
    public MakananController07162(){
    }
    public void insert(String namamakanan,String jumlah_makanan,String harga){
        AllObjectModel07162.makanan.insert(new
MakananEntity07162(namamakanan,jumlah_makanan,harga)
);
    }
    public void view(){
        AllObjectModel07162.makanan.view();
    }
    public void hapus(String namamakanan){

AllObjectModel07162.makanan.hapus(namamakanan);
    }
    public void update(String namamakanan,String jumlah_makanan,String harga){

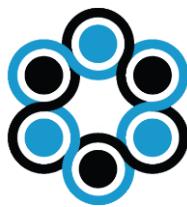
AllObjectModel07162.makanan.update1(namamakanan,jumlah_makanan, harga);
    }
}
```

Analisa

Di class MakananController07162 terdapat method insert untuk inputan nama makanan, jumlah makanan, dan harga. Juga terdapat method view untuk melihat inputan makanan, method hapus untuk menhapus makanan yang akan di hapus, dan method update untuk mengupdate jumlah makanan dan harga.

Langkah Keempat

Membuat class dari package Controller07162 dengan nama Pembeli07162



Tugas Resmi

Source Code

```
package Controller07162;
import Entity07162.PembeliEntity07162;
public class Pembeli07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public Pembeli07162() {
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.pembeliModel.cekData(nama,
password);
    }
    public void insertPembeli(String nama, String
password, String alamat, String no_telp) {
        PembeliEntity07162 pembeliModel = new
PembeliEntity07162(nama, password, alamat, no_telp);

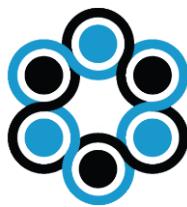
        allobject.pembeliModel.insertPembeli(pembeliModel);
    }
}
```

Analisa

Di class Pembeli07162 terdapat method login untuk login pembeli. Juga terdapat method insertPembeli untuk inputan nama password alamat dan no telp.

Langkah Kelima

Membuat class dari package Controller07162 dengan nama admin07162



Tugas Resmi

Source Code

```
package Controller07162;

import Entity07162.AdminEntity07162;
import Entity07162.MakananEntity07162;
import Entity07162.PembeliEntity07162;
import java.util.ArrayList;
public class admin07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public admin07162() {
    }
    public void dataAdmin() {
        String passwordAdmin[] = {"123"};
        String namaAdmin[] = {"Anindya"};
        String noTelpAdmin[] = {"088805577083"};
        for(int i = 0; i < passwordAdmin.length; i++)
    {

        AllObjectModel07162.adminModel.insertAdmin(new
AdminEntity07162(passwordAdmin[i], namaAdmin[i] ));
    }
}
    public void viewAdmin() {
        AllObjectModel07162.adminModel.view();
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.adminModel.cekData(nama,
password);
    }
}
```

Analisa

Di class admin07162 terdapat method dataAdmin yang berisi password, nama, dan no telpon. Juga terdapat method viewAdmin yang digunakan untuk melihat data admin dan method login untuk login ke admin

Langkah Keenam



Tugas Resmi

Membuat Package, lalu membuat class dari package Entity07162 dengan nama AbstractEntity07162

Source Code

```
package Entity07162;

public abstract class AbstractEntity07162 {
    public String nama,password;

    public AbstractEntity07162(String nama,String password){
        this.nama = nama;
        this.password = password;
    }
    public AbstractEntity07162 (){
    }

    public abstract String getNama();

    public void setNama(String nama) {
        this.nama = nama;
    }

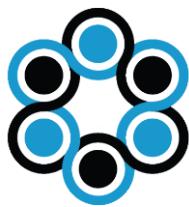
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}
```

Analisa

Di class AbstratcEntity07162 terdapat variabel nama dan password juga terdapat parameter dan gether nya.

Langkah Ketujuh

Membuat class dari package Entity07162 dengan nama AdminEntity07162



Tugas Resmi

Source Code

```
package Entity07162;

public class AdminEntity07162 extends
AbstractEntity07162 {

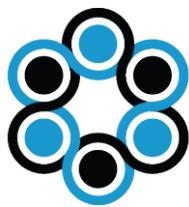
    public String harga;
    public String jumlah_makanan;
    public AdminEntity07162(String nama,String
password) {
        super (nama,password);
    }
    @Override
    public String getNama(){
        return nama;
    }
}
```

Analisa

Di class AdminEntit07162 terdapat variabel nama dan password. Juga terdapat parameter dan gether nya.

Langkah Kedelapan

Membuat Package, lalu membuat class dari package Entity07162 dengan nama MakananEntity07162.



Tugas Resmi

Source Code

```
package Entity07162;

public class MakananEntity07162 {
    public String namamakanan,jumlah_makanan,harga;

    public MakananEntity07162 (String
namamakanan,String jumlah_makanan,String harga){
        this.namamakanan = namamakanan;
        this.jumlah_makanan = jumlah_makanan;
        this.harga = harga;
    }

    public String getNamaMakanan() {
        return namamakanan;
    }
    public String getJumlahMakanan() {
        return jumlah_makanan;
    }

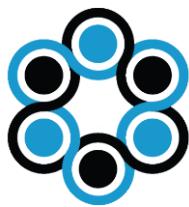
    public String getHarga() {
        return harga;
    }
}
```

Analisa

Di class MakananEntity07162 terdapat variabel nama makanan, jumlah makana, dan harga. Juga terdapat parameter dan gether nya.

Langkah keSembilan

Membuat class dari package Entity07162 dengan nama PembeliEntity07162.



Tugas Resmi

Source Code

```
package Entity07162;

public class PembeliEntity07162 extends
AbstractEntity07162 {
    String alamat;
    String no_telp;

    public PembeliEntity07162(String nama, String
password, String alamat, String no_telp){
        super (nama,password);
        this.alamat = alamat;
        this.no_telp = no_telp;
    }

    @Override
    public String getNama(){
        return nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public String getNoTelp() {
        return no_telp;
    }
}
```

Analisa

Di class PembeliEntity07162 terdapat variabel nama, password, alamat dan no telp. Juga terdapat parameter dan gether nya.

Langkah KeSepuluh

Membuat package Model dengan nama Model07162 lalu membuat class dengan nama AdminModel07162

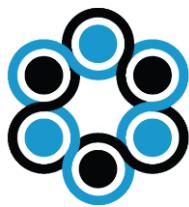


Tugas Resmi

Source Code

```
package Model07162;
import static
Controller07162.AllObjectModel07162.adminModel;
import Entity07162.AdminEntity07162;
import java.util.ArrayList;
public class AdminModel07162 implements
ModelInterface07162
{
    private ArrayList<AdminEntity07162>
adminEntityArrayList;

    public AdminModel07162()
    {
        adminEntityArrayList = new
ArrayList<AdminEntity07162>();
    }
    public void insertAdmin(AdminEntity07162 admin)
    {
        adminEntityArrayList.add(admin);
    }
    public void view()
    {
        for (AdminEntity07162 admin :
adminEntityArrayList)
        {
            System.out.print(admin.getNama());
            System.out.print(admin.getPassword());
            System.out.println();
        }
    }
    public int cekData(String nama, String password){
        int loop = 0;
        for(AdminEntity07162 admin :
adminEntityArrayList){
            if(admin.getNama().equals(nama) &&
admin.getPassword().equals(password)){
                break;
            }else{
                loop++;
            }
        }
        return loop;
    }
}
```



Tugas Resmi

Analisa

Di kelas AdminModel07162 terdapat method admin yang berisi data admin. Juga terdapat method view untuk melihat data admin dan method cekData untuk login.

Langkah keSembelas

Membuat class dengan nama MakananModel07162.

Source Code

```
package Model07162;
import Entity07162.MakananEntity07162;
import java.util.ArrayList;
public class MakananModel07162 {
    public ArrayList<MakananEntity07162> datamakanan;

    public MakananModel07162(){
        datamakanan = new
ArrayList<MakananEntity07162>();
    }
    public void Insert(MakananEntity07162 makanan){
        datamakanan.add(makanan);
    }
    public void view()
    {

        for (MakananEntity07162 makanan : datamakanan)
        {
            System.out.println("Nama Makanan =
"+makanan.getNamaMakanan());
            System.out.println("Jumlah Makanan =
"+makanan.getJumlahMakanan());
            System.out.println("Harga =
"+makanan.getHarga());

System.out.println("=====");
        }
    }
    public void hapus(String namakanan)
    {
        for(int i=0;i<datamakanan.size();i++)
        {
```



Tugas Resmi

Source Code

```
public void hapus(String namamakanan)
{
    for(int i=0;i<datamakanan.size();i++)
    {

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
    datamakanan.remove(i);
    System.out.println("Makanan Telah DI
Hapus");

System.out.println("=====
=====");
}
else
{
    System.out.println("Makanan Tidak
Tersedia");

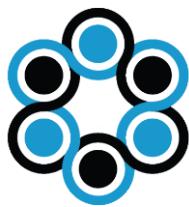
System.out.println("=====
=====");
}
}
}

public void update(String namamakanan,String
newjumlah_makanan,String newharga)
{
    for(int i=0;i<datamakanan.size();i++)
    {

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
    datamakanan.get(i).harga=newharga;

datamakanan.get(i).jumlah_makanan=newjumlah_makanan;
    System.out.println("Data Telah DI
Update");

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



Tugas Resmi

Source Code

```
System.out.println("=====");
=====
}
}
public void update1(String namamakanan, String newjumlah_makanan, String newharga){
    for(int i=0;i<datamakanan.size();i++)
    {

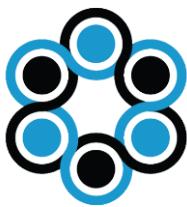
        if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
        {
            update(namamakanan,newjumlah_makanan,newharga);
        }
        else
        {
            System.out.println("Makanan
Tidak Tersedia");
        }
    }
}
```

Analisa

Di class MakananModel07162 terdapat method insert yang digunakan untuk menampilkan data makanan. Juga terdapat method view untuk melihat data makanan, method hapus untuk menghapus data makanan, method update untuk mengupdate jumlah makanan dan harga.

Langkah keDuaBebelas

Membuat class interface dengan nama ModelInterface0716207162.



Tugas Resmi

Source Code

```
package Model07162;

public interface ModelInterface07162 {
    public void view();
    public int cekData (String nama, String
password);
}
```

Analisa

Yang berisi method view untuk melihat dan method cekData untuk login

Langkah KeTigaBelas

Membuat class dengan nama PembeliModel07162.

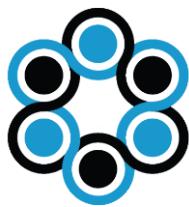
Source Code

```
package Model07162;
import Entity07162.PembeliEntity07162;
import java.util.ArrayList;
public class PembeliModel07162 implements
ModelInterface07162 {
    private ArrayList<PembeliEntity07162>
pembeliEntityArrayList;

    public PembeliModel07162()
    {
        pembeliEntityArrayList = new
ArrayList<PembeliEntity07162>();
    }

    public void insertPembeli(PembeliEntity07162
pembeli)
    {
        pembeliEntityArrayList.add(pembeli);
    }

    @Override
    public void view(){
        for(PembeliEntity07162 pembeli :
pembeliEntityArrayList)
        {
            System.out.print(pembeli.getNama());
            System.out.print(pembeli.getPassword());
        }
    }
}
```



Tugas Resmi

Source Code

```
public void view(){
    for(PembeliEntity07162 pembeli :
pembeliEntityArrayList)
    {
        System.out.print(pembeli.getNama());
        System.out.print(pembeli.getPassword());
        System.out.print(pembeli.getAlamat());
        System.out.print(pembeli.getNoTelp());
        System.out.println();
    }
}

public int cekData(String nama, String password){
    int loop = 0;
    while
(!pembeliEntityArrayList.get(loop).getNama().equals(nama)
) &&

!pembeliEntityArrayList.get(loop).getPassword().equals(p
assword)){
        loop++;
    }
    return loop;
}
```

Analisa

Di class PembeliModel07162 terdapat method insert untuk menampilkan data pembeli, method view untuk melihat dan method cekData untuk Login.

Langkah keEmpatBelas

Mengisi class main dengan perintah :

Source Code

```
package modul3_07162;
import Entity07162.*;
import Controller07162.*;
import java.util.ArrayList;
import java.util.Scanner;

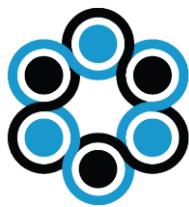
public class Modul3_07162 {
    private static admin07162 admin = new admin07162();
    . . . . .
```



Tugas Resmi

Source Code

```
public class Modul3_07162 {  
    private static admin07162 admin = new admin07162();  
    private static Pembeli07162 pembeli = new  
Pembeli07162();  
    private static MakananController07162 makanan = new  
MakananController07162();  
    private static Scanner input = new Scanner  
(System.in);  
    static int cekAdmin;  
    static int cekPembeli;  
    public static void main(String[] args) {  
        System.out.println("SELAMAT DATANG DI RUMAH  
MAKAN !!");  
  
        System.out.println("=====");  
        int pil;  
        do{  
            System.out.println("1. Daftar");  
            System.out.println("2. Login");  
            System.out.println("3. Exit");  
            System.out.print("Pilih = ");  
            pil = input.nextInt();  
  
            System.out.println("=====");  
            switch(pil){  
                case 1:  
                    if(pil == 1){  
                        daftarpembeli() ;  
                    }  
                    break;  
                case 2 : if(pil == 2){  
                    int pilLogin = 0;  
                    System.out.print("1. Login Admin" + "\n2.  
Login pembeli" + "\n Pilih : ");  
                    pilLogin = input.nextInt();  
                    if(pilLogin == 1){  
                        loginAdmin();  
                    }else{  
                        loginPembeli();  
                    }  
                }  
            }  
        }  
    }  
}
```



Tugas Resmi

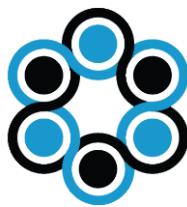
Source Code

```
break;

}System.out.println("=====");
=====
}while (pil!=3);
}
static void daftarpembeli()
{
    try{
        System.out.print("Input Nama = ");
        String nama = input.next();
        System.out.print("Input Password = ");
        String password = input.next();
        System.out.print("Alamat = ");
        String alamat = input.next();
        System.out.print("Input notelp = ");
        String no_telp = input.next();
        pembeli.insertPembeli(nama, password, alamat,
no_telp);
        System.out.println("Daftar Sukses !!");
    }catch (Exception exception){
        System.out.println("Format Salah !!");
    }
}
static void makanan(){
    System.out.print("INPUT NAMA MAKANAN = ");
    String namamakanan = input.next();
    System.out.print("INPUT JUMLAH MAKANAN = ");
    String jumlah_makanan = input.next();
    System.out.print("INPUT HARGA = ");
    String harga = input.next();
    makanan.Insert
(namamakanan,jumlah_makanan,harga);

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

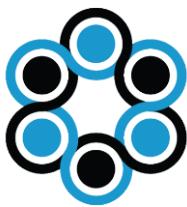
static void view()
{
    for(int
i=0;i<makanan.cekDaftarMakanan(cekmakanan).size();i++)
    {
        System.out.println("Nama Makanan =
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getNamaMak
anan());
    }
}
```



Tugas Resmi

Source Code

```
System.out.println("Nama Makanan =  
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getNamaMakanan());  
        System.out.println("Jumlah Makanan =  
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getJumlahMakanan());  
        System.out.println("Harga =  
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getHarga());  
  
System.out.println("=====");  
}  
}  
static void remove(String namamakanan)  
{  
    for(int  
i=0;i<makanan.cekDaftarMakanan(cekAdmin).size();i++)  
    {  
  
if(namamakanan.equals(makanan.cekDaftarMakanan(i).get(i)  
.getNamaMakanan()))  
    {  
        makanan.remove(i);  
        System.out.println("Makanan Telah Di  
Hapus");  
  
System.out.println("=====");  
    }  
    else  
    {  
        System.out.println("Makanan Tidak  
Tersedia");  
  
System.out.println("=====");  
    }  
    }  
}  
static void update(String namamakanan,String  
jumlah_makanan, String harga)  
{  
    makanan.update(namamakanan,jumlah_makanan,  
harga);
```



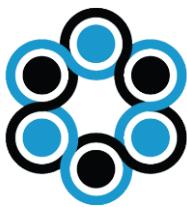
Tugas Resmi

Source Code

```
static void update(String namamakanan, String jumlah_makanan, String harga)
{
    makanan.update(namamakanan, jumlah_makanan,
harga);
}

static void loginPembeli()
{
    int pil;
    System.out.print("Nama Pembeli : ");
    String nama = input.next();
    System.out.print("Password : ");
    String password = input.next();
    cekPembeli=
AllObjectModel07162.pembeliModel.cekData(nama,
password);
    do
    {
        System.out.println("1. Data Makanan Yang
Tersedia");
        System.out.println("2. Exit");
        System.out.print("Pilih = ");
        pil = input.nextInt();

System.out.println("=====");
=====
        switch(pil)
        {
            case 1:view();
                ;
                break;
            }
        while (pil!=2);
    }
static void loginAdmin(){
    int pil;
    System.out.print("Masukkan Nama Admin : ");
    String nama = input.next();
    System.out.print("Password Admin : ");
    String password = input.next();
    cekAdmin =
AllObjectModel07162.adminModel.cekData(nama, password);
```

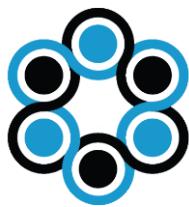


Tugas Resmi

Source Code

```
do{
    System.out.println("1. Input Data Makanan");
    System.out.println("2. View Data Makanan");
    System.out.println("3. Hapus Data Makanan");
    System.out.println("4. Update Data
Makanan");
    System.out.println("5. Exit");
    System.out.print("Pilih = ");
    pil = input.nextInt();

System.out.println("=====");
=====
switch(pil){
    case 1 : makanan();
        break;
    case 2 : view();
        break;
    case 3 : System.out.print("Inputkan Nama
makanan = ");
        String namamakanan =
input.next();
        remove(namamakanan);
        break;
    case 4 :
        for(int
i=0;i<makanan.cekDaftarMakanan(cekmakanan).size();i++)
        {
            System.out.print("Inputkan
Nama Makanan : ");
            namamakanan = input.next();
            System.out.print("Inputkan
Harga Baru : ");
            String newharga =
input.next();
            System.out.print("Inputkan
Jumlah Makanan : ");
            String newjumlah_makanan =
input.next();
            makanan.Insert(namamakanan,newjumlah_makanan,newharga);
            break;
        }
    }
}
while(pil!=5);
```



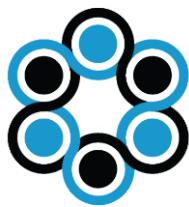
Tugas Resmi

Source Code

```
makanan.Insert(namamakanan,newjumlah_makanan,newharga);
break;
}
}
while(pil!=5);
}
}
```

Analisa :

Terdapat pilihan Daftar dan Login. Jika memilih Daftar akan diarahkan ke method daftar pembeli dengan menginputkan nama, password, alamat dan notelp. Jika memilih Login akan diarahkan ke Method Login Admin dan Login Pembeli. Jika memilih login Admin diarahkan ke method login admin lalu memilih pilihan 1 untuk menginputkan nama makanan, jumlah makanan, dan harga. Lalu pilih 2 untuk melihat data makanan, pilih 4 untuk mengupdate, di method update akan mengupdate jumlah makanan dan harga, tetapi sebelum itu mengisi nama makanan yang akan dihapus lalu pilih 2 untuk melihat makanan yang sudah diupdate. Pilih 3 untuk menghapus makanan yang akan dihapus. Jika memilih LoginPembeli maka akan diarahkan ke method login pembeli dengan mengisi nama dan password.



Tugas Resmi

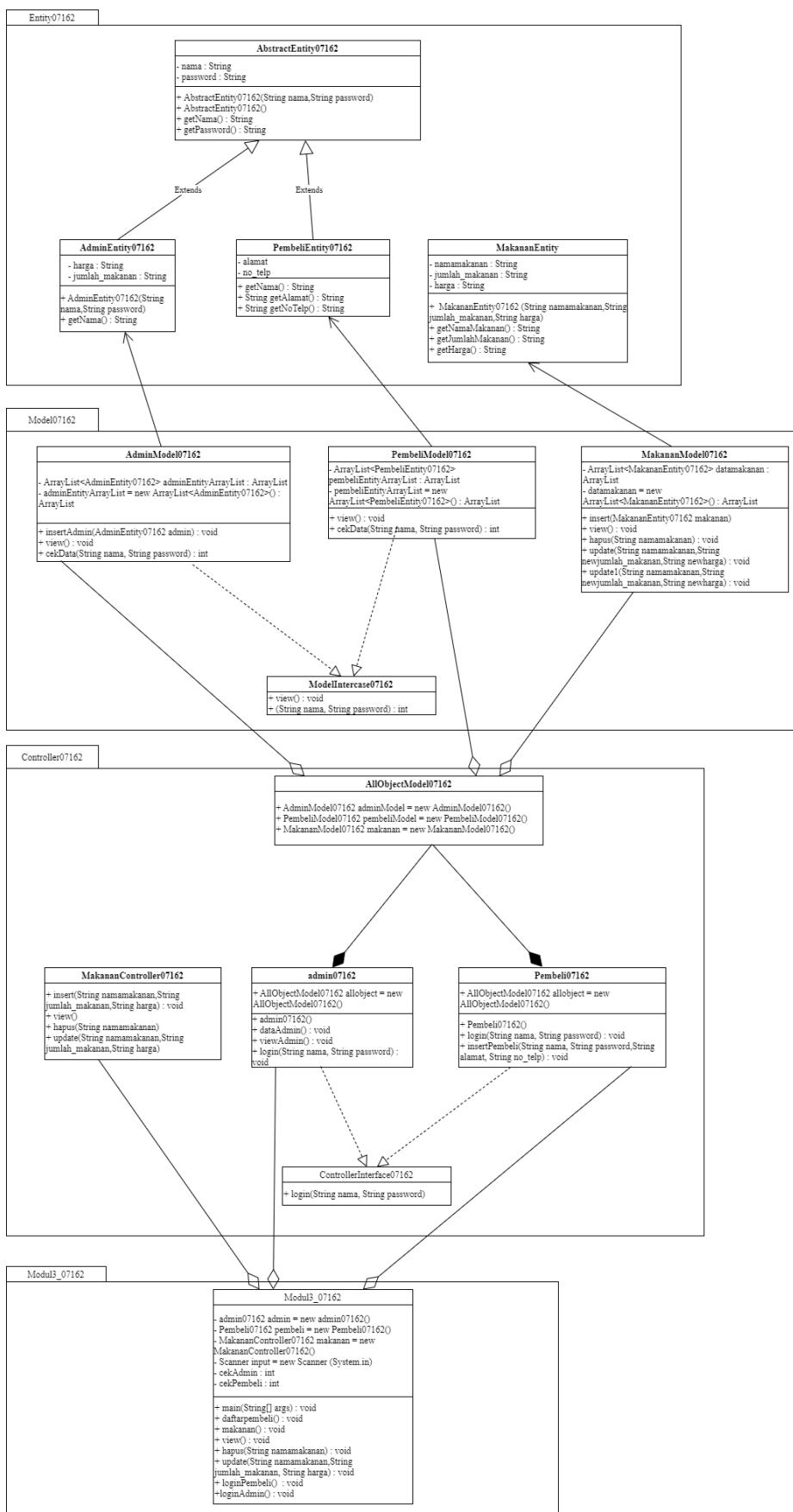
Screenshot

```
SELAMAT DATANG DI SUMAH MAKAN !!  
1. Daftar  
2. Login  
3. Exit  
Pilih = 1  
  
Input Name = annin  
Input Password = 123  
Alamat = mjk  
Input notelp = 088808577083  
Daftar Sukses !!  
1. Daftar  
2. Login  
3. Exit  
Pilih = 2  
  
1. Login Admin  
2. Login pembeli  
Pilih = 1  
Masukkan Nama Admin : Anindyaa  
Password Admin : 123  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 1  
  
INPUT NAMA MAKANAN = bakso  
INPUT JUMLAH MAKANAN = 1  
INPUT HARGA = 10000  
=====  
1. Input Data Makanan  
2. View Data Makanan  
  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 2  
  
Nama Makanan = bakso  
Jumlah Makanan = 1  
Harga = 10000  
=====  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 4  
  
Inputkan Nama Makanan : bakso  
Inputkan Harga Bakso : 18000  
Inputkan Jumlah Makanan : 3  
Data Telah Di Update  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 2  
  
Nama Makanan = bakso  
Jumlah Makanan = 3  
Harga = 18000  
=====  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 1  
  
Inputkan Nama makanan = bakso  
Makanan Telah Di Hapus  
=====  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 2  
  
1. Input Data Makanan  
2. View Data Makanan  
3. Hapus Data Makanan  
4. Update Data Makanan  
5. Exit  
Pilih = 6  
=====  
1. Daftar  
2. Login  
3. Exit  
Pilih = BUILD STOPPED (total time: 1 minute 31 seconds)
```

Class Diagram



Tugas Resmi

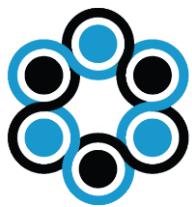




RESMI

MODUL 4

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



Tugas Resmi

Soal Praktikum

1. Terapkan *GUI* sesuai dengan *project* masing-masing.
2. Terapkan *CRUD* di *GUI* *project* kalian.
3. Terapkan *JOptionPane* ke *GUI* kalian.



Tugas Resmi

Langkah Pertama

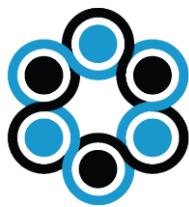
Membuat package Controller07162r, lalu membuat class dengan nama AllObjectModel07162

Source Code

```
package Controller07162;
import Model07162.*;
public class AllObjectModel07162 {
    public static DaftarPembeliModel07162
daftarPembelimodel = new DaftarPembeliModel07162();
    public static AdminModel07162 adminModel = new
AdminModel07162 ();
    public static PembeliModel07162 pembeliModel = new
PembeliModel07162();
}
```

Analisa

Kelas yang diciptakan sebagai penghubung antara class – class yang berada pada package Model dan Entity



Tugas Resmi

Langkah Kedua

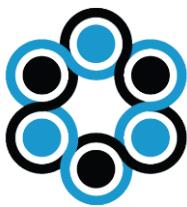
Membuat class dari package Controller07162 dengan nama ControllerInterface07162

Source Code

```
package Controller07162;
public interface ControllerInterface07162 {
    public void login(String nama, String password);
}
```

Analisa

Di class ControllerInterface07162 terdapat void login yang berisi nama dan password untuk masuk sebagai admin/pembeli.



Tugas Resmi

Langkah Ketiga

Membuat class dari package Controller07162 dengan nama AdminController07162

Source Code

```
package Controller07162;
import Entity.*;
import java.util.ArrayList;
public class AdminController07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public AdminController07162() {
}
    public void dataAdmin() {
        String passwordAdmin[] = {"123"};
        String namaAdmin[] = {"Anindya"};
        for(int i = 0; i < namaAdmin.length; i++) {

AllObjectModel07162.adminModel.insertAdmin(new
AdminEntity07162 (namaAdmin[i],passwordAdmin[i]));
        }
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.adminModel.cekData(nama, password);
    }
    public AdminEntity07162 AdminEntity07162() {
        return
AllObjectModel07162.adminModel.showDataAdmin(indexLogin)
    ;
    }
    public void updateIsVerified(int index,int
indexMakanan,PembeliEntity07162 pembeli ) {

AllObjectModel07162.daftarPembeliModel.updateIsVerified(
index, new
        DaftarPembeliEntity07162 (indexMakanan,
pembeli,true));
    }
    public ArrayList<DaftarPembeliEntity07162>
cekDaftarPembeliModel() {
        return
AllObjectModel07162.daftarPembeliModel.getdaftarpembeliA
rrayList();
}
```



Tugas Resmi

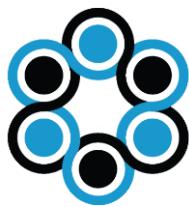
Source Code

```
public ArrayList<DaftarPembeliEntity07162>
cekDaftarPembeliModel() {
    return
AllObjectModel07162.daftarPembelimodel.getdaftarpembeliA
rrayList();
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ControllerInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ControllerInterface07162.

Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Resmi

Langkah Keempat

Membuat class dari package Controller07162 dengan nama
PembeliController07162

Source Code

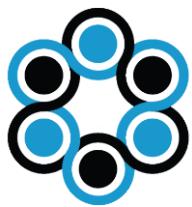
```
package Controller07162;
import Entity.*;
import javax.swing.table.DefaultTableModel;
import java.util.ArrayList;
public class PembeliController07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public PembeliController07162() {
    }
    public PembeliEntity07162 getData() {
        return
AllObjectModel07162.pembeliModel.getpembeliEntityArrayList(indexLogin);
    }

    private static class pembeli {

        public pembeli() {
        }
    }

    public void DaftarMakanan (int indexMakanan,
PembeliEntity07162 pembeli, boolean isVerified) {

AllObjectModel07162.daftarPembelimodel.insertDataPembeli
(new
            DaftarPembeliEntity07162 (indexMakanan, pembeli,
isVerified));
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.pembeliModel.cekData(nama,
password);
    }
    public void insert(String nama, String password,
String alamat) {
        AllObjectModel07162.pembeliModel.insert(new
PembeliEntity07162(nama,password,alamat));
    }
}
```



Tugas Resmi

Source Code

```
}

    public int cekDaftarPembeli (String nama ) {
        int cek =
AllObjectModel07162.daftarPembelimodel.cekData(nama,
null);
        return cek;
    }

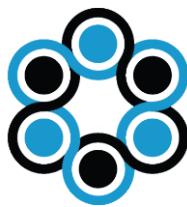
    public DaftarPembeliEntity07162
showDaftarPembeli(int index) {
        return
AllObjectModel07162.daftarPembelimodel.showDaftarPembeli
(index);
    }

    public void update(String makanan, int
Updatemakanan) {

AllObjectModel07162.daftarPembelimodel.update(makanan,
Updatemakanan);
}
    public void deleteDataPembeli(int index) {

AllObjectModel07162.daftarPembelimodel.hapusPembeliModel
(index);
}
    public DefaultTableModel daftarpem(){
        DefaultTableModel dtmdaftarpem = new
DefaultTableModel();
        Object[] kolom
={"Password","Nama","Alamat","Makanan","Verified"};
        dtmdaftarpem.setColumnIdentifiers(kolom);
        int size =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().
size();
        String verif = null;
        for(int i=0;i<size;i++){

if(AllObjectModel07162.daftarPembelimodel.alldatapembeli
().get(i).isIsVerified()==false){
            verif = "no";
        }else{
            verif = "yes";
        }
    }
}
```



Tugas Resmi

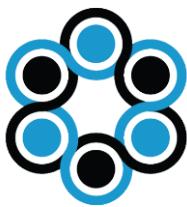
Source Code

```
        }else{
            verif = "yes";
        }
        Object [] data = new Object[6];
        data[0] =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().get(i).getPembeli().getNama();
            data[1] =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().get(i).getPembeli().getPassword();
            data[2] =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().get(i).getPembeli().getAlamat();
            data[3] =
MakananEntity07162.makanan[AllObjectModel07162.daftarPembelimodel.alldatapembeli().get(i).getIndexMakanan()];
            data[4] = verif;
            dtmdaftarpem.addRow(data);
        }
        return dtmdaftarpem;
    }
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ControllerInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ControllerInterface07162.

Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override. Kemudian terdapat method daftarpel, ini berguna untuk membuat sebuah tabel untuk daftar pelanggan pada gui.



Tugas Resmi

Langkah Kelima

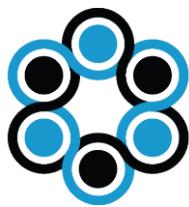
Membuat Package Entity lalu membuat class dengan nama MakananEntity07162

Source Code

```
package Entity;
public class MakananEntity07162 {
    public static String [] makanan = { "bakso = 5000 ",
    "mie ayam = 8000", "soto = 10000 ", "pecel = 7000" ,
    "rawon = 10000" };
```

Analisa

Di class MakananEntity07162 terdapat Class data array yang berisi daftar pilihan makanan, bertipe data String. Class tidak menerapkan inheritance dan polymorphism. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu public. Public memiliki sifat bahwa properti dan method bisa diakses oleh semua class.



Tugas Resmi

Langkah Keenam

Membuat class dari package Entity dengan nama AbstractEntity07162

Source Code

```
package Entity;

public class AbstractEntity07162 {
    protected String nama,password;

    public AbstractEntity07162(String nama, String password) {
        this.nama = nama;
        this.password = password;
    }

    public AbstractEntity07162(){
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
}
```

Analisa

Class AbstractEntity07162 digunakan sebagai superclass/parent class , untuk penerapan inheritance pada project. Terdapat penerapan encapsulasi dalam project, yaitu menggunakan Protected pada properti.dimana protected dapat diakses oleh class itu sendiri dan class turunan.



Tugas Resmi

Langkah Ketujuh

Membuat class dari package Entity dengan nama AdminEntity07162

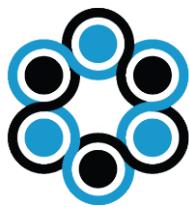
Source Code

```
package Entity;

public class AdminEntity07162 extends
AbstractEntity07162 {
    public AdminEntity07162(String nama,String
password) {
        super (nama,password);
    }
    @Override
    public String getNama(){
        return nama;
    }
}
```

Analisa

Pada class AdminEntity07162 menerapkan inheritance/pewarisan, dimana class AbstractEntity07162 menurunkan method dan property kepada AdminEntity07162. Pada keyword super, ini digunakan untuk memanggil konstruktor dari class AbstractEntity07162. Override ini untuk pendefinisian ulang method yang sama pada class AbstractEntity07162



Tugas Resmi

Langkah Kedelapan

Membuat class dari package Entity dengan nama PembeliEntity07162.

Source Code

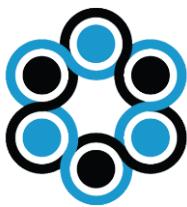
```
package Entity;

public class PembeliEntity07162 extends AbstractEntity07162
{
    private String alamat;

    public PembeliEntity07162(String nama, String
password, String alamat) {
        super(nama, password);
        this.alamat = alamat;
    }
    @Override
    public String getNama() {
        return nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }
}
```

Analisa

Pada class PembeliEntity07162 menerapkan inheritance/pewarisan, dimana class AbstractEntity07162 menurunkan method dan property kepada PembeliEntity07162 . Pada keyword super, ini digunakan untuk memanggil konstruktor dari class AbstractEntity07162. Override ini untuk pendefinisian ulang method yang sama pada class AbstractEntity07162.



Tugas Resmi

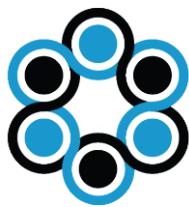
Langkah keSembilan

Membuat class dari package Entity dengan nama DaftarPembeliEntity07162.
Source Code

```
package Entity;
public class DaftarPembeliEntity07162 {
    private PembeliEntity07162 pembeli;
    private boolean isVerified;
    private int indexMakanan;
    public DaftarPembeliEntity07162(int indexMakanan,
                                    PembeliEntity07162 pembeli, boolean
    isVerified) {
        this.indexMakanan = indexMakanan;
        this.pembeli = pembeli;
        this.isVerified = isVerified;
    }
    public PembeliEntity07162 getPembeli() {
        return pembeli;
    }
    public boolean isIsVerified() {
        return isVerified;
    }
    public int getIndexMakanan() {
        return indexMakanan;
    }
    public void setPembeli(PembeliEntity07162 pembeli) {
        this.pembeli= pembeli;
    }
    public void setIsVerified(boolean isVerified) {
        this.isVerified = isVerified;
    }
    public void setIndexMakanan(int indexMakanan) {
        this.indexMakanan = indexMakanan;
    }
}
```

Analisa

Di class DaftarPembeliEntity07162 terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat variable untuk menampung data pelanggan, serta isVerified untuk mengonfirmasi status pelanggan nantinya, dan juga indexMakanan untuk menginput data pilihan makanan.



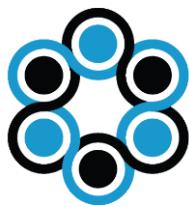
Tugas Resmi

Langkah KeSepuluh

Membuat package Model dengan nama Model07162 lalu membuat class dengan nama AdminModel07162

Source Code

```
package Model07162;
import Entity.AdminEntity07162;
import java.util.ArrayList;
public class AdminModel07162 implements
ModelInterface07162
{
    private ArrayList<AdminEntity07162>
adminEntityArrayList;
    public AdminModel07162()
    {
        adminEntityArrayList = new
ArrayList<AdminEntity07162>();
    }
    public void insertAdmin(AdminEntity07162 admin)
    {
        adminEntityArrayList.add(admin);
    }
    public void view() {
        for (AdminEntity07162 adminEntity :
adminEntityArrayList) {
            System.out.println(adminEntity.getNama());
            System.out.println(adminEntity.getPassword());
        }
    }
    @Override
    public int cekData(String nama, String password){
        int loop = 0;
        for(AdminEntity07162 admin :
adminEntityArrayList){
            if(admin.getNama().equals(nama) &&
admin.getPassword().equals(password)){
                break;
            }else{
                loop++;
            }
        }
        return loop;
    }
    public AdminEntity07162 showDataAdmin(int index){
```



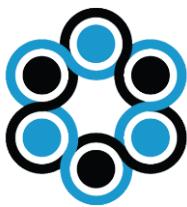
Tugas Resmi

Source Code

```
public AdminEntity07162 showDataAdmin(int index){  
    return adminEntityArrayList.get(index);  
}  
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ModelInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ModelInterface07162. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pen definisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Resmi

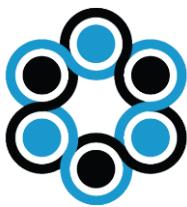
Langkah keSembelas

Membuat class dengan nama DaftarPembeliModel07162.

Source Code

```
package Model07162;
import Entity.*;
import java.util.ArrayList;
import java.text.ParseException;
public class DaftarPembeliModel07162 implements
ModelInterface07162 {
    private ArrayList<DaftarPembeliEntity07162>
daftarpembeliArrayList;
    public DaftarPembeliModel07162(){
        daftarpembeliArrayList = new
ArrayList<DaftarPembeliEntity07162>();
    }
    public void
insertDataPembeli(DaftarPembeliEntity07162
daftarpembeli){
        daftarpembeliArrayList.add(daftarpembeli);
    }
    public ArrayList<DaftarPembeliEntity07162>
getdaftarpembeliArrayList(){
        return daftarpembeliArrayList;
    }
    public int size() {
        return daftarpembeliArrayList.size();
    }
    @Override
    public void view() {
        for (DaftarPembeliEntity07162 pembelientity :
daftarpembeliArrayList) {

System.out.println("=====");
System.out.println("nama : " +
pembelientity.getPembeli().getNama()+"\nPassword : " +
pembelientity.getPembeli().getPassword() + "\nAlamat :
"+pembelientity.getPembeli().getAlamat()+"\n Makanan :
"+MakananEntity07162.makanan[pembelientity.getIndexMakanan()] + "\n isVerified : ");
        if(pembelientity.isIsVerified()==false) {
            System.out.println("Belum Di Verifikasi
Admin");
        } else {
```



Tugas Resmi

Source Code

```
System.out.println("Sudah Di Verifikasi Admin");
    }

System.out.println("=====");
=====
}
@Override
public int cekData(String nama, String password ){
    int loop = 0;
    if(daftarpembeliArrayList.size() ==0){
        loop = -1;//data kosong
    }else{
        for(int i = 0; i <daftarpembeliArrayList.size(); i++){
            if(daftarpembeliArrayList.get(i).getPembeli().getPassword().equals(password)){
                loop = i;
                break;
            }else{
                loop = -2;
            }
        }
    }    return loop;
}
public DaftarPembeliEntity07162
showDaftarPembeli(int index){
    return daftarpembeliArrayList.get(index);
//menampilkan data pembeli
}
public void hapusPembeliModel(int index) {

daftarpembeliArrayList.remove(daftarpembeliArrayList.get(index)); //menghapus data pembeli
}
public void updateIsVerified(int
index, DaftarPembeliEntity07162 daftarpembeli){
    daftarpembeliArrayList.set(index, daftarpembeli);
}
public ArrayList <DaftarPembeliEntity07162>
alldatapembeli(){
    return daftarpembeliArrayList;
}
private int cariMakanan(String makanan){
```



Tugas Resmi

Source Code

```
private int cariMakanan(String makanan){
    int index=-1;
    for(int i=0;i<daftarpembeliArrayList.size();i++) {

        if(makanan.equals(daftarpembeliArrayList.get(i).getPembeli().getNama()))
            index=i;
    }
    return index;
}
public void update(String makanan, int Updatemakanan){
    int data;
    if(cariMakanan(makanan)!=-1){

        daftarpembeliArrayList.get(cariMakanan(makanan)).setIndexMakanan(Updatemakanan);
        data = 1;
    }
    else {
        data = 0;
    }
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ModelInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ModelInterface07162. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Resmi

Langkah KeSebelas

Membuat class interface class dengan nama Modelinterface07162.

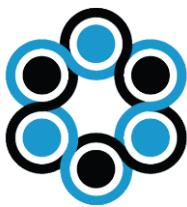
Source Code

```
package Model07162;

public interface ModelInterface07162 {
    public void view();
    public int cekData (String nama, String
password);
}
```

Analisa

Kelas interface hanya terdiri dari method-method kosong (abstract, tanpa body method). Interface biasanya lebih digunakan untuk mengelompokkan behavior menjadi grup behavior. Kelas interface menggambarkan kemampuan, bukan entitas/konsep objek



Tugas Resmi

Langkah keDuaBebelas

Membuat class interface dengan nama PembeliModel07162

Source Code

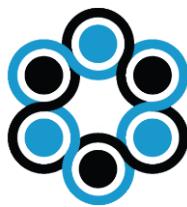
```
package Model07162;
import Entity.PembeliEntity07162;
import java.util.ArrayList;
public class PembeliModel07162 implements
ModelInterface07162
{
    private ArrayList<PembeliEntity07162>
pembeliEntityArrayList;

    public PembeliModel07162()
    {
        pembeliEntityArrayList = new
ArrayList<PembeliEntity07162>();
    }
    public void insert(PembeliEntity07162
pembeliEntity){ //menambah data pembeli
    pembeliEntityArrayList.add(pembeliEntity);
}

@Override
public void view() {
    for (PembeliEntity07162 pembeliEntity :
pembeliEntityArrayList) {

System.out.println("=====");
System.out.println("nama : "+
pembeliEntity.getNama()+"\npassword : "+
pembeliEntity.getPassword()+"\nAlamat :
"+pembeliEntity.getAlamat()));

System.out.println("=====");
}
}
public int cekData(String nama, String password){
    int loop = 0;
    while
(!pembeliEntityArrayList.get(loop).getNama().equals(nam
a) &&
!pembeliEntityArrayList.get(loop).getPassword().equals(p
assword))
loop++;
}
}
```



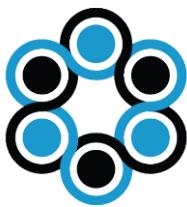
Tugas Resmi

Source Code

```
public int cekData(String nama, String password){  
    int loop = 0;  
    while  
(!pembeliEntityArrayList.get(loop).getNama().equals(nam  
a) &&  
  
!pembeliEntityArrayList.get(loop).getPassword().equals(  
password)){  
        loop++;  
    }  
    return loop;  
}  
public PembeliEntity07162  
getpembeliEntityArrayList(int index){ //  
    return pembeliEntityArrayList.get(index);  
}  
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ModelInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ModelInterface07162. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Resmi

Langkah KeTigaBelas

Membuat Package dengan nama View07162 lalu buat class dengan nama AdminGUI07162.

Source Code

```
package View07162;
import Controller07162.AllObjectModel07162;
import Controller07162.PembeliController07162;
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import javax.swing.*;
public class AdminGUI07162 {
    private static PembeliController07162 pembeli = new
PembeliController07162();
    JFrame admin = new JFrame();
    JTable tabelpembeli = new JTable();
    JScrollPane scrolladmin = new
JScrollPane(tabelpembeli);
    JButton back,verif,hapus;
    JLabel Admin,passlabel,namalabel;
    JTextField passtext,namatext;
    public AdminGUI07162(){
        admin.setSize(650,500);
        admin.setLayout(null);

        admin.getContentPane().setBackground(Color.ORANGE);

        Admin = new JLabel("Admin");
        admin.setBounds(250, 30, 200, 30);
        admin.setFont(new Font("Timer New
Roman",Font.BOLD,40));
        admin.add(Admin);

        scrolladmin.setBounds(30, 100, 570, 200);
        tabelpembeli.setModel(pembeli.daftarpem());
        admin.add(scrolladmin);

        namalabel = new JLabel("NAMA");
        namalabel.setBounds(50, 320, 100, 30);
        admin.add(namalabel);

        passtext = new JTextField("PASSWORD");
        passtext.setBounds(150, 320, 150, 30);
        admin.add(passtext);

        verif = new JButton("VERIFIKASI");
        verif.setBounds(300, 320, 100, 30);
        admin.add(verif);

        h

```



Tugas Resmi

Source Code

```
namalabel = new JLabel("NAMA");
namalabel.setBounds(50, 320, 100, 30);
admin.add(namalabel);

namatext = new JTextField();
namatext.setBounds(50, 350, 100, 30);
admin.add(namatext);

passlabel = new JLabel("Password");
passlabel.setBounds(200, 320, 100, 30);
admin.add(passlabel);

passtext = new JTextField();
passtext.setBounds(200, 350, 100, 30);
admin.add(passtext);

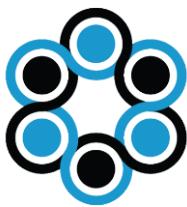
verif = new JButton("verif");
verif.setBounds(350, 350, 100, 30);
verif.setBackground(Color.GRAY);
admin.add(verif);

back = new JButton("back");
back.setBounds(50, 400, 100, 30);
back.setBackground(Color.GRAY);
admin.add(back);

hapus = new JButton("Hapus");
hapus.setBounds(350, 400, 100, 30);
hapus.setBackground(Color.GRAY);
admin.add(hapus);

admin.setLocationRelativeTo(null);
admin.setVisible(true);

admin.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
back.addActionListener(new ActionListener()
{
    @Override
    public void actionPerformed(ActionEvent
ae){
        admin.dispose();
        GUI07162 men = new GUI07162(); //nama
objek men
    }
    ..
})
```



Tugas Resmi

Source Code

```
    }
    });
    verif.addActionListener(new ActionListener(){
        @Override
        public void actionPerformed(ActionEvent ae){ //untuk verifikasi nama dan password
            String nama = namatext.getText();
            String pass = passtext.getText();
            int index =
AllObjectModel07162.daftarPembelimodel.cekData(nama,
pass); //dicek

Allobjctrl07162.admin.updateIsVerified(index,
Allobjctrl07162.Pembeli.showDaftarPembeli(index).getInd
exMakanan(),Allobjctrl07162.Pembeli.showDaftarPembeli(i
ndex).getPembeli());

tabelpembeli.setModel(pembeli.daftarpem());
}
});
hapus.addActionListener(new ActionListener(){
    @Override
    public void actionPerformed(ActionEvent ae){

Allobjctrl07162.Pembeli.deleteDataPembeli(0);

tabelpembeli.setModel(pembeli.daftarpem());
}
});
tabelpembeli.addMouseListener(new
MouseAdapter(){
    @Override
    public void mouseClicked(MouseEvent e){
//method untuk kursor
        int i = tabelpembeli.getSelectedRow();

namatext.setText(Allobjctrl07162.Pembeli.daftarpem().ge
tValueAt(i, 0).toString());

passtext.setText(Allobjctrl07162.Pembeli.daftarpem().ge
tValueAt(i, 1).toString());
}
});
}
```



Tugas Resmi

Source Code

```
public void mouseClicked(MouseEvent e){ //method  
untuk kursor  
    int i = tabelpembeli.getSelectedRow();  
  
    namatext.setText(Allobjctrl07162.Pembeli.daftarpem().get  
ValueAt(i, 0).toString());  
  
    passtext.setText(Allobjctrl07162.Pembeli.daftarpem().get  
ValueAt(i, 1).toString());  
    }  
});  
}  
}
```

Analisa

Pada class AdminGUI07162 merupakan tampilan GUI pada saat kita login sebagai admin, kita bisa verifikasi data pembeli dan hapus pembeli



Tugas Resmi

Langkah keEmpatBelas

Membuat kelas dari Package view07162 dengan nama PembeliGUI07162
Source Code

```
package View07162;
import Entity.MakananEntity07162;
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.text.SimpleDateFormat;
import javax.swing.*;
public class PembeliGUI07162 {
    JFrame Pembeli = new JFrame();
    JButton back,daftarbtn;
    JTextArea area = new JTextArea();
    JLabel datapembeli,daftarmakanan;
    JComboBox pilmakanan = new
    JComboBox(MakananEntity07162.makanan);
    int cek = Allobjctrl07162.Pembeli.cekDaftarPembeli
    (Allobjctrl07162.Pembeli.getData().getPassword());
    public PembeliGUI07162(){
        Pembeli.setSize(720, 600);
        Pembeli.setLayout(null);

        Pembeli.getContentPane().setBackground(Color.PINK);

        datapembeli = new JLabel("Data Pembeli");
        datapembeli.setFont(new Font("Timer New
Roman",Font.BOLD,30));
        datapembeli.setBounds(90, 30, 200, 30);
        Pembeli.add(datapembeli);

        area.setBounds(30, 90, 400, 300);
        Pembeli.add(area);

        daftarmakanan = new JLabel("Daftar Makanan");
        daftarmakanan.setBounds(450, 30, 250, 30);
        daftarmakanan.setFont(new Font("Times New
Roman",Font.BOLD,30));
        Pembeli.add(daftarmakanan);

        pilmakanan.setBounds(450, 90, 230, 30);
        Pembeli.add(pilmakanan);
```



Tugas Resmi

```
daftarbtn = new JButton("Daftar");
daftarbtn.setBounds(500, 200, 100, 30);
daftarbtn.setBackground(Color.GRAY);
Pembeli.add(daftarbtn);

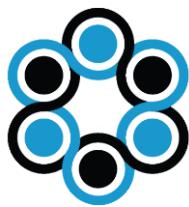
back = new JButton("Back");
back.setBounds(30, 500, 100, 30);
back.setBackground(Color.red);

Pembeli.add(back);
Pembeli.setVisible(true);
Pembeli.setLocationRelativeTo(null);

Pembeli.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
if(cek== -1){ //cek data, apabila kita belum
daftar pembeli
    JOptionPane.showMessageDialog(null, "Anda
Belum Daftar sebagai pembeli", "Information",
JOptionPane.INFORMATION_MESSAGE);
} else if(cek== -2){
    JOptionPane.showMessageDialog(null, "Anda
Belum Daftar sebagai pembeli", "Information",
JOptionPane.INFORMATION_MESSAGE);
} else{
    area.setText(datapembeli());
}
daftarbtn.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent ae){
    int indexmakanan =
pilmakanan.getSelectedIndex();
    Allobjctrl07162.Pembeli.DaftarMakanan

(indexmakanan,Allobjctrl07162.Pembeli.getData(), false);
    area.setText(datapembeli());
}
});

back.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent ae){
    Pembeli.dispose();
    GUI07162 men = new GUI07162(); //nama
objek menu
}
})
```



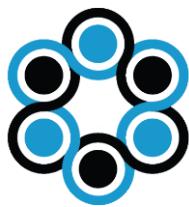
Tugas Resmi

```
        }
    });
}
String datapembeli(){
    int cek =
Allobjctrl07162.Pembeli.cekDaftarPembeli(Allobjctrl0716
2.Pembeli.getData().getNama());
    String cekverif;

if(Allobjctrl07162.Pembeli.showDaftarPembeli(cek).isIsV
erified()==false){
    cekverif = "belum diverifikasi";
}else{
    cekverif = "sudah diverifikasi";
}
String text = "Nama =
"+Allobjctrl07162.Pembeli.showDaftarPembeli(cek).getPem
beli().getNama()+"\n"
+ "Alamat =
"+Allobjctrl07162.Pembeli.showDaftarPembeli(cek).getPem
beli().getAlamat()+"\n"
+ "Verifikasi = "+cekverif+"\n"
+ "Makanan =
"+MakananEntity07162.makanan[Allobjctrl07162.Pembeli.sh
owDaftarPembeli(cek).getIndexMakanan()];
    return text;
}
}
```

Analisa :

Pada class PembeliGUI07162, pada kelas ini merupakan tampilan GUI pada saat kita login sebagai pembeli, kita bisa daftar makanan terlebih dahulu, ketika sudah daftar, makan akan muncul data pembeli



Tugas Resmi

Langkah keLimaBelas

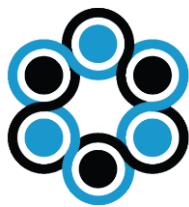
Membuat class dari package view07162 lalu beri nama Allobjctrl07162

Source Code

```
package View07162;
import Controller07162AdminController07162;
import Controller07162PembeliController07162;
public class Allobjctrl07162 {
    public static AdminController07162 admin = new
AdminController07162();
    public static PembeliController07162 Pembeli = new
PembeliController07162();
}
```

Analisa :

*Class Allobjctrl07162 untuk memanggil objek AdminController07162 dan
PembeliController07162*



Tugas Resmi

Langkah keEnamBelas

Membuat class dari package view07162 lalu beri nama GUI07162

Source Code

```
package View07162;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;

public class GUI07162 {
    JFrame TampilanAwal = new JFrame(); //tampilan awal
    JLabel login,daftar,top;
    JRadioButton radioadmin,radio_pembeli; //tombol
    pilihan
    JTextField textnamalogin,textnamadaftar, textalamat;
    JLabel
    labelnamalogin,labelpasswordlogin,labelnamadaftar, labelpa
    ssworddaftar,labelalamat;
    JButton check,reg,update;
    JPasswordField passwordlogin,passworddaftar;

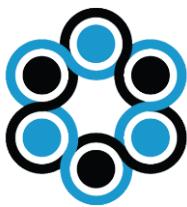
    private boolean ceklogin = false;
    public GUI07162(){ //komponen
        TampilanAwal.setSize(700, 630); //ukuran
        TampilanAwal.setLayout(null);

        TampilanAwal.getContentPane().setBackground(Color.YELLOW)
        ; //warna background

        top = new JLabel("Rumah Makan");
        top.setBounds(225, 10, 600, 50);
        top.setFont(new Font("Times New
        Roman",Font.BOLD,40));
        TampilanAwal.add(top);

        login = new JLabel("Login");
        login.setBounds(100, 50, 100, 100);
        login.setFont(new Font("Times New
        Roman",Font.BOLD,30));
        TampilanAwal.add(login);

        radioadmin = new JRadioButton("admin");
        radioadmin.setBounds(30, 150, 100, 30);
        radioadmin.setBackground(Color.CYAN);
        - - - - -
```



Tugas Resmi

Source Code

```
TampilanAwal.add(radioadmin);

radiopembeli = new JRadioButton("pembeli");
radiopembeli.setBounds(140, 150, 100, 30);
radiopembeli.setBackground(Color.CYAN);
TampilanAwal.add(radiopembeli);

labelnamalogin = new JLabel("Nama");
labelnamalogin.setBounds(30, 180, 50, 30);
TampilanAwal.add(labelnamalogin);

textnamalogin = new JTextField();
textnamalogin.setBounds(30, 210, 200, 30);
TampilanAwal.add(textnamalogin);

labelpasswordlogin = new JLabel("Password");
labelpasswordlogin.setBounds(30, 240, 100, 30);
TampilanAwal.add(labelpasswordlogin);

passwordlogin = new JPasswordField();
passwordlogin.setBounds(30, 270, 200, 30);
TampilanAwal.add(passwordlogin);

check = new JButton("check");
check.setBounds(90, 320, 100, 40);
check.setBackground(Color.CYAN);
TampilanAwal.add(check);

update = new JButton("update");
update.setBounds(90, 400, 100, 40);
update.setBackground(Color.CYAN);
TampilanAwal.add(update);

daftar = new JLabel("Buat Akun");
daftar.setFont(new Font("Times New
Roman",Font.BOLD,30));
daftar.setBounds(440, 50, 150, 100);
TampilanAwal.add(daftar);
labelnamadaftar = new JLabel("Nama");
labelnamadaftar.setBounds(400, 210, 100, 30);
TampilanAwal.add(labelnamadaftar);
textnamadaftar = new JTextField();
textnamadaftar.setBounds(400, 240, 200, 30);
TampilanAwal.add(textnamadaftar);
....
```



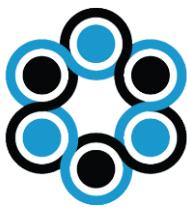
Tugas Resmi

Source Code

```
labelpassworddaftar = new JLabel("Password");
    labelpassworddaftar.setBounds(400, 270, 100,
30);
    TampilanAwal.add(labelpassworddaftar);
    passworddaftar = new JPasswordField();
    passworddaftar.setBounds(400, 300, 200, 30);
    TampilanAwal.add(passworddaftar);
    labelalamat = new JLabel("Alamat");
    labelalamat.setBounds(400, 330, 100, 30);
    TampilanAwal.add(labelalamat);
    textalamat = new JTextField();
    textalamat.setBounds(400, 360, 200, 30);
    TampilanAwal.add(textalamat);

    reg = new JButton("Daftar");
    reg.setBounds(460, 520, 100, 40);
    reg.setBackground(Color.CYAN);
    TampilanAwal.add(reg);

TampilanAwal.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    TampilanAwal.setVisible(true);
    TampilanAwal.setLocationRelativeTo(null);
    radioadmin.addActionListener(new
ActionListener(){
        @Override
        public void actionPerformed(ActionEvent ae){
            if(radioadmin.isSelected()){
                radiopembeli.setSelected(false);
                ceklogin=true;
            }
        }
    });
    radiopembeli.addActionListener(new
ActionListener(){ //untuk pembeli
        @Override
        public void actionPerformed(ActionEvent ae){
            if(radiopembeli.isSelected()){
                radioadmin.setSelected(false);
                ceklogin=false;
            }
        }
    });
    check.addActionListener(new ActionListener(){
        @Override
```



Tugas Resmi

Source Code

```
public void actionPerformed(ActionEvent ae){ // login untuk admin
    if(ceklogin==true){
        try{
            Allobjctrl07162.admin.dataAdmin();

            Allobjctrl07162.admin.login(textnamalogin.getText(),
            passwordlogin.getText());
            String nama =
            Allobjctrl07162.admin.AdminEntity07162().getNama();

            JOptionPane.showMessageDialog(null, "Selamat datang "+nama,"information", JOptionPane.INFORMATION_MESSAGE);
            AdminGUI07162 admin = new AdminGUI07162();
            TampilanAwal.dispose();
        }catch(Exception exception){

            JOptionPane.showMessageDialog(null, "Nama atau Password salah", "information", JOptionPane.INFORMATION_MESSAGE);
        }
        kosong(); //method kosong, jika datanya kosong
    }else{
        try{ //login untuk pembeli
            Allobjctrl07162.Pembeli.login(textnamalogin.getText(),passwordlogin.getText());
            String nama =
            Allobjctrl07162.Pembeli.getData().getNama();
            //memasukkan nama dan password jika benar

            JOptionPane.showMessageDialog(null, "Selamat datang "+nama,"information", JOptionPane.INFORMATION_MESSAGE);
            PembeliGUI07162 prak = new PembeliGUI07162();
            TampilanAwal.dispose();
        }catch(Exception eception){ //jika nama atau password salah makan akan muncul tulisan

            JOptionPane.showMessageDialog(null, "Nama atau Password salah", "information", JOptionPane.INFORMATION_MESSAGE);
            kosong();
        }
    }
}
```



Tugas Resmi

Source Code

```
        }
    }
});
update.addActionListener(new ActionListener(){

    @Override
    public void actionPerformed(ActionEvent ae)
{
    Update07162 update =new Update07162();
    TampilanAwal.dispose();
}

});

reg.addActionListener(new ActionListener(){
    @Override
    public void actionPerformed(ActionEvent ae){
        try{
            String nama =
textnamadaftar.getText(); //registrasi
            String password =
passworddaftar.getText();
            String alamat =
textalamat.getText();

Allobjctrl07162.Pembeli.insert(nama,password,alamat);
//jika sukses maka muncul tulisan
            JOptionPane.showMessageDialog(null,
"Registrasi sukses", "information",
JOptionPane.INFORMATION_MESSAGE);
            kosong();
        }catch(Exception exception){
            JOptionPane.showMessageDialog(null,
"Format Penulisan Salah", "Registrasi Gagal",
JOptionPane.INFORMATION_MESSAGE);
        }
    }
});
void kosong(){
    textnamadaftar.setText(null); //settext null
semua
    passworddaftar.setText(null);
}
```



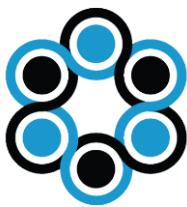
Tugas Resmi

Source Code

```
void kosong(){
    textnamadaftar.setText(null); //settext null
semua
    passworddaftar.setText(null);
    textalamat.setText(null);
    textnamalogin.setText(null);
    passwordlogin.setText(null);
}
}
```

Analisa :

Pada class GUI07162, pada kelas ini merupakan tampilan GUI Utama pada saat pertama kali Run. pada Gui ini kita dapat daftar pembelin, login sebagai admin dan login sebagai pembeli



Tugas Resmi

Langkah keTujuhBelas

Membuat class dari package view07162 lalu beri nama Update07162

Source Code

```
package View07162;
import Entity.MakananEntity07162;
import java.awt.Color;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.Font;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
public class Update07162 {
    JComboBox pilmakanan = new
JComboBox(MakananEntity07162.makanan);
    JButton konfirmasi,kembali;
    JFrame TampilanAwal = new JFrame();
    JTextArea area = new JTextArea();
    JLabel update,labelmakananUpdate;
    JTextField textmakananUpdate;
    public Update07162() {
        TampilanAwal.setSize(720,600);
        TampilanAwal.setLayout(null);

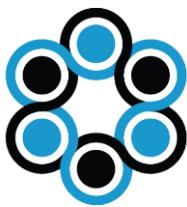
        TampilanAwal.getContentPane().setBackground(Color.LIGHT
_GRAY);

        update = new JLabel("Update Data");
        update.setFont(new Font("Times New
Roman",Font.BOLD,43));
        update.setBounds(240,30,300,80);
        TampilanAwal.add(update);

        labelmakananUpdate = new JLabel("Nama
Pembeli");
        labelmakananUpdate.setBounds(300,180,150,30);
        TampilanAwal.add(labelmakananUpdate);

        textmakananUpdate = new JTextField();
        textmakananUpdate.setBounds(250,210,200,40);
        TampilanAwal.add(textmakananUpdate);

        pilmakanan.setBounds(276,270,150,40);
        TampilanAwal.add(pilmakanan);
```



Tugas Resmi

Source Code

```
pilmakanan.setBounds(276,270,150,40);
    TampilanAwal.add(pilmakanan);

    konfirmasi = new JButton("KONFIRMASI");
    konfirmasi.setBounds(290,330,120,40);
    konfirmasi.setBackground(Color.RED);
    TampilanAwal.add(konfirmasi);

    kembali = new JButton("KEMBALI");
    kembali.setBounds(290,380,120,40);
    kembali.setBackground(Color.GREEN);
    TampilanAwal.add(kembali);

TampilanAwal.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    TampilanAwal. setVisible(true);
    TampilanAwal.setLocationRelativeTo(null);

    konfirmasi.addActionListener(new
ActionListener(){

        @Override
        public void actionPerformed(ActionEvent ae)
{
            int makananUpdate =
pilmakanan.getSelectedIndex();

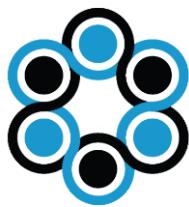
Allobjctr107162.Pembeli.update(textmakananUpdate.getText
(), makananUpdate);

JOptionPane.showMessageDialog(null,"BERHASIL DI
UPDATE","INFORMATION", JOptionPane.INFORMATION_MESSAGE);
            kosong();
}
});

    kembali.addActionListener(new ActionListener(){

        @Override
        public void actionPerformed(ActionEvent ae)
{
            TampilanAwal.dispose();
GUJT07162 men = new GUJT07162():

}});
```



Tugas Resmi

Source Code

```
@Override  
    public void  
actionPerformed(ActionEvent ae) {  
    TampilanAwal.dispose();  
    GUI07162 men = new GUI07162();  
}  
  
});  
}  
void kosong() {  
    textmakananUpdate.setText(null);  
}  
}
```

Analisa

Pada class *UpdateGUI07162*, pada kelas ini merupakan tampilan GUI pada saat kita update dengan memasukkan nama lalu memilih makanan yang akan di update



Tugas Resmi

Langkah keDelapanBelas

Membuat class main pada Package AnindyaModul4_07162

Source Code

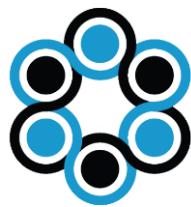
```
package anindyamodul4_07162;
import View07162.GUI07162;

public class AnindyaModul4_07162 {

    public static void main(String[] args) {
        GUI07162 gui = new GUI07162();
    }
}
```

Analisa

Pada main program, kita bisa memilih Gui, jika Gui akan ditujukan ke class GUI07162.



Tugas Resmi

ScreenShoot

The screenshot shows a Windows application window titled "Rumah Makan". The window is divided into two main sections: "Login" on the left and "Buat Akun" (Create Account) on the right.

Login Section:

- Contains radio buttons for "admin" and "pembeli".
- Text input fields for "Nama" and "Password".
- Buttons for "check" and "update".

Buat Akun Section:

- Text input fields for "Nama" (Anindya), "Password" (***), and "Alamat" (kwatu).
- A "Daftar" button.

System Tray:

- Windows taskbar with search bar, pinned icons (File Explorer, Mail, Edge, File History, Task View, Taskbar settings), and system status (0:50, 15/01/2021).

Second Screenshot:

An information dialog box is displayed in the center of the screen, stating "Registrasi sukses" (Registration successful) with an "OK" button. The rest of the application interface is identical to the first screenshot.

Third Screenshot:

An information dialog box is displayed in the center of the screen, showing a warning icon and the text "Maaf, ada kesalahan pada sistem" (Sorry, there is an error in the system). The "OK" button is visible at the bottom right of the dialog. The application interface remains the same as the previous screenshots.



Tugas Resmi

ScreenShoot

Rumah Makan

Login

admin pembeli

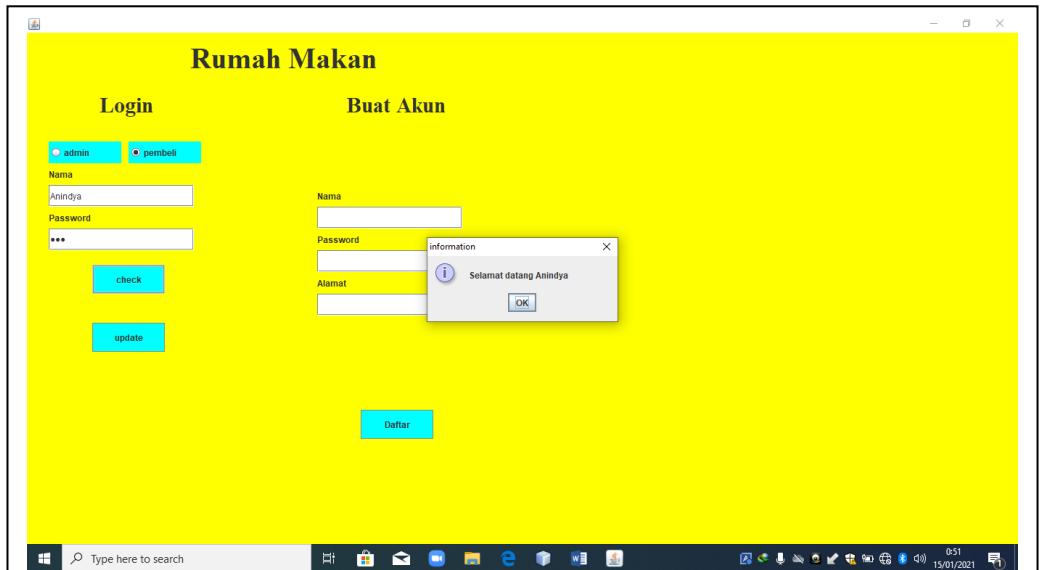
Nama: Anindya
Password: ***

Buat Akun

Nama: [empty]
Password: [empty]
Alamat: [empty]

Information dialog: Selamat datang Anindya

Daftar



Rumah Makan

Login

admin pembeli

Nama: Anindya
Password: ***

Data Pembeli

bakso = 5000

Information dialog: Anda Belum Daftar sebagai pembeli

Daftar Makanan

Daftar

Back



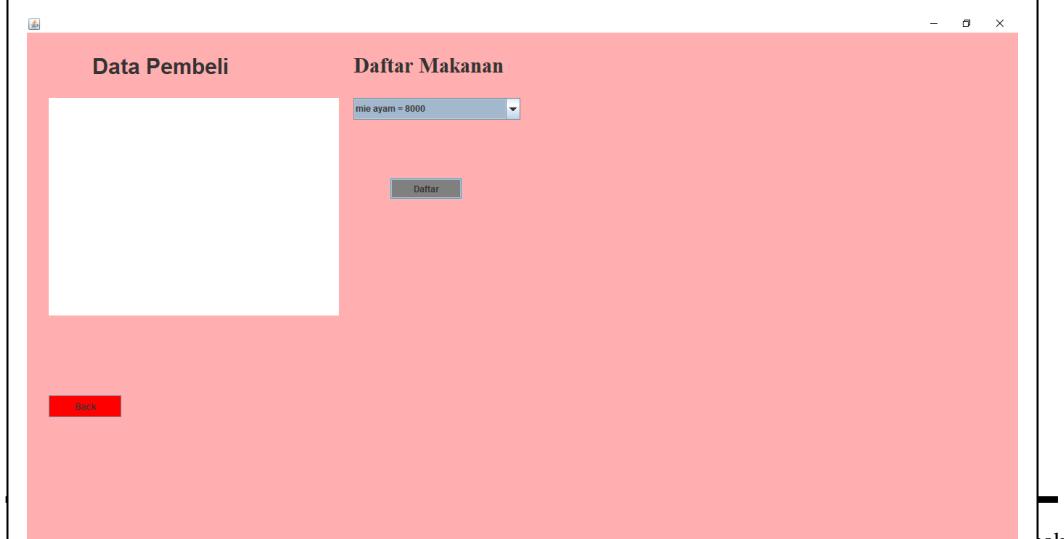
Data Pembeli

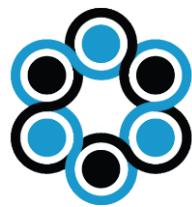
mie ayam = 8000

Daftar Makanan

Daftar

Back





Tugas Resmi

ScreenShoot

Data Pembeli

Nama = Anin
Alamat = kawau
Verifikasi = belum di verifikasi
Makanan = mie ayam = 8000

Daftar Makanan

mie ayam = 8000

Daftar

Back

Type here to search 0:53 15/01/2021

Update Data

Nama Pembeli
Anindya

bakso = 5000

KONFIRMASI
KEMBALI

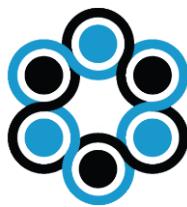
Update Data

INFORMATION

BERHASIL DI UPDATE

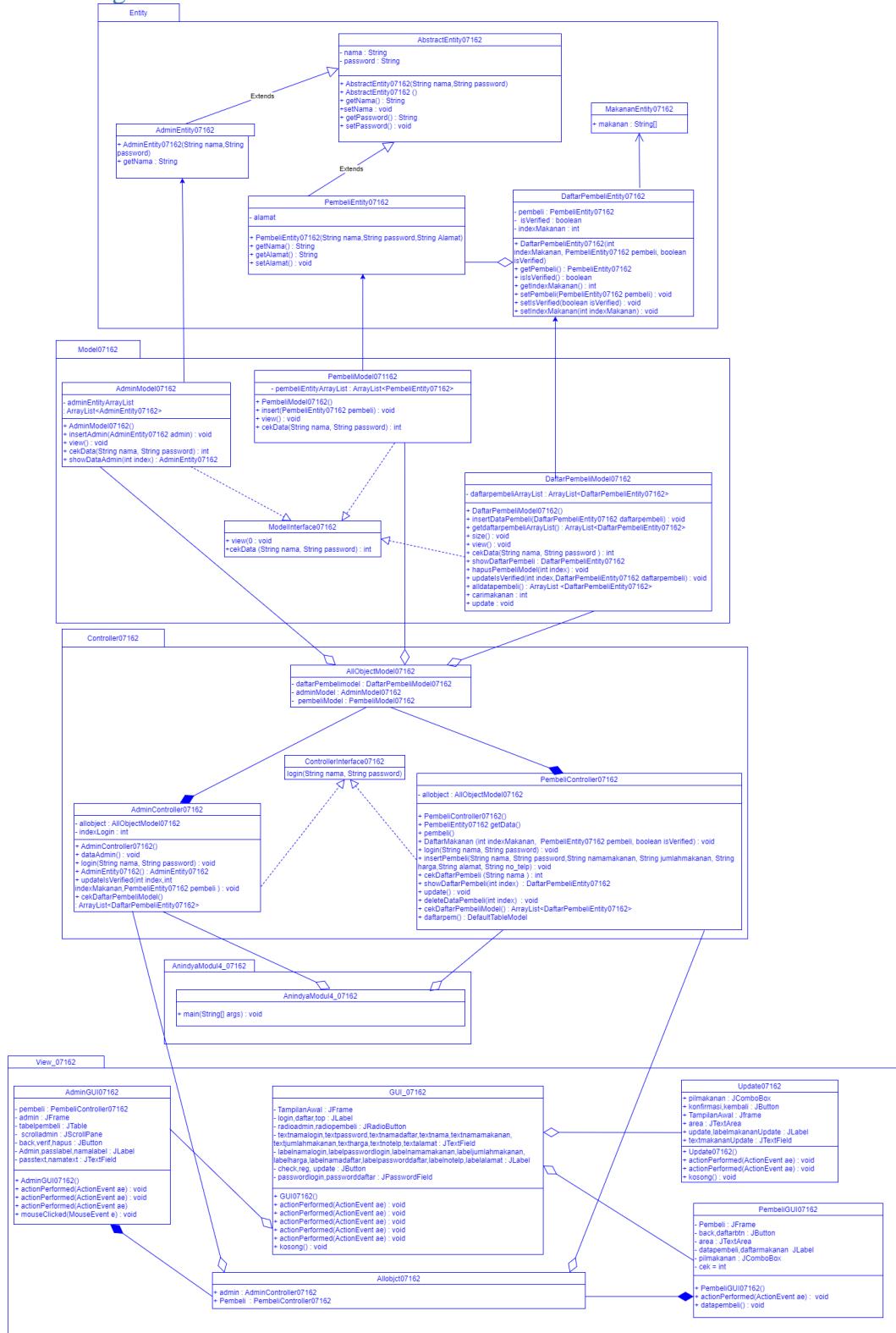
OK

KONFIRMASI
KEMBALI



Tugas Resmi

Class Diagram





PRAKTIKUM

MODUL 1

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



Tugas Praktikum

TUGAS PRAKTIKUM



**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
PERIODE VI - Tahun 2020/2021**

NAMA : Anindya Berlinani S.
NPM : 06.2019.1.07162
MODUL : 1



Tugas Praktikum

Soal Resmi

Sistem Informasi Menu Makanan

Langkah Pertama

Buatlah file baru dengan nama PublicClassAnindya07162. Pastikan file yang telah terbuat nama kelas dan nama file harus sama. Buatlah sebuah void main di dalam class yang sudah dibuat tersebut. Buat class anindyaEntity yang digunakan untuk menyimpan data menu makanan dengan kode sebagai berikut:



Tugas Praktikum

Source Code

```
package pkgpublic.pkgclass.anindya.pkg07162;

public class anindyaEntity {
    String anindya07162_nama;
    String anindya07162_pesan;
    String anindya07162_makanan;
    String anindya07162_harga;

    public anindyaEntity(String anindya07162_nama,
String anindya07162_pesan, String
anindya07162_makanan, String anindya07162_harga)
    {
        this.anindya07162_nama=anindya07162_nama;
        this.anindya07162_pesan=anindya07162_pesan;
        this.anindya07162_makanan=anindya07162_makanan;
        this.anindya07162_harga=anindya07162_harga;

    }
    public String getanindya07162_nama(){
        return anindya07162_nama;
    }
    public String getanindya07162_pesan(){
        return anindya07162_pesan;
    }
    public String getanindya07162_makanan(){
        return anindya07162_makanan;
    }
    public String getanindya07162_harga(){
        return anindya07162_harga;
    }
}
```



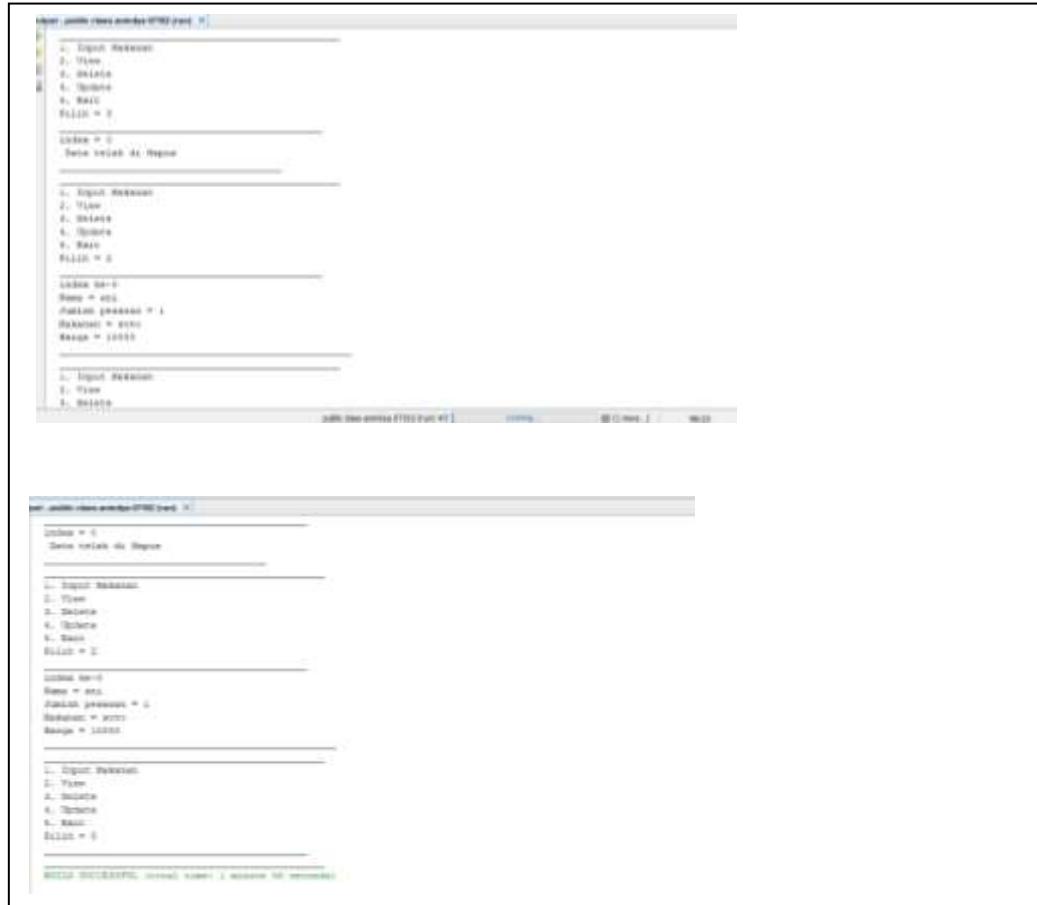
Tugas Praktikum

Screenshot



Tugas Praktikum

Screenshot



Analisa

Pada class `anindyaEntity` didalamnya terdapat attribute: `String anindya07162_nama, String anindya07162_pesanan, String anindya07162_makanan, String anindya07162_harga.` Lalu terdapat Constructor dengan parameternya: `public anindyaEntity(String anindya07162_nama, String anindya07162_pesanan, String anindya07162_makanan, String anindya07162_harga).` Juga terdapat get:

```
public String getanindya07162_nama(){
    return anindya07162_nama;
}
public String getanindya07162_pesanan(){
    return anindya07162_pesanan;
}
public String getanindya07162_makanan(){
    return anindya07162_makanan;
}
public String getanindya07162_harga(){
    return anindya07162_harga
}
```



Tugas Praktikum



Tugas Praktikum

Langkah Kedua

Pada main class yang sudah di buat sebelumnya tulis kode seperti berikut :

Source Code

```
package pkgpublic.pkgclass.anindya.pkg07162;
import java.util.Scanner;
import java.util.ArrayList;
public class PublicClassAnindya07162 {
    static Scanner input = new Scanner (System.in);
    static ArrayList<anindyaEntity>datamakanan = new
ArrayList();

    static void anindya07162_makanan()
    {
        System.out.print("Inputkan Nama = ");
        String anindya07162_nama = input.next();
        System.out.print("Inputkan jumlah pesanan = ");
        String anindya07162_pesan = input.next();
        System.out.print("Inputkan makanan = ");
        String anindya07162_makanan = input.next();
        System.out.print("Inputkan harga = ");
        String anindya07162_harga = input.next();
        datamakanan.add(new
anindyaEntity(anindya07162_nama,anindya07162_pesan,anind
ya07162_makanan,anindya07162_harga));
    }
    static void anindya07162_view()
    {
        for(int i=0;i<datamakanan.size();i++)
        {
            System.out.println("index ke- "+(i));
            System.out.println("Nama =
"+datamakanan.get(i).getanindya07162_nama());
            System.out.println("Jumlah pesanan =
"+datamakanan.get(i).getanindya07162_pesan());
            System.out.println("Makanan =
"+datamakanan.get(i).getanindya07162_makanan());
            System.out.println("Harga =
"+datamakanan.get(i).getanindya07162_harga());

            System.out.println(" _____");
            System.out.println(" _____");
        }
    }
}
```



Tugas Praktikum

Source Code

```
static int carimakanan(String anindya07162_nama)
{
    int indexmakanan = -1;
    for(int i=0;i<datamakanan.size();i++)
    {

        if(anindya07162_nama.equals(datamakanan.get(i).getanindya07162_nama()))
        {
            indexmakanan = i;
        }
    }
    return indexmakanan;
}

static void anindya07162_hapus()
{
    System.out.print("index = ");
    int index = input.nextInt();
    datamakanan.remove(index);
    System.out.println(" Data telah di Hapus ");

    System.out.println("_____");
}

static void anindya07162_update()
{
    System.out.print("index = ");
    int index = input.nextInt();
    for(int i=0;i<datamakanan.size();i++)
    {
        if(index == i)
        {
            System.out.print("Inputkan nama = ");
            String anindya07162_nama = input.next();
            System.out.print("Inputkan jumlah pesanan = ");
            String anindya07162_pesan =
            input.next();
            System.out.print("Inputkan menu makanan = ");
            String anindya07162_makanan =
            input.next();
            System.out.print("Inputkan Harga = ");
            String anindya07162_harga =
        }
    }
}
```



Tugas Praktikum

Source Code

```
System.out.print("Inputkan Harga = ");
String anindya07162_harga =
input.nextInt();

datamakanan.get(i).anindya07162_nama=anindya07162_nama;

datamakanan.get(i).anindya07162_pesan=anindya07162_pesan
;

datamakanan.get(i).anindya07162_makanan=anindya07162_mak
anan;

datamakanan.get(i).anindya07162_harga=anindya07162_harga
;
System.out.println("Data Telah di
Update");
}
}
}

public static void main(String[] args) {
System.out.println(" SELAMAT DATANG !!");

System.out.println("_____");
int pil;
do{
    System.out.println("1. Input Makanan");
    System.out.println("2. View");
    System.out.println("3. Delete");
    System.out.println("4. Update");
    System.out.println("5. Exit");
    System.out.print("Pilih = ");
    pil = input.nextInt();

System.out.println("_____");
_____);
switch(pil){
    case 1 : anindya07162_makanan();
    break;
    case 2 : anindya07162_view();
    break;
    case 3 : anindya07162_hapus();
    break;
    case 4 : anindya07162_update();
    break;
}
```



Tugas Praktikum

Source Code

```
}

System.out.println("_____
_____
}while(pil!=5);
}
}
```

Screenshot

```
Belanj - public class belanj {ST222.html
}
1. Input Belanxx
2. Tampilkan Belanxx
3. Total
4. Uraian
5. Exit
Pilih = 5

Inputan Nama = mardiyah
Inputan jumlah penabuh = 2
Inputan belanxx = 2000
Inputan harga = 1000

1. Input Belanxx
2. Tampilkan Belanxx
3. Total
4. Uraian
5. Exit
Pilih = 1

Inputan Nama = ana
Inputan jumlah penabuh = 2
Inputan belanxx = poci
Inputan harga = 5000

1. Input Belanxx
2. Tampilkan Belanxx
3. Total
4. Uraian
5. Exit
Pilih = 2

class Belanj {
    String Nama;
    int penabuh;
    double penabuh;
    double belanxx;
    double harga;

    public Belanj() {
        Nama = "mardiyah";
        penabuh = 2;
        belanxx = 2000;
        harga = 1000;
    }

    public void tampil() {
        Nama = "ana";
        penabuh = 2;
        penabuh = 5000;
        belanxx = poci;
        harga = 5000;
    }

    public void uraian() {
        Nama = "ana";
        penabuh = 2;
        penabuh = 5000;
        belanxx = poci;
        harga = 5000;
    }

    public void total() {
        Nama = "ana";
        penabuh = 2;
        penabuh = 5000;
        belanxx = poci;
        harga = 5000;
    }

    public void exit() {
        Nama = "ana";
        penabuh = 2;
        penabuh = 5000;
        belanxx = poci;
        harga = 5000;
    }
}
```



Tugas Praktikum

Screenshot





Tugas Praktikum

Analisa

Pada class main terdapat: `java.util.ArrayList` yang digunakan untuk menyimpan data berupa array. `Java.util.Scanner` yang digunakan untuk menginputkan data. Scanner dengan sistem scanner digunakan untuk sistem inputan `int`. `ArrayList` dari kelas `anindyaEntity` dengan nama datamakanan dan disimpan dengan new `ArrayList`.

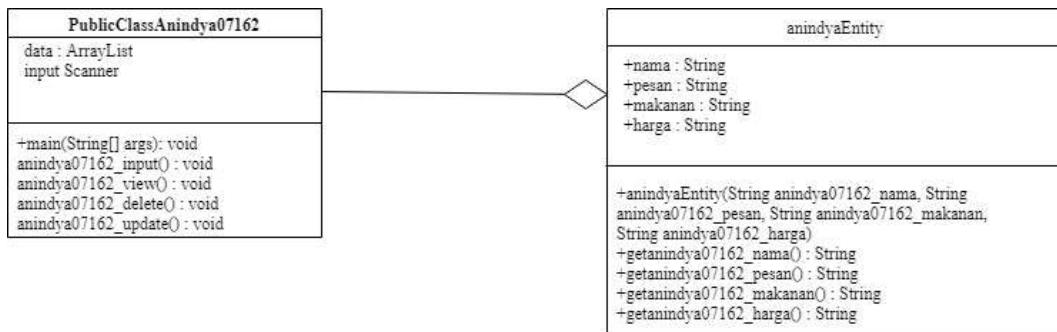
```
public static void main(String[] args) {  
    System.out.println(" SELAMAT DATANG !!");  
    System.out.println(" _____ ");  
    int pil;  
    do{  
        System.out.println("1. Input Makanan");  
        System.out.println("2. View");  
        System.out.println("3. Delete");  
        System.out.println("4. Update");  
        System.out.println("5. Exit");  
        System.out.print("Pilih = ");  
        pil = input.nextInt();  
  
        System.out.println(" _____ ");  
        switch(pil){  
            case 1 : anindya07162_makanan();  
            break;  
            case 2 : anindya07162_view();  
            break;  
            case 3 : anindya07162_hapus();  
            break;  
            case 4 : anindya07162_update();  
            break;  
        }  
  
        System.out.println(" _____ ");  
    }  
    }while(pil!=5);  
}
```

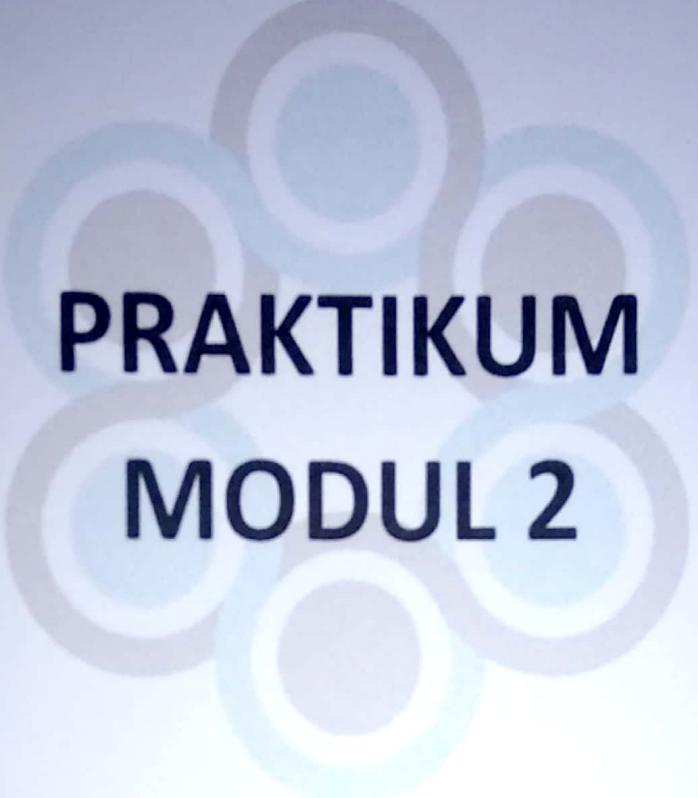
Source code diatas untuk masuk ke menu utama. Di menu utama terdapat 5 pilihan. Pilihan pertama akan darahkan ke method `anindya07162_makanan`. Di method `anindya07162_makanan` kita menginputkan data berupa nama, jumlah pesanan, makanan, harga. Jika sudah akan disimpan di datamakanan dengan new class `anindyaEntity`. Jika kita memilih pilihan 2 maka akan ditampilkan semua data yang sudah diinputkan tadi. Jika memilih pilihan 3 maka di uji dulu jika ada maka menginputkan index dari view jika inputannya sesuai maka `equalsnya` berarti dia berada di index 0 atau benar dan jika diuji `equalsnya` tidak sesuai maka akan muncul tulisan data telah dihapus. Lalu kita memilih pilihan 4, disini untuk mengupdate data lama menjadi data baru.



Tugas Praktikum

Class Diagram :

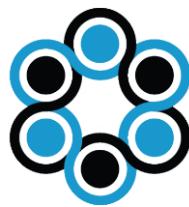




PRAKTIKUM

MODUL 2

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



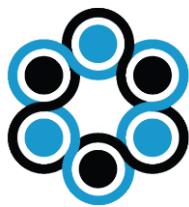
Tugas Praktikum

LAPORAN PRAKTIKUM



**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
PERIODE VI - Tahun 2020/2021**

NAMA : Anindya Berlinani S
NPM : 06.2019.1.07162
MODUL : 2



Tugas Praktikum

Soal Praktikum

Sistem Informasi Menu Makanan

Langkah Pertama

membuat class abstract

Source Code

```
package Entity07162;

public abstract class AbstractEntity07162 {
    public String nama,password;

    public AbstractEntity07162(String nama,String password){
        this.nama = nama;
        this.password = password;
    }
    public AbstractEntity07162 (){
    }

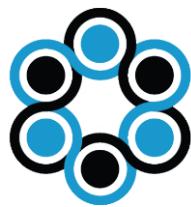
    public abstract String getNama();

    public void setNama(String nama) {
        this.nama = nama;
    }

    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}
```

Analisa :

Class AbstarcetEntity digunakan untuk menginputkan login dengan menginputkan nama dan password.



Tugas Praktikum

Langkah Kedua

Membuat class AdminEntity

Source Code

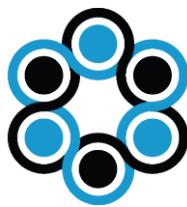
```
package Entity07162;

public class AdminEntity07162 extends
AbstractEntity07162 {

    public AdminEntity07162(String nama,String
password) {
        super (nama,password);
    }
    @Override
    public String getNama(){
        return nama;
    }
}
```

Analisa :

Class AdminEntity digunakan untuk menginputkan nama dan password pada saat login admin



Tugas Praktikum

Source Code

```
package Entity07162;

public class MakananEntity07162 {
    public String namamakanan,jumlah_makanan,harga;

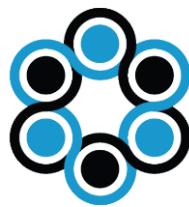
    public MakananEntity07162 (String
namamakanan,String jumlah_makanan,String harga){
        this.namamakanan = namamakanan;
        this.jumlah_makanan = jumlah_makanan;
        this.harga = harga;
    }

    public String getNamaMakanan() {
        return namamakanan;
    }
    public String getJumlahMakanan() {
        return jumlah_makanan;
    }

    public String getHarga() {
        return harga;
    }
}
```

Analisa :

Class MakananEntity berisi constructor String namamakanan, String jumlah_makanan, String harga



Tugas Praktikum

Source Code

```
package Entity07162;

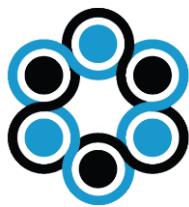
public class PembeliEntity07162 extends
AbstractEntity07162 {
    String alamat;
    String no_telp;

    public PembeliEntity07162(String
nama,String password,String alamat, String
no_telp){
        super (nama,password);
        this.alamat = alamat;
        this.no_telp = no_telp;
    }

    @Override
    public String getNama(){
        return nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public String getNoTelp() {
        return no_telp;
    }
}
```

Analisa :

Class AdminEntity digunakan untuk menginputkan nama dan password pada saat login pembeli

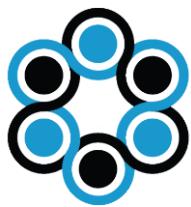


Tugas Praktikum

Source Code

```
package Model07162;
import Entity07162.AdminEntity07162;
import java.util.ArrayList;
public class AdminModel07162 implements
ModelInterface07162
{
    private ArrayList<AdminEntity07162>
adminEntityArrayList;

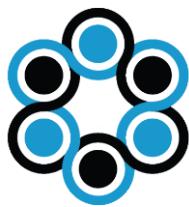
    public AdminModel07162()
    {
        adminEntityArrayList = new
ArrayList<AdminEntity07162>();
    }
    public void insertAdmin(AdminEntity07162
admin)
    {
        adminEntityArrayList.add(admin);
    }
    @Override
    public void view()
    {
        for (AdminEntity07162 admin :
adminEntityArrayList)
        {
            System.out.print(admin.getNama());
            System.out.print(admin.getPassword());
            System.out.println();
        }
    }
    public int cekData(String nama, String password){
        int loop = 0;
        for(AdminEntity07162 admin :
adminEntityArrayList){
            if(admin.getNama().equals(nama) &&
admin.getPassword().equals(password)){
                break;
            }else{
                loop++;
            }
        }
        return loop;
    }
}
```



Tugas Praktikum

Analisa :

Pada class **AdminModel** berisikan data yang ada pada class
AdminEntity



Tugas Praktikum

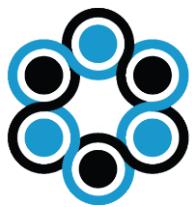
Source Code

```
package Model07162;
import Entity07162.PembeliEntity07162;
import java.util.ArrayList;
public class PembeliModel07162 implements
ModelInterface07162 {
    private ArrayList<PembeliEntity07162>
pembeliEntityArrayList;

    public PembeliModel07162()
    {
        pembeliEntityArrayList = new
ArrayList<PembeliEntity07162>();
    }
    public void insertPembeli(PembeliEntity07162
pembeli)
    {
        pembeliEntityArrayList.add(pembeli);
    }

    @Override
    public void view(){
        for(PembeliEntity07162 pembeli :
pembeliEntityArrayList)
        {
            System.out.print(pembeli.getNama());
            System.out.print(pembeli.getPassword());
            System.out.print(pembeli.getAlamat());
            System.out.print(pembeli.getNoTelp());
            System.out.println();
        }
    }

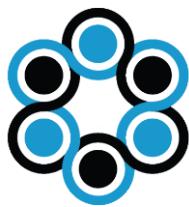
    public int cekData(String nama, String password){
        int loop = 0;
        while
(!pembeliEntityArrayList.get(loop).getNama().equals(nama
) &&
!pembeliEntityArrayList.get(loop).getPassword().equals(p
assword)){
            loop++;
        }
        return loop;
    }
}
```



Tugas Praktikum

Analisa :

Pada class **PembeliModel** berisikan data yang ada pada class
PembeliEntity



Tugas Praktikum

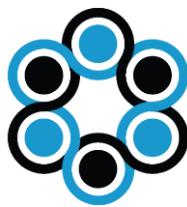
Source Code

```
package Model07162;

public interface ModelInterface07162 {
    public void view();
    public int cekData (String nama, String
password);
}
```

Analisa :

Pada class Interface terdapat cekData untuk mengecek nama dan password
Serta terdapat View untuk melihat data.



Tugas Praktikum

Source Code

```
package modul2_07162;
import Entity07162.*;
import Model07162.*;
import java.util.ArrayList;
import java.util.Scanner;
public class Modul2_07162 {
private static AdminModel07162 adminModel = new
AdminModel07162();
private static PembeliModel07162 pembeliModel = new
PembeliModel07162();
static public ArrayList <MakananEntity07162>
datamakanan= new ArrayList();
static int cekAdmin;
static int cekPembeli;
static Scanner input = new Scanner(System.in);
    public static void main(String[] args) {
        System.out.println(" SELAMAT DATANG DI RUMAH
MAKAN !!");

System.out.println("=====");
=====
        int pilih;
        do{
            System.out.println("1. Daftar");
            System.out.println("2. Login");
            System.out.println("3. Exit");
            System.out.print("Pilih = ");
            pilih = input.nextInt();

System.out.println("=====");
=====
            switch(pilih){
                case 1:
                    if(pilih == 1){
                        daftpembeli();
                    }
                    break;
                case 2 : if(pilih == 2){
                    int pilLogin = 0;
                    System.out.print("1. Login Admin" + "\n2.
Login pembeli" + "\n Pilih : ");
                    pilLogin = input.nextInt();
                    if(pilLogin == 1){
                        loginAdmin();
```



Tugas Praktikum

Source Code

```
        }else{
            loginPembeli();
        }
    }
    break;

}System.out.println("=====
=====");
}while (pilih!=3);
}
static void daftarpembeli()
{
    System.out.print("Input Nama = ");
    String nama = input.next();
    System.out.print("Input Password = ");
    String password = input.next();
    System.out.print("Alamat = ");
    String alamat = input.next();
    System.out.print("Input notelp = ");
    String no_telp = input.next();
    pembeliModel.insertPembeli(new
PembeliEntity07162(nama,password,alamat,no_telp));
}

static void dataAdmin()
{
    String nama[] = {"anindya","anin"};
    String password[] = {"123","123"};
    for(int i=0;i<nama.length;i++)
    {
        adminModel.insertAdmin(new
AdminEntity07162(nama[i],password[i]));
    }
}
static void makanan(){
    System.out.print("INPUT NAMA MAKANAN = ");
    String namamakanan = input.next();
    System.out.print(" INPUT JUMLAH MAKANAN = ");
    String jumlah_makanan = input.next();
    System.out.print("INPUT HARGA = ");
    String harga = input.next();

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



Tugas Praktikum

Source Code

```
datamakanan.add(new MakananEntity07162
(namamakanan,jumlah_makanan,harga));
}
static void view(){
    System.out.println(" MAKANAN YANG TERSEDIA ");

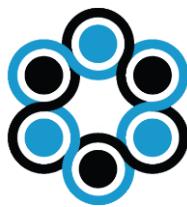
System.out.println("=====");
for(int i=0;i<datamakanan.size();i++)
{
    System.out.println("Nama Makanan =
"+datamakanan.get(i).getNamaMakanan());
    System.out.println("Jumlah Makanan =
"+datamakanan.get(i).getJumlahMakanan());
    System.out.println("Harga =
"+datamakanan.get(i).getHarga());

System.out.println("=====");
}
}
static void hapus(String namamakanan)
{
    for(int i=0;i<datamakanan.size();i++)
{

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
    datamakanan.remove(i);
    System.out.println("Data Telah Di
Hapus");

System.out.println("=====");
}
else
{
    System.out.println("Makanan Tidak
Tersedia");

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



Tugas Praktikum

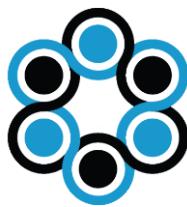
Source Code

```
    }
        static void update(String namamakanan, String jumlah_makanan, String newharga)
    {
        for(int i=0;i<datamakanan.size();i++)
        {

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
            datamakanan.get(i).jumlah_makanan =
jumlah_makanan;
            datamakanan.get(i).harga = newharga;
            System.out.println("Data Telah DI
Update");

System.out.println("=====
=====");
        }
    }
}
static void loginAdmin(){
    int pil;
    System.out.print("Masukkan Nama Admin : ");
    String nama = input.next();
    System.out.print("Password Admin : ");
    String password = input.next();
    cekAdmin = adminModel.cekData(nama,password);
    do{
        System.out.println("1. Input Data
Makanan");
        System.out.println("2. View Data Makanan");
        System.out.println("3. Hapus Data
Makanan");
        System.out.println("4. Update Data
Makanan");
        System.out.println("5. Exit");
        System.out.print("Pilih = ");
        pil = input.nextInt();

System.out.println("=====
=====");
switch(pil){
    case 1 : makanan();
    break;
    .....
```



Tugas Praktikum

Source Code

```
case 2 : view();
           break;
    case 3 : System.out.print("Inputkan Nama
Makanan = ");
               String namamakanan =
input.next();
               hapus(namamakanan);
               break;
    case 4 :
               System.out.print("Inputkan Nama
Makanan = ");
               namamakanan = input.next();
               for(int
i=0;i<datamakanan.size();i++)
{
if(namamakanan.equals(datamakanan.get(i).getNamaMakanan(
)))
{
               System.out.print("Inputkan
Harga Baru = ");
               String newharga =
input.next();
               System.out.print("Inputkan
jumlah makanan baru = ");
               String jumlah_makanan =
input.next();

System.out.println("=====
=====");
update(namamakanan,jumlah_makanan,newharga);
}
else
{
               System.out.println("Makanan
Tidak Tersedia");
System.out.println("=====
=====");
}
break;
}
```

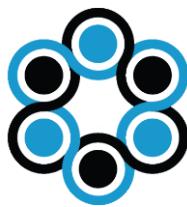


Tugas Praktikum

Source Code

```
        }
            while(pil!=5);
        }
    static void loginPembeli()
    {
        int pilihan;
        System.out.print("Nama Pelanggan : ");
        String nama = input.next();
        System.out.print("Password : ");
        String password = input.next();
        cekPembeli = pembeliModel.cekData(nama,
password);
        do
        {
            System.out.println("1. Data Makanan
Yang Tersedia");
            System.out.println("2. Exit");
            System.out.print("Pilih = ");
            pilihan = input.nextInt();

            System.out.println("=====
=====");
            switch(pilihan)
            {
                case 1: view();
                ;
                break;
            }
        }
        while (pilihan!=2);
    }
}
```



Tugas Praktikum

Screenshot

```
SELAMAT DATANG DI RUMAH MAKAN !!
1. Daftar
2. Login
3. Exit
Pilih = 1

Input Name = anindya
Input Password = 123
Input Email = anindya@gmail.com
Input nohp = 088005577083

1. Daftar
2. Login
3. Exit
Pilih = 2

1. Login Admin
2. Login pembeli
Pilih = 1
Masukkan Name Admin : admin
Password Admin : 123
1. Input Data Makanan
2. View Data Makanan
3. Update Data Makanan
4. Delete Data Makanan
5. Exit
Pilih = 1

INPUT NAMA MAKANAN = baks
INPUT JUMLAH MAKANAN = 18000
INPUT HARGA = 10000

1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 1

MAKANAN YANG TERSEDIA
Nama Makanan = baks
Jumlah Makanan = 18000
Harga = 10000
=====
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 4

Inputkan Nama Makanan = batso
Inputkan Harga Baru = 10000
Inputkan jumlah makanan baru = 2
Data Telah Di Update
=====
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 2

MAKANAN YANG TERSEDIA
Nama Makanan = baks
Jumlah Makanan = 2

Modul2_07162 (run) | running... | (2 more...) (1)
```



```
MAKANAN YANG TERSEDIA
Nama Makanan = baks
Jumlah Makanan = 2
Harga = 10000
=====
1. Input Data Makanan
2. View Data Makanan
3. Update Data Makanan
4. Delete Data Makanan
5. Exit
Pilih = 3

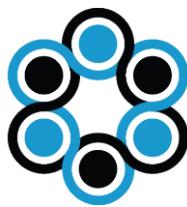
Inputkan Name Makanan = batso
Data Telah Di Hapus
=====
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 2

MAKANAN YANG TERSEDIA
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 5
=====
1. Daftar
2. Login
3. Exit
Pilih =
```



```
MAKANAN YANG TERSEDIA
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 1

1. Daftar
2. Login
3. Exit
Pilih =
```



Tugas Praktikum

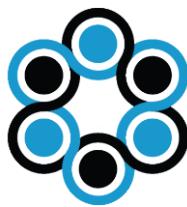
Analisa :

Pada class main terdapat: `java.util.ArrayList` yang digunakan untuk menyimpan data berupa array. `Java.util.Scanner` yang digunakan untuk menginputkan data. Scanner dengan sistem scanner digunakan untuk sistem inputan int. `ArrayList` dari kelas `MakananEntity07162` dengan nama datamakanan dan disimpan dengan new `ArrayList`.

```
public static void main(String[] args) {
    System.out.println(" SELAMAT DATANG DI RUMAH MAKAN !!!");

    System.out.println("=====");
    int pilih;
    do{
        System.out.println("1. Daftar");
        System.out.println("2. Login");
        System.out.println("3. Exit");
        System.out.print("Pilih = ");
        pilih = input.nextInt();

        System.out.println("=====");
        switch(pilih){
            case 1:
                if(pilih == 1){
                    daftarpembeli();
                }
                break;
            case 2 : if(pilih == 2){
                int pilLogin = 0;
                System.out.print("1. Login Admin" + "\n2. Login pembeli" + "\n
Pilih : ");
                pilLogin = input.nextInt();
                if(pilLogin == 1){
                    loginAdmin();
                }else{
                    loginPembeli();
                }
            }
            break;
        }
    }while (pilih!=3);
}
```



Tugas Praktikum

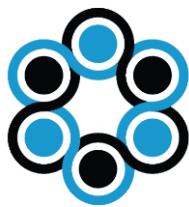
Pada source code diatas terdapat 3 pilihan yaitu Daftar untuk mendaftar nama pembeli, Login untuk login admin dan pembeli , serta exit untuk keluar.

```
static void loginAdmin(){
    int pil;
    System.out.print("Masukkan Nama Admin : ");
    String nama = input.next();
    System.out.print("Password Admin : ");
    String password = input.next();
    cekAdmin = adminModel.cekData(nama,password);
    do{
        System.out.println("1. Input Data Makanan");
        System.out.println("2. View Data Makanan");
        System.out.println("3. Hapus Data Makanan");
        System.out.println("4. Update Data Makanan");
        System.out.println("5. Exit");
        System.out.print("Pilih = ");
        pil = input.nextInt();

    System.out.println("=====");
    switch(pil){
        case 1 : makanan();
        break;
        case 2 : view();
        break;
        case 3 : System.out.print("Inputkan Nama Makanan = ");
            String namamakanan = input.next();
            hapus(namamakanan);
        break;
        case 4 :
            System.out.print("Inputkan Nama Makanan = ");
            namamakanan = input.next();
            for(int i=0;i<datamakanan.size();i++)
            {

        if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
        {
            System.out.print("Inputkan Harga Baru = ");
            String newharga = input.next();
            System.out.print("Inputkan jumlah makanan baru = ");
            String jumlah_makanan = input.next();

        System.out.println("=====");
        update(namamakanan,jumlah_makanan,newharga);
```



Tugas Praktikum

```
        }
    else
    {
        System.out.println("Barang Tidak Tersedia");

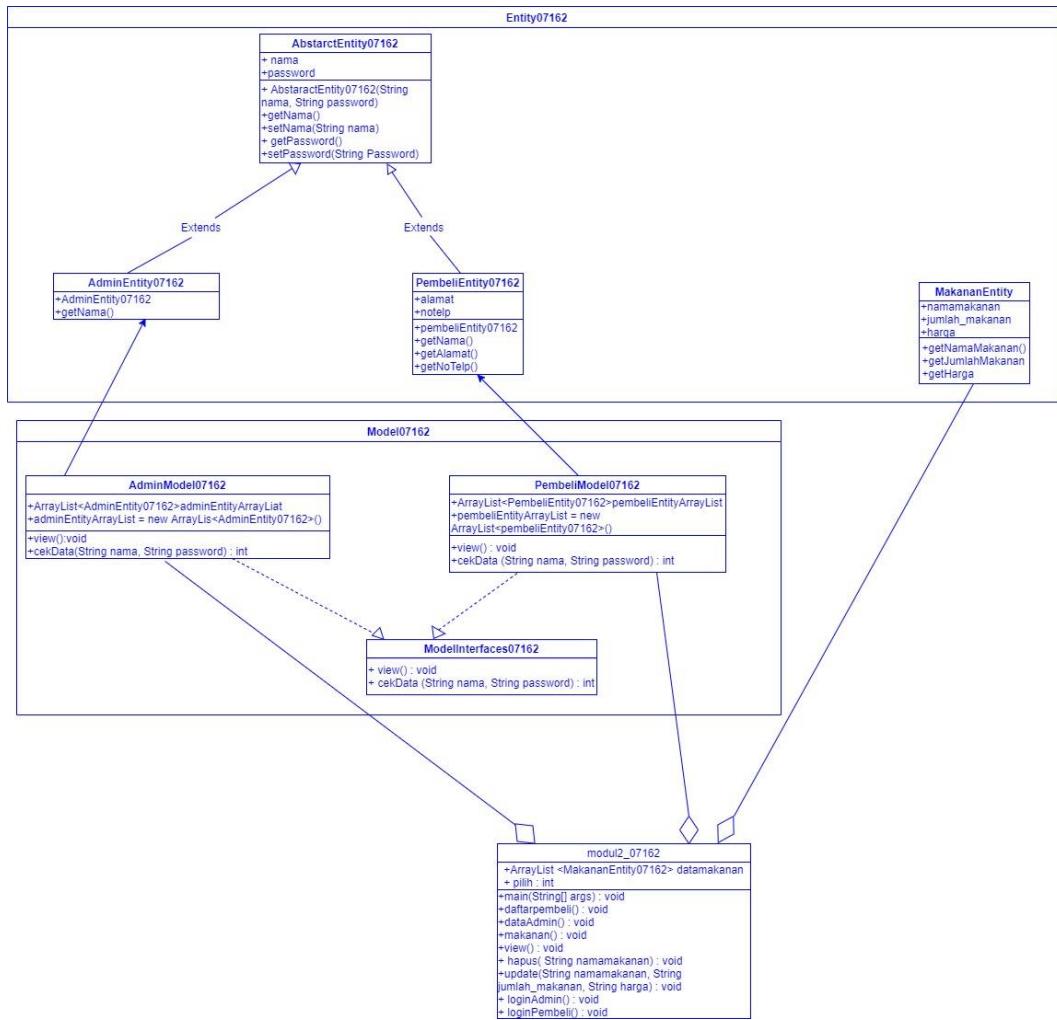
System.out.println("=====");
}
}
break;
}
}
while(pil!=5);
}
```

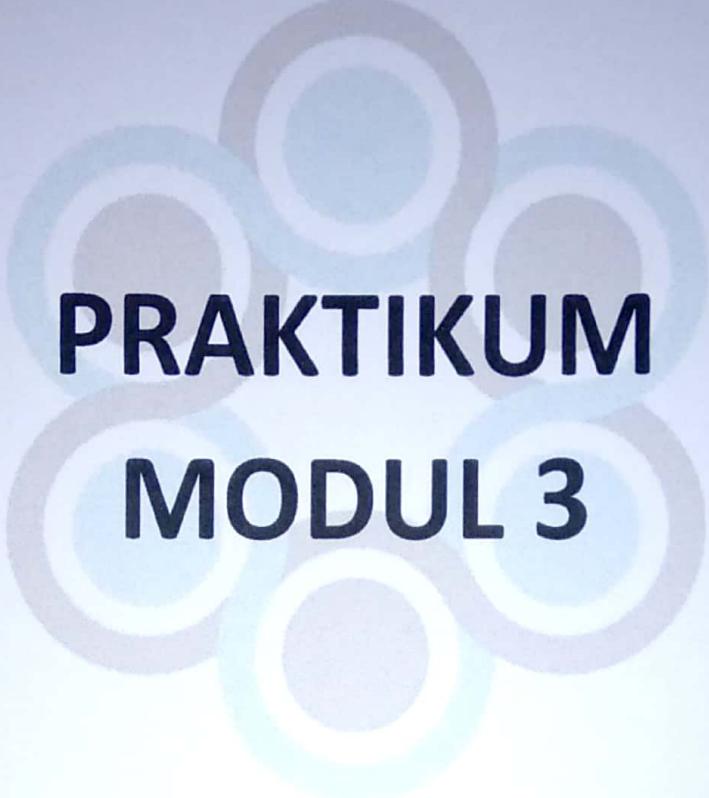
Source code diatas untuk masuk ke menu utama. Di menu utama terdapat 5 pilihan. Pilihan pertama akan darahkan ke method_makanan. Di method makanan kita menginputkan data berupa nama, jumlah pesanan, harga. Jika sudah akan disimpan di datamakanan. Jika kita memilih pilihan 2 maka akan ditampilkan semua data yang sudah diinputkan tadi. Jika memilih pilihan 3 maka di uji dulu jika ada maka menginputkan index dari view.jika inputannya sesuai maka equallsnya berarti dia berada di index 0 atau benar. Lalu kita memilih pilihan 4, disini untuk mengupdate data lama menjadi data baru.



Tugas Praktikum

Class diagram :

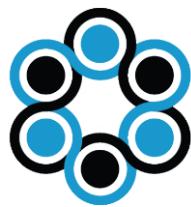




PRAKTIKUM

MODUL 3

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



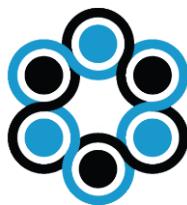
Tugas Praktikum

LAPORAN PRAKTIKUM



**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
PERIODE VI - Tahun 2020/2021**

NAMA : Anindya Berlinani S
NPM : 06.2019.1.07162
MODUL : 3



Tugas Praktikum

Soal Praktikum

Sistem Informasi menu Makanan

Langkah Pertama

Membuat package Controller07162r, lalu membuat class dengan nama AllObjectModel07162

Source Code

```
package Controller07162;
import Model07162.*;
public class AllObjectModel07162{
    public static AdminModel07162 adminModel
= new AdminModel07162();
    public static PembeliModel07162
pembeliModel = new PembeliModel07162();
    public static MakananModel07162 makanan =
new MakananModel07162();
}
```

Analisa

Di class AllObjectModel07162 berisi import an dari package Model07162

Langkah Kedua

Membuat class dari package Controller07162 dengan nama ControllerInterface07162

Source Code

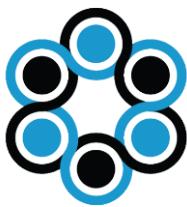
```
package Controller07162;
import java.util.Date;
public interface ControllerInterface07162{
    public void login(String nama, String
password);
}
```

Analisa

Di class ControllerInterface07162 terdapat void login yang berisi nama dan password untuk masuk sebagai admin/pembeli.

Langkah Ketiga

Membuat class dari package Controller07162 dengan nama MakananController07162



Tugas Praktikum

Source Code

```
package Controller07162;
import Entity07162.MakananEntity07162;
public class MakananController07162 {
    int index1 = 0;
    public MakananController07162(){
    }
    public void insert(String namamakanan,String jumlah_makanan,String harga){
        AllObjectModel07162.makanan.insert(new
MakananEntity07162(namamakanan,jumlah_makanan,harga)
);
    }
    public void view(){
        AllObjectModel07162.makanan.view();
    }
    public void hapus(String namamakanan){

AllObjectModel07162.makanan.hapus(namamakanan);
    }
    public void update(String namamakanan,String jumlah_makanan,String harga){

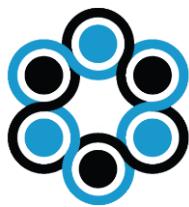
AllObjectModel07162.makanan.update1(namamakanan,jumlah_makanan, harga);
    }
}
```

Analisa

Di class MakananController07162 terdapat method insert untuk inputan nama makanan, jumlah makanan, dan harga. Juga terdapat method view untuk melihat inputan makanan, method hapus untuk menhapus makanan yang akan di hapus, dan method update untuk mengupdate jumlah makanan dan harga.

Langkah Keempat

Membuat class dari package Controller07162 dengan nama Pembeli07162



Tugas Praktikum

Source Code

```
package Controller07162;
import Entity07162.PembeliEntity07162;
public class Pembeli07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public Pembeli07162() {
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.pembeliModel.cekData(nama,
password);
    }
    public void insertPembeli(String nama, String
password, String alamat, String no_telp) {
        PembeliEntity07162 pembeliModel = new
PembeliEntity07162(nama, password, alamat, no_telp);

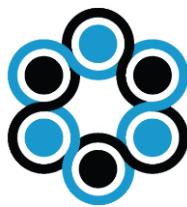
        allobject.pembeliModel.insertPembeli(pembeliModel);
    }
}
```

Analisa

Di class Pembeli07162 terdapat method login untuk login pembeli. Juga terdapat method insertPembeli untuk inputan nama password alamat dan no telp.

Langkah Kelima

Membuat class dari package Controller07162 dengan nama admin07162



Tugas Praktikum

Source Code

```
package Controller07162;

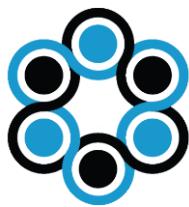
import Entity07162.AdminEntity07162;
import Entity07162.MakananEntity07162;
import Entity07162.PembeliEntity07162;
import java.util.ArrayList;
public class admin07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public admin07162() {
    }
    public void dataAdmin() {
        String passwordAdmin[] = {"123"};
        String namaAdmin[] = {"Anindya"};
        String noTelpAdmin[] = {"088805577083"};
        for(int i = 0; i < passwordAdmin.length; i++)
    {

        AllObjectModel07162.adminModel.insertAdmin(new
AdminEntity07162(passwordAdmin[i], namaAdmin[i] ));
    }
}
    public void viewAdmin() {
        AllObjectModel07162.adminModel.view();
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.adminModel.cekData(nama,
password);
    }
}
```

Analisa

Di class admin07162 terdapat method dataAdmin yang berisi password, nama, dan no telpon. Juga terdapat method viewAdmin yang digunakan untuk melihat data admin dan method login untuk login ke admin

Langkah Keenam



Tugas Praktikum

Membuat Package, lalu membuat class dari package Entity07162 dengan nama AbstractEntity07162

Source Code

```
package Entity07162;

public abstract class AbstractEntity07162 {
    public String nama,password;

    public AbstractEntity07162(String nama,String password){
        this.nama = nama;
        this.password = password;
    }
    public AbstractEntity07162 (){
    }

    public abstract String getNama();

    public void setNama(String nama) {
        this.nama = nama;
    }

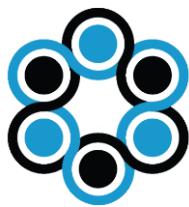
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}
```

Analisa

Di class AbstratcEntity07162 terdapat variabel nama dan password juga terdapat parameter dan gether nya.

Langkah Ketujuh

Membuat class dari package Entity07162 dengan nama AdminEntity07162



Tugas Praktikum

Source Code

```
package Entity07162;

public class AdminEntity07162 extends
AbstractEntity07162 {

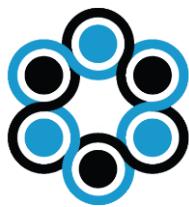
    public String harga;
    public String jumlah_makanan;
    public AdminEntity07162(String nama,String
password) {
        super (nama,password);
    }
    @Override
    public String getNama(){
        return nama;
    }
}
```

Analisa

Di class AdminEntit07162 terdapat variabel nama dan password. Juga terdapat parameter dan gether nya.

Langkah Kedelapan

Membuat Package, lalu membuat class dari package Entity07162 dengan nama MakananEntity07162.



Tugas Praktikum

Source Code

```
package Entity07162;

public class MakananEntity07162 {
    public String namamakanan,jumlah_makanan,harga;

    public MakananEntity07162 (String
namamakanan,String jumlah_makanan,String harga){
        this.namamakanan = namamakanan;
        this.jumlah_makanan = jumlah_makanan;
        this.harga = harga;
    }

    public String getNamaMakanan() {
        return namamakanan;
    }
    public String getJumlahMakanan() {
        return jumlah_makanan;
    }

    public String getHarga() {
        return harga;
    }
}
```

Analisa

Di class MakananEntity07162 terdapat variabel nama makanan, jumlah makana, dan harga. Juga terdapat parameter dan gether nya.

Langkah keSembilan

Membuat class dari package Entity07162 dengan nama PembeliEntity07162.



Tugas Praktikum

Source Code

```
package Entity07162;

public class PembeliEntity07162 extends
AbstractEntity07162 {
    String alamat;
    String no_telp;

    public PembeliEntity07162(String nama, String
password, String alamat, String no_telp){
        super (nama,password);
        this.alamat = alamat;
        this.no_telp = no_telp;
    }

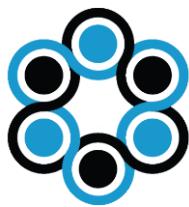
    @Override
    public String getNama(){
        return nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public String getNoTelp() {
        return no_telp;
    }
}
```

Analisa

Di class PembeliEntity07162 terdapat variabel nama, password, alamat dan no telp. Juga terdapat parameter dan gether nya.

Langkah KeSepuluh

Membuat package Model dengan nama Model07162 lalu membuat class dengan nama AdminModel07162

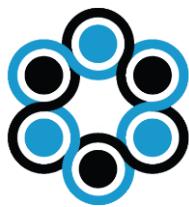


Tugas Praktikum

Source Code

```
package Model07162;
import static
Controller07162.AllObjectModel07162.adminModel;
import Entity07162.AdminEntity07162;
import java.util.ArrayList;
public class AdminModel07162 implements
ModelInterface07162
{
    private ArrayList<AdminEntity07162>
adminEntityArrayList;

    public AdminModel07162()
    {
        adminEntityArrayList = new
ArrayList<AdminEntity07162>();
    }
    public void insertAdmin(AdminEntity07162 admin)
    {
        adminEntityArrayList.add(admin);
    }
    public void view()
    {
        for (AdminEntity07162 admin :
adminEntityArrayList)
        {
            System.out.print(admin.getNama());
            System.out.print(admin.getPassword());
            System.out.println();
        }
    }
    public int cekData(String nama, String password){
        int loop = 0;
        for(AdminEntity07162 admin :
adminEntityArrayList){
            if(admin.getNama().equals(nama) &&
admin.getPassword().equals(password)){
                break;
            }else{
                loop++;
            }
        }
        return loop;
    }
}
```



Tugas Praktikum

Analisa

Di kelas AdminModel07162 terdapat method admin yang berisi data admin. Juga terdapat method view untuk melihat data admin dan method cekData untuk login.

Langkah keSembelas

Membuat class dengan nama MakananModel07162.

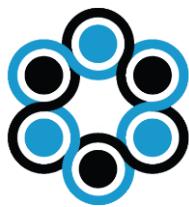
Source Code

```
package Model07162;
import Entity07162.MakananEntity07162;
import java.util.ArrayList;
public class MakananModel07162 {
    public ArrayList<MakananEntity07162> datamakanan;

    public MakananModel07162(){
        datamakanan = new
ArrayList<MakananEntity07162>();
    }
    public void Insert(MakananEntity07162 makanan){
        datamakanan.add(makanan);
    }
    public void view()
    {

        for (MakananEntity07162 makanan : datamakanan)
        {
            System.out.println("Nama Makanan =
"+makanan.getNamaMakanan());
            System.out.println("Jumlah Makanan =
"+makanan.getJumlahMakanan());
            System.out.println("Harga =
"+makanan.getHarga());

System.out.println("=====");
        }
    }
    public void hapus(String namamakanan)
    {
        for(int i=0;i<datamakanan.size();i++)
        {
```



Tugas Praktikum

Source Code

```
public void hapus(String namamakanan)
{
    for(int i=0;i<datamakanan.size();i++)
    {

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
    datamakanan.remove(i);
    System.out.println("Makanan Telah DI
Hapus");

System.out.println("=====
=====");
}
else
{
    System.out.println("Makanan Tidak
Tersedia");

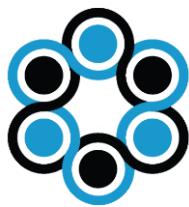
System.out.println("=====
=====");
}
}
}

public void update(String namamakanan,String
newjumlah_makanan,String newharga)
{
    for(int i=0;i<datamakanan.size();i++)
    {

if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
{
    datamakanan.get(i).harga=newharga;

datamakanan.get(i).jumlah_makanan=newjumlah_makanan;
    System.out.println("Data Telah DI
Update");

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



Tugas Praktikum

Source Code

```
System.out.println("=====");
=====
}
}
public void update1(String namamakanan, String newjumlah_makanan, String newharga){
    for(int i=0;i<datamakanan.size();i++)
    {

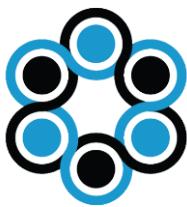
        if(namamakanan.equals(datamakanan.get(i).getNamaMakanan()))
        {
            update(namamakanan,newjumlah_makanan,newharga);
        }
        else
        {
            System.out.println("Makanan
Tidak Tersedia");
        }
    }
}
```

Analisa

Di class MakananModel07162 terdapat method insert yang digunakan untuk menampilkan data makanan. Juga terdapat method view untuk melihat data makanan, method hapus untuk menghapus data makanan, method update untuk mengupdate jumlah makanan dan harga.

Langkah keDuaBebelas

Membuat class interface dengan nama ModelInterface0716207162.



Tugas Praktikum

Source Code

```
package Model07162;

public interface ModelInterface07162 {
    public void view();
    public int cekData (String nama, String
password);
}
```

Analisa

Yang berisi method view untuk melihat dan method cekData untuk login

Langkah KeTigaBelas

Membuat class dengan nama PembeliModel07162.

Source Code

```
package Model07162;
import Entity07162.PembeliEntity07162;
import java.util.ArrayList;
public class PembeliModel07162 implements
ModelInterface07162 {
    private ArrayList<PembeliEntity07162>
pembeliEntityArrayList;

    public PembeliModel07162()
    {
        pembeliEntityArrayList = new
ArrayList<PembeliEntity07162>();
    }

    public void insertPembeli(PembeliEntity07162
pembeli)
    {
        pembeliEntityArrayList.add(pembeli);
    }

    @Override
    public void view(){
        for(PembeliEntity07162 pembeli :
pembeliEntityArrayList)
        {
            System.out.print(pembeli.getNama());
            System.out.print(pembeli.getPassword());
        }
    }
}
```



Tugas Praktikum

Source Code

```
public void view(){
    for(PembeliEntity07162 pembeli :
pembeliEntityArrayList)
    {
        System.out.print(pembeli.getNama());
        System.out.print(pembeli.getPassword());
        System.out.print(pembeli.getAlamat());
        System.out.print(pembeli.getNoTelp());
        System.out.println();
    }
}

public int cekData(String nama, String password){
    int loop = 0;
    while
(!pembeliEntityArrayList.get(loop).getNama().equals(nama)
) &&

!pembeliEntityArrayList.get(loop).getPassword().equals(p
assword)){
        loop++;
    }
    return loop;
}
```

Analisa

Di class PembeliModel07162 terdapat method insert untuk menampilkan data pembeli, method view untuk melihat dan method cekData untuk Login.

Langkah keEmpatBelas

Mengisi class main dengan perintah :

Source Code

```
package modul3_07162;
import Entity07162.*;
import Controller07162.*;
import java.util.ArrayList;
import java.util.Scanner;

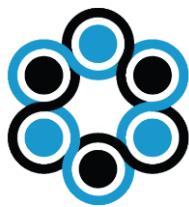
public class Modul3_07162 {
    private static admin07162 admin = new admin07162();
    . . . . .
```



Tugas Praktikum

Source Code

```
public class Modul3_07162 {  
    private static admin07162 admin = new admin07162();  
    private static Pembeli07162 pembeli = new  
Pembeli07162();  
    private static MakananController07162 makanan = new  
MakananController07162();  
    private static Scanner input = new Scanner  
(System.in);  
    static int cekAdmin;  
    static int cekPembeli;  
    public static void main(String[] args) {  
        System.out.println("SELAMAT DATANG DI RUMAH  
MAKAN !!");  
  
        System.out.println("=====");  
        int pil;  
        do{  
            System.out.println("1. Daftar");  
            System.out.println("2. Login");  
            System.out.println("3. Exit");  
            System.out.print("Pilih = ");  
            pil = input.nextInt();  
  
            System.out.println("=====");  
            switch(pil){  
                case 1:  
                    if(pil == 1){  
                        daftarpembeli() ;  
                    }  
                    break;  
                case 2 : if(pil == 2){  
                    int pilLogin = 0;  
                    System.out.print("1. Login Admin" + "\n2.  
Login pembeli" + "\n Pilih : ");  
                    pilLogin = input.nextInt();  
                    if(pilLogin == 1){  
                        loginAdmin();  
                    }else{  
                        loginPembeli();  
                    }  
                }  
            }  
        }  
    }  
}
```



Tugas Praktikum

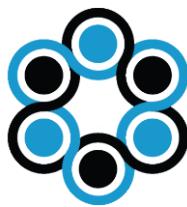
Source Code

```
break;

}System.out.println("=====");
=====
}while (pil!=3);
}
static void daftarpembeli()
{
    try{
        System.out.print("Input Nama = ");
        String nama = input.next();
        System.out.print("Input Password = ");
        String password = input.next();
        System.out.print("Alamat = ");
        String alamat = input.next();
        System.out.print("Input notelp = ");
        String no_telp = input.next();
        pembeli.insertPembeli(nama, password, alamat,
no_telp);
        System.out.println("Daftar Sukses !!");
    }catch (Exception exception){
        System.out.println("Format Salah !!");
    }
}
static void makanan(){
    System.out.print("INPUT NAMA MAKANAN = ");
    String namamakanan = input.next();
    System.out.print("INPUT JUMLAH MAKANAN = ");
    String jumlah_makanan = input.next();
    System.out.print("INPUT HARGA = ");
    String harga = input.next();
    makanan.Insert
(namamakanan,jumlah_makanan,harga);

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

static void view()
{
    for(int
i=0;i<makanan.cekDaftarMakanan(cekmakanan).size();i++)
    {
        System.out.println("Nama Makanan =
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getNamaMak
anan());
    }
}
```



Tugas Praktikum

Source Code

```
System.out.println("Nama Makanan =  
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getNamaMakanan());  
        System.out.println("Jumlah Makanan =  
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getJumlahMakanan());  
        System.out.println("Harga =  
"+makanan.cekDaftarMakanan(cekmakanan).get(i).getHarga());  
  
System.out.println("=====");  
}  
}  
static void remove(String namamakanan)  
{  
    for(int  
i=0;i<makanan.cekDaftarMakanan(cekAdmin).size();i++)  
    {  
  
if(namamakanan.equals(makanan.cekDaftarMakanan(i).get(i)  
.getNamaMakanan()))  
    {  
        makanan.remove(i);  
        System.out.println("Makanan Telah Di  
Hapus");  
  
System.out.println("=====");  
    }  
    else  
    {  
        System.out.println("Makanan Tidak  
Tersedia");  
  
System.out.println("=====");  
    }  
    }  
}  
static void update(String namamakanan,String  
jumlah_makanan, String harga)  
{  
    makanan.update(namamakanan,jumlah_makanan,  
harga);
```



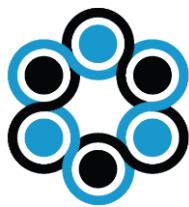
Tugas Praktikum

Source Code

```
static void update(String namamakanan, String jumlah_makanan, String harga)
{
    makanan.update(namamakanan, jumlah_makanan,
harga);
}

static void loginPembeli()
{
    int pil;
    System.out.print("Nama Pembeli : ");
    String nama = input.next();
    System.out.print("Password : ");
    String password = input.next();
    cekPembeli=
AllObjectModel07162.pembeliModel.cekData(nama,
password);
    do
    {
        System.out.println("1. Data Makanan Yang
Tersedia");
        System.out.println("2. Exit");
        System.out.print("Pilih = ");
        pil = input.nextInt();

System.out.println("=====");
=====
        switch(pil)
        {
            case 1:view();
                ;
                break;
            }
        while (pil!=2);
    }
static void loginAdmin(){
    int pil;
    System.out.print("Masukkan Nama Admin : ");
    String nama = input.next();
    System.out.print("Password Admin : ");
    String password = input.next();
    cekAdmin =
AllObjectModel07162.adminModel.cekData(nama, password);
```

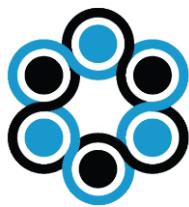


Tugas Praktikum

Source Code

```
do{
    System.out.println("1. Input Data Makanan");
    System.out.println("2. View Data Makanan");
    System.out.println("3. Hapus Data Makanan");
    System.out.println("4. Update Data
Makanan");
    System.out.println("5. Exit");
    System.out.print("Pilih = ");
    pil = input.nextInt();

System.out.println("=====");
=====
switch(pil){
    case 1 : makanan();
        break;
    case 2 : view();
        break;
    case 3 : System.out.print("Inputkan Nama
makanan = ");
        String namamakanan =
input.next();
        remove(namamakanan);
        break;
    case 4 :
        for(int
i=0;i<makanan.cekDaftarMakanan(cekmakanan).size();i++)
        {
            System.out.print("Inputkan
Nama Makanan : ");
            namamakanan = input.next();
            System.out.print("Inputkan
Harga Baru : ");
            String newharga =
input.next();
            System.out.print("Inputkan
Jumlah Makanan : ");
            String newjumlah_makanan =
input.next();
            makanan.Insert(namamakanan,newjumlah_makanan,newharga);
            break;
        }
    }
}
while(pil!=5);
```



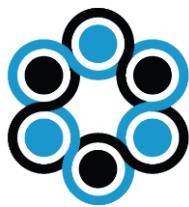
Tugas Praktikum

Source Code

```
makanan.Insert(namamakanan,newjumlah_makanan,newharga);
break;
}
}
while(pil!=5);
}
```

Analisa :

Terdapat pilihan Daftar dan Login. Jika memilih Daftar akan diarahkan ke method daftar pembeli dengan menginputkan nama, password, alamat dan notelp. Jika memilih Login akan diarahkan ke Method Login Admin dan Login Pembeli. Jika memilih login Admin diarahkan ke method login admin lalu memilih pilihan 1 untuk menginputkan nama makanan, jumlah makanan, dan harga. Lalu pilih 2 untuk melihat data makanan, pilih 4 untuk mengupdate, di method update akan mengupdate jumlah makanan dan harga, tetapi sebelum itu mengisi nama makanan yang akan dihapus lalu pilih 2 untuk melihat makanan yang sudah diupdate. Pilih 3 untuk menghapus makanan yang akan dihapus. Jika memilih LoginPembeli maka akan diarahkan ke method login pembeli dengan mengisi nama dan password.



Tugas Praktikum

Screenshot

The screenshot displays three terminal windows (cmd) showing the execution of a Java program named Modul3_07162.java. The program is a simple command-line application for managing food data.

Terminal 1:

```
PS C:\Users\Anindya\OneDrive\Belajar\Pemrograman Berorientasi Objek\Tugas Praktikum> java Modul3_07162.java
SELAMAT DATANG DI SUNDAH MAKAN !!
-----
1. Daftar
2. Login
3. Exit
Pilih = 1

Input Name = annin
Input Password = 123
Alamat = mjk
Input notelp = 088808577083
Daftar Sukses !!
-----
1. Daftar
2. Login
3. Exit
Pilih = 2
-----
1. Login Admin
2. Login pembeli
Pilih = 1
Masukkan Nama Admin : Anindya
Password Admin : 123
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 1
-----
INPUT NAMA MAKANAN = bakso
INPUT JUMLAH MAKANAN = 1
INPUT HARGA = 10000
-----
1. Input Data Makanan
2. View Data Makanan
```

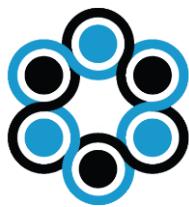
Terminal 2:

```
artPage [Output-Modul3_07162(run)] > Modul3_07162.java [x]
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 2
-----
Nama Makanan = bakso
Jumlah Makanan = 1
Harga = 10000
-----
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 4
-----
Inputkan Nama Makanan : bakso
Inputkan Harga Bakso : 18000
Inputkan Jumlah Makanan : 3
Data Telah Di Update
-----
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 2
-----
Nama Makanan = bakso
Jumlah Makanan = 3
Harga = 18000
-----
1. Input Data Makanan
2. View Data Makanan
```

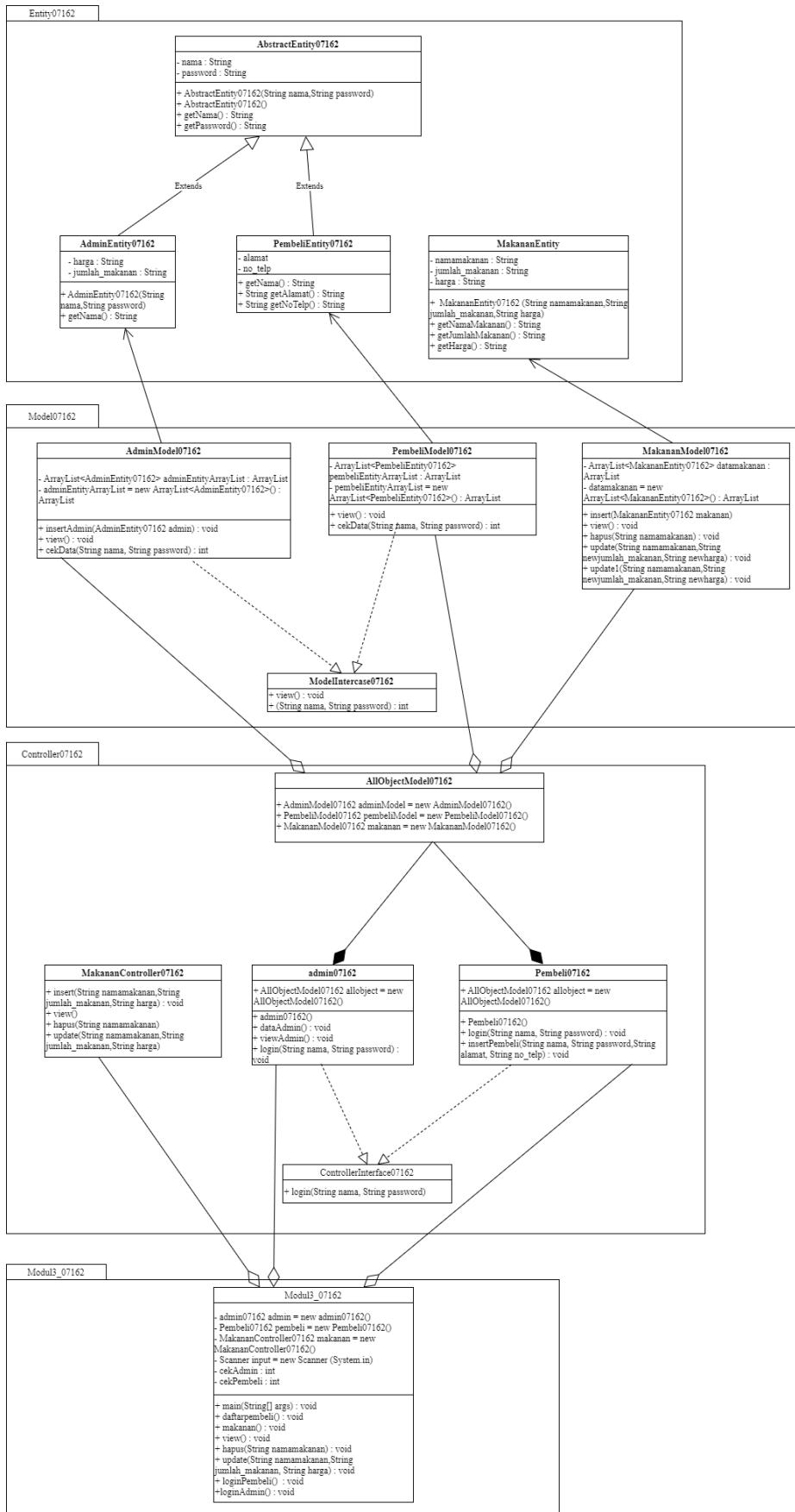
Terminal 3:

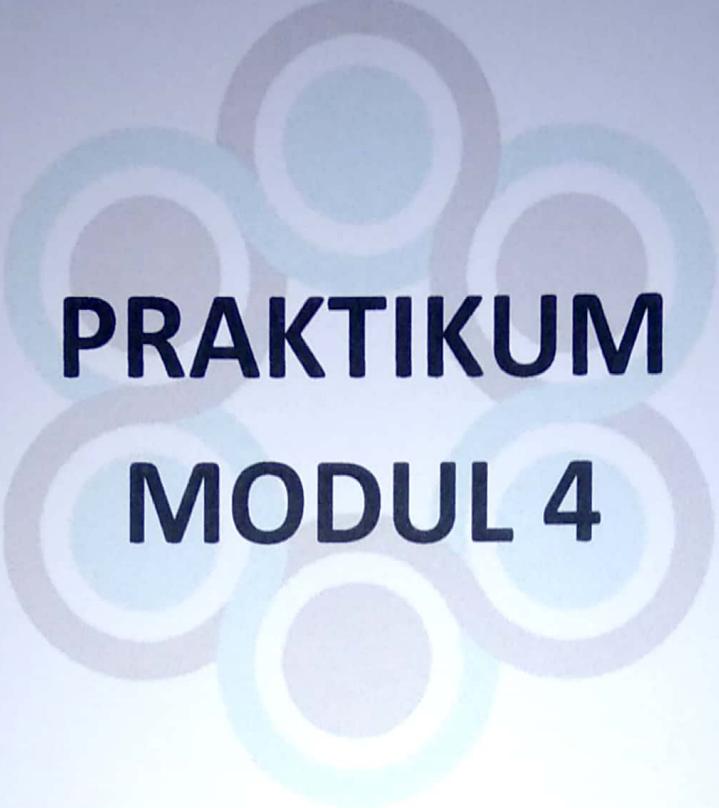
```
StartPage [Output-Modul3_07162(run)] > Modul3_07162.java [x]
5. Exit
Pilih = 3
-----
Nama Makanan = bakso
Jumlah Makanan = 3
Harga = 18000
-----
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 3
-----
Inputkan Nama makanan = bakso
Makanan Telah Di Hapus
-----
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 2
-----
1. Input Data Makanan
2. View Data Makanan
3. Hapus Data Makanan
4. Update Data Makanan
5. Exit
Pilih = 6
-----
1. Daftar
2. Login
3. Exit
Pilih = BUILD STOPPED (total time: 1 minute 31 seconds)
```

Class Diagram



Tugas Praktikum

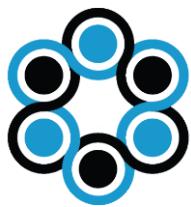




PRAKTIKUM

MODUL 4

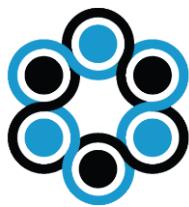
LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



Tugas Praktikum

Soal Praktikum

1. Terapkan *GUI* sesuai dengan *project* masing-masing.
2. Terapkan *CRUD* di *GUI* *project* kalian.
3. Terapkan *JOptionPane* ke *GUI* kalian.



Tugas Praktikum

Langkah

Pertama

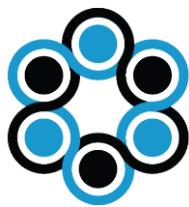
Membuat package Controller07162r, lalu membuat class dengan nama AllObjectModel07162

Source Code

```
package Controller07162;
import Model07162.*;
public class AllObjectModel07162 {
    public static DaftarPembeliModel07162
daftarPembelimodel = new DaftarPembeliModel07162();
    public static AdminModel07162 adminModel = new
AdminModel07162 ();
    public static PembeliModel07162 pembeliModel = new
PembeliModel07162();
}
```

Analisa

Kelas yang diciptakan sebagai penghubung antara class – class yang berada pada package Model dan Entity



Tugas Praktikum

Langkah Kedua

Membuat class dari package Controller07162 dengan nama ControllerInterface07162

Source Code

```
package Controller07162;
public interface ControllerInterface07162 {
    public void login(String nama, String password);
}
```

Analisa

Di class ControllerInterface07162 terdapat void login yang berisi nama dan password untuk masuk sebagai admin/pembeli.



Tugas Praktikum

Langkah Ketiga

Membuat class dari package Controller07162 dengan nama AdminController07162

Source Code

```
package Controller07162;
import Entity.*;
import java.util.ArrayList;
public class AdminController07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public AdminController07162() {
}
    public void dataAdmin() {
        String passwordAdmin[] = {"123"};
        String namaAdmin[] = {"Anindya"};
        for(int i = 0; i < namaAdmin.length; i++) {

AllObjectModel07162.adminModel.insertAdmin(new
AdminEntity07162 (namaAdmin[i],passwordAdmin[i]));
        }
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.adminModel.cekData(nama, password);
    }
    public AdminEntity07162 AdminEntity07162() {
        return
AllObjectModel07162.adminModel.showDataAdmin(indexLogin)
    ;
}
    public void updateIsVerified(int index,int
indexMakanan,PembeliEntity07162 pembeli ) {

AllObjectModel07162.daftarPembeliModel.updateIsVerified(
index, new
        DaftarPembeliEntity07162 (indexMakanan,
pembeli,true));
}
    public ArrayList<DaftarPembeliEntity07162>
cekDaftarPembeliModel() {
        return
AllObjectModel07162.daftarPembeliModel.getdaftarpembeliA
rrayList();
}
```



Tugas Praktikum

Source Code

```
public ArrayList<DaftarPembeliEntity07162>
cekDaftarPembeliModel() {
    return
AllObjectModel07162.daftarPembelimodel.getdaftarpembeliA
rrayList();
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ControllerInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ControllerInterface07162. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Praktikum

Langkah Keempat

Membuat class dari package Controller07162 dengan nama PembeliController07162

Source Code

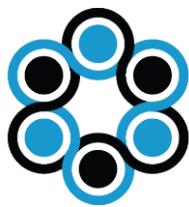
```
package Controller07162;
import Entity.*;
import javax.swing.table.DefaultTableModel;
import java.util.ArrayList;
public class PembeliController07162 implements
ControllerInterface07162 {
    AllObjectModel07162 allobject = new
AllObjectModel07162();
    int indexLogin = 0;
    public PembeliController07162() {
    }
    public PembeliEntity07162 getData() {
        return
AllObjectModel07162.pembeliModel.getpembeliEntityArrayList(indexLogin);
    }

    private static class pembeli {

        public pembeli() {
        }
    }

    public void DaftarMakanan (int indexMakanan,
PembeliEntity07162 pembeli, boolean isVerified) {

AllObjectModel07162.daftarPembelimodel.insertDataPembeli
(new
            DaftarPembeliEntity07162 (indexMakanan, pembeli,
isVerified));
    }
    @Override
    public void login(String nama, String password) {
        indexLogin =
AllObjectModel07162.pembeliModel.cekData(nama,
password);
    }
    public void insert(String nama, String password,
String alamat) {
        AllObjectModel07162.pembeliModel.insert(new
PembeliEntity07162(nama,password,alamat));
    }
}
```



Tugas Praktikum

Source Code

```
}

    }

    public int cekDaftarPembeli (String nama ) {
        int cek =
AllObjectModel07162.daftarPembelimodel.cekData(nama,
null);
        return cek;
    }

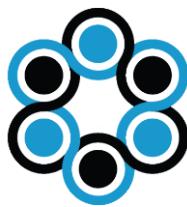
    public DaftarPembeliEntity07162
showDaftarPembeli(int index) {
        return
AllObjectModel07162.daftarPembelimodel.showDaftarPembeli
(index);
    }

    public void update(String makanan, int
Updatemakanan) {

AllObjectModel07162.daftarPembelimodel.update(makanan,
Updatemakanan);
    }
    public void deleteDataPembeli(int index) {

AllObjectModel07162.daftarPembelimodel.hapusPembeliModel
(index);
    }
    public DefaultTableModel daftarpem(){
        DefaultTableModel dtmdaftarpem = new
DefaultTableModel();
        Object[] kolom
={"Password","Nama","Alamat","Makanan","Verified"};
        dtmdaftarpem.setColumnIdentifiers(kolom);
        int size =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().size();
        String verif = null;
        for(int i=0;i<size;i++){

if(AllObjectModel07162.daftarPembelimodel.alldatapembeli
().get(i).isIsVerified()==false){
            verif = "no";
        }else{
            verif = "yes";
        }
    }
}
```



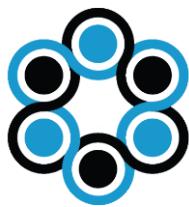
Tugas Praktikum

Source Code

```
        }else{
            verif = "yes";
        }
        Object [] data = new Object[6];
        data[0] =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().
get(i).getPembeli().getNama();
        data[1] =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().
get(i).getPembeli().getPassword();
        data[2] =
AllObjectModel07162.daftarPembelimodel.alldatapembeli().
get(i).getPembeli().getAlamat();
        data[3] =
MakananEntity07162.makanan[AllObjectModel07162.daftarPem
belimodel.alldatapembeli().get(i).getIndexMakanan()];
        data[4] = verif;
        dtmdaftarpem.addRow(data);
    }
    return dtmdaftarpem;
}
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ControllerInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ControllerInterface07162. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override. Kemudian terdapat method daftarpel, ini berguna untuk membuat sebuah tabel untuk daftar pelanggan pada gui.



Tugas Praktikum

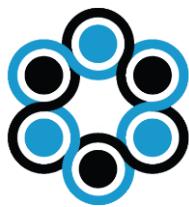
Langkah Kelima

Membuat Package Entity lalu membuat class dengan nama MakananEntity07162
Source Code

```
package Entity;
public class MakananEntity07162 {
    public static String [] makanan = { "bakso = 5000",
    "mie ayam = 8000", "soto = 10000 ", "pecel = 7000" ,
    "rawon = 10000" };
```

Analisa

Di class MakananEntity07162 terdapat Class data array yang berisi daftar pilihan makanan, bertipe data String. Class tidak menerapkan inheritance dan polymorphism. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu public. Public memiliki sifat bahwa properti dan method bisa diakses oleh semua class.



Tugas Praktikum

Langkah Keenam

Membuat class dari package Entity dengan nama AbstractEntity07162
Source Code

```
package Entity;

public class AbstractEntity07162 {
    protected String nama,password;

    public AbstractEntity07162(String nama, String password) {
        this.nama = nama;
        this.password = password;
    }

    public AbstractEntity07162(){}
}

public String getNama() {
    return nama;
}

public void setNama(String nama) {
    this.nama = nama;
}

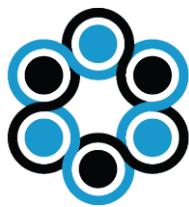
public String getPassword() {
    return password;
}

public void setPassword(String password) {
    this.password = password;
}

}
```

Analisa

Class AbstractEntity07162 digunakan sebagai superclass/parrent class , untuk penerapan inheritance pada project. Terdapat penerapan encapsulasi dalam project, yaitu menggunakan Protected pada properti.dimana protected dapat diakses oleh class itu sendiri dan class turunan.



Tugas Praktikum

Langkah Ketujuh

Membuat class dari package Entity dengan nama AdminEntity07162

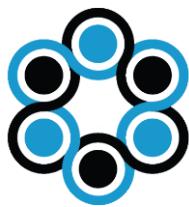
Source Code

```
package Entity;

public class AdminEntity07162 extends
AbstractEntity07162 {
    public AdminEntity07162(String nama,String
password) {
        super (nama,password);
    }
    @Override
    public String getNama(){
        return nama;
    }
}
```

Analisa

Pada class AdminEntity07162 menerapkan inheritance/pewarisan, dimana class AbstractEntity07162 menurunkan method dan property kepada AdminEntity07162. Pada keyword super, ini digunakan untuk memanggil konstruktor dari class AbstractEntity07162. Override ini untuk pendefinisian ulang method yang sama pada class AbstractEntity07162



Tugas Praktikum

Langkah Kedelapan

Membuat class dari package Entity dengan nama PembeliEntity07162.

Source Code

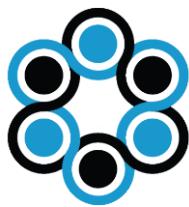
```
package Entity;

public class PembeliEntity07162 extends AbstractEntity07162
{
    private String alamat;

    public PembeliEntity07162(String nama, String
password, String alamat) {
        super(nama, password);
        this.alamat = alamat;
    }
    @Override
    public String getNama() {
        return nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }
}
```

Analisa

Pada class PembeliEntity07162 menerapkan inheritance/pewarisan, dimana class AbstractEntity07162 menurunkan method dan property kepada PembeliEntity07162 . Pada keyword super, ini digunakan untuk memanggil konstruktor dari classAbstractEntity07162. Override ini untuk pendefinisian ulang method yang sama pada class AbstractEnttity07162.



Tugas Praktikum

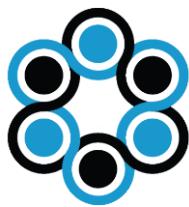
Langkah keSembilan

Membuat class dari package Entity dengan nama DaftarPembeliEntity07162.
Source Code

```
package Entity;
public class DaftarPembeliEntity07162 {
    private PembeliEntity07162 pembeli;
    private boolean isVerified;
    private int indexMakanan;
    public DaftarPembeliEntity07162(int indexMakanan,
                                    PembeliEntity07162 pembeli, boolean
    isVerified) {
        this.indexMakanan = indexMakanan;
        this.pembeli = pembeli;
        this.isVerified = isVerified;
    }
    public PembeliEntity07162 getPembeli() {
        return pembeli;
    }
    public boolean isIsVerified() {
        return isVerified;
    }
    public int getIndexMakanan() {
        return indexMakanan;
    }
    public void setPembeli(PembeliEntity07162 pembeli) {
        this.pembeli= pembeli;
    }
    public void setIsVerified(boolean isVerified) {
        this.isVerified = isVerified;
    }
    public void setIndexMakanan(int indexMakanan) {
        this.indexMakanan = indexMakanan;
    }
}
```

Analisa

Di class DaftarPembeliEntity07162 terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat variable untuk menampung data pelanggan, serta isVerified untuk mengonfirmasi status pelanggan nantinya, dan juga indexMakanan untuk menginput data pilihan makanan.



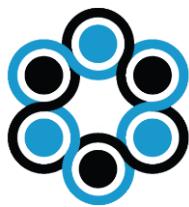
Tugas Praktikum

Langkah KeSepuluh

Membuat package Model dengan nama Model07162 lalu membuat class dengan nama AdminModel07162

Source Code

```
package Model07162;
import Entity.AdminEntity07162;
import java.util.ArrayList;
public class AdminModel07162 implements
ModelInterface07162
{
    private ArrayList<AdminEntity07162>
adminEntityArrayList;
    public AdminModel07162()
    {
        adminEntityArrayList = new
ArrayList<AdminEntity07162>();
    }
    public void insertAdmin(AdminEntity07162 admin)
    {
        adminEntityArrayList.add(admin);
    }
    public void view() {
        for (AdminEntity07162 adminEntity :
adminEntityArrayList) {
            System.out.println(adminEntity.getNama());
            System.out.println(adminEntity.getPassword());
        }
    }
    @Override
    public int cekData(String nama, String password){
        int loop = 0;
        for(AdminEntity07162 admin :
adminEntityArrayList){
            if(admin.getNama().equals(nama) &&
admin.getPassword().equals(password)){
                break;
            }else{
                loop++;
            }
        }
        return loop;
    }
    public AdminEntity07162 showDataAdmin(int index){
```



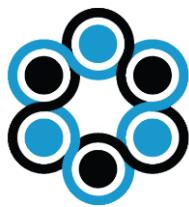
Tugas Praktikum

Source Code

```
public AdminEntity07162 showDataAdmin(int index){  
    return adminEntityArrayList.get(index);  
}  
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ModelInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ModelInterface07162. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Praktikum

Langkah keSembelas

Membuat class dengan nama DaftarPembeliModel07162.

Source Code

```
package Model07162;
import Entity.*;
import java.util.ArrayList;
import java.text.ParseException;
public class DaftarPembeliModel07162 implements
ModelInterface07162 {
    private ArrayList<DaftarPembeliEntity07162>
daftarpembeliArrayList;
    public DaftarPembeliModel07162(){
        daftarpembeliArrayList = new
ArrayList<DaftarPembeliEntity07162>();
    }
    public void
insertDataPembeli(DaftarPembeliEntity07162
daftarpembeli){
        daftarpembeliArrayList.add(daftarpembeli);
    }
    public ArrayList<DaftarPembeliEntity07162>
getdaftarpembeliArrayList(){
        return daftarpembeliArrayList;
    }
    public int size() {
        return daftarpembeliArrayList.size();
    }
    @Override
    public void view() {
        for (DaftarPembeliEntity07162 pembelientity :
daftarpembeliArrayList) {

System.out.println("=====");
System.out.println("nama : " +
pembelientity.getPembeli().getNama()+"\nPassword : " +
pembelientity.getPembeli().getPassword() + "\nAlamat :
"+pembelientity.getPembeli().getAlamat()+"\n Makanan :
"+MakananEntity07162.makanan[pembelientity.getIndexMakanan()] + "\n isVerified : ");
        if(pembelientity.isIsVerified()==false) {
            System.out.println("Belum Di Verifikasi
Admin");
        } else {
```



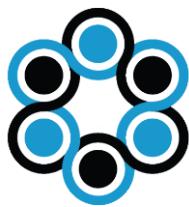
Tugas Praktikum

Source Code

```
System.out.println("Sudah Di Verifikasi Admin");
    }

System.out.println("=====");
=====
}
@Override
public int cekData(String nama, String password ){
    int loop = 0;
    if(daftarpembeliArrayList.size() ==0){
        loop = -1;//data kosong
    }else{
        for(int i = 0; i <daftarpembeliArrayList.size(); i++){
            if(daftarpembeliArrayList.get(i).getPembeli().getPassword().equals(password)){
                loop = i;
                break;
            }else{
                loop = -2;
            }
        }
    }    return loop;
}
public DaftarPembeliEntity07162
showDaftarPembeli(int index){
    return daftarpembeliArrayList.get(index);
//menampilkan data pembeli
}
public void hapusPembeliModel(int index) {

daftarpembeliArrayList.remove(daftarpembeliArrayList.get(index)); //menghapus data pembeli
}
public void updateIsVerified(int
index, DaftarPembeliEntity07162 daftarpembeli){
    daftarpembeliArrayList.set(index, daftarpembeli);
}
public ArrayList <DaftarPembeliEntity07162>
alldatapembeli(){
    return daftarpembeliArrayList;
}
private int cariMakanan(String makanan){
```



Tugas Praktikum

Source Code

```
private int cariMakanan(String makanan){
    int index=-1;
    for(int i=0;i<daftarpembeliArrayList.size();i++) {

        if(makanan.equals(daftarpembeliArrayList.get(i).getPembeli().getNama()))
            index=i;
    }
    return index;
}
public void update(String makanan, int Updatemakanan){
    int data;
    if(cariMakanan(makanan)!=-1){

        daftarpembeliArrayList.get(cariMakanan(makanan)).setIndexMakanan(Updatemakanan);
        data = 1;
    }
    else {
        data = 0;
    }
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ModelInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ModelInterface07162. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Praktikum

Langkah KeSebelas

Membuat class interface class dengan nama Modelinterface07162.

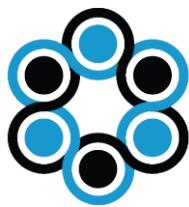
Source Code

```
package Model07162;

public interface ModelInterface07162 {
    public void view();
    public int cekData (String nama, String
password);
}
```

Analisa

Kelas interface hanya terdiri dari method-method kosong (abstract, tanpa body method). Interface biasanya lebih digunakan untuk mengelompokkan behaviorbehavior menjadi grup behavior. Kelas interface menggambarkan kemampuan, bukan entitas/konsep objek



Tugas Praktikum

Langkah keDuaBebelas

Membuat class interface dengan nama PembeliModel07162

Source Code

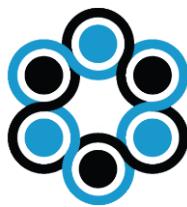
```
package Model07162;
import Entity.PembeliEntity07162;
import java.util.ArrayList;
public class PembeliModel07162 implements
ModelInterface07162
{
    private ArrayList<PembeliEntity07162>
pembeliEntityArrayList;

    public PembeliModel07162()
    {
        pembeliEntityArrayList = new
ArrayList<PembeliEntity07162>();
    }
    public void insert(PembeliEntity07162
pembeliEntity){ //menambah data pembeli
    pembeliEntityArrayList.add(pembeliEntity);
}

@Override
public void view() {
    for (PembeliEntity07162 pembeliEntity :
pembeliEntityArrayList) {

System.out.println("=====");
System.out.println("nama : "+
pembeliEntity.getNama()+"\npassword : "+
pembeliEntity.getPassword()+"\nAlamat :
"+pembeliEntity.getAlamat()));

System.out.println("=====");
}
}
public int cekData(String nama, String password){
    int loop = 0;
    while
(!pembeliEntityArrayList.get(loop).getNama().equals(nam
a) &&
!pembeliEntityArrayList.get(loop).getPassword().equals(p
assword))
loop++;
}
}
```



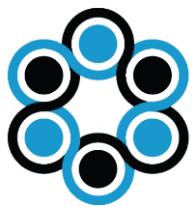
Tugas Praktikum

Source Code

```
public int cekData(String nama, String password){  
    int loop = 0;  
    while  
(!pembeliEntityArrayList.get(loop).getNama().equals(nam  
a) &&  
  
!pembeliEntityArrayList.get(loop).getPassword().equals(  
password)){  
    loop++;  
}  
return loop;  
}  
public PembeliEntity07162  
getpembeliEntityArrayList(int index){ //  
    return pembeliEntityArrayList.get(index);  
}  
}
```

Analisa

Terdapat implementasi yang diterapkan pada class, class ini mengimplementasi dari class ModelInterface07162. Class melakukan implementasi agar class bisa mengadopsi behavior-behavior dari class ModelInterface07162. Terdapat enkapsulasi yang diterapkan pada class ini, yaitu private dan public. Private memiliki sifat bahwa properti dan method hanya bisa diakses oleh class itu sendiri, sedangkan public dapat diakses oleh semua class. Terdapat polymorphism yang diterapkan pada class ini yaitu method overriding, method overriding digunakan untuk pendefinisian ulang method yang sama pada subclass, yang ditandai dengan keyword @Override.



Tugas Praktikum

Langkah KeTigaBelas

Membuat Package dengan nama View07162 lalu buat class dengan nama AdminGUI07162.

Source Code

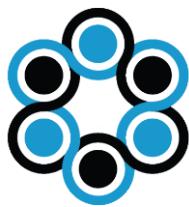
```
package View07162;
import Controller07162.AllObjectModel07162;
import Controller07162.PembeliController07162;
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import javax.swing.*;
public class AdminGUI07162 {
    private static PembeliController07162 pembeli = new
PembeliController07162();
    JFrame admin = new JFrame();
    JTable tabelpembeli = new JTable();
    JScrollPane scrolladmin = new
JScrollPane(tabelpembeli);
    JButton back,verif,hapus;
    JLabel Admin,passlabel,namalabel;
    JTextField passtext,namatext;
    public AdminGUI07162(){
        admin.setSize(650,500);
        admin.setLayout(null);

admin.getContentPane().setBackground(Color.ORANGE);

        Admin = new JLabel("Admin");
        admin.setBounds(250, 30, 200, 30);
        admin.setFont(new Font("Timer New
Roman",Font.BOLD,40));
        admin.add(Admin);

        scrolladmin.setBounds(30, 100, 570, 200);
        tabelpembeli.setModel(pembeli.daftarpem());
        admin.add(scrolladmin);

        namalabel = new JLabel("NAMA");
        namalabel.setBounds(50, 320, 100, 30);
        admin.add(namalabel);
```



Tugas Praktikum

Source Code

```
namalabel = new JLabel("NAMA");
namalabel.setBounds(50, 320, 100, 30);
admin.add(namalabel);

namatext = new JTextField();
namatext.setBounds(50, 350, 100, 30);
admin.add(namatext);

passlabel = new JLabel("Password");
passlabel.setBounds(200, 320, 100, 30);
admin.add(passlabel);

passtext = new JTextField();
passtext.setBounds(200, 350, 100, 30);
admin.add(passtext);

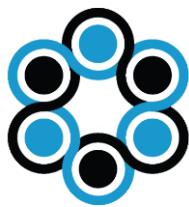
verif = new JButton("verif");
verif.setBounds(350, 350, 100, 30);
verif.setBackground(Color.GRAY);
admin.add(verif);

back = new JButton("back");
back.setBounds(50, 400, 100, 30);
back.setBackground(Color.GRAY);
admin.add(back);

hapus = new JButton("Hapus");
hapus.setBounds(350, 400, 100, 30);
hapus.setBackground(Color.GRAY);
admin.add(hapus);

admin.setLocationRelativeTo(null);
admin.setVisible(true);

admin.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
back.addActionListener(new ActionListener()
{
    @Override
    public void actionPerformed(ActionEvent
ae){
        admin.dispose();
        GUI07162 men = new GUI07162(); //nama
objek men
    }
    ..
})
```



Tugas Praktikum

Source Code

```
    }
    });
    verif.addActionListener(new ActionListener(){
        @Override
        public void actionPerformed(ActionEvent ae){ //untuk verifikasi nama dan password
            String nama = namatext.getText();
            String pass = passtext.getText();
            int index =
AllObjectModel07162.daftarPembelimodel.cekData(nama,
pass); //dicek

Allobjctrl07162.admin.updateIsVerified(index,
Allobjctrl07162.Pembeli.showDaftarPembeli(index).getInd
exMakanan(),Allobjctrl07162.Pembeli.showDaftarPembeli(i
ndex).getPembeli());

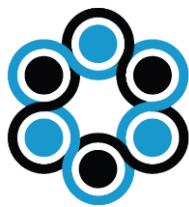
tabelpembeli.setModel(pembeli.daftarpem());
}
});
hapus.addActionListener(new ActionListener(){
    @Override
    public void actionPerformed(ActionEvent ae){

Allobjctrl07162.Pembeli.deleteDataPembeli(0);

tabelpembeli.setModel(pembeli.daftarpem());
}
});
tabelpembeli.addMouseListener(new
MouseAdapter(){
    @Override
    public void mouseClicked(MouseEvent e){
//method untuk kursor
        int i = tabelpembeli.getSelectedRow();

namatext.setText(Allobjctrl07162.Pembeli.daftarpem().ge
tValueAt(i, 0).toString());

passtext.setText(Allobjctrl07162.Pembeli.daftarpem().ge
tValueAt(i, 1).toString());
}
});
}
```



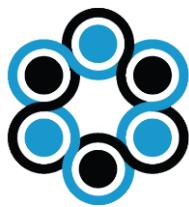
Tugas Praktikum

Source Code

```
public void mouseClicked(MouseEvent e){ //method  
untuk kursor  
    int i = tabelpembeli.getSelectedRow();  
  
    namatext.setText(Allobjctrl07162.Pembeli.daftarpem().get  
ValueAt(i, 0).toString());  
  
    passtext.setText(Allobjctrl07162.Pembeli.daftarpem().get  
ValueAt(i, 1).toString());  
    }  
});  
}  
}
```

Analisa

Pada class AdminGUI07162 merupakan tampilan GUI pada saat kita login sebagai admin, kita bisa verifikasi data pembeli dan hapus pembeli



Tugas Praktikum

Langkah keEmpatBelas

Membuat kelas dari Package view07162 dengan nama PembeliGUI07162
Source Code

```
package View07162;
import Entity.MakananEntity07162;
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.text.SimpleDateFormat;
import javax.swing.*;
public class PembeliGUI07162 {
    JFrame Pembeli = new JFrame();
    JButton back,daftarbtn;
    JTextArea area = new JTextArea();
    JLabel datapembeli,daftarmakanan;
    JComboBox pilmakanan = new
    JComboBox(MakananEntity07162.makanan);
    int cek = Allobjctrl07162.Pembeli.cekDaftarPembeli
    (Allobjctrl07162.Pembeli.getData().getPassword());
    public PembeliGUI07162(){
        Pembeli.setSize(720, 600);
        Pembeli.setLayout(null);

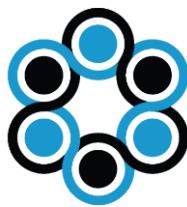
        Pembeli.getContentPane().setBackground(Color.PINK);

        datapembeli = new JLabel("Data Pembeli");
        datapembeli.setFont(new Font("Timer New
Roman",Font.BOLD,30));
        datapembeli.setBounds(90, 30, 200, 30);
        Pembeli.add(datapembeli);

        area.setBounds(30, 90, 400, 300);
        Pembeli.add(area);

        daftarmakanan = new JLabel("Daftar Makanan");
        daftarmakanan.setBounds(450, 30, 250, 30);
        daftarmakanan.setFont(new Font("Times New
Roman",Font.BOLD,30));
        Pembeli.add(daftarmakanan);

        pilmakanan.setBounds(450, 90, 230, 30);
        Pembeli.add(pilmakanan);
```



Tugas Praktikum

```
daftarbtn = new JButton("Daftar");
daftarbtn.setBounds(500, 200, 100, 30);
daftarbtn.setBackground(Color.GRAY);
Pembeli.add(daftarbtn);

back = new JButton("Back");
back.setBounds(30, 500, 100, 30);
back.setBackground(Color.red);

Pembeli.add(back);
Pembeli.setVisible(true);
Pembeli.setLocationRelativeTo(null);

Pembeli.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
if(cek== -1){ //cek data, apabila kita belum
daftar pembeli
    JOptionPane.showMessageDialog(null, "Anda
Belum Daftar sebagai pembeli", "Information",
JOptionPane.INFORMATION_MESSAGE);
} else if(cek== -2){
    JOptionPane.showMessageDialog(null, "Anda
Belum Daftar sebagai pembeli", "Information",
JOptionPane.INFORMATION_MESSAGE);
} else{
    area.setText(datapembeli());
}
daftarbtn.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent ae){
    int indexmakanan =
pilmakanan.getSelectedIndex();
    Allobjctrl07162.Pembeli.DaftarMakanan

(indexmakanan,Allobjctrl07162.Pembeli.getData(), false);
    area.setText(datapembeli());
}
});

back.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent ae){
    Pembeli.dispose();
    GUI07162 men = new GUI07162(); //nama
objek menu
}
})
```



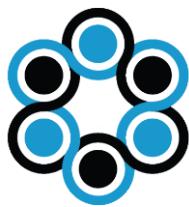
Tugas Praktikum

```
        }
    });
}
String datapembeli(){
    int cek =
Allobjctrl07162.Pembeli.cekDaftarPembeli(Allobjctrl0716
2.Pembeli.getData().getNama());
    String cekverif;

if(Allobjctrl07162.Pembeli.showDaftarPembeli(cek).isIsV
erified()==false){
    cekverif = "belum diverifikasi";
}else{
    cekverif = "sudah diverifikasi";
}
String text = "Nama =
"+Allobjctrl07162.Pembeli.showDaftarPembeli(cek).getPem
beli().getNama()+"\n"
+ "Alamat =
"+Allobjctrl07162.Pembeli.showDaftarPembeli(cek).getPem
beli().getAlamat()+"\n"
+ "Verifikasi = "+cekverif+"\n"
+ "Makanan =
"+MakananEntity07162.makanan[Allobjctrl07162.Pembeli.sh
owDaftarPembeli(cek).getIndexMakanan()];
    return text;
}
}
```

Analisa :

Pada class *PembeliGUI07162*, pada kelas ini merupakan tampilan GUI pada saat kita login sebagai pembeli, kita bisa daftar makanan terlebih dahulu, ketika sudah daftar, makan akan muncul data pembeli



Tugas Praktikum

Langkah keLimaBelas

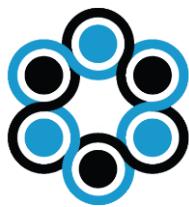
Membuat class dari package view07162 lalu beri nama Allobjctrl07162

Source Code

```
package View07162;
import Controller07162AdminController07162;
import Controller07162PembeliController07162;
public class Allobjctrl07162 {
    public static AdminController07162 admin = new
AdminController07162();
    public static PembeliController07162 Pembeli = new
PembeliController07162();
}
```

Analisa :

Class Allobjctrl07162 untuk memanggil objek AdminController07162 dan PembeliController07162



Tugas Praktikum

Langkah keEnamBelas

Membuat class dari package view07162 lalu beri nama GUI07162

Source Code

```
package View07162;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;

public class GUI07162 {
    JFrame TampilanAwal = new JFrame(); //tampilan awal
    JLabel login,daftar,top;
    JRadioButton radioadmin,radio_pembeli; //tombol
    pilihan
    JTextField textnamalogin,textnamadaftar, textalamat;
    JLabel
    labelnamalogin,labelpasswordlogin,labelnamadaftar, labelpa
    ssworddaftar,labelalamat;
    JButton check,reg,update;
    JPasswordField passwordlogin,passworddaftar;

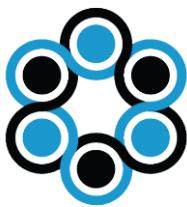
    private boolean ceklogin = false;
    public GUI07162(){ //komponen
        TampilanAwal.setSize(700, 630); //ukuran
        TampilanAwal.setLayout(null);

        TampilanAwal.getContentPane().setBackground(Color.YELLOW)
        ; //warna background

        top = new JLabel("Rumah Makan");
        top.setBounds(225, 10, 600, 50);
        top.setFont(new Font("Times New
        Roman",Font.BOLD,40));
        TampilanAwal.add(top);

        login = new JLabel("Login");
        login.setBounds(100, 50, 100, 100);
        login.setFont(new Font("Times New
        Roman",Font.BOLD,30));
        TampilanAwal.add(login);

        radioadmin = new JRadioButton("admin");
        radioadmin.setBounds(30, 150, 100, 30);
        radioadmin.setBackground(Color.CYAN);
        - - - - -
```



Tugas Praktikum

Source Code

```
TampilanAwal.add(radioadmin);

radiopembeli = new JRadioButton("pembeli");
radiopembeli.setBounds(140, 150, 100, 30);
radiopembeli.setBackground(Color.CYAN);
TampilanAwal.add(radiopembeli);

labelnamalogin = new JLabel("Nama");
labelnamalogin.setBounds(30, 180, 50, 30);
TampilanAwal.add(labelnamalogin);

textnamalogin = new JTextField();
textnamalogin.setBounds(30, 210, 200, 30);
TampilanAwal.add(textnamalogin);

labelpasswordlogin = new JLabel("Password");
labelpasswordlogin.setBounds(30, 240, 100, 30);
TampilanAwal.add(labelpasswordlogin);

passwordlogin = new JPasswordField();
passwordlogin.setBounds(30, 270, 200, 30);
TampilanAwal.add(passwordlogin);

check = new JButton("check");
check.setBounds(90, 320, 100, 40);
check.setBackground(Color.CYAN);
TampilanAwal.add(check);

update = new JButton("update");
update.setBounds(90, 400, 100, 40);
update.setBackground(Color.CYAN);
TampilanAwal.add(update);

daftar = new JLabel("Buat Akun");
daftar.setFont(new Font("Times New
Roman",Font.BOLD,30));
daftar.setBounds(440, 50, 150, 100);
TampilanAwal.add(daftar);
labelnamadaftar = new JLabel("Nama");
labelnamadaftar.setBounds(400, 210, 100, 30);
TampilanAwal.add(labelnamadaftar);
textnamadaftar = new JTextField();
textnamadaftar.setBounds(400, 240, 200, 30);
TampilanAwal.add(textnamadaftar);
....
```



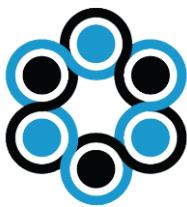
Tugas Praktikum

Source Code

```
labelpassworddaftar = new JLabel("Password");
    labelpassworddaftar.setBounds(400, 270, 100,
30);
    TampilanAwal.add(labelpassworddaftar);
    passworddaftar = new JPasswordField();
    passworddaftar.setBounds(400, 300, 200, 30);
    TampilanAwal.add(passworddaftar);
    labelalamat = new JLabel("Alamat");
    labelalamat.setBounds(400, 330, 100, 30);
    TampilanAwal.add(labelalamat);
    textalamat = new JTextField();
    textalamat.setBounds(400, 360, 200, 30);
    TampilanAwal.add(textalamat);

    reg = new JButton("Daftar");
    reg.setBounds(460, 520, 100, 40);
    reg.setBackground(Color.CYAN);
    TampilanAwal.add(reg);

TampilanAwal.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    TampilanAwal.setVisible(true);
    TampilanAwal.setLocationRelativeTo(null);
    radioadmin.addActionListener(new
ActionListener(){
        @Override
        public void actionPerformed(ActionEvent ae){
            if(radioadmin.isSelected()){
                radiopembeli.setSelected(false);
                ceklogin=true;
            }
        }
    });
    radiopembeli.addActionListener(new
ActionListener(){ //untuk pembeli
        @Override
        public void actionPerformed(ActionEvent ae){
            if(radiopembeli.isSelected()){
                radioadmin.setSelected(false);
                ceklogin=false;
            }
        }
    });
    check.addActionListener(new ActionListener(){
        @Override
```



Tugas Praktikum

Source Code

```
public void actionPerformed(ActionEvent ae){ // login untuk admin
    if(ceklogin==true){
        try{
            Allobjctrl07162.admin.dataAdmin();

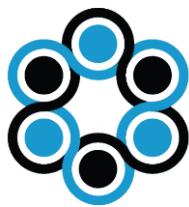
            Allobjctrl07162.admin.login(textnamalogin.getText(),
            passwordlogin.getText());
            String nama =
            Allobjctrl07162.admin.AdminEntity07162().getNama();

            JOptionPane.showMessageDialog(null, "Selamat datang "+nama,"information", JOptionPane.INFORMATION_MESSAGE);
            AdminGUI07162 admin = new AdminGUI07162();
            TampilanAwal.dispose();
        }catch(Exception exception){

            JOptionPane.showMessageDialog(null, "Nama atau Password salah", "information", JOptionPane.INFORMATION_MESSAGE);
        }
        kosong(); //method kosong, jika datanya kosong
    }else{
        try{ //login untuk pembeli
            Allobjctrl07162.Pembeli.login(textnamalogin.getText(),passwordlogin.getText());
            String nama =
            Allobjctrl07162.Pembeli.getData().getNama();
            //memasukkan nama dan password jika benar

            JOptionPane.showMessageDialog(null, "Selamat datang "+nama,"information", JOptionPane.INFORMATION_MESSAGE);
            PembeliGUI07162 prak = new PembeliGUI07162();
            TampilanAwal.dispose();
        }catch(Exception eception){ //jika nama atau password salah akan muncul tulisan

            JOptionPane.showMessageDialog(null, "Nama atau Password salah", "information", JOptionPane.INFORMATION_MESSAGE);
            kosong();
        }
    }
}
```



Tugas Praktikum

Source Code

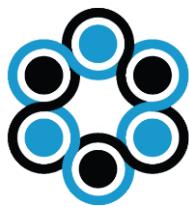
```
        }
    }
});
update.addActionListener(new ActionListener(){

    @Override
    public void actionPerformed(ActionEvent ae)
{
    Update07162 update =new Update07162();
    TampilanAwal.dispose();
}

});

reg.addActionListener(new ActionListener(){
    @Override
    public void actionPerformed(ActionEvent ae){
        try{
            String nama =
textnamadaftar.getText(); //registrasi
            String password =
passworddaftar.getText();
            String alamat =
textalamat.getText();

Allobjctrl07162.Pembeli.insert(nama,password,alamat);
//jika sukses maka muncul tulisan
            JOptionPane.showMessageDialog(null,
"Registrasi sukses", "information",
JOptionPane.INFORMATION_MESSAGE);
            kosong();
        }catch(Exception exception){
            JOptionPane.showMessageDialog(null,
"Format Penulisan Salah", "Registrasi Gagal",
JOptionPane.INFORMATION_MESSAGE);
        }
    }
});
void kosong(){
    textnamadaftar.setText(null); //settext null
semua
    passworddaftar.setText(null);
}
```



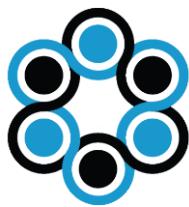
Tugas Praktikum

Source Code

```
void kosong(){
    textnamadaftar.setText(null); //settext null
semua
    passworddaftar.setText(null);
    textalamat.setText(null);
    textnamalogin.setText(null);
    passwordlogin.setText(null);
}
}
```

Analisa :

Pada class GUI07162, pada kelas ini merupakan tampilan GUI Utama pada saat pertama kali Run. pada Gui ini kita dapat daftar pembelin, login sebagai admin dan login sebagai pembeli



Tugas Praktikum

Langkah keTujuhBelas

Membuat class dari package view07162 lalu beri nama Update07162

Source Code

```
package View07162;
import Entity.MakananEntity07162;
import java.awt.Color;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.Font;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
public class Update07162 {
    JComboBox pilmakanan = new
    JComboBox(MakananEntity07162.makanan);
    JButton konfirmasi,kembali;
    JFrame TampilanAwal = new JFrame();
    JTextArea area = new JTextArea();
    JLabel update,labelmakananUpdate;
    JTextField textmakananUpdate;
    public Update07162() {
        TampilanAwal.setSize(720,600);
        TampilanAwal.setLayout(null);

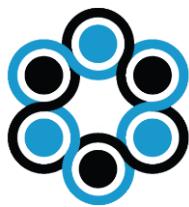
        TampilanAwal.getContentPane().setBackground(Color.LIGHT
        _GRAY);

        update = new JLabel("Update Data");
        update.setFont(new Font("Times New
        Roman",Font.BOLD,43));
        update.setBounds(240,30,300,80);
        TampilanAwal.add(update);

        labelmakananUpdate = new JLabel("Nama
        Pembeli");
        labelmakananUpdate.setBounds(300,180,150,30);
        TampilanAwal.add(labelmakananUpdate);

        textmakananUpdate = new JTextField();
        textmakananUpdate.setBounds(250,210,200,40);
        TampilanAwal.add(textmakananUpdate);

        pilmakanan.setBounds(276,270,150,40);
        TampilanAwal.add(pilmakanan);
```



Tugas Praktikum

Source Code

```
pilmakanan.setBounds(276,270,150,40);
    TampilanAwal.add(pilmakanan);

    konfirmasi = new JButton("KONFIRMASI");
    konfirmasi.setBounds(290,330,120,40);
    konfirmasi.setBackground(Color.RED);
    TampilanAwal.add(konfirmasi);

    kembali = new JButton("KEMBALI");
    kembali.setBounds(290,380,120,40);
    kembali.setBackground(Color.GREEN);
    TampilanAwal.add(kembali);

TampilanAwal.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    TampilanAwal. setVisible(true);
    TampilanAwal.setLocationRelativeTo(null);

    konfirmasi.addActionListener(new
ActionListener(){

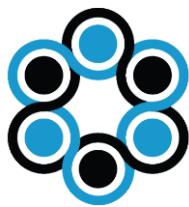
        @Override
        public void actionPerformed(ActionEvent ae)
{
            int makananUpdate =
pilmakanan.getSelectedIndex();

Allobjctr107162.Pembeli.update(textmakananUpdate.getText
(), makananUpdate);

JOptionPane.showMessageDialog(null,"BERHASIL DI
UPDATE","INFORMATION", JOptionPane.INFORMATION_MESSAGE);
            kosong();
}
});

    kembali.addActionListener(new ActionListener(){

        @Override
        public void actionPerformed(ActionEvent ae)
{
            TampilanAwal.dispose();
GUJT07162 men = new GUJT07162();
```



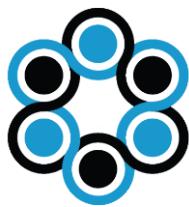
Tugas Praktikum

Source Code

```
@Override  
    public void  
actionPerformed(ActionEvent ae) {  
    TampilanAwal.dispose();  
    GUI07162 men = new GUI07162();  
}  
  
});  
}  
void kosong() {  
    textmakananUpdate.setText(null);  
}  
}
```

Analisa

Pada class *UpdateGUI07162*, pada kelas ini merupakan tampilan GUI pada saat kita update dengan memasukkan nama lalu memilih makanan yang akan di update



Tugas Praktikum

Langkah keDelapanBelas

Membuat class main pada Package AnindyaModul4_07162

Source Code

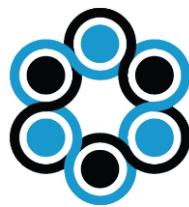
```
package anindyamodul4_07162;
import View07162.GUI07162;

public class AnindyaModul4_07162 {

    public static void main(String[] args) {
        GUI07162 gui = new GUI07162();
    }
}
```

Analisa

Pada main program, kita bisa memilih Gui, jika Gui akan ditujukan ke class GUI07162.



Tugas Praktikum

ScreenShoot

The screenshot shows a Windows application window titled "Rumah Makan". The window is divided into two main sections: "Login" on the left and "Buat Akun" (Create Account) on the right.

Login Section:

- Contains radio buttons for "admin" and "pembeli".
- Text input fields for "Nama" (Name) and "Password".
- Buttons for "check" and "update".

Buat Akun Section:

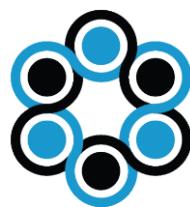
- Text input fields for "Nama" (Name), "Password", and "Alamat" (Address). The "Nama" field contains "Anindya", the "Password" field contains "***", and the "Alamat" field contains "kwatu".
- A "Daftar" (Register) button.

System Tray and Taskbar:

- Windows taskbar at the bottom with various pinned icons.
- Date and time: 15/01/2021, 05:50.

The screenshot shows the same application window after a successful registration. A modal dialog box titled "information" appears in the center, displaying the message "Registrasi sukses" (Registration successful) with an "OK" button. The "Buat Akun" section's input fields now show the registered values: "Anindya" in the Name field, "***" in the Password field, and "kwatu" in the Address field.

The screenshot shows the application window again, but the "Buat Akun" section's input fields have been cleared. The "Nama" field now contains "Anindya", the "Password" field contains "***", and the "Alamat" field is empty.



Tugas Praktikum

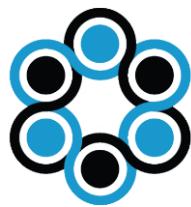
ScreenShoot

The screenshots show a Windows application interface for a restaurant system named "Rumah Makan".

Screenshot 1: The main window titled "Rumah Makan" contains two sections: "Login" and "Buat Akun". The "Login" section has radio buttons for "admin" (selected) and "pembeli", and fields for "Nama" (Anindya) and "Password" (***). Buttons include "check", "update", and "Daftar". The "Buat Akun" section has fields for "Nama", "Password", and "Alamat", with a "Dafat" button. A message box is displayed: "Selamat datang Anindya".

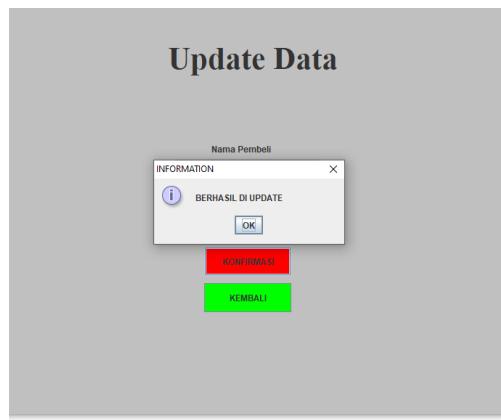
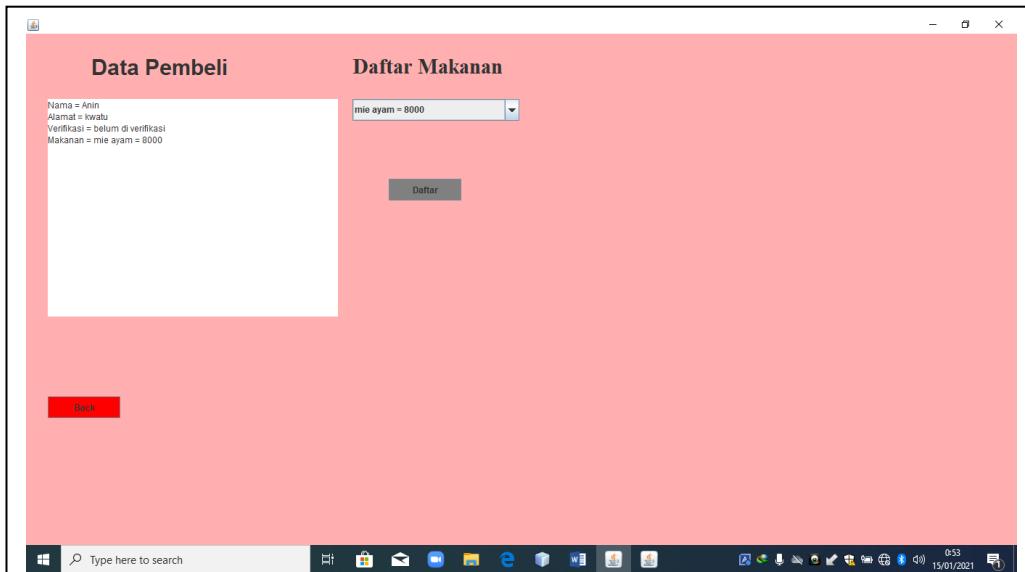
Screenshot 2: The main window shows the "Data Pembeli" and "Daftar Makanan" sections. The "Data Pembeli" section has a dropdown menu showing "bakso = 5000". A message box says "Anda Belum Daftar sebagai pembeli". The "Daftar" button is visible. A "Back" button is at the bottom.

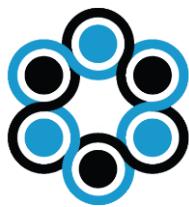
Screenshot 3: The main window shows the "Data Pembeli" and "Daftar Makanan" sections. The "Data Pembeli" section has a dropdown menu showing "mie ayam = 8000". A "Daftar" button is visible. A "Back" button is at the bottom.



Tugas Praktikum

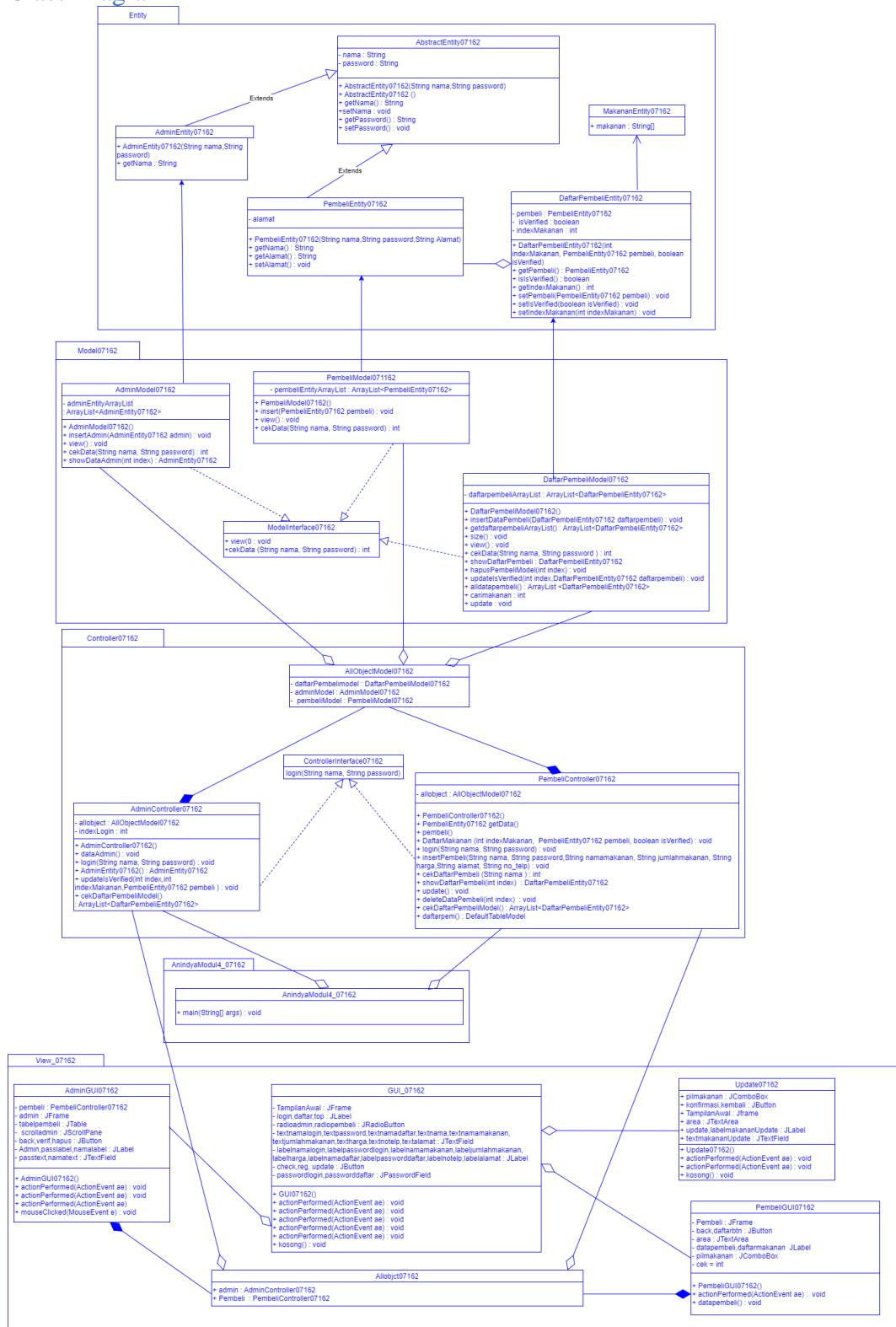
ScreenShoot





Tugas Praktikum

Class Diagram





KRITIK & SARAN

LABORATORIUM REKAYASA PERANGKAT LUNAK
FAKULTAS TEKNIK ELEKTRO DAN
TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI ADHI TAMA
SURABAYA
2020



Kritik & Saran

Kritik dan saran untuk laboratorium RPL :

Kritik: Saya bingung mau kritik apa. Karena praktikum kali ini tidak bisa langsung di lama. Jadi semoga kesempatan nanti bisa masuk dan merasakan praktikum langsung di Laboratorium RPL

Saran: Mungkin untuk pembayaran bisa langsung online tanpa harus menitipkan kepada teman. Mungkin itu saja Selebihnya best. Terimakasih



Kritik & Saran

Kritik dan Saran untuk Asisten Lab:

- Eric Wahyu Amiruddin

Kritik: Tidak ada yang dikritik - Orangnya cepat ngasih

Solusi permasalahan, sabar dan friendly

Saran: Selalu sabar menghadapi praktikan dan tetap friendly.

- Khisy Al Ghafari

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.

- Achmad Fadlan Bahrul Sy

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa.

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.

- Michael Araona Willy

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa.

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.

- Orla Windy Astuti Halimahing

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa.

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.



Kritik & Saran

- Fernanda Putra Aditya

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa.

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.

- Fitria Risqina

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa.

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.

- Sita Fara Yunanda

Kritik: Saya tidak pernah berinteraksi . Jadi tidak tau mau kritik apa.

Saran: Semoga ada kesempatan bertemu dan berbagi ilmu.