



TUGAS PERTEMUAN KE - 3

NAMA	Tesalonika Dua Nurak
NIM	22180015
KELAS	A
PEMBERI TUGAS	Raden Adityo Priyo Harjuno (2218093)

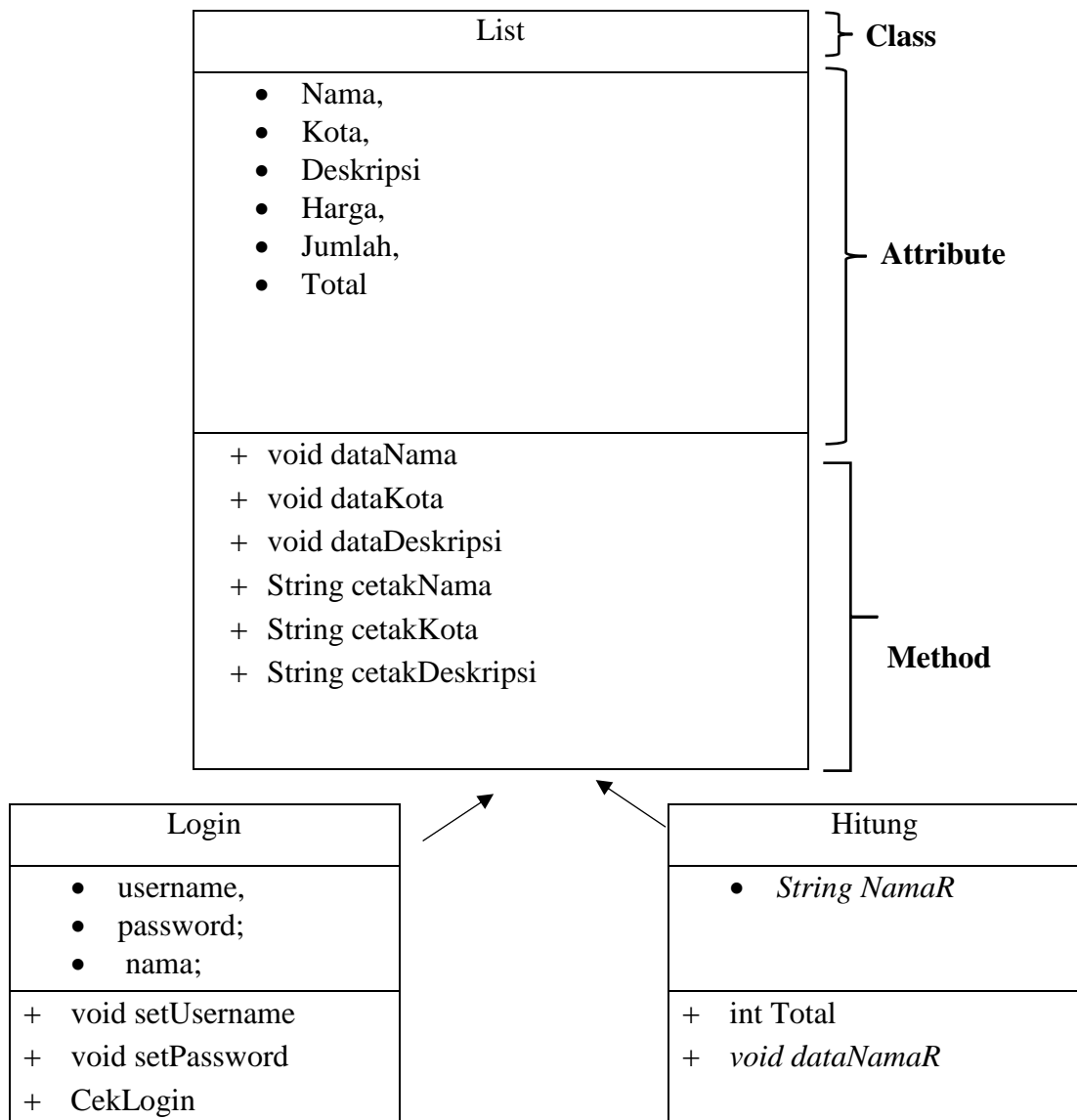
(PRAKTIKUM OOP 2023-2024)

3.1 Tugas Rumah 3 :

Judul : Sistem Informasi Wisata

Tugas : Membuat 2 Class Tambahan ,GUI, dan Menerapkan Inheritance

Class Diagram :





Source Code Object Class/Abstract (List) :

```
public class List {  
    String Nama, Kota, Deskripsi;  
    int Harga, Jumlah, Total;  
  
    void dataNama(String Nama){  
        this.Nama = Nama;  
    }  
    void dataKota(String Kota){  
        this.Kota = Kota;  
    }  
    void dataDeskripsi(String Deskripsi){  
        this.Deskripsi = Deskripsi;  
    }  
}
```



Desain *from* (GUI_Wisata.java) :

Gambar 3.1 Desain GUI_Wisata.java

Tabel 3.1 Properti Desain GUI_Wisata.java

No	Nama Komponen	Properti	Value
1	jLabel1	Text	Wisata
2	jLabel2	Text	Nama
3	jLabel3	Text	Kota
4	jLabel4	Text	Deskripsi
5	jTextField1	Name	txtNama
		Text	“ “
6	jTextField2	Name	txtKota
		Text	“ “
7	jTextField3	Name	txtDes
		Text	“ “
8	jButton1	Name	Add
		Text	btnData
9	jScrollPane1	Name	memoDta
		Text	

Source Code button Add :

```
private void btnDataActionPerformed(java.awt.event.ActionEvent  
evt) {  
    // TODO add your handling code here:
```



```
memoData.setText("");  
//    Wisata wst = new Wisata();  
List wst = new List();  
wst.dataNama(txtNama.getText());  
wst.dataKota(txtKota.getText());  
wst.dataDeskripsi(txtDes.getText());  
  
memoData.append("List Wisata\n");  
memoData.append("-----\n");  
memoData.append(" Nama : " + wst.Nama + "\n");  
memoData.append(" Kota: " + wst.Kota + "\n");  
memoData.append(" Deskripsi: " + wst.Deskripsi + "\n");  
  
Reservasi r = new Reservasi();  
r.setVisible(true);  
}
```

Hasil Tampilan :



Gambar 3.2 Tampilan GUI

Source Code Object Class/Abstract (Login.java) :

```
private void btnLoginActionPerformed(java.awt.event.ActionEvent  
evt) {  
    // TODO add your handling code here:  
    Login l = new Login();  
    String username = txtNama.getText();  
    String password = txtPass.getText();  
    boolean Authenticated = l.CekLogin(username, password);  
    if (Authenticated)  
    {  
        JOptionPane.showMessageDialog(rootPane, "LOGIN
```



```
BERHASIL, "+l.nama + "!");
    Wisata w = new Wisata();
    w.setVisible(true);
    this.dispose();
}else
{
    JOptionPane.showMessageDialog(rootPane, "LOGIN
GAGAL. Silahkan periksa kembali username dan password Anda.");
}
}
```

Desain form (Login.java) :

Gambar 3.3 Desain CostumerData.java

Tabel 3.2 Properti Desain Login.java

No	Nama Komponen	Properti	Value
1	jLabel1	Text	Login
2	jLabel2	Text	Nama :
3	jLabel3	Text	Password
4	jTextField1	Name	txtnama
		Text	-
5	jTextField2	Name	txtPass
		Text	-
6	JButton1	Name	btnLogin
		Text	Login

Source code Button/combobox btnLogin :

```
private void
btnLoginActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Login l = new Login();
    String username = txtNama.getText();
}
```



```
String password = txtPass.getText();
boolean    Authenticated    =    l.CekLogin(username,
password);
    if (Authenticated)
    {
        JOptionPane.showMessageDialog(rootPane,    "LOGIN
BERHASIL, "+l.nama + "!");
        Wisata w = new Wisata();
        w.setVisible(true);
        this.dispose();
    }else
    {
        JOptionPane.showMessageDialog(rootPane,    "LOGIN
GAGAL. Silahkan    periksa    kembali    username    dan    password
Anda.");
    }
}
```

Hasil Tampilan :



Gambar 3.4 Hasil Tampilan Login.java

Analisa :

Program diatas merupakan tampilan gui wisata dari class list,pada bagian ini akan tampil nama,kota dan deskripsi.pada tampilan gui kedua tampilan login.pada login ada nama dan password.

Source Code Object Class/Abstract (Hitung.java) :

```
public class Hitung extends List{
    String NamaR;

    void dataNamaR(String NamaR)
    {
        this.NamaR = NamaR;
    }

    public Hitung(){
        this.Harga = 25000;
        this.Jumlah = Jumlah;
        this.Total = Total;
    }
    public int Total()
    {
```



```
Total = (Harga * Jumlah);  
return Total;  
}  
}
```

Desain *form* (GUI_Reservasi.java) :

Gambar 3.5 Desain GUI_Reservasi.java

Tabel 3.3 Properti Desain GUI_Reservasi.java

No	Nama Komponen	Properti	Value
1	jLabel1	Text	Reservasi
2	jLabel2	Text	Nama
3	jLabel3	Text	Harga Tiket
4	jLabel4	Text	Jumlah Orang
7	jTextField1	Name	txtR
		Text	-
8	jTextField2	Name	txtHarga
		Text	-
9	jTextField3	Name	txtJumlah
		Text	-
12	JButton1	Name	btnHitung
		Text	Hitung
14	jScrollPane1	Name	memo
		Text	-



Source code Button/combobox btnHitung :

```
private void  
btnDataActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    memoData.setText("");  
    //    Wisata wst = new Wisata();  
    List wst = new List();  
    wst.dataNama(txtNama.getText());  
    wst.dataKota(txtKota.getText());  
    wst.dataDeskripsi(txtDes.getText());  
  
    memoData.append("List Wisata\n");  
    memoData.append("-----\n");  
    memoData.append(" Nama : " + wst.Nama + "\n");  
    memoData.append(" Kota: " + wst.Kota + "\n");  
    memoData.append(" Deskripsi: " + wst.Deskripsi +  
    "\n");  
  
    Reservasi r = new Reservasi();  
    r.setVisible(true);  
}
```

Hasil Tampilan :

RESERVASI

Nama Tesa

Harga Tiket 25000

Jumlah Orang 5

Hitung

Reservasi
Nama : Tesa
Jumlah Orang : 5
Total :125000

Gambar 3.6 Hasil Tampilan GUI_Reservasi.java

Analisa :

Program diatas merupakan implementasi dari class hitung yang di hubungkan gui reservasi.Tampil yang amuncul nama pemesan,harga tiket dan jumlah orang.Setelah itu akan muncul di memo sesuai apa yang di inputkan.



3.2 Kesimpulan

1. Inheritance atau Pewarisan/Penurunan adalah konsep pemrograman dimana sebuah class dapat ‘menurunkan’ properti dan method yang dimilikinya kepada class lain.
2. Konsep inheritance digunakan untuk memanfaatkan fitur ‘code reuse ’ untuk menghindari duplikasi kode program. Fungsi dari inheritance memperluas fungsi dari parent class.
3. Pada class yang memiliki class turunan disebut dengan parent class atau base class, sedangkan class turunan itu sendiri kerap disebut dengan subclass atau child class yang bisa menurunkan atau mewariskan apa pun dari parent class.