



Bab 4

Enkapsulasi

Pengertian Enkapsulasi



Enkapsulasi merupakan proses **pemaketan** objek beserta methodnya untuk **menyembunyikan** rincian **implementasi** dari pemakai/objek lainnya. Enkapsulasi juga dikenal dengan istilah '*information hiding*'. Dengan enkapsulasi, maka programmer akan **dibatasi** dalam mengakses suatu atribut dan method yang dimiliki oleh suatu class. Dengan menghalangi kode program lain untuk mengubah property tertentu, class menjadi lebih terintegrasi, dan menghindari kesalahan ketika seseorang '**mencoba**' mengubahnya. Salah satu implementasi dari enkapsulasi adalah adanya **setter** dan **getter** untuk suatu atribut dalam suatu kelas. Method **set** digunakan untuk **menetapkan nilai** pada atribut sedangkan method **get** **mengembalikan nilai** dari variable yang sudah kita set sebelumnya.

.

Jenis Enkapsulasi



a. Public

Dengan mendeklarasikan data(atribut) dan method dengan tingkat akses public, maka data dan method dapat diakses semua kelas yang ada di dalam program, baik yang merupakan kelas turunan maupun kelas yang tidak mempunyai hubungan sama sekali.

```
public String atribut1; //pada atribut  
public int Luas() {...} // pada methode
```

Jenis Enkapsulasi



b. Private

Dengan mendeklarasikan data(atribut) dan method menggunakan tingkat akses private, maka data dan method tersebut hanya dapat diakses oleh kelas tersebut. Sehingga data dan method tersebut tidak dapat diakses oleh kelas lain.

```
private String atribut1; //pada atribut  
private int Luas() {...} // pada methode
```

Jenis Enkapsulasi



c. Protected

Dengan mendeklarasikan data(atribut) dan method menggunakan tingkat akses protected, maka data dan method tersebut hanya dapat diakses oleh kelas yang memilikinya dan kelas-kelas yang masih memiliki hubungan turunan.

```
protected String atribut1; //pada atribut  
protected int Luas() {...} // pada methode
```

Jenis Enkapsulasi



d. Default (tidak ada modifier)

Untuk hak akses default ini, sebenarnya hanya ditujukan untuk class yang ada dalam satu paket, atau istilahnya hak akses yang berlaku untuk satu folder saja (tidak berlaku untuk class yang tidak satu folder/package)

```
String atribut1; //pada atribut  
int Luas() {...} // pada methode
```

Tabel symbol Enkapsulasi



<u>Hak Akses</u>	Symbol
Public	+
Protected	#
Private	-
Default	~

Tabel hak akses dari tiap jenis Enkapsulasi:



<u>Hak Akses</u>	Public	Protected	Default	Private
Dari <u>kelas yang sama</u>	✓	✓	✓	✓
Dari <u>kelas lain didalam paket</u>	✓	✓	✓	✗
Dari <u>kelas lain diluar paket</u>	✓	✗	✗	✗
Dari <u>kelas turunan didalam paket</u>	✓	✓	✓	✗
Dari <u>kelas turunan diluar paket</u>	✓	✓	✗	✗

Implementasi 1



```
11 public class pembayaranGaji {
12     private String nomorRekening;
13     private String input;
14     private double saldo;
15     public pembayaranGaji() {
16         this.nomorRekening = "G12345";
17         this.saldo = 1000000;
18     }
19     public String getNomorRekening() {
20         return nomorRekening;
21     }
22     public void setNomorRekening(String nomorRekening) {
23         this.nomorRekening = nomorRekening;
24     }
25     public double getSaldo() {
26         return saldo;
27     }
28     public void setSaldo(double saldo) {
29         this.saldo = saldo;
30     }
31     public String getInput() {
32         return input;
33     }
34     public void setInput(String input) {
35         this.input = input;
36     }
37     public double cekSaldo() {
38         if (getNomorRekening().equals(anObject:getInput())) {
39             return getSaldo();
40         } else {
41             return 0;
42         }
43     }
44 }
45 }
```

Implementasi



```
11 public class mainGaji {  
12     public static void main(String[] args) {  
13         pembayaranGaji gaji = new pembayaranGaji();  
14         gaji.setInput( input: "G123456");  
15         System.out.println("Rekening yang anda masukkan : "+gaji.getInput());  
16         System.out.println("Jumlah saldo anda : "+gaji.cekSaldo());  
17     }  
18 }  
19 }  
20 }
```

Hasil :

A screenshot of an IDE's output window. The window has a title bar with red, yellow, and green window control buttons. The title is "Output - PraktikumOOP_2118112 (run) x". On the left side of the output area, there are four icons: a green double arrow, a yellow double arrow, a red square, and a gear icon. The output text is as follows:

```
run:  
Rekening yang anda masukkan : G12345  
Jumlah saldo anda : 1000000.0  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Implementasi 2



```
11 public class Login {
12
13     private String username, password;
14     public String nama;
15
16     public Login() {
17         nama = "Firman Frezy Pradana";
18         username = "Firman";
19         password = "Firman123";
20     }
21
22     public String getUsername() {
23         return username;
24     }
25
26     public void setUsername(String username) {
27         this.username = username;
28     }
29
30     public String getPassword() {
31         return password;
32     }
33
34     public void setPassword(String password) {
35         this.password = password;
36     }
37
38     //
39     boolean CekLogin(String Username , String password){
40         if(username.equals( anObject:getUsername()) && password.equals( anObject:getPassword())){
41             return true;
42         }
43         return false;
44     }
```

Implementasi 2



LOGIN

Username

Password

Login

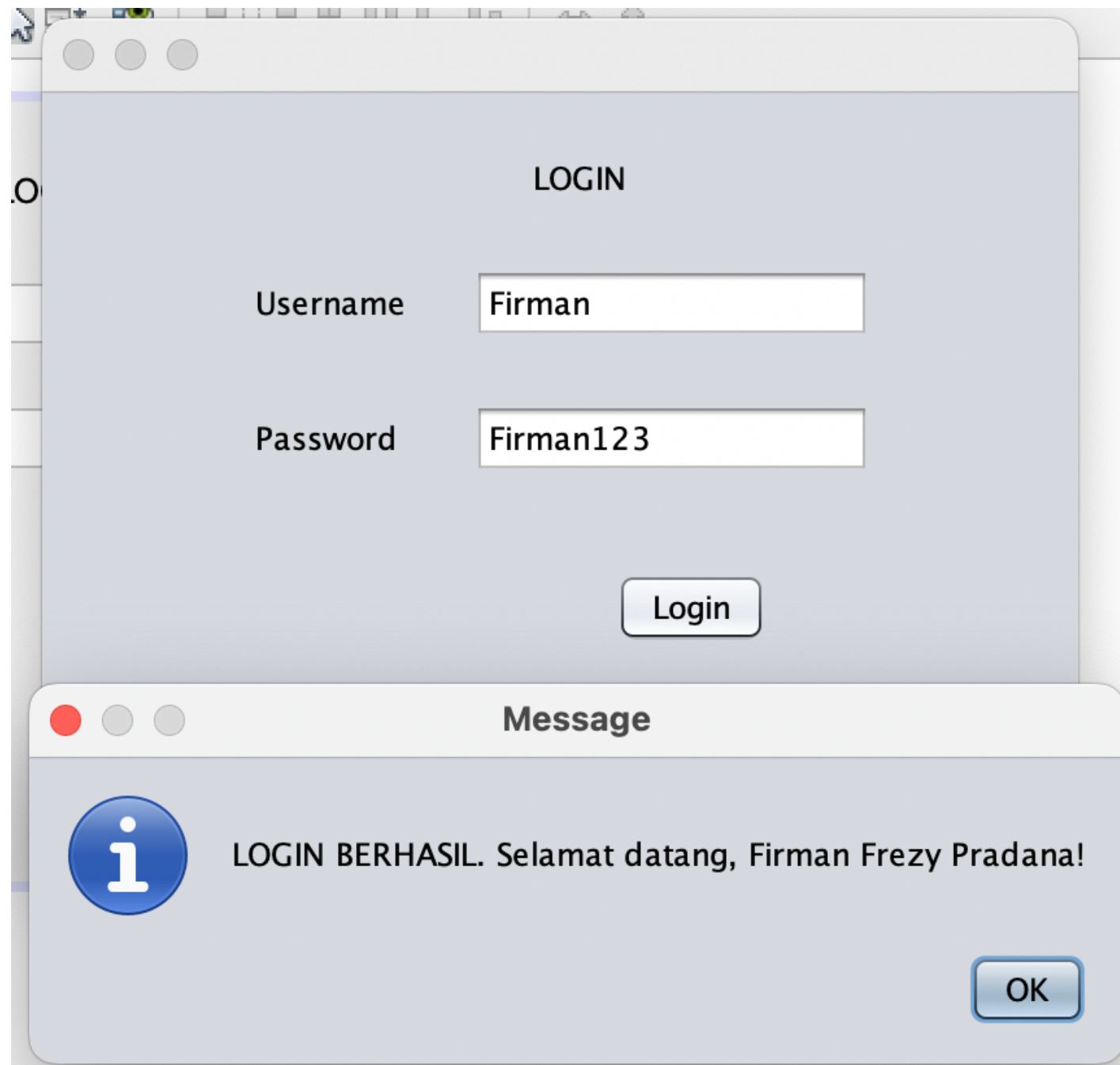
No	Objek	Properti	Nilai
1	jLabel1	Text	LOGIN
2	jLabel2	Text	USERNAME
3	jLabel3	Text	password
4	Button	Name	btn_login
5	JTextField	Name	txtUsername
		Text	" "
6	JTextField	Name	txtPassword
		Text	""

Implementasi



```
99 private void btn_loginActionPerformed(java.awt.event.ActionEvent evt) {  
100     // TODO add your handling code here:  
101     Login login = new Login();  
102     String username = txtUsername.getText();  
103     String password = txtPassword.getText();  
104     boolean isAuthenticated = login.CekLogin(Username:username, password);  
105     if (isAuthenticated) {  
106         JOptionPane.showMessageDialog(parentComponent:rootPane, "LOGIN BERHASIL. Selamat datang, " + login.nama + "!");  
107     } else {  
108         JOptionPane.showMessageDialog(parentComponent:rootPane, message:"Login Gagal . Silakan periksa kembali username dan password Anda.");  
109     }  
}
```


Hasil Implementasi





"Open NetBeans, and let's write code Java"

