

LAPORAN TUGAS ALGORITMA PEMROGRAMAN  
SISTEM REGISTRASI KREDENSIAL PENGGUNA MENGGUNAKAN  
OBJECT STRING PADA JAVA



Oleh: Afif Naufal Zahran

NIM: 2511533009

DOSEN PENGAMPU: DR.WAHYUDI, S.T, M.T

FAKULTAS TEKNOLOGI INFORMASI  
DEPARTEMEN INFORMATIKA  
UNIVERSITAS ANDALAS

# DAFTAR ISI

BAB I PENDAHULUAN.....	
1.1 Latar Belakang .....	
1.2 Tujuan .....	
1.3 Manfaat Praktikum.....	
BAB II PEMBAHASAN .....	
2,1 Pengertian Object dan String .....	
2,2 Penjelasan Program.....	
2,3 FlowChart, Pseudocode dan kode java .....	
BAB III KESIMPULAN.....	
DAFTAR PUSTAKA .....	

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

*String* adalah tipe data dalam bahasa pemrograman yang digunakan untuk merepresentasikan teks, yakni serangkaian karakter seperti huruf, angka, simbol, dan spasi. *String* sendiri merupakan urutan karakter, bisa berupa kata, kalimat atau bahkan urutan karakter acak seperti “8923 asd9asd9298 \*@#”. Di java sendiri *String* tidak termasuk ke tipe data primitif, melainkan sebuah *class* yang disediakan oleh *Java* yang dapat dideklarasikan dengan (`String s = “halo dunia”`). *String Class* sendiri menyediakan *method* yang dapat digunakan untuk memanipulasi isi dari *string*.

Tidak hanya itu, kali ini kita akan mempelajari sedikit dasar OOP(*Object-Oriented Programming*) dimana kita membuat suatu *class* yang memiliki *attribute*, *getter method*, *setter method*, dan *helper method*. Untuk memanggil object yang kita buat adalah (`ClassObjectKita namaVar = new ClassObjectKita()`) setelah itu kita dapat mengakses fungsi pada class tersebut.

### 1.2 Tujuan

Tujuan dari laporan tugas praktikum ini agar pembaca dan penulis dapat mendalami dan memahami tentang Object dan String dan implementasinya di bahasa pemograman *Java*.

### 1.3 Manfaat

Manfaat dari tugas praktikum tentang perulangan agar pembaca mendapatkan pengetahuan tentang OOP(*Object-Oriented Programming*) dan String. Manfaat untuk penulis agar dapat memahami OOP(*Object-Oriented Programming*) dan String pada bahasa pemograman *Java*.

## BAB II

### PEMBAHASAN

#### 2,1 Pengertian Object Dan String

*String* adalah tipe data dalam bahasa pemrograman yang digunakan untuk merepresentasikan teks, yakni serangkaian karakter seperti huruf, angka, simbol, dan spasi. *String* sendiri merupakan urutan karakter, bisa berupa kata, kalimat atau bahkan urutan karakter acak seperti “8923 asd9asd9298 \*@#”. Di java sendiri *String* tidak termasuk ke tipe data primitif, melainkan sebuah *class* yang disediakan oleh *Java* yang dapat dideklarasikan dengan (`String s = “halo dunia”`). *String* memiliki beberapa method seperti “`.toUpperCase()`” untuk memanipulasi huruf menjadi kapital dan “`.toLowerCase()`” kebalikan dari method “`.toUpperCase()`”, “`.concat()`” untuk menggabungkan suatu string ke string yang memanggil fungsi *concat*.

*Class* merupakan *blueprint* dari sebuah objek atau cetakan untuk membuat objek. *Class* akan merepresentasikan objek yang mau dibuat. *Class* berisi dari beberapa kumpulan definisi variabel dan fungsi yang menggambarkan sebuah objek. Kita dapat menambahkan atribut, method pada *class* yang kita buat. Untuk membuat *object* pada java kita dapat melakukannya dengan (`ClassObjectKita namaVar = new ClassObjectKita()`) setelah itu kita dapat mengakses fungsi pada class tersebut. Atribut sendiri bisa juga disebut sebagai property dari sebuah *class*.

## 2.2 Penjelasan Program

Pada program kali ini akan terdapat 2 *class* yang dibuat yakni *main class* dan *class* untuk *object* yakni *tugasAlproPekan7\_2511533009.java* sebagai *main class* dan *Akun\_2511533009.java* sebagai *class* untuk *object*. Pada *object class* tersebut terdapat atribut seperti *username*, *email*, *password*, dan *pinAngka*. Untuk memanipulasi atribut tersebut kita akan membuat fungsi *Setter*. Sedangkan untuk mengambil dari dari attribute tersebut kita akan membuat fungsi *Getter*. Lalu membuat fungsi pembantu untuk melakukan pengecekan validasi password dan email.

Untuk *helper method* terdapat dua pengecekan yakni jika email valid “*isEmailValid()*” dan jika password valid “*isPasswordValid()*” seperti pada ambar 2.1 dan 2.2 di bawah.

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: afif2511533009
Masukkan Password: 2513
Masukkan Email: afif_2511533009@nasgo.enak
Masukkan PIN (6 digit): 1123
--- REGISTRASI GAGAL ---
Email Anda "afif2511533009" tidak valid (panjang password minimal 8 karakter).
Silakan coba lagi.
```

Gambar 2.1

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: afif2511533009
Masukkan Password: admin123kominform
Masukkan Email: afif2511533009
Masukkan PIN (6 digit): 123456
|--- REGISTRASI GAGAL ---
Email Anda "afif2511533009" tidak valid (harus mengandung '@' dan '.').
Silakan coba lagi.
```

Gambar 2.2

Jika email dan password valid maka output yang didapatkan akan menjadi seperti pada gambar 2.3

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: afif2511533009
Masukkan Password: adminkominfo123
Masukkan Email: afif2511533009@lapangan.kerja
Masukkan PIN (6 digit): 123456
--- REGISTRASI BERHASIL ---
Akun untuk "afif2511533009" telah berhasil dibuat

--- Detail Akun ---
Username (Lowecase): afif2511533009
Email (Uppercase): AFIF2511533009@LAPANGAN.KERJA
ID Pengguna (Gabungan): afif2511533009123456

--- uji Tipe Data (PIN Anda: 123456) ---
PIN (int) + 10      = 123466
PIN (String) + "10" =12345610
```

Gambar 2.3

Pada program juga terdapat pengujian data seperti operasi logaritma, penggabungan string dan penggunaan fungsi string seperti `toLowerCase()`, `toUpperCase()` dan `concat()`.

## 2,3 Flowchart, Pseudocode Dan Kode Java

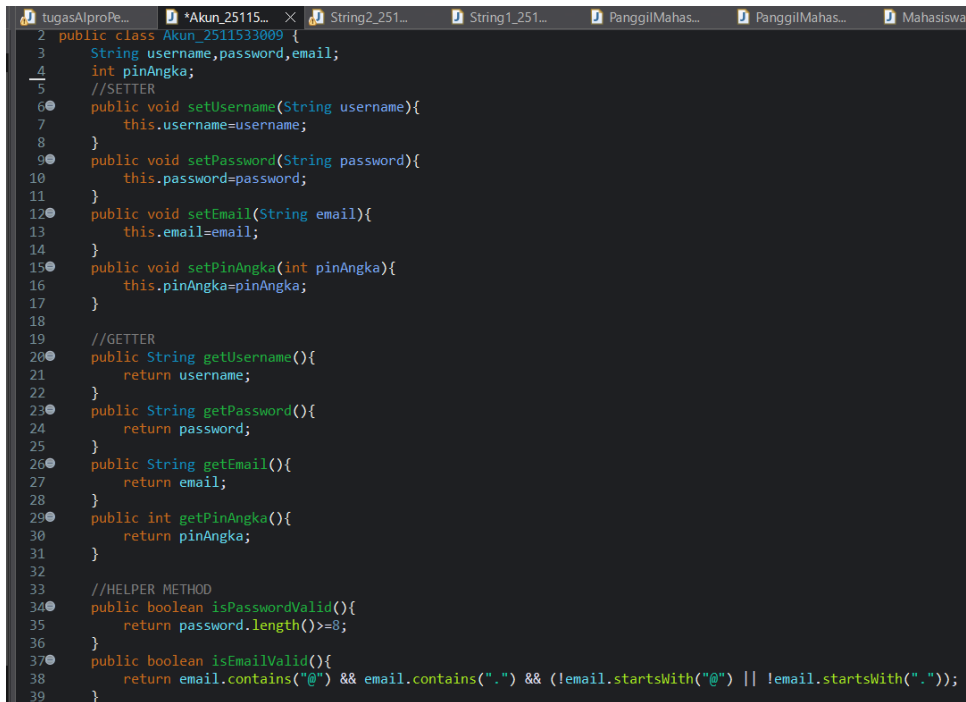
Berikut kode java untuk tugas kali ini

```
3 public class tugasAlproPekan7_2511533009 {
4
5     public static void main(String[] args) {
6         String username,email,password;
7         int pinAngka;
8         Akun_2511533009 akun = new Akun_2511533009();
9         Scanner input = new Scanner(System.in);
10        System.out.println("==== REGISTRASI AKUN BARU =====");
11        System.out.print("Masukkan Username: ");
12        username = input.nextLine();
13        akun.setUsername(username);
14        System.out.print("Masukkan Password: ");
15        password = input.nextLine();
16        akun.setPassword(password);
17        System.out.print("Masukkan Email: ");
18        email = input.nextLine();
19        akun.setEmail(email);
20        System.out.print("Masukkan PIN (6 digit): ");
21        pinAngka = input.nextInt();
22        akun.setPinAngka(pinAngka);
23        if (akun.isPasswordValid()){
24            if (akun.isEmailValid()){
25                System.out.println("--- REGISTRASI BERHASIL ---");
26                System.out.println("Akun untuk \"" + akun.getUsername() + "\" telah berhasil dibuat");
27
28                System.out.println("\n--- Detail Akun ---");
29                System.out.println("Username (Lowercase): " + akun.getUsername().toLowerCase());
30
31                System.out.println("Email (Uppercase): " + akun.getEmail().toUpperCase());
32
33                String concat1 = akun.getUsername() + akun.getPinAngka();
34                System.out.println("ID Pengguna (Gabungan): " + concat1);
35
36                System.out.println("\n--- uji Tipe Data (PIN Anda: " + akun.getPinAngka() + ") ---");
37
38                System.out.println("\n--- uji Tipe Data (PIN Anda: " + akun.getPinAngka() + ") ---");
39
40                int penjumlahan = akun.getPinAngka() + 10;
41                System.out.println("PIN (int) + 10      = " + penjumlahan);
42
43                String concat2 = akun.getPinAngka() + "10";
44                System.out.println("PIN (String) + \"10\" = " + concat2);
45            }else {
46                System.out.println("--- REGISTRASI GAGAL ---");
47                System.out.println("Email Anda \"" + akun.getUsername() + "\" tidak valid (harus mengandung '@' dan '.').");
48                System.out.println("Silakan coba lagi.");
49            }
50        }else {
51            System.out.println("--- REGISTRASI GAGAL ---");
52            System.out.println("Email Anda \"" + akun.getUsername() + "\" tidak valid (panjang password minimal 8 karakter).");
53            System.out.println("Silakan coba lagi.");
54        }
55    }
56 }
```

Gambar 2.4

Pada program utama kita akan menginisiasikan 4 variabel dimana variabel *username*, *email*, dan *password* ber-tipe data string dan *pinAngka* bertipe data integer. Lalu kita membuat objek dari *object* class yang telah kita buat. Setelah itu, membuat objek *Scanner* agar dapat menerima input dari pengguna. Setelah dilakukan pengecekan apakah password dan email valid maka akan muncul *output* “REGISTRASI BERHASIL”.

Untuk *object class* sendiri seperti pada gambar 2.5 dibawah



```
2 public class Akun_2511533009 {
3     String username,password,email;
4     int pinAngka;
5     //SETTER
6     public void setUsername(String username){
7         this.username=username;
8     }
9     public void setPassword(String password){
10        this.password=password;
11    }
12    public void setEmail(String email){
13        this.email=email;
14    }
15    public void setPinAngka(int pinAngka){
16        this.pinAngka=pinAngka;
17    }
18
19    //GETTER
20    public String getUsername(){
21        return username;
22    }
23    public String getPassword(){
24        return password;
25    }
26    public String getEmail(){
27        return email;
28    }
29    public int getPinAngka(){
30        return pinAngka;
31    }
32
33    //HELPER METHOD
34    public boolean isPasswordValid(){
35        return password.length()>=8;
36    }
37    public boolean isValidEmail(){
38        return email.contains("@") && email.contains(".") && (!email.startsWith("@") || !email.startsWith("."));
39    }
}
```

Gambar 2.5

Terdapat 4 atribut untuk *object class* ini yakni *username*, *password*, *email*, dan *pinAngka* dan terdapat 4 fungsi *getter*, 4 fungsi *setter*, dan 2 fungsi pembantu. Fungsi *setter* berfungsi untuk memanipulasi nilai dari atribut dan fungsi *getter* berfungsi untuk mengambil nilai dari attribute tersebut.

- a. Flowchart
- b. Pseudocode



## **BAB III**

### **KESIMPULAN**

Kesimpulan dari praktikum kali ini adalah perulangan pada bahasa pemrograman *Java* begitu penting karena dapat digunakan untuk melakukan *looping* pada data.

## **DAFTAR PUSTAKA**