

**TUGAS PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
TUGAS PEKAN 7
REGISTRASI KREDENSIAL PENGGUNA**



Disusun oleh:

Ghinada Fathanawafa Algma
2511533008

Kelas B Informatika

Dosen Pengampu:
Dr. Wahyudi S.T. M.T

Asisten Praktikum:
Rahmad Dwirizki Olders

**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
2025**

1. Soal

Sistem Registrasi Kredensial Pengguna

Buatlah program untuk mensimulasikan proses registrasi akun baru. Program harus dapat mengambil input pengguna, menyimpannya dalam sebuah object, dan melakukan validasi data berdasarkan aturan yang ditentukan sebelum menampilkan hasilnya.

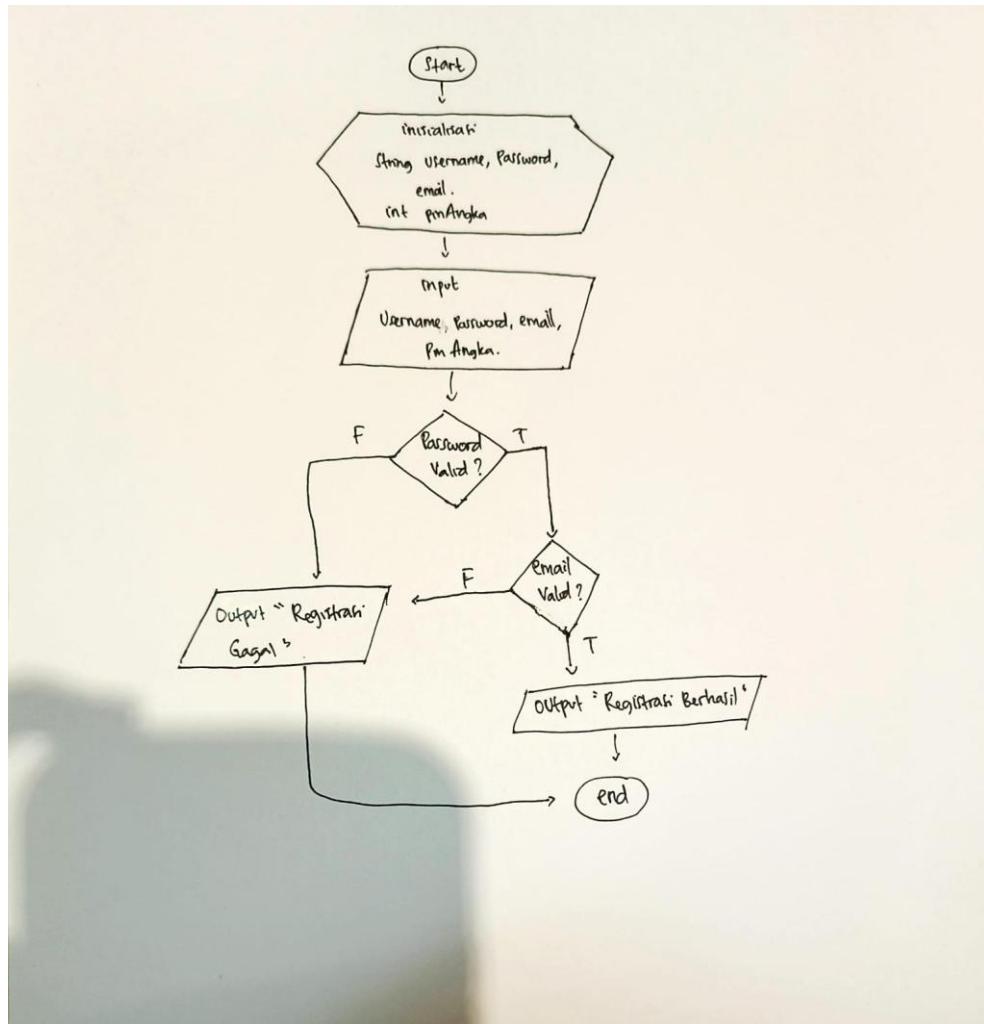
2. Pseudocode

Judul Registrasi Kredensial Pengguna
Deklarasi Scan : Scanner akun : method username, password, email : String pinAngka, pinInt, pinPlus : Integer pinString, pinStringPlus : String passwordValid, emailValid : Boolean
Pseudocode <ol style="list-style-type: none">1. Creat akun; method2. Print “===== REGISTRASI AKUN BARU =====”3. Print “Masukkan Username: ”4. Read username5. akun.setUsername(username)6. Print “Masukkan Password: ”7. Read password8. akun.setPassword(password)9. Print “Masukkan Email: ”10. Read email11. akun.setEmail(email)

12. Print “Masukkan PIN Angka (misal 6 digit): ”
13. Read pinAngka
14. akun.setPinAngka(pinAngka)
15. pinInt \leftarrow akun.getPinAngka()
16. pinString \leftarrow konversi_string(pinInt)
17. pinPlus \leftarrow pinInt + 10
18. pinStringPlus \leftarrow pinString + “10”
19. passwordValid \leftarrow akun.isPasswordValid()
20. emailValid \leftarrow akun.isEmailValid()
21. If passwordValid = False atau emailValid = false then
22. Print “Registrasi Gagal”
23. If passwordValid = false then
24. Print “Error: Password harus memiliki Panjang minimal 8 karakter.”
25. Print “Silahkan coba lagi.”
26. End if
27. If emailValid = false then
28. Print “Error: Email harus mengandung karakter ‘@’ dan ‘.’”
29. Print “Silahkan coba lagi.”
30. End if
31. Else
32. Print “---- REGISTRASI BERHASIL ----”
33. Print “akun untuk ” + username + “ telah berhasil dibuat.”
34. Print “---- Detail Akun ----”
35. Print “Username: ” + lowercase(akun.getUsername())
36. Print “Password: ” + repeat(“*”, Panjang(password)) + “(Panjang ” + panjang(password) + “)”
37. Print “Email: ” + uppercase(akun.getEmail())

38. Print "ID Pengguna: " + lowercase(akun.getUsername()) + akun.getPinAngka()
39. Print "---- Uji Tipe Data (PIN Anda: " + get.pinAngka() + ")" ----"
40. Print "PIN (int) + 10 = " + pinPlus
41. Print "PIN (String) + \"10\" = " + pinStringPlus
42. End if
43. End

3. Flowchart



Gambar 2.1 flowchart

4. Kode program

4.1 class akun

```
1 package pekan7_2511533008;
2
3 public class akun {
4     private String username;
5     private String password;
6     private String email;
7     private int pinAngka;
8
9     public void setUsername(String username) {
10         this.username = username;
11     }
12
13     public void setPassword(String password) {
14         this.password = password;
15     }
16
17     public void setEmail(String email) {
18         this.email = email;
19     }
20
21     public void setPinAngka(int pinAngka) {
22         this.pinAngka = pinAngka;
23     }
24
25     public String getUsername() {
26         return username;
27     }
28
29     public String getPassword() {
30         return password;
31     }
32
33     public String getEmail() {
34         return email;
35     }
```

Kode program 3.1 kode akun 1

```
37     public int getPinAngka() {
38         return pinAngka;
39     }
40
41     public boolean isPasswordValid() {
42         return password.length() >= 8;
43     }
44
45     public boolean isEmailValid() {
46         return email.contains "@" && email.contains ".";
47     }
48 }
49
```

Kode program 3.2 kode akun 2

Pada *class* akun itu berisikan method method yang nantinya akan dipanggil pada *class* tugasAlproPekan7_2511533008.

4.2 *class* utama

```
1 package pekan7_2511533008;
2
3 import java.util.Scanner;
4
5 public class tugasAlproPekan7_2511533008 {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9
10        akun akun = new akun();
11
12        System.out.println("===== REGISTRASI AKUN BARU =====");
13
14        System.out.print("Masukkan Username: ");
15        String username = scan.nextLine();
16        akun.setUsername(username);
17
18        System.out.print("Masukkan Password: ");
19        String password = scan.nextLine();
20        akun.setPassword(password);
21
22        System.out.print("Masukkan Email: ");
23        String email = scan.nextLine();
24        akun.setEmail(email);
25
26        System.out.print("Masukkan PIN Angka (misal 6 digit): ");
27        int pinAngka = scan.nextInt();
28        akun.setPinAngka(pinAngka);
29
30        System.out.println("");
31
32        int pinInt = akun.getPinAngka();
33        String pinString = String.valueOf(pinInt);
34
35        int pinPlus = pinInt + 10;
36        String pinStringPlus = pinString + "10";
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
```

Kode program 3.3 kode tugasAlproPekan7 1

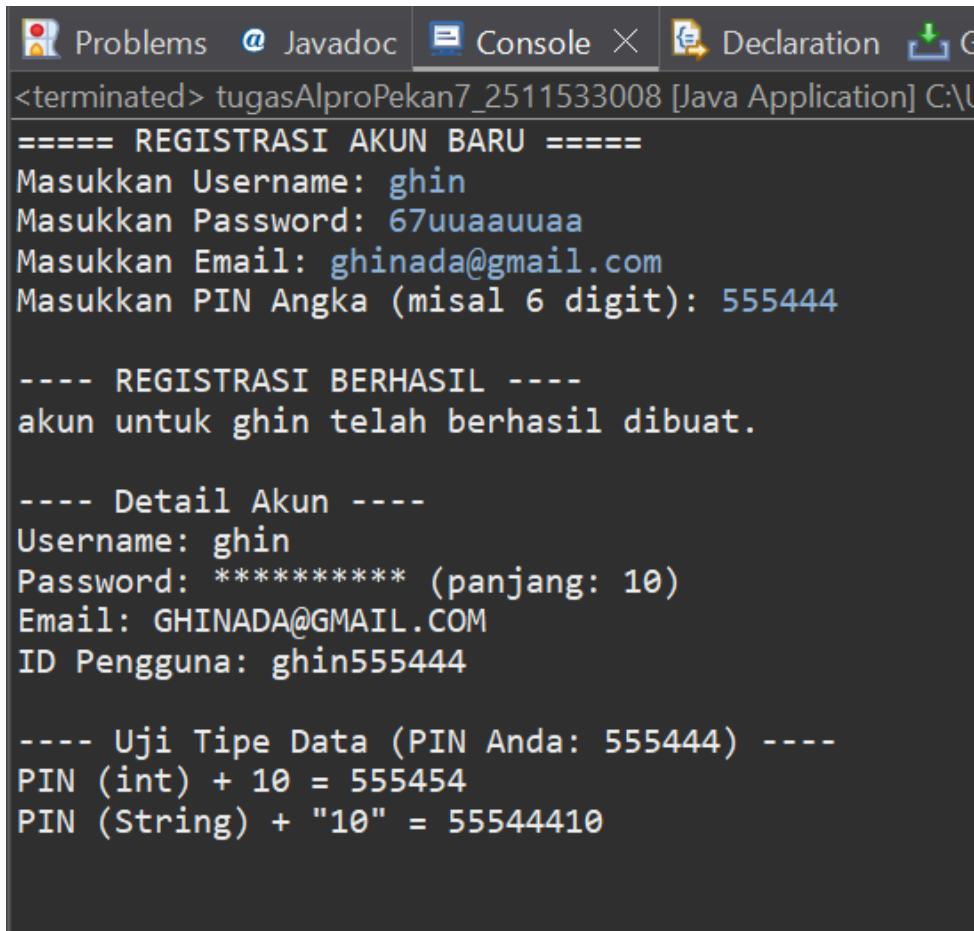
```
37
38    boolean passwordValid = akun.isPasswordValid();
39    boolean emailValid = akun.isEmailValid();
40
41    if (!passwordValid || !emailValid) {
42        System.out.println("Registrasi Gagal!");
43        if (!passwordValid) {
44            System.out.println("Error: Password harus memiliki panjang minimal 8 karakter.");
45            System.out.println("Silahkan coba lagi.");
46        }
47        if (!emailValid) {
48            System.out.println("Error: Email harus mengandung karakter '@' dan '.'.");
49            System.out.println("Silahkan coba lagi.");
50        }
51    } else {
52        System.out.println("===== REGISTRASI BERHASIL =====");
53        System.out.println("akun untuk " + username + " telah berhasil dibuat.");
54        System.out.println("");
55        System.out.println("===== Detail Akun =====");
56        System.out.println("Username: " + akun.getUsername().toLowerCase());
57        System.out.println("Password: " + "*".repeat(akun.getPassword().length()) + " (panjang: " + akun.getPassword().length() + ")");
58        System.out.println("Email: " + akun.getEmail().toUpperCase());
59        System.out.println("ID Pengguna: " + akun.getUsername().toLowerCase() + akun.getPinAngka());
60        System.out.println("");
61        System.out.println("===== Uji Tipe Data (PIN Anda: " + akun.getPinAngka() + ") =====");
62        System.out.println("PIN (int) + 10 = " + pinPlus);
63        System.out.println("PIN (String) + \"10\" = " + pinStringPlus);
64    }
65    scan.close();
66
67
68 }
```

Kode program 3.4 kode tugasAlproPekan7 2

Pada *class* ini kita akan menggunakan method yang telah dibuat pada *class* akun. Kita akan membuat program registrasi akun baru menggunakan string dan lainnya.

5. Output

5.1 output berhasil



```
Problems @ Javadoc Console × Declaration Git G
<terminated> tugasAlproPekan7_2511533008 [Java Application] C:\U
===== REGISTRASI AKUN BARU =====
Masukkan Username: ghin
Masukkan Password: 67uuuaauua
Masukkan Email: ghinada@gmail.com
Masukkan PIN Angka (misal 6 digit): 555444

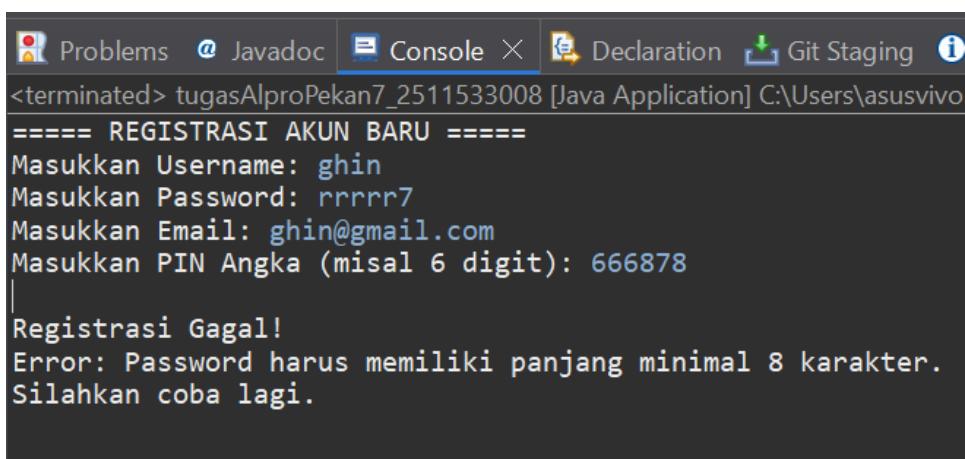
----- REGISTRASI BERHASIL -----
akun untuk ghin telah berhasil dibuat.

----- Detail Akun -----
Username: ghin
Password: ***** (panjang: 10)
Email: GHINADA@GMAIL.COM
ID Pengguna: ghin555444

----- Uji Tipe Data (PIN Anda: 555444) -----
PIN (int) + 10 = 555454
PIN (String) + "10" = 55544410
```

Gambar 4.1 output berhasil

5.2 output gagal



```
Problems @ Javadoc Console × Declaration Git Staging i
<terminated> tugasAlproPekan7_2511533008 [Java Application] C:\Users\asusvivo
===== REGISTRASI AKUN BARU =====
Masukkan Username: ghin
Masukkan Password: rrrrrr7
Masukkan Email: ghin@gmail.com
Masukkan PIN Angka (misal 6 digit): 666878

Registrasi Gagal!
Error: Password harus memiliki panjang minimal 8 karakter.
Silahkan coba lagi.
```

Gambar 4.2 output gagal

6. Penjelasan Singkat

Dari tugas ini dapat diketahui bahwa pemahaman tentang method sangat meningkatkan modularitas dan keterbacaan program karena memungkinkan pemisahan tugas ke dalam blok kode yang dapat dipanggil ulang; pemilihan antara *static* dan *instance* method menentukan cara pemanggilan dan pengelolaan state objek, sedangkan perbedaan *void* dan *return type* mempengaruhi cara data diproses dan diuji. Penggunaan parameter dan overloading menambah fleksibilitas desain API, dan pemahaman bahwa String adalah objek dengan banyak metode bawaan (seperti pengambilan substring, perbandingan isi dengan equals, serta perubahan kapitalisasi) penting untuk menghindari bug logika. Secara keseluruhan, penerapan method dan manipulasi String yang benar berkontribusi pada pembuatan program yang lebih efisien, mudah dipelihara, dan lebih tangguh terhadap kesalahan input teks.