

LAPORAN TUGAS ALGORITMA DAN PEMROGRAMAN

ALGORITMA DAN PEMROGRAMAN

TUGAS PEKAN 7

Disusun Oleh:

MUHAMMAD YASIN HABIBURRAHMAN

2511532016

Kelas-B

Dosen Pengampu:

Dr. Wahyudi. S.T.M.T



DEPARTEMEN INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS ANDALAS

TAHUN 2025

TUGAS PEKAN 7 PRAKTIKUM ALGORITMA DAN PEMROGRAMAN

Sistem Registrasi Kredensial Pengguna Penjelasan Singkat

SOAL

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.

Struktur Program:

1. Class Data: Akun.java
 - Class ini bertugas sebagai *blueprint* untuk menyimpan data akun.

-
- Atribut yang harus ada: username, password, email (semua String), dan pinAngka (int).

- Gunakan setter dan getter jika dibutuhkan untuk mengelola data privat tersebut.

- Wajib membuat *helper method* validasi berikut di dalam *class* ini:

- a. public boolean isPasswordValid(): Mengembalikan true jika panjang password 8 karakter atau lebih.

- b. public boolean isEmailValid(): Mengembalikan true jika email mengandung karakter @ dan “.”

2. Class Utama: Registrasi.java

- Class ini memiliki main method.

- Bertugas mengambil input, membuat *object* Akun, dan memanggil *setter*.

- Meminta input berikut dari pengguna:

- a. Username (String)

- b. Password (String)

- c. Email (String)

- d. PIN Angka (int, misal 6 digit)

- Melakukan pengecekan if-else dengan memanggil method isPasswordValid() dan isEmailValid() dari *object* Akun.
-

1. Pseudocode

1.1 Main Class

Judul

Program yang digunakan untuk menerima input berupa username, password, email, dan PIN dari pengguna untuk proses registrasi akun mereka ke suatu forum.

Deklarasi

```
VAR username, password, email; ← STRING  
VAR pinAngka; ← INT  
CLASS NEW Registrasi; ← Akun()
```

Algoritma

```
OUTPUT("==== REGISTRASI AKUN BARU ===")
```

```
OUTPUT("Masukkan Username: ")  
INPUT(username)
```

```
OUTPUT("Masukkan Password: ")  
INPUT(password)
```

```
OUTPUT("Masukkan Email: ")  
INPUT(email)
```

```
OUTPUT("Masukkan PIN (6 digit): ")  
INPUT(pinAngka)
```

```
REGISTRASI.setEmail ← email  
REGISTRASI.setPassword ← password  
REGISTRASI.setUsername ← username  
REGISTRASI.setPin ← pinAngka
```

```
VAR ujiString ← pinAngka + "10"  
VAR ujiPIN ← pinAngka + 10
```

```
IF(REGISTRASI.isEmailValid ← email == FALSE) THEN  
    OUTPUT("Email Anda \" " + password + "\" tidak valid (harus mengandung '@' dan '.').")  
    OUTPUT("Silahkan coba lagi.")
```

```
ELSE IF(REGISTRASI.isPasswordValid == FALSE) THEN  
    OUTPUT("Password Anda\" " + password + "\" tidak valid (harus terdiri dari 8 karakter atau lebih).")  
    OUTPUT("Silahkan coba lagi.")  
END ELSE IF
```

```
ELSE {  
    OUTPUT.LINE()  
    OUTPUT("--- REGISTRASI BERHASIL ---")  
    OUTPUT("Akun untuk \" " + username + "\" telah berhasil dibuat.")
```

```
OUTPUT.LINE()

OUTPUT("--- DETAIL AKUN ---")
OUTPUT("Username (Lowercase) : " + LOWERCASE(username))
OUTPUT("Email (Uppercase) : " + UPPERCASE(email))
OUTPUT("ID Pengguna (Gabungan) : " + (username + pinAngka))

OUTPUT.LINE()
OUTPUT("--- Uji Tipe Data (PIN Anda " + pinAngka + ") ---")
OUTPUT("PIN (int) + 10 = " + ujiPIN)
OUTPUT("PIN (String) + \"10\" = " + ujiString)
END ELSE
END IF
```

1.2 Data Class

Judul

Kumpulan variabel, fungsi, dan prosedur yang digunakan untuk menyimpan data yang akan dipakai pada kelas utama.

Deklarasi

VAR username, password, email; ← STRING
VAR pinAngka; ← INT

Algoritma

CLASS Akun

```
PRIVATE username : STRING
PRIVATE password : STRING
PRIVATE email   : STRING
PRIVATE pinAngka : INTEGER
PROCEDURE setUsername(username : STRING)
    username ← username
END PROCEDURE
```

```
PROCEDURE setPassword(password : STRING)
    password ← password
END PROCEDURE
```

```
PROCEDURE setEmail(email : STRING)
    email ← email
END PROCEDURE
```

```
PROCEDURE setPin(pin : INTEGER)
    pinAngka ← pin
END PROCEDURE
```

```
FUNCTION getUsername() : STRING
    RETURN username
END FUNCTION
```

```
FUNCTION getPassword() : STRING
    RETURN password
END FUNCTION

FUNCTION getEmail() : STRING
    RETURN email
END FUNCTION

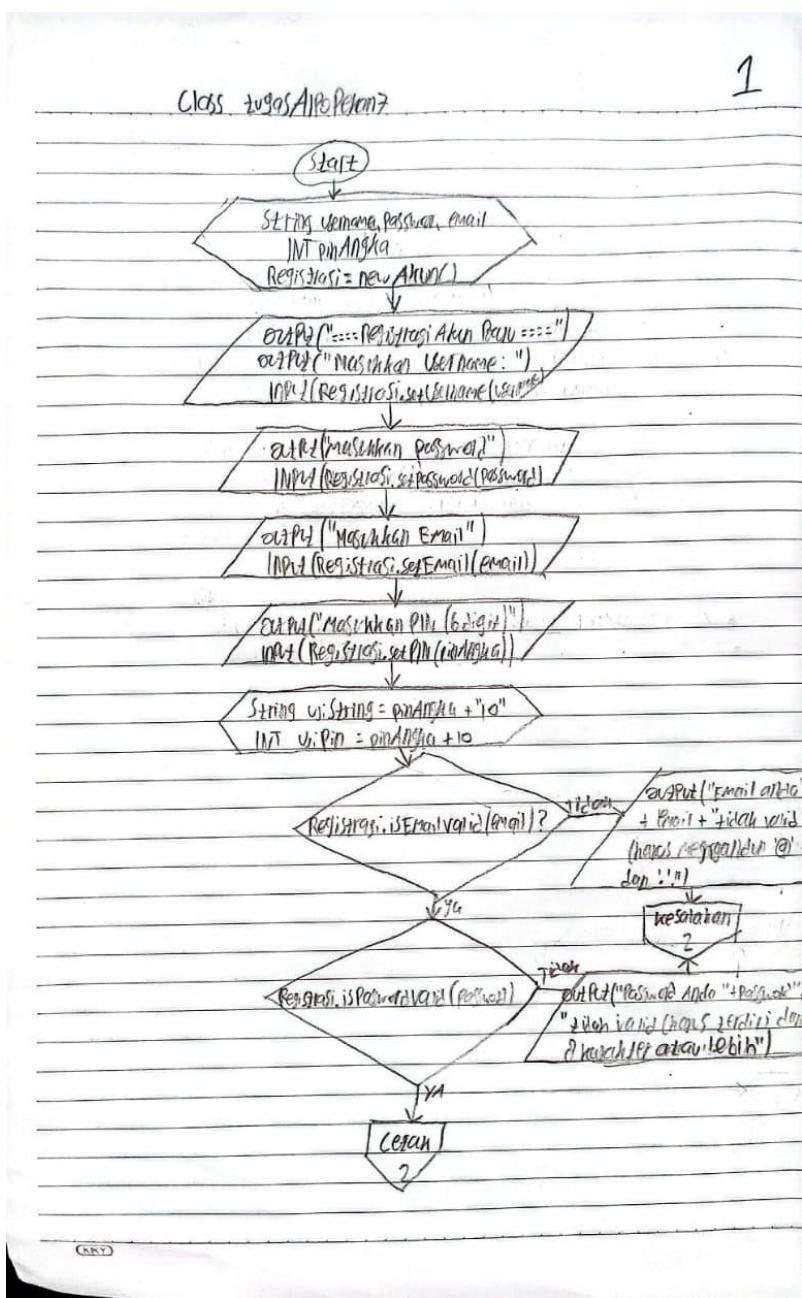
FUNCTION getPin() : INT
    RETURN pinAngka
END FUNCTION

FUNCTION isEmailValid(e : STRING) : BOOLEAN
    IF (email CONTAINS("@")) AND (email CONTAINS(".")) THEN RETURN TRUE
    ELSE RETURN FALSE
    END IF
END FUNCTION

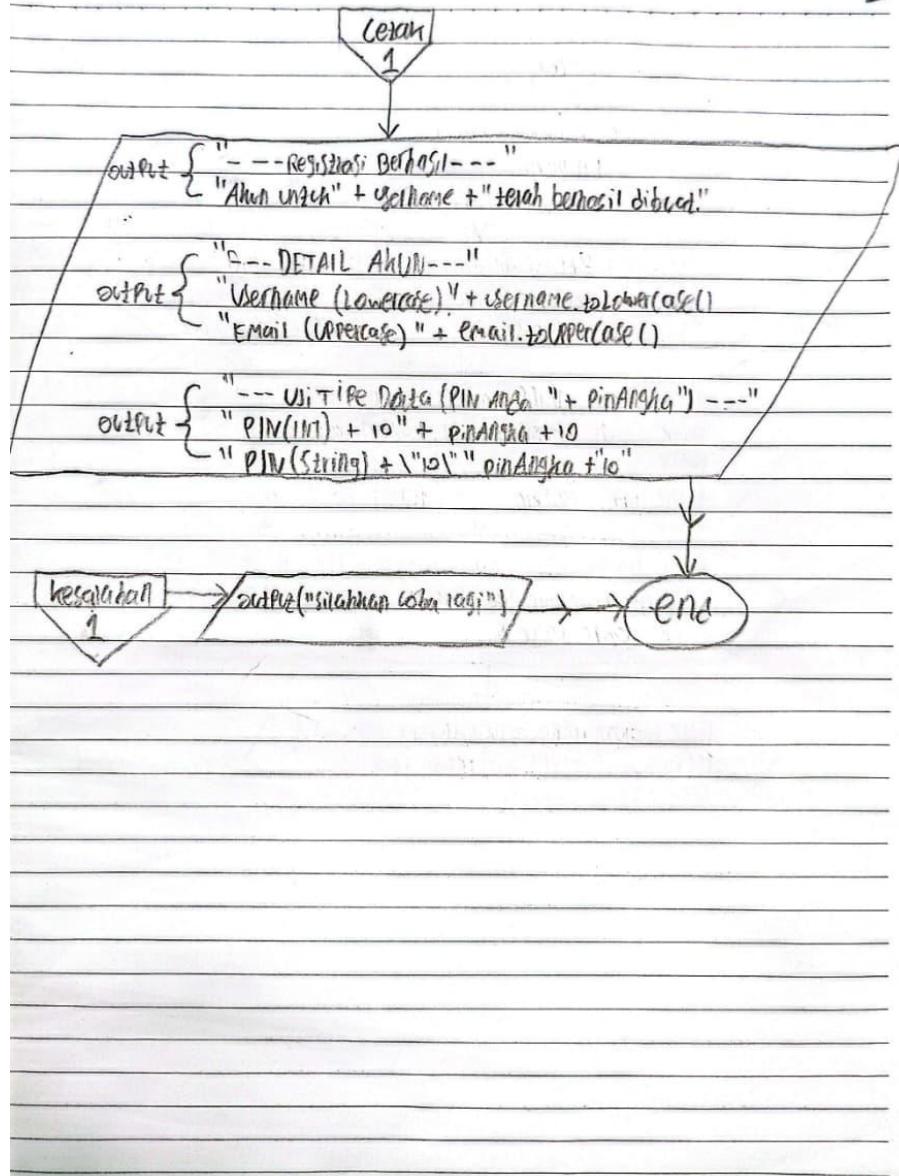
FUNCTION isPasswordValid(password : STRING) : BOOLEAN
    IF LENGTH(password) ≥ 8 THEN
        RETURN TRUE
    ELSE
        RETURN FALSE
    END IF
END FUNCTION

END CLASS
```

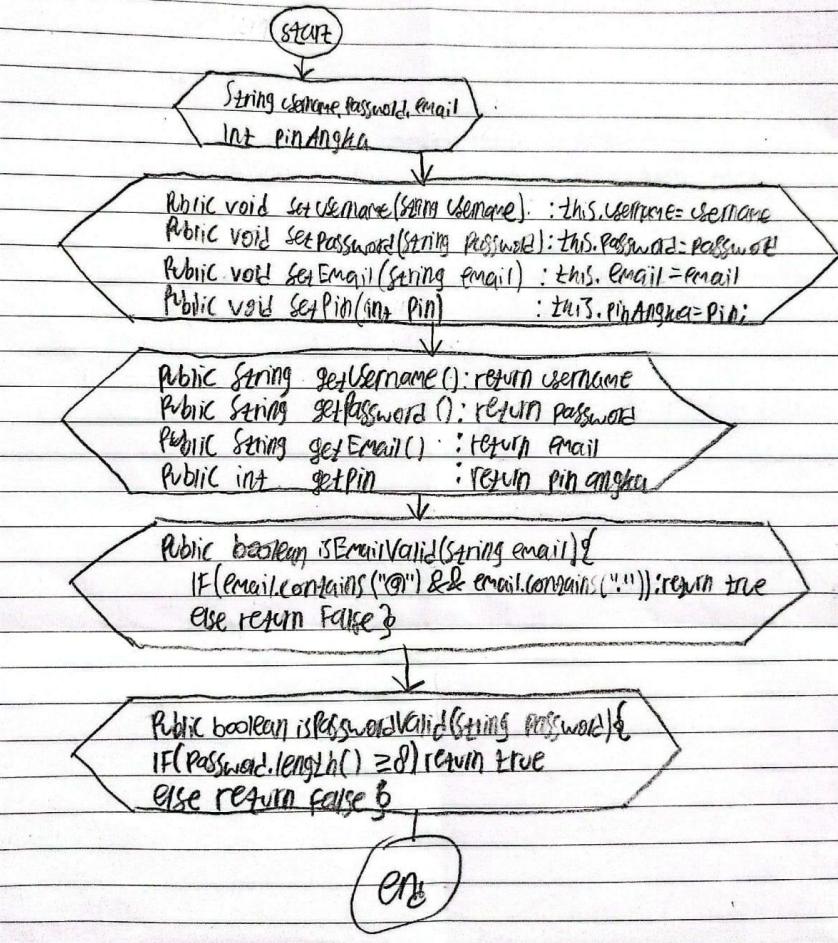
2. Flowchart



2



Class Aku



3.Program Java

3.1 Kelas Akun

```
1 package pekan7_2511532016;
2
3 public class Akun{
4
5     //BAGIAN PENYIMPANAN DATA
6     private String username, password, email;
7     private int pinAngka;
8     public void setUsername(String username) {
9         this.username = username;
10    }
11    public void setPassword(String password) {
12        this.password = password;
13    }
14    public void setEmail(String email) {
15        this.email = email;
16    }
17    public void setPin(int pin) {
18        this.pinAngka = pin;
19    }
20
21    public String getUsername() {
22        return username;
23    }
24    public String getPassword() {
25        return password;
26    }
27    public String getEmail() {
28        return email;
29    }
30    public int getPin() {
31        return pinAngka;
32    }
33    //BAGIAN VALIDASI DATA
34    public boolean isValidEmail (String email) {
35        if(email.contains("@") && email.contains(".")) return true;
36        else return false;
37    }
38
39    public boolean isValidPassword (String password) {
40        if(password.length() >= 8) return true;
41        else return false;
42    }
43 }
```

Bagian kode ini adalah kelas Akun yang dipakai sebagai “tempat penyimpanan data” untuk objek akun yang dibuat di program utama. Kelas ini menyimpan objek-objek seperti username, password, email, dan PIN yang akan dipakai nantinya pada kelas utama.

Pada bagian paling atas terdapat empat variabel: username, password, email, dan pinAngka. Keempatnya diberi akses *private*, artinya variabel ini hanya boleh diakses dari dalam kelas Akun sendiri, tidak boleh diubah langsung dari luar. Karena itulah dibuat metode *setter* seperti setUsername, setPassword, setEmail, dan setPin. Metode-metode ini berfungsi untuk mengisi atau mengubah nilai variabel-variabel tersebut. Jadi, ketika bagian program utama ingin mengisi data username, ia

memanggil setUsername(username), dan metode inilah yang menyimpan nilai itu ke dalam variabel milik objek. Begitu pula untuk data yang akan diisi lainnya.

Selain *setter*, ada juga *getter*, yaitu getUsername, getPassword, getEmail, dan getPin. *Getter* berfungsi sebagai cara mengambil atau membaca isi variabel private tadi. Ini memastikan bahwa data di dalam objek tetap terkontrol dan tidak sembarangan diubah atau diakses dari luar kelas. Dengan sistem setter dan getter seperti ini, kelas Akun menjadi lebih rapi, aman, dan mengikuti prinsip OOP.

Di bagian paling bawah, terdapat dua fungsi yang memiliki nilai kembalian *boolean*. Yaitu isEmailValid() dan isPasswordValid yang memiliki parameter berupa String. Kedua fungsi ini memiliki fungsi untuk mengecek apakah suatu password/email memenuhi kriteria atau tidak. Password yang memenuhi syarat adalah password yang memiliki panjang 8 karakter atau lebih. Sedangkan email yang memenuhi syarat adalah email yang memiliki tanda "@" dan "." di dalamnya.

3.2 Input Pengguna

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    Akun Registrasi = new Akun();

    //BAGIAN TERIMA INPUT
    System.out.println("==== REGISTRASI AKUN BARU ====");
    System.out.print("Masukkan Username: ");
    String username = input.nextLine();
    System.out.print("Masukkan Password: ");
    String password = input.nextLine();
    System.out.print("Masukkan Email: ");
    String email = input.nextLine();
    System.out.print("Masukkan PIN (6 digit): ");
    int pinAngka = input.nextInt();

    input.close();

    Registrasi.setEmail(email);
    Registrasi.setPassword(password);
    Registrasi.setUsername(username);
    Registrasi.setPin(pinAngka);
```

Bagian awal program dari kelas utama di sini bertugas menerima input pengguna dan mengirim data itu ke objek Akun. Program dimulai dengan membuat objek Scanner untuk membaca input dari keyboard, kemudian membuat objek Registrasi dari kelas Akun sebagai tempat menyimpan informasi akun yang baru. Setelah itu, program menampilkan judul registrasi dan mulai meminta pengguna memasukkan empat data berupa username, password, email, dan PIN mereka. Setiap data yang dimasukkan disimpan sementara ke dalam variabel lokal (username, password, email, dan pinAngka).

3.3 Output Program

```
28 //BAGIAN HASIL KELUARAN
29 if(!(Registrasi.isEmailValid(email))) {
30     System.out.println ("Email Anda\" " + email + "\" tidak valid (harus mengandung '@' dan '.').");
31     System.out.println("Silahkan coba lagi.");
32 }
33
34 else if(!(Registrasi.isPasswordValid(password))) {
35     System.out.println("Password Anda\" " + password + "\" tidak valid (harus terdiri dari 8 karakter atau lebih).");
36     System.out.println("Silahkan coba lagi.");
37 }
38 else{
39     System.out.println();
40     System.out.println("--- REGISTRASI BERHASIL ---");
41     System.out.println("Akun untuk \"'" + username + "\" telah berhasil dibuat.");
42     System.out.println();
43
44     System.out.println("--- DETAIL AKUN ---");
45     System.out.printf("%-25s: %s %s\n", "Username (Lowercase) ", username.toLowerCase(), "");
46     System.out.printf("%-25s: %s %s\n", "Email (Uppercase) " , email.toUpperCase(), "");
47     System.out.printf("%-25s: %s %s\n", "ID Pengguna (Gabungan) " , username + pinAngka, "");
48
49     System.out.println();
50     System.out.println("--- Uji Tipe Data (PIN Anda " + pinAngka + ") ---");
51     System.out.printf("%-25s= %s %s\n" , "PIN (int) + 10 " , (pinAngka + 10), "");
52     System.out.printf("%-25s= %s %s\n" , "PIN (String) + \"10\" " , pinAngka+ "10", "");
```

Bagian kode ini adalah tahap hasil keluaran, yang di mana program mengecek apakah data yang dimasukkan pengguna sudah benar, lalu menampilkan hasilnya.

Pertama, program memeriksa email yang dimasukkan pengguna dengan memanggil Registrasi.isEmailValid(email). Jika email tidak mengandung tanda "@" dan ".", maka nilai !(Registrasi.isEmailValid(email)) menjadi benar, sehingga program langsung menampilkan pesan bahwa email tidak valid dan meminta pengguna mencoba lagi. Jika email valid, barulah program masuk ke pengecekan kedua, yaitu memeriksa password melalui Registrasi.isPasswordValid(password). Bila panjang password kurang dari delapan karakter, program menampilkan pesan bahwa password tidak memenuhi syarat, dan pengguna diminta untuk mengulang kembali.

Jika kedua pengecekan tersebut lolos, program masuk ke bagian akhir yang merupakan blok keberhasilan. Di sini program menampilkan pesan bahwa registrasi berhasil dan mencetak detail akun milik pengguna. Username ditampilkan dalam huruf kecil, email ditampilkan dalam huruf kapital, dan ID pengguna dibuat dengan menggabungkan username dan PIN. Setelah itu program menunjukkan dua hasil uji tipe data: hasil penjumlahan angka pinAngka + 10 dan hasil penggabungan string pinAngka + "10", yang menegaskan perbedaan antara operasi matematika dan penggabungan teks. Setelah semua informasi selesai ditampilkan, bagian keluaran ini menandai akhir proses program.

4. Contoh Keluaran(Output) dari Program Java

```
<terminated> tugasAlproPekan7_2511532016 [Java Application] C:\Users\USER\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full  
===== REGISTRASI AKUN BARU =====  
Masukkan Username: fufufafa67  
Masukkan Password: 67676767  
Masukkan Email: fufufafa67@wapres.paman  
Masukkan PIN (6 digit): 676767  
  
--- REGISTRASI BERHASIL ---  
Akun untuk "fufufafa67" telah berhasil dibuat.  
  
--- DETAIL AKUN ---  
Username (Lowercase) : fufufafa67  
Email (Uppercase) : FUFUFAFA67@WAPRES.PAMAN  
ID Pengguna (Gabungan) : fufufafa67676767  
  
--- Uji Tipe Data (PIN Anda 676767) ---  
PIN (int) + 10 = 676777  
PIN (String) + "10" = 67676710
```

```
<terminated> tugasAlproPekan7_2511532016 [Java Application] C:\Users\USER\p2\pool\plugin  
===== REGISTRASI AKUN BARU =====  
Masukkan Username: Abique  
Masukkan Password: MahasiswaUnand  
Masukkan Email: Abiquitoast@gmail.com  
Masukkan PIN (6 digit): 12342016  
  
--- REGISTRASI BERHASIL ---  
Akun untuk "Abique" telah berhasil dibuat.  
  
--- DETAIL AKUN ---  
Username (Lowercase) : abique  
Email (Uppercase) : ABIQUITOAST@GMAIL.COM  
ID Pengguna (Gabungan) : Abique12342016  
  
--- Uji Tipe Data (PIN Anda 12342016) ---  
PIN (int) + 10 = 12342026  
PIN (String) + "10" = 1234201610
```

```
<terminated> tugasAlproPekan7_2511532016 [Java Application] C:\Users\USER\p2\pool\plugins\org.eclipse.ju  
===== REGISTRASI AKUN BARU =====  
Masukkan Username: Bah-Lil'Gremlin  
Masukkan Password: BahlilEt4n01  
Masukkan Email: BahlilLup@kasihTitikcom  
Masukkan PIN (6 digit): 943185  
Email Anda" BahlilLup@kasihTitikcom" tidak valid (harus mengandung '@' dan '.').  
Silahkan coba lagi.
```

```
<terminated> tugasAlproPekan7_2511532016 [Java Application] C:\Users\USER\p2\pool\plugins\org.eclipse.ju  
===== REGISTRASI AKUN BARU =====  
Masukkan Username: Cecep Suhendar  
Masukkan Password: Kurang  
Masukkan Email: Cecep@gmail.com  
Masukkan PIN (6 digit): 129463  
Password Anda" Kurang" tidak valid (harus terdiri dari 8 karakter atau lebih).  
Silahkan coba lagi.
```

```
<terminated> tugasAlproPekan7_2511532016 [Java Application] C:\Users\USER\p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_21.0.8.v20250724-1412\jre\bin\javaw.exe (Nov 16, 2023, 10:45:45 AM)  
==== REGISTRASI AKUN BARU ====  
Masukkan Username: Ren Amamiya  
Masukkan Password: DidYouKnowThatRenAmamiyaWasExpelledFromHisSchoolAndThenMovedToADifferentCityWhereHeBecameThePhantomThiefForAlmostAYear?That'sCrazy  
Masukkan Email: JokerAuraFarmer@phan.com  
Masukkan PIN (6 digit): 150901  
  
--- REGISTRASI BERHASIL ---  
Akun untuk "Ren Amamiya" telah berhasil dibuat.  
  
--- DETAIL AKUN ---  
Username (Lowercase) : ren amamiya  
Email (Uppercase) : JOKERAURAFARMER@PHAN.COM  
ID Pengguna (Gabungan) : Ren Amamiya150901  
  
--- Uji Tipe Data (PIN Anda 150901) ---  
PIN (int) + 10 = 150911  
PIN (String) + "10" = 15090110
```