

Nama : Viona Asya Arinda
NIM : 2109106108
Kelas : INFORMATIKA B 2021

POSTTEST 1

INSTRUKSI

- Tentukan Tema Program Kalian (**Tidak Mengandung SARA , Pornografi dan Sejenisnya**)
- Kemudian Tentukan Judul program kalian, Setiap Individu **tidak boleh memiliki judul program yang sama dengan Individu lainnya.**
- **Silahkan list judul dan tema kalian pada link dibawah ini:** <https://docs.google.com/spreadsheets/d/1N5WfVFKOrDDr1CxKb9YtnH2oIFyG5K8FpRA6NViw5jU/edit?usp=sharing>

INSTRUKSI PADA PROGRAM JAVA

- Program dapat melakukan CRUD (Create, Read, Update, Delete) data dari sebuah ArrayList.
- Memiliki menu pilihan dan program dapat berjalan berulang - ulang sampai kita memilih menu exit program.
- Data yang dikelola merupakan data dari class sesuai dengan judul masing - masing

PERHATIAN !!!

1. Bersifat individu dan wajib dikerjakan bagi setiap peserta praktikum
2. Kerjakan dengan sungguh-sungguh dan semaksimal mungkin
3. Percaya pada diri sendiri , Tidak perlu mencontek Teman kalian
4. Jika terindikasi meniru teman (hanya ganti tema, hanya ganti variabel, hanya ganti tipe data, hanya ganti tampilan), Keduanya mendapat nilai E.
5. Semangat Mengerjakan

Dikumpul melalui github, diclassroom hanya link repository terkait

Code Program :

```
package rumahsakit;  
  
import java.io.*;  
import java.lang.reflect.Array;  
import java.time.Year;  
import java.util.Arrays;
```

```
System.out.println("
");
System.out.println("LIST SELURUH DATA PASIEN");
```

```

System.out.println("
                \n");
                tampilkanData();
                break;
            case "2":

System.out.println("
                ");
                System.out.println("                CARI DATA PASIEN                ");

System.out.println("
                \n");
                cariData();
                break;
            case "3":

System.out.println("
                ");
                System.out.println("                TAMBAH DATA PASIEN                ");

System.out.println("
                \n");
                tambahData();
                tampilkanData();
                break;
            case "4":

System.out.println("
                ");
                System.out.println("                UBAH DATA PASIEN                ");

System.out.println("
                \n");
                updateData();
                break;
            case "5":

System.out.println("
                ");
                System.out.println("                HAPUS DATA PASIEN                ");

System.out.println("
                \n");
                deleteData();
                break;
            default:
                System.err.println("\nInput anda tidak ditemukan\nSilahkan pilih [1-5]");

```

```

    }

    isLanjutkan = getYesorNo("Apakah Anda Ingin Melanjutkan");
}

private static void updateData() throws IOException{
// database original
File database = new File("database.txt");
FileReader fileInput = new FileReader(database);
BufferedReader bufferedInput = new BufferedReader(fileInput);

// database sementara
File tempDB = new File("tempDB.txt");
FileWriter fileOutput = new FileWriter(tempDB);
BufferedWriter bufferedOutput = new BufferedWriter(fileOutput);

// tampilkan data
System.out.println("List pasien");
tampilkanData();

// ambil user input / pilihan data
Scanner terminalInput = new Scanner(System.in);
System.out.print("\nMasukan Nomor Pasien Yang Akan DiUpdate : ");

System.out.println("
");
int updateNum = terminalInput.nextInt();

// tampilkan data yang ingin diupdate

String data = bufferedInput.readLine();
int entryCounts = 0;

while (data != null){
    entryCounts++;

    StringRumahSakit st = new StringRumahSakit(data,",");

// tampilkan entrycounts == updateNum
if (updateNum == entryCounts){
    System.out.println("\nData yang ingin anda update adalah:");

System.out.println("
");

    System.out.println("dokter      : " + st.nextToken());
    System.out.println("kamar       : " + st.nextToken());
    System.out.println("nama        : " + st.nextToken());

```

```

System.out.println("penyakit      : " + st.nextToken());
System.out.println("umur        : " + st.nextToken());

// update data

// mengambil input dari user

String[] fieldData = {"kamar", "nama", "penyakit", "umur"};
String[] tempData = new String[4];

st = new StringRumahSakit(data, ",");
String originalData = st.nextToken();

for(int i=0; i < fieldData.length ; i++) {
    boolean isUpdate = getYesorNo("apakah anda ingin merubah " + fieldData[i]);
    originalData = st.nextToken();
    if (isUpdate){
        //user input

        if (fieldData[i].equalsIgnoreCase("kamar")){
            System.out.print("masukan kamar, format=(YYYY): ");
            tempData[i] = ambilkamar();
        } else {
            terminalInput = new Scanner(System.in);
            System.out.print("\nMasukan " + fieldData[i] + " baru: ");
            tempData[i] = terminalInput.nextLine();
        }

    } else {
        tempData[i] = originalData;
    }
}

// tampilkan data baru ke layar
st = new StringRumahSakit(data, ",");
st.nextToken();
System.out.println("\nData baru anda adalah ");

System.out.println("
");

System.out.println("kamar      : " + st.nextToken() + "--> " + tempData[0]);
System.out.println("nama      : " + st.nextToken() + "--> " + tempData[1]);
System.out.println("penyakit   : " + st.nextToken() + "--> " + tempData[2]);
System.out.println("umur      : " + st.nextToken() + "--> " + tempData[3]);

boolean isUpdate = getYesorNo("apakah anda yakin ingin mengupdate data tersebut");

if (isUpdate){

```

```

// cek data baru di database
boolean isExist = cekpasienDiDatabase(tempData,false);

if(isExist){
    System.err.println("data pasien sudah ada di database, proses update dibatalkan, \nsilahkan
delete data yang bersangkutan");
    // copy data
    bufferedOutput.write(data);

} else {

    // format data baru kedalam database
    String kamar = tempData[0];
    String nama = tempData[1];
    String penyakit = tempData[2];
    String umur = tempData[3];

    // kita bikin primary key
    long nomorEntry = ambilEntryPerkamar(nama, kamar) + 1;

    String punulisTanpaSpasi = nama.replaceAll("\\s+", "");
    String primaryKey = punulisTanpaSpasi+"_"+kamar+"_"+nomorEntry;

    // tulis data ke database
    bufferedOutput.write(primaryKey + "," + kamar + "," + nama + "," + penyakit + "," + umur);
}
} else {
    // copy data
    bufferedOutput.write(data);
}
} else {
    // copy data
    bufferedOutput.write(data);
}
bufferedOutput.newLine();

data = bufferedInput.readLine();
}

// menulis data ke file
bufferedOutput.flush();

// delete original database
database.delete();
// rename file tempDB menjadi database
tempDB.renameTo(database);

```

```
private static void deleteData() throws IOException{
    // kita ambil database original
    File database = new File("database.txt");
    FileReader fileInput = new FileReader(database);
    BufferedReader bufferedInput = new BufferedReader(fileInput);

    // kita buat database sementara
    File tempDB = new File("tempDB.txt");
    FileWriter fileOutput = new FileWriter(tempDB);
    BufferedWriter bufferedOutput = new BufferedWriter(fileOutput);

    // tampilkan data
    System.out.println("List pasien");
    tampilkanData();

    // kita ambil user input untuk mendelete data
    Scanner terminalInput = new Scanner(System.in);
    System.out.print("\nMasukan nomor pasien yang akan dihapus: ");
    int deleteNum = terminalInput.nextInt();

    // looping untuk membaca tiap data baris dan skip data yang akan didelete

    boolean isFound = false;
    int entryCounts = 0;

    String data = bufferedInput.readLine();

    while (data != null){
        entryCounts++;
        boolean isDelete = false;

        StringRumahSakit st = new StringRumahSakit(data, "");

        // tampilkan data yang ingin di hapus
        if (deleteNum == entryCounts){
            System.out.println("\nData Yang Ingin Anda Hapus Adalah:");

            System.out.println("
            ");

            System.out.println("dokter      : " + st.nextToken());
            System.out.println("kamar      : " + st.nextToken());
            System.out.println("nama      : " + st.nextToken());
            System.out.println("penyakit   : " + st.nextToken());
            System.out.println("umur      : " + st.nextToken());

            isDelete = getYesorNo("Apakah anda yakin akan menghapus?");
        }
        bufferedOutput.write(data + "\n");
        bufferedOutput.flush();
        bufferedInput.readLine();
    }
}
```

```

        isFound = true;
    }

    if(isDelete){
        //skip pindahkan data dari original ke sementara
        System.out.println("Data berhasil dihapus");
    } else {
        // kita pindahkan data dari original ke sementara
        bufferedOutput.write(data);
        bufferedOutput.newLine();
    }
    data = bufferedInput.readLine();
}

if(!isFound){
    System.err.println("pasien tidak ditemukan");
}

// menulis data ke file
bufferedOutput.flush();
// delete original file
database.delete();
// rename file sementara ke database
tempDB.renameTo(database);
}

private static void tampilkanData() throws IOException{

    FileReader fileInput;
    BufferedReader bufferInput;

    try {
        fileInput = new FileReader("database.txt");
        bufferInput = new BufferedReader(fileInput);
    } catch (Exception e){
        System.err.println("Database Tidak ditemukan");
        System.err.println("Silahkan tambah data terlebih dahoe");
        tambahData();
        return;
    }

    System.out.println("\n| No |tkamar |tnama          |tpenyakit          |tumur pasien");

    System.out.println("
");

    String data = bufferInput.readLine();

```



```

int nomorData = 0;
while(data != null) {
    nomorData++;

    StringRumahSakit stringToken = new StringRumahSakit(data, ",");

    stringToken.nextToken();
    System.out.printf("| %2d ", nomorData);
    System.out.printf("\t%4s ", stringToken.nextToken());
    System.out.printf("\t%-20s ", stringToken.nextToken());
    System.out.printf("\t%-20s ", stringToken.nextToken());
    System.out.printf("\t%s ", stringToken.nextToken());
    System.out.print("\n");

    data = bufferInput.readLine();
}

```

```

System.out.println("
");
}

```

```

private static void cariData() throws IOException{

    // membaca database ada atau tidak

    try {
        File file = new File("database.txt");
    } catch (Exception e){
        System.err.println("Database Tidak ditemukan");
        System.err.println("Silahkan tambah data terlebih dahoeloe");
        tambahData();
        return;
    }

    // kita ambil keyword dari user

    Scanner terminalInput = new Scanner(System.in);
    System.out.print("Masukan kata kunci untuk mencari pasien: ");
    String cariString = terminalInput.nextLine();
    String[] keywords = cariString.split("\\s+");

    // kita cek keyword di database
    cekpasienDiDatabase(keywords,true);

}

```

```

private static void tambahData() throws IOException{

```

```

FileWriter fileOutput = new FileWriter("database.txt",true);
BufferedWriter bufferOutput = new BufferedWriter(fileOutput);

// mengambil input dari user
Scanner terminalInput = new Scanner(System.in);
String nama, umur, penyakit, kamar;

System.out.print("masukan dokter nama: ");
nama = terminalInput.nextLine();
System.out.print("masukan umur pasien: ");
umur = terminalInput.nextLine();
System.out.print("masukan dokter penyakit: ");
penyakit = terminalInput.nextLine();
System.out.print("masukan kamar, format=(YYYY): ");
kamar = ambilkamar();

// cek pasien di database

String[] keywords = {kamar+", "+nama+", "+penyakit+", "+umur};
System.out.println(Arrays.toString(keywords));

boolean isExist = cekpasienDiDatabase(keywords,false);

// menulis pasien di databse
if (!isExist){
//    fiersabesari_2012_1,2012,fiersa besari,media kita,jejak langkah
    System.out.println(ambilEntryPerkamar(nama, kamar));
    long nomorEntry = ambilEntryPerkamar(nama, kamar) + 1;

    String punulisTanpaSpasi = nama.replaceAll("\\s+", "");
    String primaryKey = punulisTanpaSpasi + "_" + kamar + "_" + nomorEntry;
    System.out.println("\nData yang akan anda masukan adalah");

System.out.println("
");

    System.out.println("primary key : " + primaryKey);
    System.out.println("kamar : " + kamar);
    System.out.println("nama : " + nama);
    System.out.println("umur : " + umur);
    System.out.println("penyakit : " + penyakit);

    boolean isTambah = getYesorNo("Apakah akan ingin menambah data tersebut? ");

    if(isTambah){
        bufferOutput.write(primaryKey + ", " + kamar + ", " + nama + ", " + penyakit + ", "+umur);
        bufferOutput.newLine();

```

```

        bufferOutput.flush();
    }

    } else {
        System.out.println("pasien yang anda akan masukan sudah tersedia di database dengan data
berikut:");
        cekpasienDiDatabase(keywords,true);
    }

    bufferOutput.close();
}

private static long ambilEntryPerkamar(String nama, String kamar) throws IOException {
    FileReader fileInput = new FileReader("database.txt");
    BufferedReader bufferInput = new BufferedReader(fileInput);

    long entry = 0;
    String data = bufferInput.readLine();
    Scanner dataScanner;
    String primaryKey;

    while(data != null){
        dataScanner = new Scanner(data);
        dataScanner.useDelimiter(",");
        primaryKey = dataScanner.next();
        dataScanner = new Scanner(primaryKey);
        dataScanner.useDelimiter("_");

        nama = nama.replaceAll("\\s+", "");

        if (nama.equalsIgnoreCase(dataScanner.next()) && kamar.equalsIgnoreCase(dataScanner.next()))
        {
            entry = dataScanner.nextInt();
        }

        data = bufferInput.readLine();
    }

    return entry;
}

private static boolean cekpasienDiDatabase(String[] keywords, boolean isDisplay) throws IOException{

    FileReader fileInput = new FileReader("database.txt");
    BufferedReader bufferInput = new BufferedReader(fileInput);

    String data = bufferInput.readLine();
    boolean isExist = false;

```

[illegible]

```

private static String ambilkamar() throws IOException{
    boolean kamarValid = false;
    Scanner terminalInput = new Scanner(System.in);
    String kamarInput = terminalInput.nextLine();
    while(!kamarValid) {
        try {
            Year.parse(kamarInput);
            kamarValid = true;
        } catch (Exception e) {
            System.out.println("Format kamar yang anda masukan salah, format=(YYYY)");
            System.out.print("silahkan masukan kamar lagi: ");
            kamarValid = false;
            kamarInput = terminalInput.nextLine();
        }
    }

    return kamarInput;
}

private static boolean getYesorNo(String message){
    Scanner terminalInput = new Scanner(System.in);
    System.out.print("\n"+message+" (y/n)? ");
    String pilihanUser = terminalInput.next();

    while(!pilihanUser.equalsIgnoreCase("y") && !pilihanUser.equalsIgnoreCase("n")) {
        System.err.println("Pilihan anda bukan y atau n");
        System.out.print("\n"+message+" (y/n)? ");
        pilihanUser = terminalInput.next();
    }

    return pilihanUser.equalsIgnoreCase("y");
}

private static void clearScreen(){
    try {
        if (System.getProperty("os.name").contains("Windows")){
            new ProcessBuilder("cmd", "/c", "cls").inheritIO().start().waitFor();
        } else {
            System.out.print("\033\143");
        }
    } catch (Exception ex){
        System.err.println("tidak bisa clear screen");
    }
}
}

```

SS OUTPUT PROGRAM :

1. Menu Utama

```
|| SISTEM || MANAJEMEN || RUMAH SAKIT ||

[1] Lihat Seluruh Data Pasien
[2] Cari Data Pasien
[3] Tambah Data Pasien
[4] Ubah Data Pasien
[5] Hapus Data Pasien
```

2. Lihat Seluruh Data Pasien

LIST SELURUH DATA PASIEN					
No	kamar	nama	penyakit	umur pasien	
1	4001	Viona	Magh	19 Tahun	
2	4002	Sahriah	Pusing	20 Tahun	
3	4003	Cici	DBD	25 Tahun	
4	4004	Dhea	Muntaber	30 Tahunssss_9000_1	

3. Cari Data Pasien

```
=====
CARI DATA PASIEN
=====

Masukan kata kunci untuk mencari pasien: 4001

| No | kamar | nama          | penyakit          | umur pasien |
-----
| 1  | 4001  | Viona         | Magh              | 19 Tahun    |
=====
```

4. Tambah Data Pasien

```
=====
TAMBAH DATA PASIEN
=====

masukan dokter nama: dodo
masukan umur pasien: nana
masukan dokter penyakit: demam
masukan kamar, format=(YYYY): 2004
[2004,dodo,demam,nana]
0

Data yang akan anda masukan adalah
=====

primary key  : dodo_2004_1
kamar       : 2004
nama        : dodo
umur        : nana
penyakit    : demam

Apakah akan ingin menambah data tersebut? (y/n)?
```

5. Ubah Data Pasien

apakah anda yakin ingin mengupdate data tersebut (y/n)?

[illegible]