

Nama: Muhammad Naufal Ihsan Maulidin

NIM: 2109106034

Informatika A2 2021

POSTTEST 4

Screenshot hasil program:

- 1) Tampilan awal program ketika dijalankan

```
run:
=====
Program Pendataan Tim Basket
=====
1. List Tim
2. Tambah Data Tim
3. Edit Data Tim
4. Hapus Data Tim
5. Exit Program
=====
Masukkan Opsi Pilihan:
```

- 2) Tampilan program pada menu list tim masih sama dengan posttest sebelumnya, dimana program akan meminta user untuk memilih data mana yang ingin ditampilkan. Adapun data terbagi menjadi 3, data tim, pemain dan pelatih. Apabila belum ada data yang ditambahkan, program akan mengeluarkan input “data kosong”.

```
=====
Masukkan Opsi Pilihan: 1
=====
1. Data Tim
2. Data Pelatih
3. Data Pemain
=====
Masukkan Opsi Pilihan: 1
=====
Data Tim Kosong
=====
```

Namun apabila terdapat data yang telah diinputkan, program akan mengeluarkan output berupa data yang ada sesuai dengan kategori data yang dipilih. Sebagai contoh user memilih data tim, maka program akan menampilkan data tim.

```
=====
Masukkan Opsi Pilihan: 1
=====
Data Tim Basket
1>Nama Tim = Lakers
Lokasi Tim = Los Angeles
Tahun Tim Terbentuk = 1947
=====
```

- 3) Tampilan menu tambah data juga masih sama dengan posttest sebelumnya. Program akan meminta user untuk memilih data mana yang ingin ditambahkan, apakah data tim, pemain atau pelatih. Jika telah memilih data mana yang ingin diinput, selanjutnya program akan meminta beberapa inputan terkait dengan data yang ingin ditambahkan.

```
=====
Masukkan Opsi Pilihan: 2
Tambah Data Tim Basket
=====
1. Tambah Data Tim
2. Tambah Data Pelatih
3. Tambah Data Pemain
=====
Masukkan Opsi Pilihan: |
```

Seperti contoh apabila user memilih opsi “tambah data tim”, program akan meminta user untuk memasukkan beberapa inputan terkait data tim.

```
=====
Opsi Tambah Data Tim Basket
=====
Masukkan Nama Tim: Lakers
Masukkan Lokasi Tim: Los Angeles
Masukkan Tahun Tim Terbentuk: 1947
=====
Data Tim Berhasil Ditambahkan
=====
```

- 4) Tampilan menu edit data. Program akan meminta user untuk memilih data mana yang ingin di edit, apakah data tim, pemain atau pelatih. Lalu program akan meminta inputan data baru yang akan digunakan dalam mengedit data. Sebagai contoh user ingin mengubah data tim, maka tampilan program adalah seperti berikut.

```
=====
1. Data Tim
2. Data Pelatih
3. Data Pemain
=====
Masukkan Opsi Pilihan: 1
=====
Data Tim Basket
=====
1.Nama Tim = Lakers
Lokasi Tim = Los Angeles
Tahun Tim Terbentuk = 1947
=====
Pilih Index Data Yang Ingin Diubah: 1
=====
Masukkan Nama Tim Baru: GSW
Masukkan Lokasi Baru: San Fransisco
Masukkan Tahun Bentuk Baru: 1946
=====
Data Berhasil Diperbarui!
=====
```

5) Tampilan hapus data. Program akan meminta user memilih data mana yang ingin dihapus.

```
=====
Masukkan Opsi Pilihan: 4
=====
1. Hapus Data Tim
2. Hapus Data Pelatih
3. Hapus Data Pemain
=====
Pilih Opsi: 1
=====
1. Data Tim
2. Data Pelatih
3. Data Pemain
=====
Masukkan Opsi Pilihan: 1
=====
Data Tim Basket
=====
1. Nama Tim = GSW
Lokasi Tim = San Fransisco
Tahun Tim Terbentuk = 1946
=====
Pilih Data yang Ingin Dihapus: 1
=====
Data Tim Berhasil Dihapus
=====
```

6) Tampilan exit program.

```
=====
Program Pendataan Tim Basket
=====
1. List Tim
2. Tambah Data Tim
3. Edit Data Tim
4. Hapus Data Tim
5. Exit Program
=====
Masukkan Opsi Pilihan: 5
=====
Terima Kasih Telah Menggunakan Program Kami
Mengeluarkan Program...
=====
```

7) Penerapan Override dan Overload merupakan pembeda program dengan posttest sebelumnya. Dimana pada posstest 4, praktikan diminta untuk menerapkan polymorphism di dalam program. Penerapannya adalah sebagai berikut.

- Overloading

Penerapan overloading pada program terletak pada fungsi read() dan addData(). Dimana pada read(), menggunakan parameter yang berbeda untuk menampilkan tampilan yang berbeda pula.

```
public static void read() throws IOException{
    System.out.println( x: "=====");
    System.out.println( x: "1. Data Tim");
    System.out.println( x: "2. Data Pelatih");
    System.out.println( x: "3. Data Pemain");
    System.out.println( x: "=====");
    System.out.print( s: "Masukkan Opsi Pilihan: ");
    int opsi = Integer.parseInt( s: buff.readLine());
    read( opsi1: opsi, opsi2: -1);
}

public static void read(int opsi1, int opsi2) throws IOException{
    if (opsi1 == 1){
        if (tim.isEmpty()){
            System.out.println( x: "=====");
            System.out.println( x: "Data Tim Kosong");
            System.out.println( x: "=====");
        }
        else{
            System.out.println( x: "=====");
            System.out.println( x: "Data Tim Basket");
            System.out.println( x: "=====");
            for (int i = 0; i < tim.size(); i++){
                System.out.print(i+1 + ". ");
                tim.get( index: i).show();
                System.out.println( x: "=====");
            }
        }
    }
}
```

read() berfungsi untuk menampilkan opsi dari data mana yang ingin ditampilkan di dalam program, sedangkan read(int opsi1, int opsi2) digunakan untuk menampilkan data dari opsi yang telah dipilih di dalam program, seperti data pelatih, pemain atau data tim basket.

Selanjutnya pada addData(), berfungsi untuk menambahkan data sesuai opsi yang telah diinputkan, apakah data termasuk ke dalam data tim, pelatih atau pemain. Parameter yang digunakan berbeda sesuai dengan attribut data yang telah diinputkan ke dalam program.

```
public static void addData(String namatim, String lokasitim, int tahunbentuk) throws IOException{
    TimBasket tb = new TimBasket(namatim, lokasitim, tahunbentuk);
    tim.add(e:tb);

    System.out.println(x:"=====");
    System.out.println(x:"Data Tim Berhasil Ditambahkan");
    System.out.println(x:"=====");
}

public static void addData(String namapelatih, int lamamelatih, String namatim, String lokasitim, int tahunbentuk) throws IOException{
    datapelatih dpel = new datapelatih(namatim, lokasitim, tahunbentuk, namapelatih, lamamelatih);
    pelatih.add(e:dpel);

    System.out.println(x:"=====");
    System.out.println(x:"Data Pelatih Berhasil Ditambahkan");
    System.out.println(x:"=====");
}

public static void addData(String nama, String posisi, int nopung, int umur, String namatim, String lokasitim, int tahunbentuk) throws IOException{
    datapemain dp = new datapemain(namatim, lokasitim, tahunbentuk, nama, umur, nopung, posisi);
    pemain.add(e:dp);

    System.out.println(x:"=====");
    System.out.println(x:"Data Pemain Berhasil Ditambahkan");
    System.out.println(x:"=====");
}
```

- Overriding

Polymorphism selanjutnya adalah overriding, yang diterapkan untuk menampilkan data yang ada di dalam program, apakah itu data pelatih, pemain atau data dari tim basket. Overriding ini terdapat di dalam beberapa file, yaitu file datapemain, datapelatih dan timbasket.

```
@Override
public void show(){
    System.out.println("Nama Pemain = " + this.getNama());
    System.out.println("Posisi Pemain = " + this.getPosisi());
    System.out.println("Nomor Punggung = " + this.getNoPung());
    System.out.println("Umur Pemain = " + this.getUmur());
    System.out.println("Nama Tim = " + this.getNamatim());
    System.out.println("Lokasi Tim = " + this.getLokasiTim());
    System.out.println("Tahun Tim Terbentuk = " + this.getTahunBentuk());
}
```

```
else{
    System.out.println(x:"=====");
    System.out.println(x:"Data Pemain Basket");
    System.out.println(x:"=====");
    for(int i = 0; i < pemain.size(); i++){
        System.out.print(i+1 + ". ");
        pemain.get(index:i).show();
        System.out.println(x:"=====");
    }
}
```