Posstest 4

Nama : Muhamad Alfarizi Edytia

Nim: 2309106105

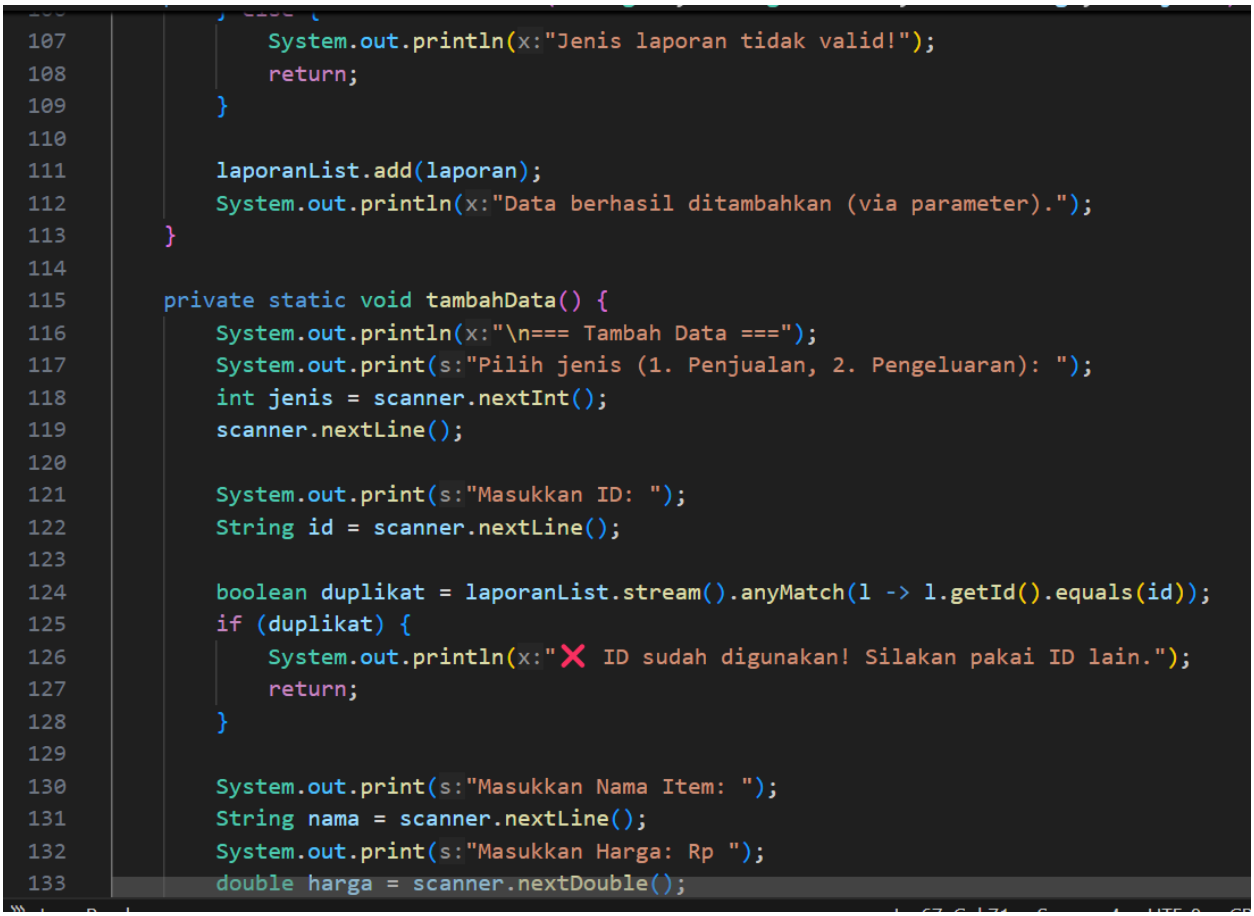
Kelas: C1 23

```java
import java.util.ArrayList;
import java.util.Iterator;
import java.util.Scanner;

abstract class Laporan {
    private String id;
    private String namaItem;
    private double harga;

    public Laporan(String id, String namaItem, double harga) {
        this.id = id;
        this.namaItem = namaItem;
        this.harga = harga;
    }

    public String getId() {
        return id;
    }

    public String getNamaItem() {
        return namaItem;
    }

    public void setNamaItem(String namaItem) {
        this.namaItem = namaItem;
    }

    public double getHarga() {
        return harga;
```

```java
    public double getHarga() {
        return harga;
    }

    public void setHarga(double harga) {
        this.harga = harga;
    }

    public abstract String getJenis();

    @Override
    public String toString() {
        return "ID: " + id + " | Jenis: " + getJenis() + " | Item: " + namaItem + " | Harga: Rp " + harga;
    }
}

class LaporanPenjualan extends Lapo String id - LaporanPenjualan.LaporanPenjualan(String, String, double)
    public LaporanPenjualan(String id, String namaItem, double harga) {
        super(id, namaItem, harga);
    }

    @Override
    public String getJenis() {
        return "Penjualan";
    }
}

class LaporanPengeluaran extends Laporan {
```

```java
class LaporanPengeluaran extends Laporan {
    public LaporanPengeluaran(String id, String namaItem, double harga) {
        super(id, namaItem, harga);
    }

    @Override
    public String getJenis() {
        return "Pengeluaran";
    }
}

public class MainLaporan {
    private static ArrayList<Laporan> laporanList = new ArrayList<>();
    private static Scanner scanner = new Scanner(System.in);

    Run | Debug
    public static void main(String[] args) {
        while (true) {
            tampilkanMenu();
            int pilihan = scanner.nextInt();
            scanner.nextLine();

            switch (pilihan) {
                case 1 -> tambahData();
                case 2 -> tampilkanData();
                case 3 -> ubahData();
                case 4 -> hapusData();
                case 5 -> {
                    System.out.println(x:"Keluar dari program...");
```

```
81                  case 5 -> {
82                      System.out.println(x:"Keluar dari program...");
83                      System.exit(status:0);
84                  }
85                  default -> System.out.println(x:"Pilihan tidak valid!");
86              }
87          }
88      }
89
90      private static void tampilkanMenu() {
91          System.out.println(x:"\n=== Menu Sistem Laporan Finansial Cafe Syafar ===");
92          System.out.println(x:"1. Tambah Data");
93          System.out.println(x:"2. Tampilkan Data");
94          System.out.println(x:"3. Ubah Data");
95          System.out.println(x:"4. Hapus Data");
96          System.out.println(x:"5. Keluar");
97          System.out.print(s:"Pilih menu (1-5): ");
98      }
99
100     public static void tambahData(String id, String namaItem, double harga, int jenis) {
101         Laporan laporan;
102         if (jenis == 1) {
103             laporan = new LaporanPenjualan(id, namaItem, harga);
104         } else if (jenis == 2) {
105             laporan = new LaporanPengeluaran(id, namaItem, harga);
106         } else {
107             System.out.println(x:"Jenis laporan tidak valid!");
```



```
            System.out.println(x:"Jenis laporan tidak valid!");
            return;
        }

        laporanList.add(laporan);
        System.out.println(x:"Data berhasil ditambahkan (via parameter).");
    }

    private static void tambahData() {
        System.out.println(x:"\n=== Tambah Data ===");
        System.out.print(s:"Pilih jenis (1. Penjualan, 2. Pengeluaran): ");
        int jenis = scanner.nextInt();
        scanner.nextLine();

        System.out.print(s:"Masukkan ID: ");
        String id = scanner.nextLine();

        boolean duplikat = laporanList.stream().anyMatch(l -> l.getId().equals(id));
        if (duplikat) {
            System.out.println(x:"❌ ID sudah digunakan! Silakan pakai ID lain.");
            return;
        }

        System.out.print(s:"Masukkan Nama Item: ");
        String nama = scanner.nextLine();
        System.out.print(s:"Masukkan Harga: Rp ");
        double harga = scanner.nextDouble();
```

```
115     private static void tambahData() {
133         double harga = scanner.nextDouble();
134         scanner.nextLine();
135 
136         tambahData(id, nama, harga, jenis);
137     }
138 
139     private static void tampilkanData() {
140         System.out.println(x:"\n=== Tampilkan Data ===");
141         if (laporanList.isEmpty()) {
142             System.out.println(x:"Belum ada data laporan.");
143         } else {
144             for (Laporan l : laporanList) {
145                 System.out.println(l);
146             }
147         }
148     }
149 
150     private static void ubahData() {
151         System.out.println(x:"\n=== Ubah Data ===");
152         System.out.print(s:"Masukkan ID yang ingin diubah: ");
153         String id = scanner.nextLine();
154 
155         for (Laporan l : laporanList) {
156             if (l.getId().equals(id)) {
157                 System.out.print(s:"Nama Item baru: ");
158                 String namaBaru = scanner.nextLine();
159                 System.out.print(s:"Harga baru: Rp ");
```

```
150     private static void ubahData() {
160                 double hargaBaru = scanner.nextDouble();
161                 scanner.nextLine();
162
163                 l.setNamaItem(namaBaru);
164                 l.setHarga(hargaBaru);
165                 System.out.println(x:"✅ Data berhasil diubah.");
166                 return;
167             }
168         }
169
170         System.out.println(x:"❌ ID tidak ditemukan.");
171     }
172
173     private static void hapusData() {
174         System.out.println(x:"\n=== Hapus Data ===");
175         System.out.print(s:"Masukkan ID yang ingin dihapus: ");
176         String id = scanner.nextLine();
177
178         Iterator<Laporan> it = laporanList.iterator();
179         while (it.hasNext()) {
180             if (it.next().getId().equals(id)) {
181                 it.remove();
182                 System.out.println(x:"✅ Data berhasil dihapus.");
183                 return;
184             }
185         }
186
```