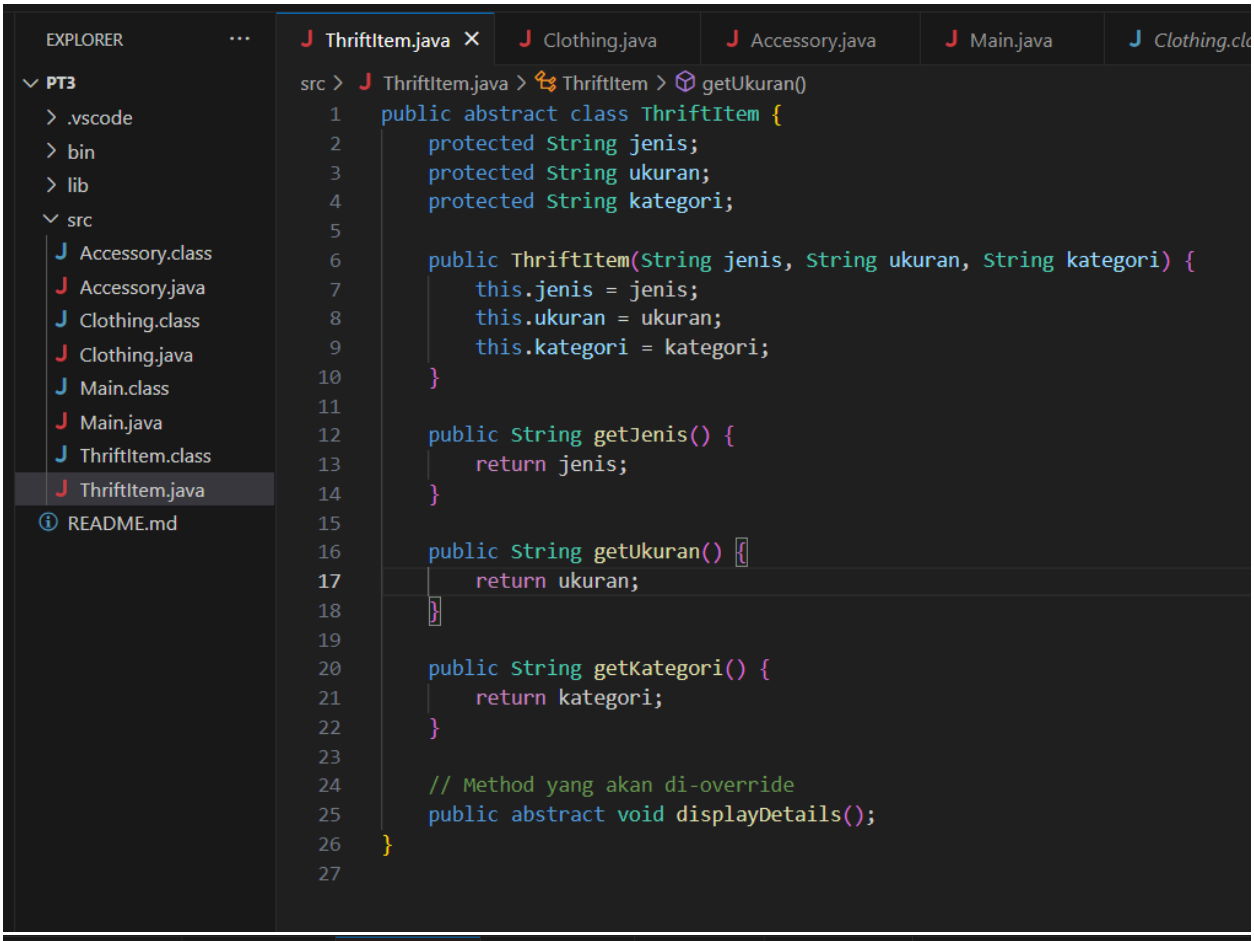
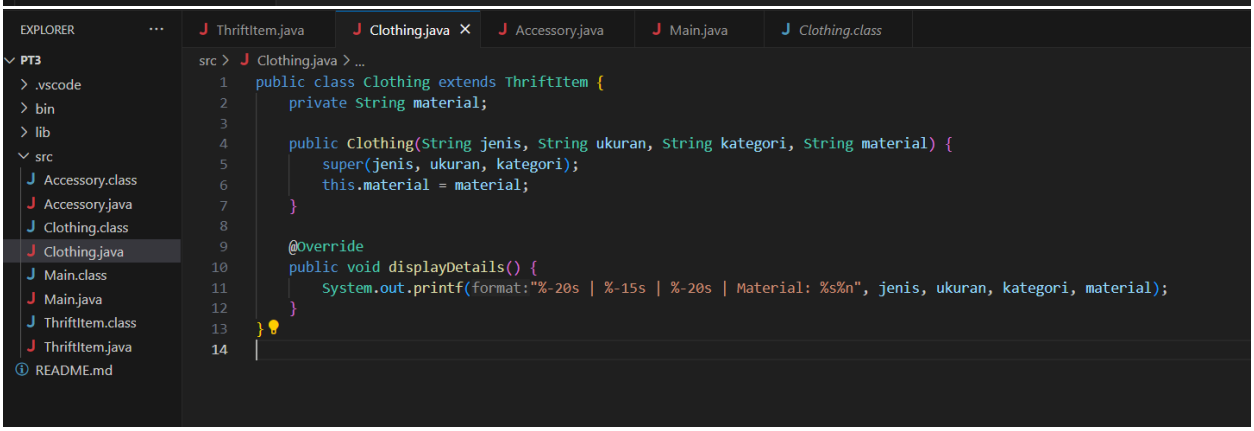
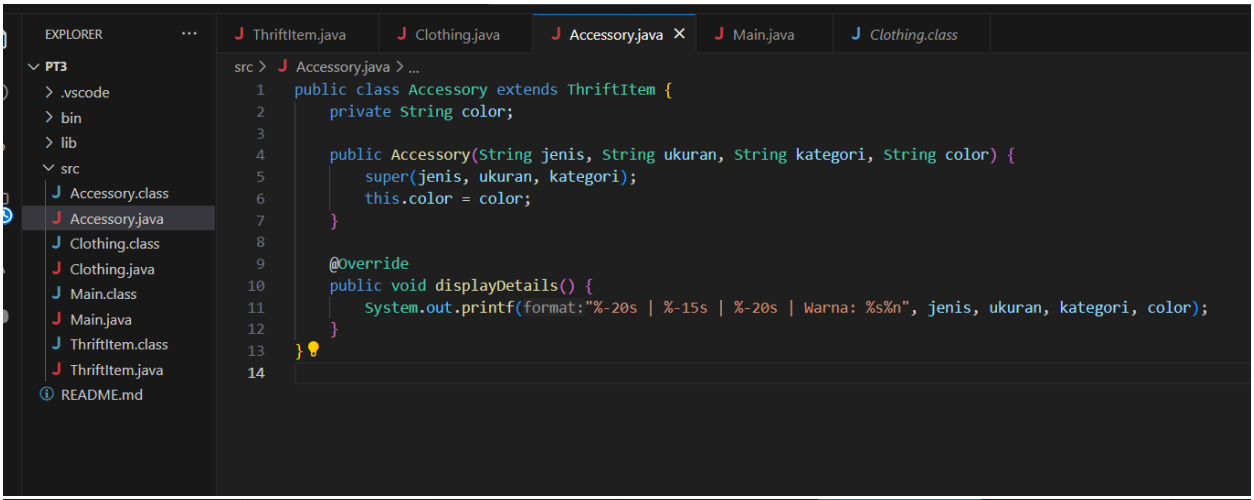
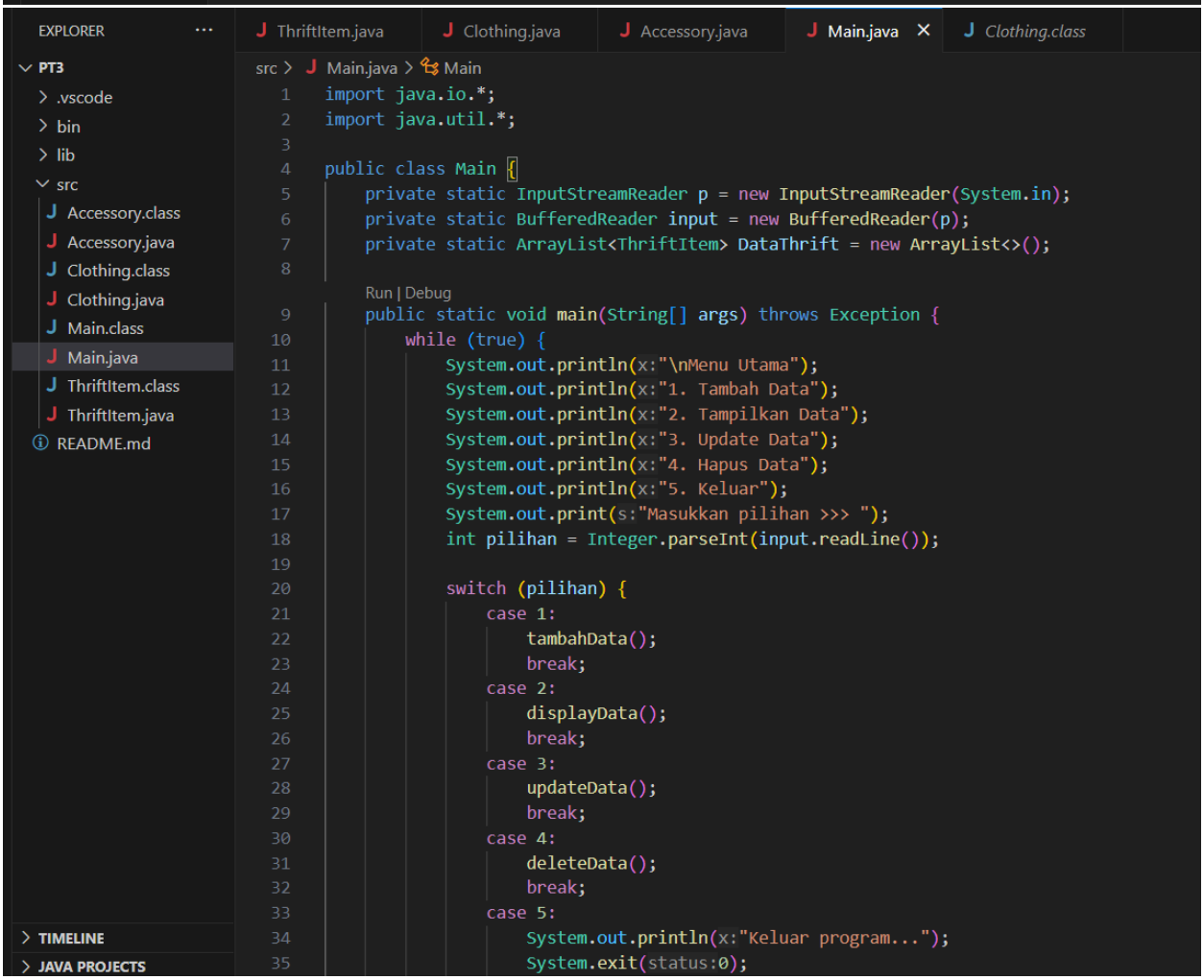
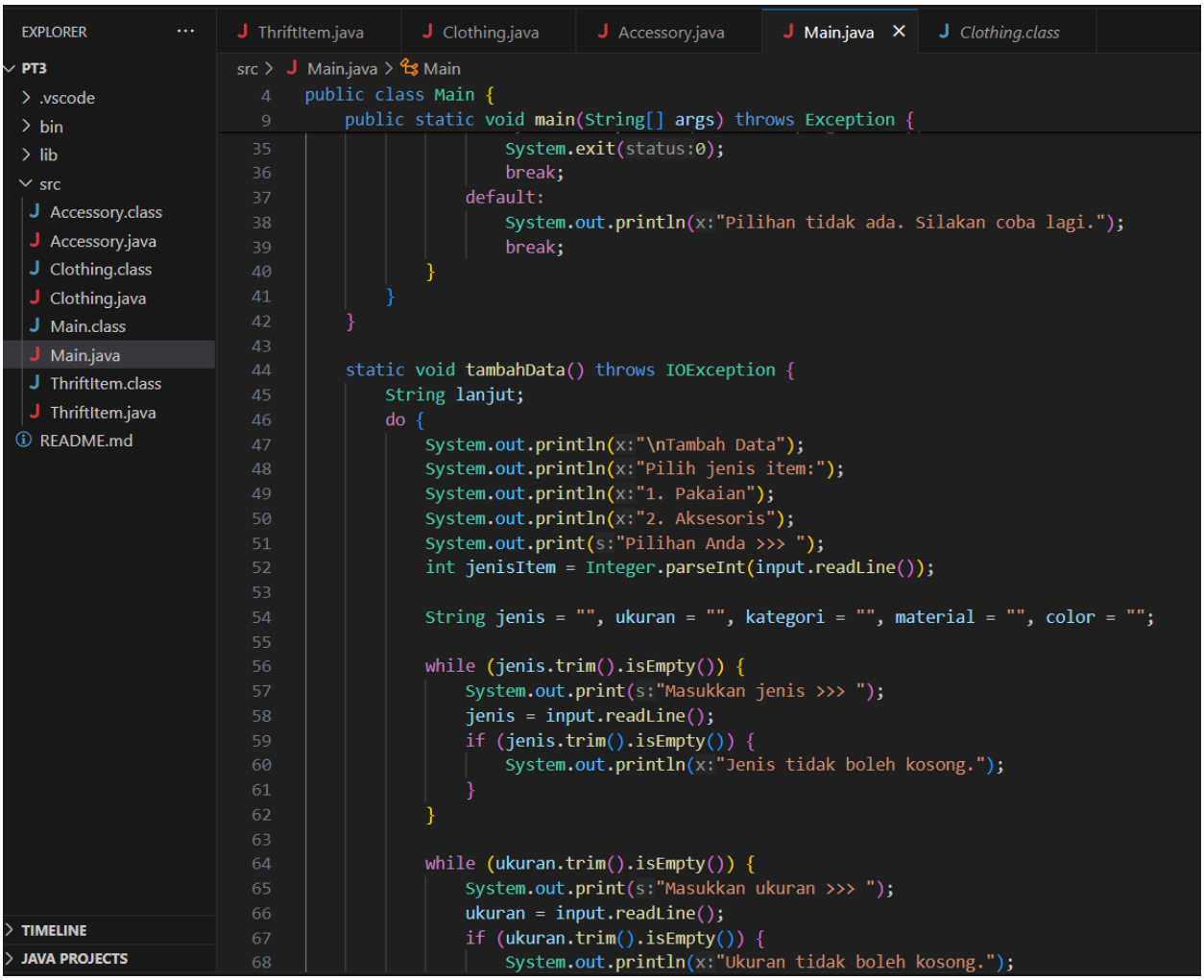
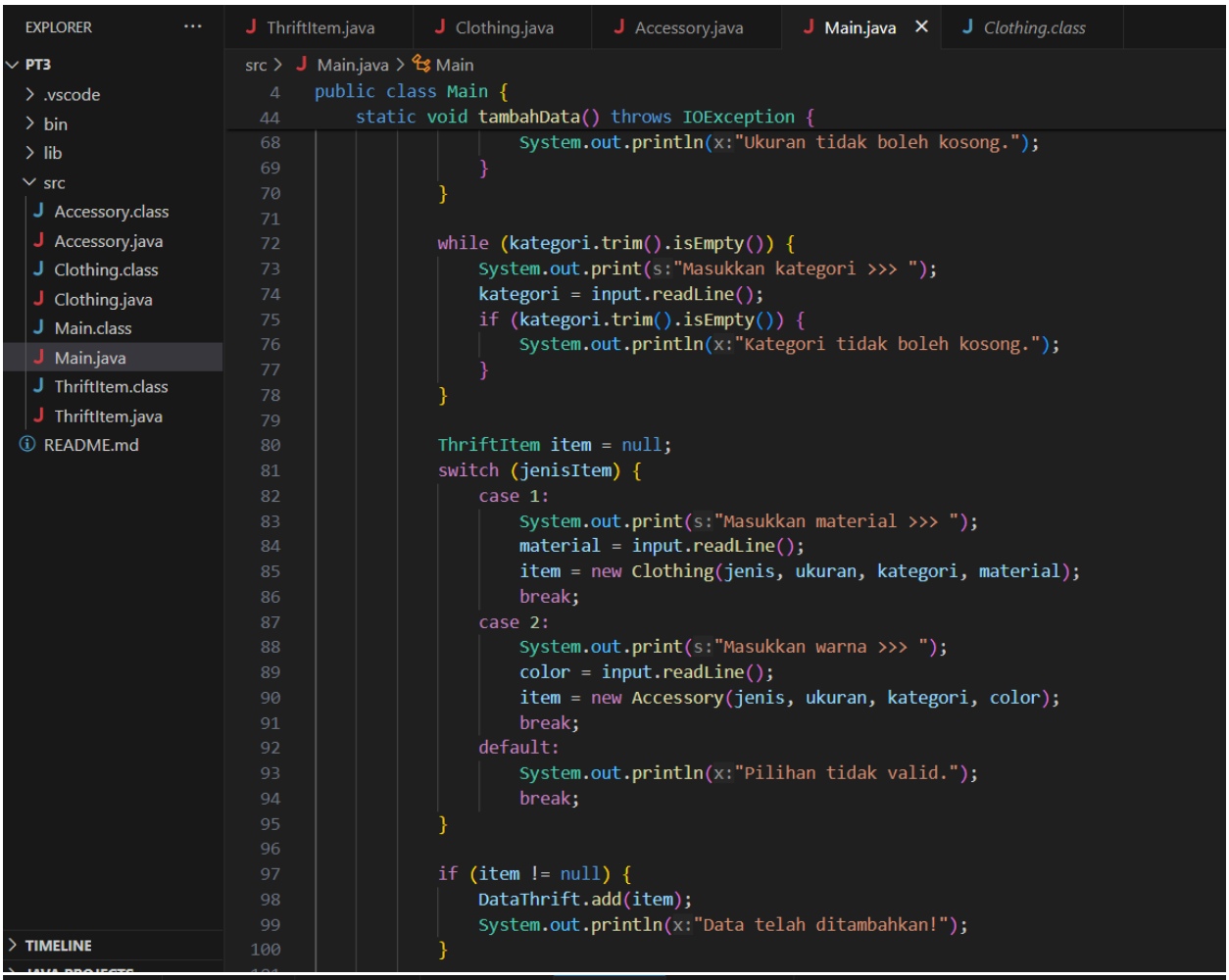

EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
ThriftItem.java
Clothing.java
Accessory.java
Main.java
src > ThriftItem.java > ThriftItem > getUkuran()
public abstract class ThriftItem {
protected String jenis;
protected String ukuran;
protected String kategori;
public ThriftItem(String jenis, String ukuran, String kategori) {
this.jenis = jenis;
this.ukuran = ukuran;
this.kategori = kategori;
}
public String getJenis() {
return jenis;
}
public String getUkuran() {
return ukuran;
}
public String getKategori() {
return kategori;
}
// Method yang akan di-override
public abstract void displayDetails();
}

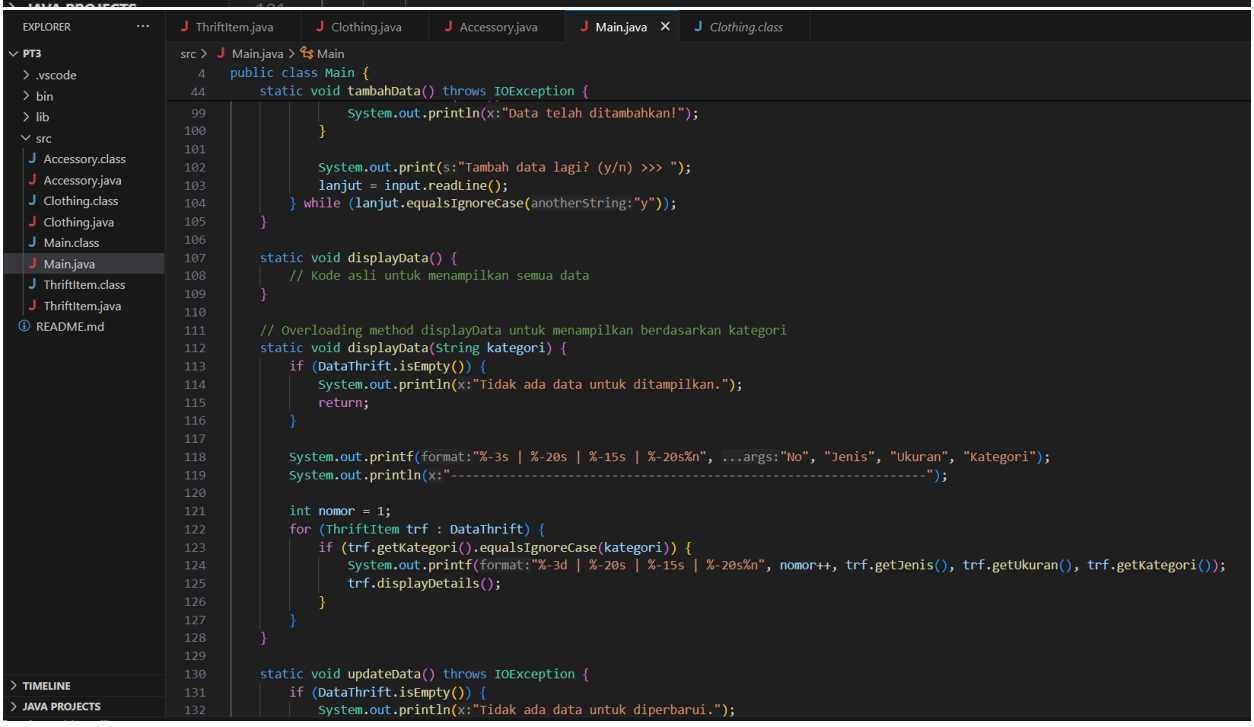

EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
ThriftItem.java
Clothing.java
Accessory.java
Main.java
src > Clothing.java > ...
public class Clothing extends ThriftItem {
private String material;
public Clothing(String jenis, String ukuran, String kategori, String material) {
super(jenis, ukuran, kategori);
this.material = material;
}
@Override
public void displayDetails() {
System.out.printf(format:"%-20s | %-15s | %-20s | Material: %s%n", jenis, ukuran, kategori, material);
}
}

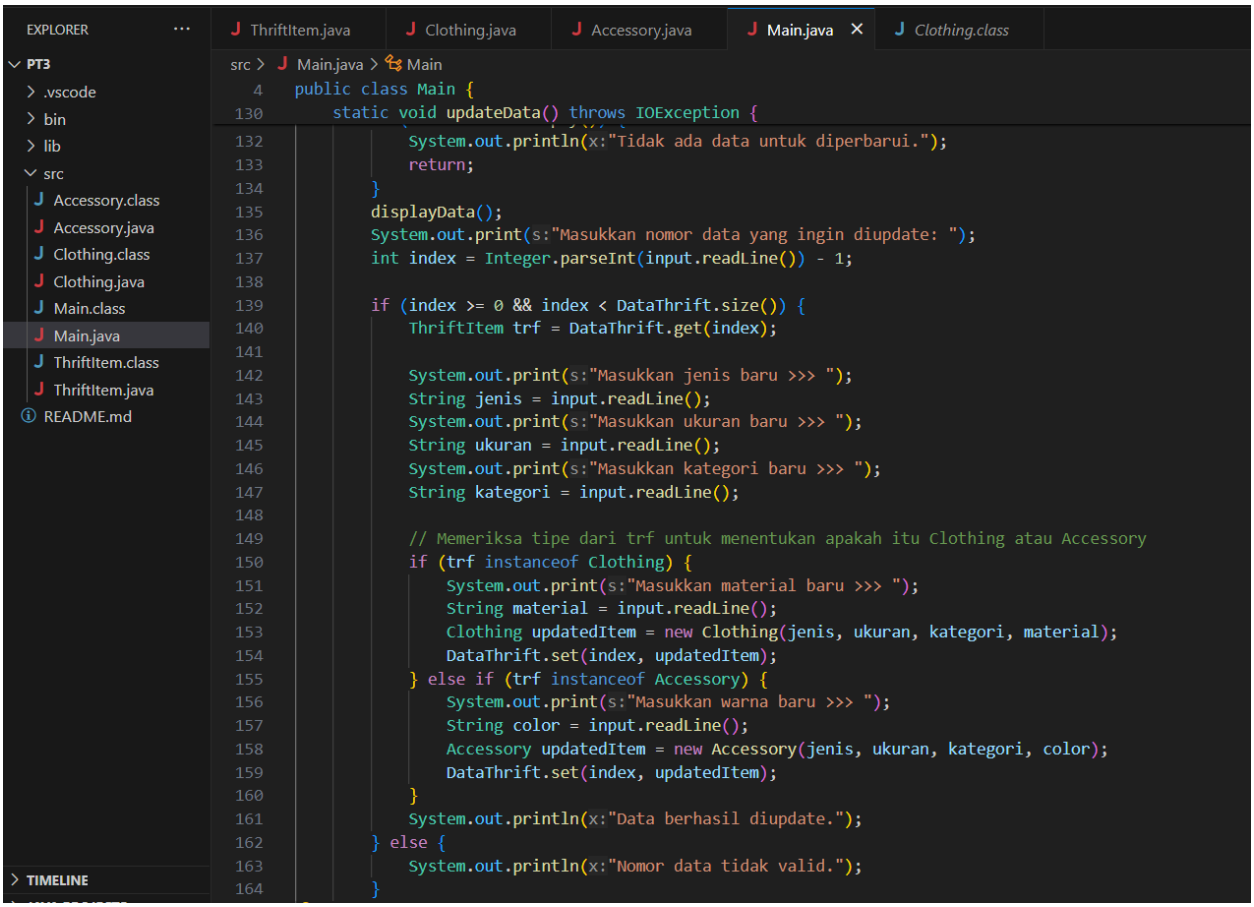

EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
src > Accessory.java > ...
public class Accessory extends ThriftItem {
private String color;
public Accessory(String jenis, String ukuran, String kategori, String color) {
super(jenis, ukuran, kategori);
this.color = color;
}
@Override
public void displayDetails() {
System.out.printf(format:"%-20s | %-15s | %-20s | Warna: %s%n", jenis, ukuran, kategori, color);
}
}

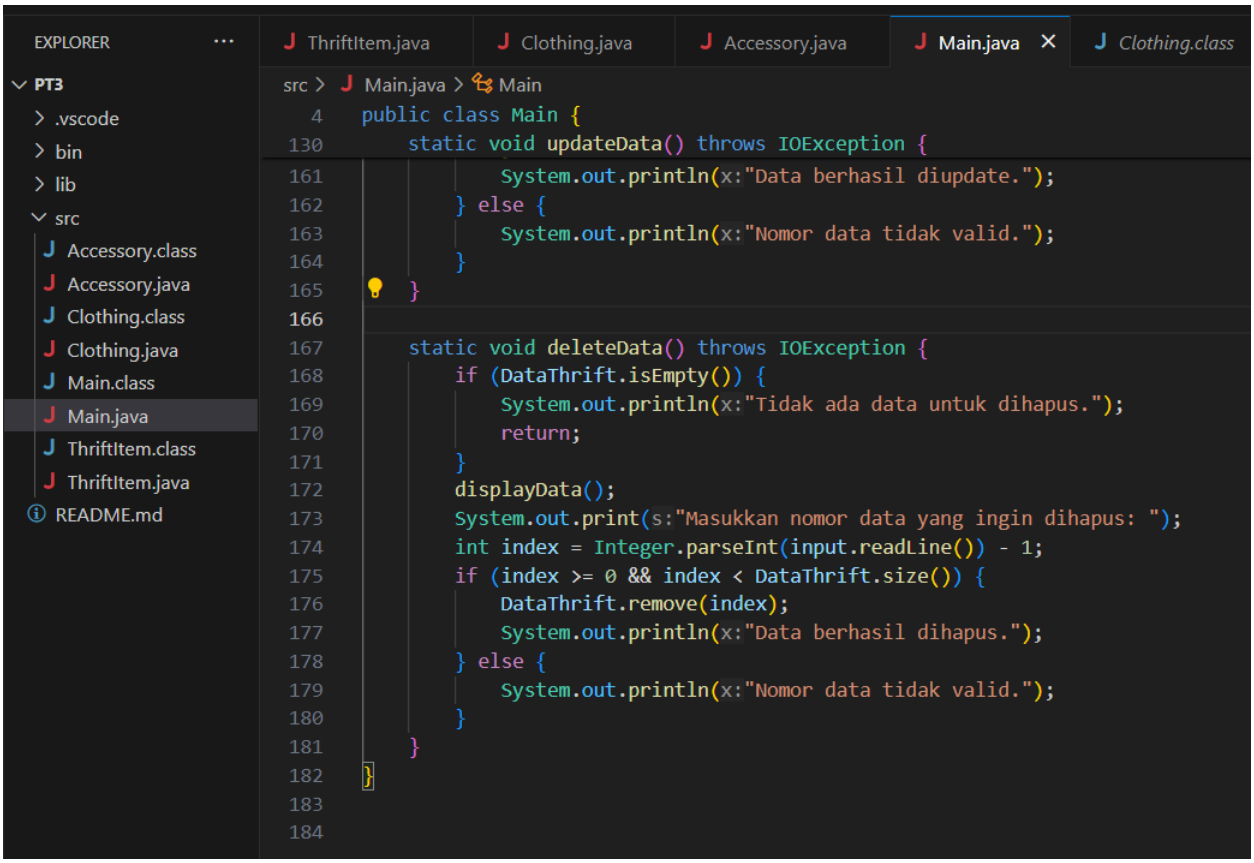

EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
TIMELINE
JAVA PROJECTS
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
src > Main.java > Main
import java.io.*;
import java.util.*;
public class Main {
private static InputStreamReader p = new InputStreamReader(System.in);
private static BufferedReader input = new BufferedReader(p);
private static ArrayList<ThriftItem> DataThrift = new ArrayList<>();
Run | Debug
public static void main(String[] args) throws Exception {
while (true) {
System.out.println(x:"\nMenu Utama");
System.out.println(x:"1. Tambah Data");
System.out.println(x:"2. Tampilkan Data");
System.out.println(x:"3. Update Data");
System.out.println(x:"4. Hapus Data");
System.out.println(x:"5. Keluar");
System.out.print(s:"Masukkan pilihan >>> ");
int pilihan = Integer.parseInt(input.readLine());
switch (pilihan) {
case 1:
tambahData();
break;
case 2:
displayData();
break;
case 3:
updateData();
break;
case 4:
deleteData();
break;
case 5:
System.out.println(x:"Keluar program...");
System.exit(status:0);


EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
TIMELINE
JAVA PROJECTS
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
src > Main.java > Main
public class Main {
public static void main(String[] args) throws Exception {
System.exit(status:0);
break;
default:
System.out.println(x:"Pilihan tidak ada. Silakan coba lagi.");
break;
}
}
}
static void tambahData() throws IOException {
String lanjut;
do {
System.out.println(x:"\nTambah Data");
System.out.println(x:"Pilih jenis item:");
System.out.println(x:"1. Pakaian");
System.out.println(x:"2. Aksesoris");
System.out.print(s:"Pilihan Anda >>> ");
int jenisItem = Integer.parseInt(input.readLine());
String jenis = "", ukuran = "", kategori = "", material = "", color = "";
while (jenis.trim().isEmpty()) {
System.out.print(s:"Masukkan jenis >>> ");
jenis = input.readLine();
if (jenis.trim().isEmpty()) {
System.out.println(x:"Jenis tidak boleh kosong.");
}
}
while (ukuran.trim().isEmpty()) {
System.out.print(s:"Masukkan ukuran >>> ");
ukuran = input.readLine();
if (ukuran.trim().isEmpty()) {
System.out.println(x:"Ukuran tidak boleh kosong.");


EXPLORER
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
TIMELINE
src > Main.java > Main
public class Main {
static void tambahData() throws IOException {
System.out.println(x:"Ukuran tidak boleh kosong.");
}
}
while (kategori.trim().isEmpty()) {
System.out.print(s:"Masukkan kategori >>> ");
kategori = input.readLine();
if (kategori.trim().isEmpty()) {
System.out.println(x:"Kategori tidak boleh kosong.");
}
}
ThriftItem item = null;
switch (jenisItem) {
case 1:
System.out.print(s:"Masukkan material >>> ");
material = input.readLine();
item = new Clothing(jenis, ukuran, kategori, material);
break;
case 2:
System.out.print(s:"Masukkan warna >>> ");
color = input.readLine();
item = new Accessory(jenis, ukuran, kategori, color);
break;
default:
System.out.println(x:"Pilihan tidak valid.");
break;
}
if (item != null) {
DataThrift.add(item);
System.out.println(x:"Data telah ditambahkan!");
}


EXPLORER
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
TIMELINE
JAVA PROJECTS
src > Main.java > Main
public class Main {
static void tambahData() throws IOException {
System.out.println(x:"Data telah ditambahkan!");
}
System.out.print(s:"Tambah data lagi? (y/n) >>> ");
lanjut = input.readLine();
} while (lanjut.equalsIgnoreCase(anotherString:"y"));
}
static void displayData() {
// Kode asli untuk menampilkan semua data
}
// Overloading method displayData untuk menampilkan berdasarkan kategori
static void displayData(String kategori) {
if (DataThrift.isEmpty()) {
System.out.println(x:"Tidak ada data untuk ditampilkan.");
return;
}
System.out.printf(format:"%-3s | %-20s | %-15s | %-20s%n", ...args:"NO", "Jenis", "Ukuran", "Kategori");
System.out.println(x:"----------------------------------------------------------------");
int nomor = 1;
for (ThriftItem trf : DataThrift) {
if (trf.getKategori().equalsIgnoreCase(kategori)) {
System.out.printf(format:"%-3d | %-20s | %-15s | %-20s%n", nomor++, trf.getJenis(), trf.getUkuran(), trf.getKategori());
trf.displayDetails();
}
}
}
static void updateData() throws IOException {
if (DataThrift.isEmpty()) {
System.out.println(x:"Tidak ada data untuk diperbarui.");


EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
TIMELINE
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
src > Main.java > Main
4 public class Main {
130 static void updateData() throws IOException {
132 System.out.println(x:"Tidak ada data untuk diperbarui.");
133 return;
134 }
135 displayData();
136 System.out.print(s:"Masukkan nomor data yang ingin diupdate: ");
137 int index = Integer.parseInt(input.readLine()) - 1;
138
139 if (index >= 0 && index < DataThrift.size()) {
140 ThriftItem trf = DataThrift.get(index);
141
142 System.out.print(s:"Masukkan jenis baru >>> ");
143 String jenis = input.readLine();
144 System.out.print(s:"Masukkan ukuran baru >>> ");
145 String ukuran = input.readLine();
146 System.out.print(s:"Masukkan kategori baru >>> ");
147 String kategori = input.readLine();
148
149 // Memeriksa tipe dari trf untuk menentukan apakah itu Clothing atau Accessory
150 if (trf instanceof Clothing) {
151 System.out.print(s:"Masukkan material baru >>> ");
152 String material = input.readLine();
153 Clothing updatedItem = new Clothing(jenis, ukuran, kategori, material);
154 DataThrift.set(index, updatedItem);
155 } else if (trf instanceof Accessory) {
156 System.out.print(s:"Masukkan warna baru >>> ");
157 String color = input.readLine();
158 Accessory updatedItem = new Accessory(jenis, ukuran, kategori, color);
159 DataThrift.set(index, updatedItem);
160 }
161 System.out.println(x:"Data berhasil diupdate.");
162 } else {
163 System.out.println(x:"Nomor data tidak valid.");
164 }


EXPLORER
PT3
.vscode
bin
lib
src
Accessory.class
Accessory.java
Clothing.class
Clothing.java
Main.class
Main.java
ThriftItem.class
ThriftItem.java
README.md
ThriftItem.java
Clothing.java
Accessory.java
Main.java
Clothing.class
src > Main.java > Main
4 public class Main {
130 static void updateData() throws IOException {
161 System.out.println(x:"Data berhasil diupdate.");
162 } else {
163 System.out.println(x:"Nomor data tidak valid.");
164 }
165 }
166
167 static void deleteData() throws IOException {
168 if (DataThrift.isEmpty()) {
169 System.out.println(x:"Tidak ada data untuk dihapus.");
170 return;
171 }
172 displayData();
173 System.out.print(s:"Masukkan nomor data yang ingin dihapus: ");
174 int index = Integer.parseInt(input.readLine()) - 1;
175 if (index >= 0 && index < DataThrift.size()) {
176 DataThrift.remove(index);
177 System.out.println(x:"Data berhasil dihapus.");
178 } else {
179 System.out.println(x:"Nomor data tidak valid.");
180 }
181 }
182 }
183
184