

## App.Java

```
1 import CRUD.PelangganCRUD;
2 import CRUD.GameCRUD;
3 import CRUD.ProdukCRUD;
4 import sigmastore.pelanggan;
5 import sigmastore.game;
6 import sigmastore.produk;
7 import sigmastore.transaksi;
8
9 import java.util.Scanner;
10
11 public class App {
12     static Scanner scanner = new Scanner(System.in);
13     static PelangganCRUD pelangganCRUD = new PelangganCRUD();
14     static GameCRUD gameCRUD = new GameCRUD();
15     static ProdukCRUD produkCRUD = new ProdukCRUD();
16
17     public static void main(String[] args) {
18         while (true) {
19             System.out.println("\n===== SIGMASTORE MENU UTAMA =====");
20             System.out.println("1. CRUD Pelanggan");
21             System.out.println("2. CRUD Game");
22             System.out.println("3. CRUD Produk");
23             System.out.println("4. Buat Transaksi");
24             System.out.println("5. Lihat Total Transaksi");
25             System.out.println("6. Keluar");
26             System.out.print("Pilih menu: ");
27
28             int pilihan = scanner.nextInt();
29             scanner.nextLine();
30
31             switch (pilihan) {
32                 case 1:
33                     menuPelanggan();
34                     break;
35                 case 2:
36                     menuGame();
37                     break;
38                 case 3:
39                     menuProduk();
40                     break;
41                 case 4:
42                     menuTransaksi();
43                     break;
44                 case 5:
45                     System.out.println("Total transaksi yang telah dilakukan: " + transaksi.getTotalTransaksi());
46                     break;
47                 case 6:
48                     System.out.println("Program Off...");
49                     System.exit(0);
50                 default:
51                     System.out.println("Tidak valid! Coba lagi");
52             }
53         }
54     }
55
56     private static void menuPelanggan() {
57         while (true) {
58             System.out.println("\n===== MENU CRUD PELANGGAN =====");
59             System.out.println("1. Tambah Pelanggan");
60             System.out.println("2. Lihat Pelanggan");
61             System.out.println("3. Ubah Pelanggan");
62             System.out.println("4. Hapus Pelanggan");
63             System.out.println("5. Kembali ke Menu Utama");
64             System.out.print("Pilih menu: ");
65
66             int pilihan = scanner.nextInt();
67             scanner.nextLine();
68
69             switch (pilihan) {
70                 case 1:
71                     System.out.print("Masukkan ID Pelanggan: ");
72                     String IDpelanggan = scanner.nextLine();
73                     System.out.print("Masukkan Nama Pelanggan: ");
74                     String Namapelanggan = scanner.nextLine();
75                     pelangganCRUD.tambah(new pelanggan(IDpelanggan, Namapelanggan));
76                     break;
77                 case 2:
78                     pelangganCRUD.lihat();
79                     break;
80                 case 3:
81                     System.out.print("Masukkan ID Pelanggan yang ingin diubah: ");
82                     String IDeditPelanggan = scanner.nextLine();
83                     System.out.print("Masukkan Nama Baru: ");
84                     String NamaBaru = scanner.nextLine();
85                     pelangganCRUD.edit(IDeditPelanggan, NamaBaru);
86                     break;
87                 case 4:
88                     System.out.print("Masukkan ID Pelanggan yang ingin dihapus: ");
89                     String IDhapusPelanggan = scanner.nextLine();
90                     pelangganCRUD.hapus(IDhapusPelanggan, pelanggan::getIDpelanggan);
91                     break;
92                 case 5:
93                     return;
94                 default:
95                     System.out.println("Tidak valid! Coba lagi");
96             }
97         }
98     }
99 }
```

```

100 private static void menuGame() {
101     while (true) {
102         System.out.println("\n===== MENU CRUD GAME =====");
103         System.out.println("1. Tambah Game");
104         System.out.println("2. Lihat Game");
105         System.out.println("3. Ubah Game");
106         System.out.println("4. Hapus Game");
107         System.out.println("5. Kembali ke Menu Utama");
108         System.out.print("Pilih menu: ");
109
110         int pilihan = scanner.nextInt();
111         scanner.nextLine();
112
113         switch (pilihan) {
114             case 1:
115                 System.out.print("Masukkan ID Game: ");
116                 String IDgame = scanner.nextLine();
117                 System.out.print("Masukkan Nama Game: ");
118                 String NamaGame = scanner.nextLine();
119                 System.out.print("Masukkan Developer Game: ");
120                 String Devgame = scanner.nextLine();
121                 game g = new game(IDgame, NamaGame, Devgame);
122                 gameCRUD.tambah(g);
123                 System.out.println("Deskripsi Singkat: " + g.deskripsisingkat());
124                 System.out.println("Deskripsi Lengkap: " + g.deskripsilengkap());
125                 break;
126             case 2:
127                 gameCRUD.lihat();
128                 break;
129             case 3:
130                 System.out.print("Masukkan ID Game yang ingin diubah: ");
131                 String IDeditGame = scanner.nextLine();
132                 System.out.print("Masukkan Nama Baru: ");
133                 String NamaGameBaru = scanner.nextLine();
134                 System.out.print("Masukkan Developer Baru: ");
135                 String DevGameBaru = scanner.nextLine();
136                 gameCRUD.edit(IDeditGame, NamaGameBaru, DevGameBaru);
137                 break;
138             case 4:
139                 System.out.print("Masukkan ID Game yang ingin dihapus: ");
140                 String IDhapusGame = scanner.nextLine();
141                 gameCRUD.hapus(IDhapusGame, game::getIDgame);
142                 break;
143             case 5:
144                 return;
145             default:
146                 System.out.println("Tidak valid! Coba lagi!");
147         }
148     }
149 }
150
151 private static void menuProduk() {
152     while (true) {
153         System.out.println("\n===== MENU CRUD PRODUK =====");
154         System.out.println("1. Tambah Produk");
155         System.out.println("2. Lihat Produk");
156         System.out.println("3. Ubah Produk");
157         System.out.println("4. Hapus Produk");
158         System.out.println("5. Kembali ke Menu Utama");
159         System.out.print("Pilih menu: ");
160
161         int pilihan = scanner.nextInt();
162         scanner.nextLine();
163
164         switch (pilihan) {
165             case 1:
166                 System.out.print("Masukkan ID Produk: ");
167                 String IDProduk = scanner.nextLine();
168                 System.out.print("Masukkan Stok: ");
169                 int stok = scanner.nextInt();
170                 scanner.nextLine();
171                 System.out.print("Masukkan Jenis Produk (diamond/item): ");
172                 String JenisProdukStr = scanner.nextLine().toUpperCase();
173                 System.out.print("Masukkan Harga: ");
174                 int harga = scanner.nextInt();
175                 scanner.nextLine();
176                 System.out.print("Masukkan ID Game: ");
177                 String IDgame = scanner.nextLine();
178                 try {
179                     produk.JenisProduk jenisProduk = produk.JenisProduk.valueOf(JenisProdukStr);
180                     produk p =
181                         if (jenisProduk == produk.JenisProduk.DIAMOND) {
182                             System.out.print("Masukkan Jumlah Diamond: ");
183                             int jumlahDiamond = scanner.nextInt();
184                             scanner.nextLine();
185                             p = new produk.diamond(IDProduk, stok, harga, IDgame, jumlahDiamond);
186                         } else if (jenisProduk == produk.JenisProduk.ITEM) {
187                             System.out.print("Masukkan Durasi Item: ");
188                             String durasiItem = scanner.nextLine();
189                             p = new produk.item(IDProduk, stok, harga, IDgame, durasiItem);
190                         } else {
191                             throw new IllegalArgumentException("Jenis Produk tidak dikenali!");
192                         }
193                     produkCRUD.tambah(p);
194                 } catch (IllegalArgumentException e) {
195                     System.out.println("Error: Jenis Produk harus 'diamond' atau 'item'!");
196                 }
197                 break;
198             case 2:
199                 System.out.println("\n===== LIHAT PRODUK =====");
200                 System.out.println("1. Lihat Semua Produk");
201                 System.out.println("2. Lihat Produk Diamond");
202                 System.out.println("3. Lihat Produk Item");
203                 System.out.print("Pilih opsi: ");
204                 int opsiLihat = scanner.nextInt();
205                 scanner.nextLine();
206                 switch (opsiLihat) {
207                     case 1:
208                         produkCRUD.lihat();
209                         break;
210                     case 2:
211                         produkCRUD.lihatByJenis(produk.JenisProduk.DIAMOND);
212                         break;
213                     case 3:
214                         produkCRUD.lihatByJenis(produk.JenisProduk.ITEM);
215                         break;
216                     default:
217                         System.out.println("Opsi tidak valid!");
218                 }
219                 break;
220             case 3:
221                 System.out.print("Masukkan ID Produk yang ingin diubah: ");
222                 String IDeditProduk = scanner.nextLine();
223                 System.out.print("Masukkan Stok Baru: ");
224                 int stokBaru = scanner.nextInt();
225                 scanner.nextLine();
226                 System.out.print("Masukkan Harga Baru: ");
227                 double hargaBaru = scanner.nextDouble();
228                 scanner.nextLine();
229                 produkCRUD.edit(IDeditProduk, stokBaru, hargaBaru);
230                 break;
231             case 4:
232                 System.out.print("Masukkan ID Produk yang ingin dihapus: ");
233                 String IDhapusProduk = scanner.nextLine();
234                 produkCRUD.hapus(IDhapusProduk, produk::getIDProduk);
235                 break;
236             case 5:
237                 return;
238             default:
239                 System.out.println("Tidak valid! Coba lagi!");
240         }
241     }
242 }

```

```

243
244     private static void menuTransaksi() {
245         System.out.print("Masukkan ID Transaksi: ");
246         String IDtransaksi = scanner.nextLine();
247         System.out.print("Masukkan ID Pelanggan: ");
248         String IDpelanggan = scanner.nextLine();
249         System.out.print("Masukkan ID Produk: ");
250         String IDproduk = scanner.nextLine();
251         System.out.print("Masukkan Total Harga: ");
252         double totalHarga = scanner.nextDouble();
253         scanner.nextLine();
254
255         // Membuat objek transaksi baru
256         transaksi t = new transaksi(IDtransaksi, IDpelanggan, IDproduk, totalHarga);
257         System.out.println("Transaksi berhasil dibuat: " + t);
258     }
259 }
260

```

crud.java

```

1  package CRUD;
2  import java.util.ArrayList;
3
4  public class crud<T> {
5      protected ArrayList<T> list = new ArrayList<>();
6
7      public void tambah(T item) {
8          list.add(item);
9          System.out.println("Data berhasil ditambahkan!");
10     }
11
12     public void lihat() {
13         if (list.isEmpty()) {
14             System.out.println("Belum ada data yang terdaftar.");
15         } else {
16             System.out.println("\nDaftar Data:");
17             for (T item : list) {
18                 System.out.println(item);
19             }
20         }
21     }
22
23     public void hapus(String id, Identifiable<T> identifier) {
24         for (T item : list) {
25             if (identifier.identify(item).equals(id)) {
26                 list.remove(item);
27                 System.out.println("Data berhasil dihapus!");
28                 return;
29             }
30         }
31         System.out.println("Error: Data tidak ditemukan.");
32     }
33
34     public interface Identifiable<T> {
35         String identify(T item);
36     }
37 }
38

```

## GameCRUD.java

```
1 package CRUD;
2 import sigmastore.game;
3
4 public class GameCRUD extends crud<game> {
5     public void edit(String IDgame, String NamaGameBaru, String DevGameBaru) {
6         for (game g : list) {
7             if (g.getIDgame().equals(IDgame)) {
8                 g.setNamagame(NamaGameBaru);
9                 g.setDevgame(DevGameBaru);
10                System.out.println("Game berhasil diperbarui!");
11                return;
12            }
13        }
14        System.out.println("Error: Game tidak ditemukan.");
15    }
16 }
17
```

## PelangganCRUD.java

```
1 package CRUD;
2 import sigmastore.pelanggan;
3
4 public class PelangganCRUD extends crud<pelanggan> {
5     public void edit(String IDpelanggan, String NamaBaru) {
6         for (pelanggan p : list) {
7             if (p.getIDpelanggan().equals(IDpelanggan)) {
8                 p.setNamapelanggan(NamaBaru);
9                 System.out.println("Pelanggan berhasil diperbarui!");
10                return;
11            }
12        }
13        System.out.println("Error: Pelanggan tidak ditemukan.");
14    }
15 }
16
```

## ProdukCRUD.java

```
1 package CRUD;
2 import sigmastore.produk;
3
4 public class ProdukCRUD extends crud<produk> {
5
6     public void tambah(produk p) {
7         for (produk existingProduct : list) {
8             if (existingProduct.getIDproduk().equals(p.getIDproduk())) {
9                 if (existingProduct.getJenisProduk() == p.getJenisProduk()) {
10                     System.out.println("Error: Produk dengan ID " + p.getIDproduk() + " sudah ada dengan jenis produk sama.");
11                     return;
12                 }
13             }
14         }
15         list.add(p);
16         System.out.println("Data produk berhasil ditambahkan!");
17     }
18
19     public void edit(String IDproduk, int stokBaru, double hargaBaru) {
20         for (produk p : list) {
21             if (p.getIDproduk().equals(IDproduk)) {
22                 p.setStok(stokBaru);
23                 p.setHarga(hargaBaru);
24                 System.out.println("Produk berhasil diperbarui!");
25                 return;
26             }
27         }
28         System.out.println("Error: Produk tidak ditemukan.");
29     }
30
31     public void lihatByJenis(produk.JenisProduk jenis) {
32         boolean ditemukan = false;
33         for (produk p : list) {
34             if (p.getJenisProduk() == jenis) {
35                 if (p instanceof produk.diamond) {
36                     produk.diamond d = (produk.diamond) p;
37                     System.out.println("Produk Diamond: " + d.deskripsiProduk());
38                 } else if (p instanceof produk.item) {
39                     produk.item i = (produk.item) p;
40                     System.out.println("Produk Item: " + i.deskripsiProduk());
41                 }
42                 ditemukan = true;
43             }
44         }
45         if (!ditemukan) {
46             System.out.println("Tidak ada produk dengan jenis " + jenis);
47         }
48     }
49 }
50
```

game.java

```
1 package sigmastore;
2
3 public class game implements Deskripsi {
4     private String IDgame;
5     private String Namagame;
6     private String Devgame;
7
8     public game(String IDgame, String Namagame, String Devgame) {
9         this.IDgame = IDgame;
10        this.Namagame = Namagame;
11        this.Devgame = Devgame;
12    }
13
14    public String getIDgame() {
15        return IDgame;
16    }
17
18    public String getNamagame() {
19        return Namagame;
20    }
21
22    public String getDevgame() {
23        return Devgame;
24    }
25
26    public void setNamagame(String Namagame) {
27        this.Namagame = Namagame;
28    }
29
30    public void setDevgame(String Devgame) {
31        this.Devgame = Devgame;
32    }
33
34    @Override
35    public String deskripsiSingkat() {
36        return Namagame + " dibuat oleh " + Devgame;
37    }
38
39    @Override
40    public String deskripsiLengkap() {
41        return "Game ID: " + IDgame + ", Nama: " + Namagame + ", Developer: " + Devgame;
42    }
43
44    @Override
45    public String toString() {
46        return deskripsiLengkap();
47    }
48 }
49
50 interface Deskripsi {
51     String deskripsiSingkat();
52     String deskripsiLengkap();
53 }
54
```

pelanggan.java



```
1  package sigmastore;
2
3  public class pelanggan {
4      private String IDpelanggan;
5      private String Namapelanggan;
6
7      public pelanggan(String IDpelanggan, String Namapelanggan) {
8          this.IDpelanggan = IDpelanggan;
9          this.Namapelanggan = Namapelanggan;
10     }
11
12     public String getIDpelanggan() {
13         return IDpelanggan;
14     }
15
16     public String getNamapelanggan() {
17         return Namapelanggan;
18     }
19
20     public void setNamapelanggan(String Namapelanggan) {
21         this.Namapelanggan = Namapelanggan;
22     }
23
24     @Override
25     public String toString() {
26         return "ID: " + IDpelanggan + ", Nama: " + Namapelanggan;
27     }
28 }
29
```

## produk.java

```
1 package sigmastore;
2
3 public abstract class produk {
4     public enum JenisProduk {
5         DIAMOND, ITEM;
6     }
7
8     private String IDproduk;
9     private int stok;
10    private JenisProduk jenisProduk;
11    private double harga;
12    private String IDgame;
13
14    public produk(String IDproduk, int stok, JenisProduk jenisProduk, double harga, String IDgame) {
15        this.IDproduk = IDproduk;
16        this.stok = stok;
17        this.jenisProduk = jenisProduk;
18        this.harga = harga;
19        this.IDgame = IDgame;
20    }
21
22    public String getIDproduk() {
23        return IDproduk;
24    }
25
26    public void setIDproduk(String IDproduk) {
27        this.IDproduk = IDproduk;
28    }
29
30    public int getstok() {
31        return stok;
32    }
33
34    public void setstok(int stok) {
35        this.stok = stok;
36    }
37
38    public JenisProduk getJenisProduk() {
39        return jenisProduk;
40    }
41
42    public void setJenisProduk(JenisProduk jenisProduk) {
43        this.jenisProduk = jenisProduk;
44    }
45
46    public double getHarga() {
47        return harga;
48    }
49
50    public void setHarga(double harga) {
51        this.harga = harga;
52    }
53
54    public void setHarga(int harga) {
55        this.harga = harga;
56    }
57
58    public String getIDgame() {
59        return IDgame;
60    }
61
62    public void setIDgame(String IDgame) {
63        this.IDgame = IDgame;
64    }
65
66 }
```



```

65
66     public abstract String deskripsiProduk();
67
68     @Override
69     public String toString() {
70         return "ID Produk: " + IDproduk + ", Stok: " + stok + ", Jenis: " + JenisProduk +
71             ", Harga: " + harga + ", ID Game: " + IDgame;
72     }
73
74     public static class diamond extends produk {
75         private int jumlahDiamond;
76
77         public diamond(String IDproduk, int stok, double harga, String IDgame, int jumlahDiamond) {
78             super(IDproduk, stok, JenisProduk.DIAMOND, harga, IDgame);
79             if (jumlahDiamond <= 0) {
80                 throw new IllegalArgumentException("Jumlah diamond harus lebih besar dari 0.");
81             }
82             this.jumlahDiamond = jumlahDiamond;
83         }
84
85         public int getJumlahDiamond() {
86             return jumlahDiamond;
87         }
88
89         public void setJumlahDiamond(int jumlahDiamond) {
90             if (jumlahDiamond <= 0) {
91                 throw new IllegalArgumentException("Jumlah diamond harus lebih besar dari 0.");
92             }
93             this.jumlahDiamond = jumlahDiamond;
94         }
95
96         @Override
97         public String deskripsiProduk() {
98             return "Diamond " + getIDproduk() + ": " + getJumlahDiamond() + " diamond untuk game " + getIDgame();
99         }
100
101         @Override
102         public String toString() {
103             return super.toString() + ", Jumlah Diamond Per Stok: " + jumlahDiamond;
104         }
105     }
106
107     public static class item extends produk {
108         private String durasiItem;
109
110         public item(String IDproduk, int stok, double harga, String IDgame, String durasiItem) {
111             super(IDproduk, stok, JenisProduk.ITEM, harga, IDgame);
112             this.durasiItem = durasiItem;
113         }
114
115         public String getDurasiItem() {
116             return durasiItem;
117         }
118
119         public void setDurasiItem(String durasiItem) {
120             this.durasiItem = durasiItem;
121         }
122
123         @Override
124         public String deskripsiProduk() {
125             return "Item " + getIDproduk() + ": durasi " + durasiItem + " untuk game " + getIDgame();
126         }
127
128         @Override
129         public String toString() {
130             return super.toString() + ", Durasi Item: " + durasiItem;
131         }
132     }
133 }
134
135

```

## transaksi.java

```

1  package sigmastore;
2
3  public final class transaksi {
4      public final String IDtransaksi;
5      public String IDpelanggan;
6      public String IDproduk;
7      public double totalHarga;
8
9      private static int totalTransaksi = 0;
10
11     public transaksi(String IDtransaksi, String IDpelanggan, String IDproduk, double totalHarga) {
12         this.IDtransaksi = IDtransaksi;
13         this.IDpelanggan = IDpelanggan;
14         this.IDproduk = IDproduk;
15         this.totalHarga = totalHarga;
16
17         totalTransaksi++;
18     }
19
20     public static int getTotalTransaksi() {
21         return totalTransaksi;
22     }
23
24     @Override
25     public final String toString() {
26         return "ID: " + IDtransaksi + ", Pelanggan: " + IDpelanggan + ", Produk: " + IDproduk + ", Total Harga: " + totalHarga;
27     }
28 }
29

```