

## Kategori.Java



```
1
2  package struktur;
3
4  public class Kategori {
5      private String namaKategori;
6
7
8      public Kategori(String namaKategori) {
9          this.namaKategori = namaKategori;
10     }
11
12
13     public String getNamaKategori() {
14         return namaKategori;
15     }
16
17
18     public void setNamaKategori(String namaKategori) {
19         this.namaKategori = namaKategori;
20     }
21
22     @Override
23     public String toString() {
24         return namaKategori;
25     }
26 }
27
```

## ManajemenProduk.java (1)

```
1 package struktur;
2
3 import java.util.ArrayList;
4 import java.util.Scanner;
5
6 public class ManajemenProduk {
7     private ArrayList<Produk> daftarProduk = new ArrayList<>();
8     private ArrayList<Kategori> daftarKategori = new ArrayList<>();
9     private Scanner scanner = new Scanner(System.in);
10
11     public ManajemenProduk() {
12         daftarKategori.add(new Kategori("Roti"));
13         daftarKategori.add(new Kategori("Pastry"));
14     }
15
16
17     public ArrayList<Kategori> getDaftarKategori() {
18         return daftarKategori;
19     }
20
21
22     public void tambahProduk() {
23         System.out.print("Masukkan nama produk: ");
24         String nama = scanner.nextLine();
25
26         System.out.print("Masukkan harga produk: ");
27         double harga = scanner.nextDouble();
28         scanner.nextLine();
29
30         System.out.print("Masukkan stok produk: ");
31         int stok = scanner.nextInt();
32         scanner.nextLine();
33
34         System.out.println("Pilih kategori:");
35         for (int i = 0; i < daftarKategori.size(); i++) {
36             System.out.println((i + 1) + ". " + daftarKategori.get(i).getNamaKategori());
37         }
38         System.out.print("Masukkan nomor kategori: ");
39         int kategoriIndex = scanner.nextInt() - 1;
40         scanner.nextLine();
41
42         if (kategoriIndex >= 0 && kategoriIndex < daftarKategori.size()) {
43             Kategori kategoriDipilih = daftarKategori.get(kategoriIndex);
44             Produk produkBaru;
45
46             if (kategoriDipilih.getNamaKategori().equalsIgnoreCase("Roti")) {
47                 System.out.print("Masukkan rasa roti: ");
48                 String rasa = scanner.nextLine();
49                 produkBaru = new ProdukRoti(nama, harga, stok, kategoriDipilih, rasa);
50             }
51             else if (kategoriDipilih.getNamaKategori().equalsIgnoreCase("Pastry")) {
52                 System.out.println("Pilih ukuran pastry:");
53                 System.out.println("1. Mini");
54                 System.out.println("2. Sedang");
55                 System.out.println("3. Jumbo");
56                 System.out.print("Masukkan pilihan (1-3): ");
57                 int ukuran = scanner.nextInt();
58                 scanner.nextLine();
59
60                 if (ukuran < 1 || ukuran > 3) {
61                     System.out.println("Ukuran tidak valid! Default ke Sedang (2)");
62                     ukuran = 2;
63                 }
64                 produkBaru = new ProdukPastry(nama, harga, stok, kategoriDipilih, ukuran);
65             }
66             else {
67                 System.out.println("Kategori tidak dikenali! Produk akan dibuat sebagai produk dasar.");
68                 produkBaru = new Produk(nama, harga, stok, kategoriDipilih);
69             }
70
71             daftarProduk.add(produkBaru);
72             System.out.println("Produk berhasil ditambahkan!");
73         } else {
74             System.out.println("Kategori tidak valid!");
75         }
76     }
77 }
```

## ManajemenProduk.java (2)

```
77
78 // Overloading 1: Tambah produk roti dengan rasa (String)
79 public void tambahProduk(String nama, double harga, int stok, Kategori kategori, String rasa) {
80     if (!kategori.getNamaKategori().equalsIgnoreCase("Roti")) {
81         System.out.println("Error: Kategori harus 'Roti'");
82         return;
83     }
84     daftarProduk.add(new ProdukRoti(nama, harga, stok, kategori, rasa));
85     System.out.println("Produk roti berhasil ditambahkan!");
86 }
87
88 // Overloading 2: Tambah produk pastry dengan ukuran (int)
89 public void tambahProduk(String nama, double harga, int stok, Kategori kategori, int ukuran) {
90     if (!kategori.getNamaKategori().equalsIgnoreCase("Pastry")) {
91         System.out.println("Error: Kategori harus 'Pastry'");
92         return;
93     }
94     if (ukuran < 1 || ukuran > 3) {
95         System.out.println("Error: Ukuran harus 1 (Mini), 2 (Sedang), atau 3 (Jumbo)");
96         return;
97     }
98     daftarProduk.add(new ProdukPastry(nama, harga, stok, kategori, ukuran));
99     System.out.println("Produk pastry berhasil ditambahkan!");
100 }
101
102
103 public void tampilkanProduk() {
104     if (daftarProduk.isEmpty()) {
105         System.out.println("Belum ada produk!");
106     } else {
107         System.out.println("\n=== Daftar Produk ===");
108         for (int i = 0; i < daftarProduk.size(); i++) {
109             System.out.println((i + 1) + ". " + daftarProduk.get(i));
110         }
111     }
112 }
113
```

### ManajemenProduk.java (3)

```
113
114     public void editProduk() {
115         tampilkanProduk();
116         if (daftarProduk.isEmpty()) return;
117
118         System.out.print("Pilih nomor produk yang ingin diedit: ");
119         int index = scanner.nextInt() - 1;
120         scanner.nextLine();
121
122         if (index >= 0 && index < daftarProduk.size()) {
123             Produk produk = daftarProduk.get(index);
124
125             System.out.print("Masukkan nama baru: ");
126             produk.setNama(scanner.nextLine());
127
128             System.out.print("Masukkan harga baru: ");
129             produk.setHarga(scanner.nextDouble());
130             scanner.nextLine();
131
132             System.out.print("Masukkan stok baru: ");
133             produk.setStok(scanner.nextInt());
134             scanner.nextLine();
135
136             System.out.println("Pilih kategori baru:");
137             for (int i = 0; i < daftarKategori.size(); i++) {
138                 System.out.println((i + 1) + ". " + daftarKategori.get(i).getNamaKategori());
139             }
140             System.out.print("Masukkan nomor kategori: ");
141             int kategoriIndex = scanner.nextInt() - 1;
142             scanner.nextLine();
143
144             if (kategoriIndex >= 0 && kategoriIndex < daftarKategori.size()) {
145                 produk.setKategori(daftarKategori.get(kategoriIndex));
146             }
147
148             System.out.println("Produk berhasil diperbarui!");
149         } else {
150             System.out.println("Nomor produk tidak valid!");
151         }
152     }
153
154     public void hapusProduk() {
155         tampilkanProduk();
156         if (daftarProduk.isEmpty()) return;
157
158         System.out.print("Pilih nomor produk yang ingin dihapus: ");
159         int index = scanner.nextInt() - 1;
160         scanner.nextLine();
161
162         if (index >= 0 && index < daftarProduk.size()) {
163             daftarProduk.remove(index);
164             System.out.println("Produk berhasil dihapus!");
165         } else {
166             System.out.println("Nomor produk tidak valid!");
167         }
168     }
169 }
```

## Produk.Java

```
1 package struktur;
2
3 public class Produk {
4     private String nama;
5     private double harga;
6     private int stok;
7     private Kategori kategori;
8
9     public Produk(String nama, double harga, int stok, Kategori kategori) {
10         this.nama = nama;
11         this.harga = harga;
12         this.stok = stok;
13         this.kategori = kategori;
14     }
15
16     public String getNama() {
17         return nama;
18     }
19
20     public void setNama(String nama) {
21         this.nama = nama;
22     }
23
24     public double getHarga() {
25         return harga;
26     }
27
28     public void setHarga(double harga) {
29         this.harga = harga;
30     }
31
32     public int getStok() {
33         return stok;
34     }
35
36     public void setStok(int stok) {
37         this.stok = stok;
38     }
39
40     public Kategori getKategori() {
41         return kategori;
42     }
43
44     public void setKategori(Kategori kategori) {
45         this.kategori = kategori;
46     }
47
48     public String getDetailTambahan() {
49         return "";
50     }
51
52     @Override
53     public String toString() {
54         return nama + " - Rp" + harga + " - Stok: " + stok + " - Kategori: " + kategori.getNamaKategori() + getDetailTambahan();
55     }
56 }
57
```

## ProdukPastry.java

```
1 package struktur;
2
3 public class ProdukPastry extends Produk {
4     private int ukuran;
5
6     public ProdukPastry(String nama, double harga, int stok, Kategori kategori, int ukuran) {
7         super(nama, harga, stok, kategori);
8         this.ukuran = ukuran;
9     }
10
11     public int getUkuran() {
12         return ukuran;
13     }
14
15     public String getUkuranString() {
16         switch (ukuran) {
17             case 1: return "Mini";
18             case 2: return "Sedang";
19             case 3: return "Jumbo";
20             default: return "Tidak Valid";
21         }
22     }
23
24     @Override
25     public String getDetailTambahan() {
26         return " - Ukuran: " + getUkuranString();
27     }
28 }
```

## ProdukRoti.java

```
1 package struktur;
2
3 public class ProdukRoti extends Produk {
4     private String rasa;
5
6     public ProdukRoti(String nama, double harga, int stok, Kategori kategori, String rasa) {
7         super(nama, harga, stok, kategori);
8         this.rasa = rasa;
9     }
10
11     public String getRasa() {
12         return rasa;
13     }
14
15     public void setRasa(String rasa) {
16         this.rasa = rasa;
17     }
18
19     @Override
20     public String getDetailTambahan() {
21         return " - Rasa: " + rasa;
22     }
23 }
24
```

## LunarBakeshop.java

```
1 import java.util.Scanner;
2 import struktur.Kategori;
3 import struktur.ManajemenProduk;
4
5 public class LunarBakeshop {
6     public static void main(String[] args) {
7         ManajemenProduk manajemenProduk = new ManajemenProduk();
8         Scanner scanner = new Scanner(System.in);
9
10        Kategori roti = manajemenProduk.getDaftarKategori().get(0);
11        Kategori pastry = manajemenProduk.getDaftarKategori().get(1);
12
13        // Tambah produk roti (overloading 1)
14        manajemenProduk.tambahProduk("Roti Coklat", 15000, 30, roti, "Coklat");
15        // Tambah produk pastry (overloading 2)
16        manajemenProduk.tambahProduk("Croissant", 25000, 20, pastry, 3);
17
18        while (true) {
19            System.out.println("\n=== Lunar Bakeshop ===");
20            System.out.println("1. Tambah Produk");
21            System.out.println("2. Tampilkan Produk");
22            System.out.println("3. Edit Produk");
23            System.out.println("4. Hapus Produk");
24            System.out.println("5. Keluar");
25            System.out.print("Pilih menu: ");
26
27            int pilihan = scanner.nextInt();
28            scanner.nextLine();
29
30            switch (pilihan) {
31                case 1:
32                    manajemenProduk.tambahProduk();
33                    break;
34                case 2:
35                    manajemenProduk.tampilkanProduk();
36                    break;
37                case 3:
38                    manajemenProduk.editProduk();
39                    break;
40                case 4:
41                    manajemenProduk.hapusProduk();
42                    break;
43                case 5:
44                    System.out.println("Terima kasih telah menggunakan sistem Lunar Bakeshop!");
45                    return;
46                default:
47                    System.out.println("Pilihan tidak valid! Silakan coba lagi.");
48            }
49        }
50    }
51 }
```