

## POSTTEST 5

### PEMROGRAMAN BERORIENTASI OBJEK


Nama : Salom sefanya onibala

NIM : 2209106084

Kelas : Informatika B2

#### SS PROGRAM

##### 1. Roti Java

```
src > Roti >  roti.java > ...  
1  package Roti;  
2  
3  public abstract class roti {  
4      protected int id;  
5      protected String nama;  
6      protected int harga;  
7      protected int stok;  
8      static int idCounter = 1;  
9  
10     public roti(String nama, int harga, int stok) {  
11         this.id = idCounter++;  
12         this.nama = nama;  
13         this.harga = harga;  
14         this.stok = stok;  
15     }  
16  
17     //abstract method untuk mendapatkan tipe roti  
18     public abstract String getTipeRoti();  
19  
20     //getter untuk ID (penerapan encapsulation)  
21     public int getId() {  
22         return id;  
23     }  
24  
25     //getter untuk NAMA  
26     public String getNama() {  
27         return nama;  
28     }  
}
```

src > Roti >  roti.java > ...

```
3    public abstract class roti {  
29        //getter untuk Harga  
30        public int getHarga() {  
31            return harga;  
32        }  
33  
34        //getter untuk stok  
35        public int getStok() {  
36            return stok;  
37        }  
38  
39        //setter untuk Nama (penerapan encapsulation)  
40        public void setName(String nama) {  
41            this.nama = nama;  
42        }  
43  
44        //setter untuk Harga  
45        public void setHarga(int harga) {  
46            this.harga = harga;  
47        }  
48  
49        //setter untuk stok  
50        public void setStok(int stok) {  
51            this.stok = stok;  
52        }  
53    }
```

## 2. Roti Manis

```
src > Roti > RotiManis.java > ...
1  package Roti;
2
3  public class RotiManis extends roti {
4      protected String rasa;
5
6      public RotiManis(String nama, int harga, int stok, String rasa) {
7          super(nama, harga, stok);
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 getTipeRoti() {
21         return "Manis";
22     }
23
24     @Override
25     public String toString() {
26         return super.toString() + " - Rasa: " + rasa;
27     }
28
29 }
```

### 3. Roti Tawar

```
src > Roti > RotiTawar.java > ...
1  package Roti;
2
3  public class RotiTawar extends roti{
4      protected String jenis;
5
6      public RotiTawar(String nama, int harga, int stok, String jenis) {
7          super(nama, harga, stok);
8          this.jenis = jenis;
9      }
10
11     public String getJenis() {
12         return jenis;
13     }
14
15     public void setJenis(String jenis) {
16         this.jenis = jenis;
17     }
18
19     @Override
20     public String getTipeRoti() {
21         return "Manis";
22     }
23
24     @Override
25     public String toString() {
26         return super.toString() + " - Jenis: " + jenis;
27     }
28 }
```

#### 4. App Java

```
src > App.java > App > main(String[])
1  import java.util.ArrayList;
2  import java.util.Scanner;
3  import java.io.IOException;
4  import Roti.*;
5
6  public final class App {
7      public static final int idCounter = 1;
8      private static Scanner scanner = new Scanner(System.in);
9      private static ArrayList<roti> daftarRoti = new ArrayList<>();
10
11      Run | Debug
12      public static final void main(String[] args) throws IOException {
13          while (true) {
14              clearScreen();
15              System.out.println(x:"=====");
16              System.out.println(x:"                Selamat datang di Toko Roti Onibala                ");
17              System.out.println(x:"=====");
18              System.out.println(x:" [1] Tambah Roti");
19              System.out.println(x:" [2] Lihat Semua Roti");
20              System.out.println(x:" [3] Perbarui Informasi Roti");
21              System.out.println(x:" [4] Hapus Roti");
22              System.out.println(x:" [5] Keluar");
23              System.out.print(s:" Pilih menu: ");
24
25              int pilihan = scanner.nextInt();
26              scanner.nextLine();
27
28              switch (pilihan) {
29                  case 1:
30                      tambahRoti();
31                      waitForEnter();
32                      break;
33                  case 2:
34                      lihatSemuaRoti();
35                      waitForEnter();
36                      break;
```

```

6   public final class App {
11      public static final void main(String[] args) throws IOException {
37          case 3:
38              perbaruiRoti();
39              waitForEnter();
40              break;
41          case 4:
42              hapusRoti();
43              waitForEnter();
44              break;
45          case 5:
46              System.out.println(x:"Terima kasih telah menggunakan layanan Toko Roti Onibala");
47              return;
48          default:
49              System.out.println(x:"Pilihan tidak valid, silakan coba lagi");
50              waitForEnter();
51      }
52  }
53  }
54
55  private static void tambahRoti() {
56      clearScreen();
57      System.out.println(x:"=====");
58      System.out.println(x:"                Selamat datang di Toko Roti Onibala                ");
59      System.out.println(x:"=====");
60      System.out.print(s:"Masukkan nama roti      : ");
61      String nama = scanner.nextLine();
62      System.out.print(s:"Masukkan harga roti   : ");
63      int harga = scanner.nextInt();
64      System.out.print(s:"Masukkan stok roti    : ");
65      int stok = scanner.nextInt();
66      scanner.nextLine();
67
68      System.out.print(s:"Roti Tawar atau Roti Manis (Tawar/Manis) : ");
69      String jenisRoti = scanner.nextLine();

```

```

6   public final class App {
55      private static void tambahRoti() {
71          if (jenisRoti.equalsIgnoreCase(anotherString:"Tawar")) {
72              System.out.print(s:"Masukkan jenis roti tawar (Misal: Gandum/Roti Sobek): ");
73              String jenis = scanner.nextLine();
74              RotiTawar roti = new RotiTawar(nama, harga, stok, jenis);
75              daftarRoti.add(roti);
76          } else if (jenisRoti.equalsIgnoreCase(anotherString:"Manis")) {
77              System.out.print(s:"Masukkan rasa roti manis (Misal: Coklat/Keju): ");
78              String rasa = scanner.nextLine();
79              RotiManis roti = new RotiManis(nama, harga, stok, rasa);
80              daftarRoti.add(roti);
81          } else {
82              System.out.println(x:"Jenis roti tidak valid");
83              return;
84          }
85          System.out.println(x:"Roti berhasil ditambahkan");
86      }
87
88      private static void lihatSemuaRoti() {
89          clearScreen();
90          System.out.println(x:"=====");
91          System.out.println(x:"                Selamat datang di Toko Roti Onibala                ");
92          System.out.println(x:"=====");
93
94          if (daftarRoti.isEmpty()) {
95              System.out.println(x:"Tidak ada roti yang tersedia");
96          } else {
97              System.out.println(x:"                DAFTAR ROTI                ");
98              for (roti roti : App.daftarRoti) {
99                  //menampilkan informasi roti
100                 System.out.println(roti.getId() + ". " + "Roti " + roti.getNama());
101                 System.out.println("        Harga      : Rp. " + roti.getHarga());
102                 System.out.println("        Stok       : " + roti.getStok());

```

```

6 public final class App {
88     private static void lihatSemuaRoti() {
103         //memeriksa apakah roti adalah RotiManis atau RotiTawar
104         if (roti instanceof RotiManis) {
105             RotiManis rotiManis = (RotiManis) roti;
106             System.out.println("    Rasa      : " + rotiManis.getRasa());
107         } else if (roti instanceof RotiTawar) {
108             RotiTawar rotiTawar = (RotiTawar) roti;
109             System.out.println("    Jenis      : " + rotiTawar.getJenis());
110         }
111     }
112 }
113 }
114
115 private static void perbaruiRoti() {
116     clearScreen();
117     System.out.println(x:"=====");
118     System.out.println(x:"                Selamat datang di Toko Roti Onibala                ");
119     System.out.println(x:"=====");
120
121     if (daftarRoti.isEmpty()) {
122         System.out.println(x:"Tidak ada roti yang tersedia untuk diperbarui");
123         return;
124     }
125
126     System.out.print(s:"Masukkan ID roti yang ingin diperbarui: ");
127     int idRoti = scanner.nextInt();
128     scanner.nextLine();
129
130     for (roti roti : daftarRoti) {
131         if (roti.getId() == idRoti) {
132             System.out.print(s:"Masukkan nama baru untuk roti      : ");
133             String newName = scanner.nextLine();
134             System.out.print(s:"Masukkan harga baru untuk roti    : ");
135             int newHarga = scanner.nextInt();
136             System.out.print(s:"Masukkan stok baru untuk roti      : ");
137             int newStok = scanner.nextInt();

```

```

src > App.java > ...
6 public final class App {
115     private static void perbaruiRoti() {
138         scanner.nextLine();
139
140         roti.setNama(newNama);
141         roti.setHarga(newHarga);
142         roti.setStok(newStok);
143         System.out.println(x:"Informasi roti berhasil diperbarui");
144         return;
145     }
146 }
147 System.out.println(x:"Roti dengan ID tersebut tidak ditemukan");
148 }
149
150 private static void hapusRoti() {
151     clearScreen();
152     System.out.println(x:"=====");
153     System.out.println(x:"          Selamat datang di Toko Roti Onibala          ");
154     System.out.println(x:"=====");
155
156     if (daftarRoti.isEmpty()) {
157         System.out.println(x:"Tidak ada roti yang tersedia untuk dihapus");
158         return;
159     }
160
161     System.out.print(s:"Masukkan ID roti yang ingin dihapus: ");
162     int idRoti = scanner.nextInt();
163     scanner.nextLine();
164
165     //mencari roti berdasarkan ID
166     roti rotiToDelete = null;
167     for (roti roti : daftarRoti) {
168         if (roti.getId() == idRoti) {
169             rotiToDelete = roti;
170             break;
171         }
172     }

```



```

6 public final class App {
150     private static void hapusRoti() {
173
174         if (rotiToDelete != null) {
175             //cek apakah stok roti lebih dari 1
176             if (rotiToDelete.getStok() > 1) {
177
178                 //jika stok lebih dari 1 kurangi stok
179                 rotiToDelete.setStok(rotiToDelete.getStok() - 1);
180                 System.out.println(x:"Stok roti berhasil dikurangi");
181             } else {
182
183                 //jika stok hanya 1 hapus roti dari daftar
184                 daftarRoti.remove(rotiToDelete);
185                 System.out.println(x:"Roti berhasil dihapus");
186             }
187         } else {
188             System.out.println(x:"Roti dengan ID tersebut tidak ditemukan");
189         }
190     }
191
192     private static void clearScreen() {
193         System.out.print(s:"\033[H\033[2J");
194         System.out.flush();
195     }
196
197     private static void waitForEnter() {
198         System.out.print(s:"Tekan Enter untuk melanjutkan...");
199         try {
200             System.in.read();
201         } catch (Exception e) {
202             e.printStackTrace();
203         }
204     }
205 }

```