

PENGEMBANGAN BERORIENTASI OBJEK PRAKTIK

KELOMPOK 9

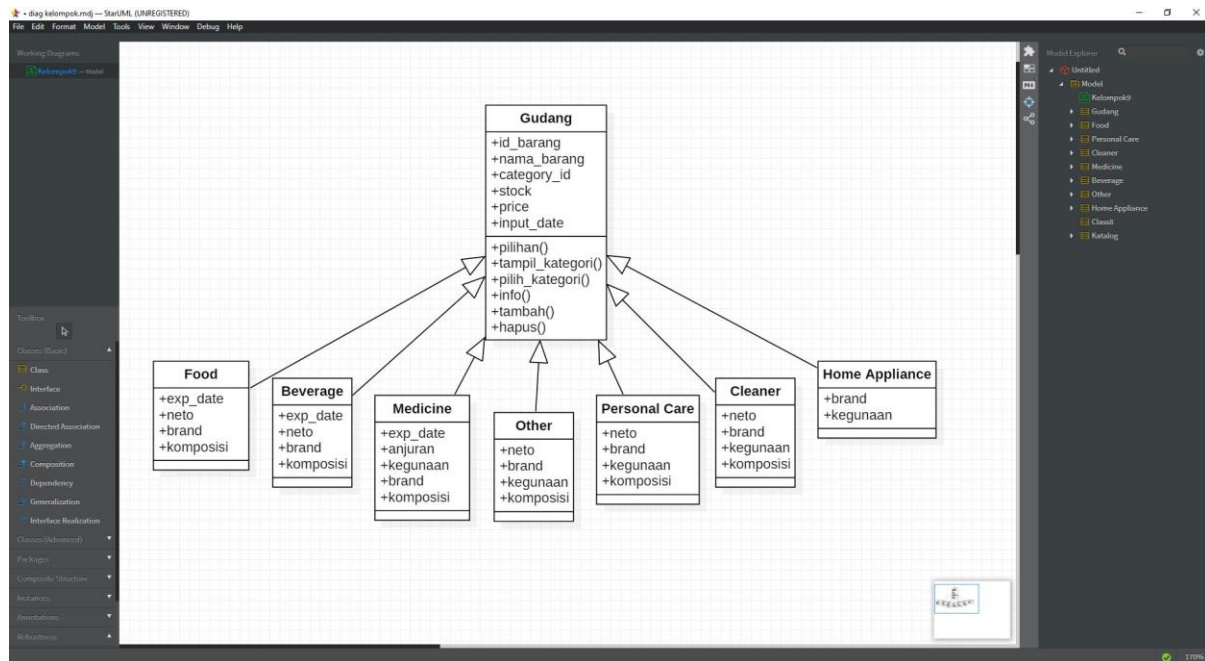


Disusun Oleh :

5210411383 Allan Bil Faqih
5210411359 Gilang Fathanatu Tantular
5210411367 Muhammad Insan Kamil
5210411378 Jaka Rahmat Sambegana

**PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS & TEKNOLOGI
UNIVERSITAS TEKNOLOGI YOGYAKARTA
YOGYAKARTA
2022**

1. Class Diagram



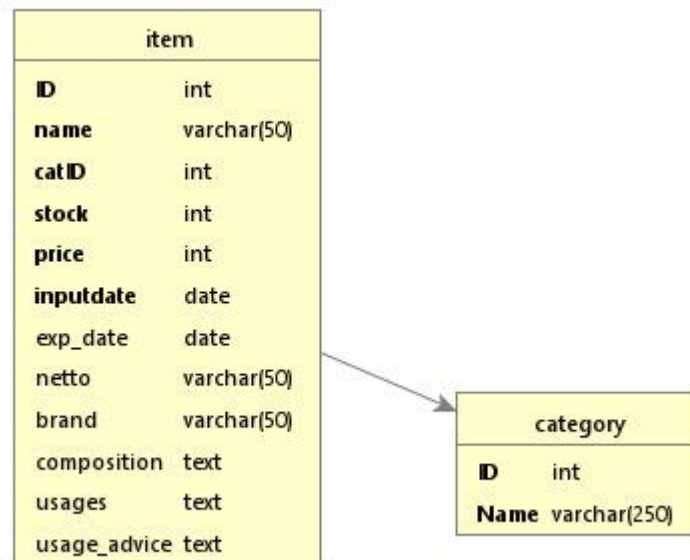
Berikut adalah class diagram dari pengelolaan gudang. Class gudang tersebut mempunyai beberapa atribut dan operation yang digunakan dapat dilihat pada gambar diatas. Pada class gudang terdapat 7 class didalamnya yang digeneralization ke class gudang antara lain :

- Food
- Beverages
- Medicine
- Other
- Personal Care
- Cleaener
- Home Application

Dari beberapa class tersebut mempunyai masing-masing attribute yang digunakan.

2. Database

Pada bagian database kami membuat 2 database yaitu item dan category. Pada masing-masing database dimasukan beberapa atribute dan fungsi sebagai berikut :



3. Input Source Code

Pada bagian coding kami membuat 2 program yaitu gudang dan app. Program gudang digunakan untuk membuat class gudang dan beberapa class yang ada didalam gudang. Kemudian program app digunakan untuk memanggil program gudang yang kemudian diinputkan beberapa fungsi sebagai berikut :

a. gudang.py

```
import mysql.connector
import datetime as dt
from prettytable import PrettyTable as pt

db = mysql.connector.connect (
    host = "localhost",
    user = "root",
    password = "",
    database = "gudang"
)

cur = db.cursor()

class Gudang:
    def __init__(self, id = 0, name = "", catid = 0, stock = 0, price = 0,
input_date = ""):
        self.id = id
        self.name = name
        self.catid = catid
        self.stock = stock
        self.price = price
        self.input_date = input_date
```

```

def pilihan(self):
    main_menu = ["[1] LIHAT DATA ITEM",
                  "[2] TAMBAH DATA ITEM",
                  "[3] HAPUS DATA ITEM",
                  "[4] KELUAR DARI APP"]

    for i in main_menu:
        print(i)

def info(self):
    colItems = ['Id', 'Nama Barang', 'Kategori', 'Stok', 'Harga',
'Tanggal Input',
                "Tanggal Kadaluarsa", "Berat Bersih", "Brand",
"Komposisi", "Kegunaan", "Anjuran Pemakaian"]
    print("\n[i] Daftar Barang\n")
    cur.execute("""select
                    item.ID as 'Id', item.name, category.name,
item.stock, item.price,item.inputdate,
                    item.exp_date, item.netto, item.brand,
item.composition, item.usages, item.usage_advice
                    from item left join category on item.catID =
category.ID
                """)

    newPT = pt()
    newPT.field_names = colItems
    for r in cur.fetchall():
        newPT.add_row(r)
    print(f"{newPT}\n")

def tampil_kategori(self):
    colCat = ['Id', 'name Kategori']
    print("\n[i] name Kategori\n")
    cur.execute("""select ID as 'Id', name as 'name Kategori' from
category""")
    newPT = pt()
    newPT.field_names = colCat
    for r in cur.fetchall():
        newPT.add_row(r)
    print(newPT)

def pilih_kategori(self):
    print("\n[i] Untuk memasukkan kategori anda bisa mengisinya dengan
Id maupun name dari kategori tersebut.")
    inpCat = input("\n[=] Masukkan Id/name Kategori : ")
    while True:
        if(inpCat == ""):
            break
        elif(inpCat.isdigit()):
            cur.execute("select * from category where ID=%s", (inpCat,))
            row = cur.fetchone()
            if row == None:

```

```

        print("\n[i] Tidak ada kategori yang ber-ID " + inpCat)
        return self.pilih_kategori()
    else:
        self.catid = row[0]
        print("\n[i] Kategori yang anda pilih adalah " + row[1])
        #break
        return self.catid
    else:
        cur.execute("select * from category where name like %s",
(inpCat,))
        row = cur.fetchone()
        if row == None:
            print("\n[i] Tidak ada kategori yang bernama " + inpCat)
            return self.pilih_kategori()
        else:
            self.catid = row[0]
            print("\n[i] ID Kategori yang anda pilih adalah " +
str(row[0]))
            #break
            return self.catid

    def tambah(self, name = "", catid = "", stock = "", price = "",
exp_date = "", netto = "", brand = "", komposisi = "", anjuran = "",
kegunaan = ""):
        print("\n[i] Anda akan memasukkan barang baru ke server.")
        cur.execute("""insert into item (name, catID, stock, price,
inputdate,
exp_date, netto, brand, composition, usages,
usage_advice)
values(%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)""",
(name, catid, stock, price, dt.date.today(),
exp_date, netto, brand, komposisi, anjuran, kegunaan),)
        db.commit()
        print("\n[i] Data berhasil ditambahkan.\n")
        self.pilihan()

    def hapus(self):
        self.info()
        print("[i] Masukkan ID atau name barang untuk menghapus.")
        print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
        while True:
            inpID = input("[=] Masukkan ID atau name barang : ")
            if(inpID == ""):
                print("[!] Input tidak boleh Kosong")
            elif(inpID.isdigit()):
                cur.execute("select * from item where ID=%s", (inpID,))
                row = cur.fetchone()
                if row == None:
                    print("\n[i] Tidak ada barang yang ber-ID " + inpID)
                else:
                    inpID = row[0]
                    print("\n[i] Barang yang anda pilih adalah " + row[1])
            elif(inpID == "0" or inpID == "exit" or inpID == "Exit" or inpID

```

```

== "EXIT"):
    return
else:
    cur.execute("select * from item where name like %s",
(inpID,))
    row = cur.fetchone()
    if row == None:
        print("\n[i] Tidak ada barang yang bernama " + inpID)
    else:
        inpID = row[0]
        print("\n[i] ID Kategori yang anda pilih adalah " +
str(row[0]))

    print("[i] Apakah anda yakin ingin menghapus data ini?")
    inpUsure = input("[=] Y/N : ")
    if(inpUsure == "y" or inpUsure == "Y"):
        cur.execute("delete from item where id=%s", (inpID,))
        db.commit()
        print("\n[i] Data berhasil dihapus.\n")
        self.pilihan()
        break
    elif(inpUsure == "n" or inpUsure == "N"):
        break
    else:
        continue

class Food(Gudang):
    def __init__(self, id, name, catid, stock, price,
        exp_date, netto, brand, komposisi):
        super().__init__(id, name, catid, stock, price)
        self.exp_date = exp_date
        self.netto = netto
        self.brand = brand
        self.komposisi = komposisi

    def tambah(self):
        Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
self.exp_date, self.netto, self.brand, self.komposisi, "", "")

class Beverage(Gudang):
    def __init__(self, id, name, catid, stock, price,
        exp_date, netto, brand, komposisi):
        super().__init__(id, name, catid, stock, price)
        self.exp_date = exp_date
        self.netto = netto
        self.brand = brand
        self.komposisi = komposisi

    def tambah(self):
        Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
self.exp_date, self.netto, self.brand, self.komposisi, "", "")

class Medicine(Gudang):

```

```

def __init__(self, id, name, catid, stock, price,
              exp_date, brand, komposisi, anjuran, kegunaan):
    super().__init__(id, name, catid, stock, price)
    self.exp_date = exp_date
    self.brand = brand
    self.komposisi = komposisi
    self.anjuran = anjuran
    self.kegunaan = kegunaan

def tambah(self):
    Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
                  self.exp_date, "", self.brand, self.komposisi, self.anjuran,
                  self.kegunaan)

class Other(Gudang):
    def __init__(self, id, name, catid, stock, price,
                  netto, brand, komposisi, kegunaan):
        super().__init__(id, name, catid, stock, price)
        self.netto = netto
        self.brand = brand
        self.komposisi = komposisi
        self.kegunaan = kegunaan

    def tambah(self):
        Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
                      "", self.netto, self.brand, self.komposisi, self.kegunaan, "")

class PersonalCare(Gudang):
    def __init__(self, id, name, catid, stock, price,
                  netto, brand, komposisi, kegunaan):
        super().__init__(id, name, catid, stock, price)
        self.netto = netto
        self.brand = brand
        self.komposisi = komposisi
        self.kegunaan = kegunaan

    def tambah(self):
        Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
                      "", self.netto, self.brand, self.komposisi, self.kegunaan, "")

class Cleaner(Gudang):
    def __init__(self, id, name, catid, stock, price,
                  netto, brand, komposisi, kegunaan):
        super().__init__(id, name, catid, stock, price)
        self.netto = netto
        self.brand = brand
        self.komposisi = komposisi
        self.kegunaan = kegunaan

    def tambah(self):
        Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
                      "", self.netto, self.brand, self.komposisi, self.kegunaan, "")

```

```

class HomeAppliance(Gudang):
    def __init__(self, id, name, catid, stock, price, brand, kegunaan):
        super().__init__(id, name, catid, stock, price)
        self.brand = brand
        self.kegunaan = kegunaan

    def tambah(self):
        Gudang.tambah(self, self.name, self.catid, self.stock, self.price,
            "", "", self.brand, "", self.kegunaan, "")

```

b. app.py

```

from gudang import *
import datetime as dt
from dateutil.relativedelta import relativedelta as rd
from prettytable import PrettyTable as pt

newGudang = Gudang()

print('\n=====')
print('[i] Select Menu')
print('=====')
newGudang.pilihan()

while True:
    inpSelect = input("\n[=] Pilih Menu: ")
    cur.execute("select count(*) as count from item")
    for r in cur.fetchall():
        ct = r[0]

    if(inpSelect == '1'):
        newGudang.info()
        newGudang.pilihan()
    elif(inpSelect == '2'):
        newGudang.tampil_kategori()
        catid = newGudang.pilih_kategori()
        while True:
            inpName = input("\n[=] Masukkan Nama Barang : ")
            inpStock = input("[=] Masukkan Stock Barang : ")
            inpPrice = input("[=] Masukkan Harga Barang : ")
            if(inpName == "" or inpStock == "" or inpPrice == ""):
                print("\n[i] Input tidak boleh kosong!\n")
            else:
                if(catid == 1):
                    while True:
                        inpNetto = input("[=] Masukkan Netto/Berat Bersih
Barang : ")

                        inpBrand = input("[=] Masukkan Brand Barang : ")
                        inpComp = input("[=] Masukkan Komposisi Barang : ")
                        if(inpNetto == "" or inpBrand == "" or inpComp ==
                        ""):
                            print("\n[i] Input tidak boleh kosong!\n")

```



```

        else:
            expdate = dt.datetime.utcnow() + rd(years = 5)
            newBev = Beverage(0, inpName, catid, inpStock,
inpPrice, expdate, inpNetto, inpBrand, inpComp)
            newBev.tambah()
            break
    elif(catid == 2):
        while True:
            inpNetto = input("[=] Masukkan Netto/Berat Bersih
Barang : ")

            inpBrand = input("[=] Masukkan Brand Barang : ")
            inpComp = input("[=] Masukkan Komposisi Barang : ")
            if(inpNetto == "" or inpBrand == "" or inpComp ==
""):
                print("\n[i] Input tidak boleh kosong!\n")
            else:
                expdate = dt.datetime.utcnow() + rd(years = 5)
                newBev = Food(0, inpName, catid, inpStock,
inpPrice, expdate, inpNetto, inpBrand, inpComp)
                newBev.tambah()
                break
    elif(catid == 3):
        while True:
            inpNetto = input("[=] Masukkan Netto/Berat Bersih
Barang : ")

            inpBrand = input("[=] Masukkan Brand Barang : ")
            inpComp = input("[=] Masukkan Komposisi Barang : ")
            inpUsage = input("[=] Masukkan Kegunaan Barang : ")
            if(inpNetto == "" or inpBrand == "" or inpComp == ""
or inpUsage == ""):
                print("\n[i] Input tidak boleh kosong!\n")
            else:
                newBev = Cleaner(0, inpName, catid, inpStock,
inpPrice, inpNetto, inpBrand, inpComp, inpUsage)
                newBev.tambah()
                break
    elif(catid == 4):
        while True:
            inpNetto = input("[=] Masukkan Netto/Berat Bersih
Barang : ")

            inpBrand = input("[=] Masukkan Brand Barang : ")
            inpComp = input("[=] Masukkan Komposisi Barang : ")
            inpUsage = input("[=] Masukkan Kegunaan Barang : ")
            if(inpNetto == "" or inpBrand == "" or inpComp == ""
or inpUsage == ""):
                print("\n[i] Input tidak boleh kosong!\n")
            else:
                newBev = PersonalCare(0, inpName, catid,
inpStock, inpPrice, inpNetto, inpBrand, inpComp, inpUsage)
                newBev.tambah()
                break
    elif(catid == 5):
        while True:

```

```

inpNetto = input("[=] Masukkan Netto/Berat Bersih
Barang : ")

inpBrand = input("[=] Masukkan Brand Barang : ")
inpComp = input("[=] Masukkan Komposisi Barang : ")
inpUsage = input("[=] Masukkan Kegunaan Barang : ")
if(inpNetto == "" or inpBrand == "" or inpComp == ""
or inpUsage == ""):
    print("\n[i] Input tidak boleh kosong!\n")
else:
    newBev = Other(0, inpName, catid, inpStock,
inpPrice, inpNetto, inpBrand, inpComp, inpUsage)
    newBev.tambah()
    break
elif(catid == 6):
    while True:
        inpBrand = input("[=] Masukkan Brand Barang : ")
        inpComp = input("[=] Masukkan Komposisi Barang : ")
        inpUAdvice = input("[=] Masukkan Anjuran Penggunaan
Barang : ")

        inpUsage = input("[=] Masukkan Kegunaan Barang : ")
        if(inpUAdvice == "" or inpBrand == "" or inpComp ==
"" or inpUsage == ""):
            print("\n[i] Input tidak boleh kosong!\n")
        else:
            expdate = dt.datetime.utcnow() + rd(years = 2)
            newBev = Medicine(0, inpName, catid, inpStock,
inpPrice, expdate, inpBrand, inpComp, inpUAdvice, inpUsage)
            newBev.tambah()
            break
    elif(catid == 7):
        while True:
            inpBrand = input("[=] Masukkan Brand Barang : ")
            inpUsage = input("[=] Masukkan Kegunaan Barang : ")
            if(inpBrand == "" or inpComp == "" or inpUsage ==
 ""):
                print("\n[i] Input tidak boleh kosong!\n")
            else:
                newBev = HomeAppliance(0, inpName, catid,
inpStock, inpPrice, inpBrand, inpUsage)
                newBev.tambah()
                break

        break

elif(inpSelect == '3'):
    if(ct == 0):
        print("\n[i] Tabel Data masih kosong!")
    else:
        print("\n[i] Anda akan mengubah beberapa atribut barang.")
        newGudang.hapus()
elif(inpSelect == '4'):
    print("\n[i] Good Bye")
    break
else:

```

```
print("\n[!] Input Salah")

db.close()
```

4. Output Program :

Setelah program gudang diimport menggunakan file program app kemudian di Run maka akan muncul output yang akan memunculkan class dan database yang telah dibuat sebagai berikut :

```
D:\Univ\Mata Kuliah\SEMESTER 2\PBOB\Pert12-13 Kelompok\tugas>app.py

=====
[i] Select Menu
=====
[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP

[=] Pilih Menu: █

[=] Pilih Menu: 1

[i] Daftar Barang

+---+---+---+---+---+---+---+---+---+---+---+---+
| Id | Nama Barang | Kategori | Stok | Harga | Tanggal Input | Tanggal Kadaluarsa | Berat Bersih | Brand | Komposisi | Kegunaan | Anjuran Pemakaian |
+---+---+---+---+---+---+---+---+---+---+---+---+

[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP

[=] Pilih Menu: █

[=] Pilih Menu: 2

[i] name Kategori

+---+---+---+
| Id | name Kategori |
+---+---+---+
| 1 | Beverage |
| 2 | Food |
| 3 | Cleaner |
| 4 | Personal Care |
| 5 | Other |
| 6 | Medicine |
| 7 | Home Appliances |
+---+---+---+

[i] Untuk memasukkan kategori anda bisa mengisinya dengan Id maupun name dari kategori tersebut.

[=] Masukkan Id/name Kategori : █
```

```
[=] Masukkan Id/name Kategori : 1

[i] Kategori yang anda pilih adalah Beverage

[=] Masukkan Nama Barang : New Coke
[=] Masukkan Stock Barang : 50
[=] Masukkan Harga Barang : 5500
[=] Masukkan Netto/Berat Bersih Barang : 250g
[=] Masukkan Brand Barang : Coca-Cola
[=] Masukkan Komposisi Barang : Soda, Gula

[i] Anda akan memasukkan barang baru ke server.

[i] Data berhasil ditambahkan.

[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP

[=] Pilih Menu: █
```

```
[=] Pilih Menu: 1
```

```
[i] Daftar Barang
```

Id	Nama Barang	Kategori	Stok	Harga	Tanggal Input	Tanggal Kadaluarsa	Berat Bersih	Brand	Komposisi	Kegunaan	Anjuran Pemakaian
26	New Coke	Beverage	50	5500	2022-05-19	2027-05-19	250g	Coca-Cola	Soda, Gula		

```
[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP
```

```
[=] Pilih Menu: █
```

```
[=] Masukkan Id/name Kategori : 2

[i] Kategori yang anda pilih adalah Food

[=] Masukkan Nama Barang : Tortilla Snack
[=] Masukkan Stock Barang : 50
[=] Masukkan Harga Barang : 7500
[=] Masukkan Netto/Berat Bersih Barang : 50g
[=] Masukkan Brand Barang : Tortilla
[=] Masukkan Komposisi Barang : Kentang, Garam

[i] Anda akan memasukkan barang baru ke server.

[i] Data berhasil ditambahkan.
```

[=] Masukkan Id/name Kategori : 3

[i] Kategori yang anda pilih adalah Cleaner

[=] Masukkan Nama Barang : Pemutih Bleach

[=] Masukkan Stock Barang : 25

[=] Masukkan Harga Barang : 2500

[=] Masukkan Netto/Berat Bersih Barang : 100g

[=] Masukkan Brand Barang : Bleach

[=] Masukkan Komposisi Barang : Pemutih

[=] Masukkan Kegunaan Barang : Untuk mencuci pakaian/kain

[i] Anda akan memasukkan barang baru ke server.

[i] Data berhasil ditambahkan.

[=] Masukkan Id/name Kategori : 4

[i] Kategori yang anda pilih adalah Personal Care

[=] Masukkan Nama Barang : Denuvo Sabun Cair

[=] Masukkan Stock Barang : 65

[=] Masukkan Harga Barang : 12000

[=] Masukkan Netto/Berat Bersih Barang : 120g

[=] Masukkan Brand Barang : Denuvo

[=] Masukkan Komposisi Barang : Sabun, Pewangi

[=] Masukkan Kegunaan Barang : Membersihkan tubuh

[i] Anda akan memasukkan barang baru ke server.

[i] Data berhasil ditambahkan.

[=] Masukkan Id/name Kategori : 5

[i] Kategori yang anda pilih adalah Other

[=] Masukkan Nama Barang : Platinum Cigar

[=] Masukkan Stock Barang : 50

[=] Masukkan Harga Barang : 25000

[=] Masukkan Netto/Berat Bersih Barang : 500g

[=] Masukkan Brand Barang : Platinum

[=] Masukkan Komposisi Barang : Tobako, Kertas

[=] Masukkan Kegunaan Barang : Rokok

[i] Anda akan memasukkan barang baru ke server.

[i] Data berhasil ditambahkan.


```
[=] Masukkan Id/name Kategori : 6

[i] Kategori yang anda pilih adalah Medicine

[=] Masukkan Nama Barang : Parahmen Sakit Kepala
[=] Masukkan Stock Barang : 25
[=] Masukkan Harga Barang : 2500
[=] Masukkan Brand Barang : Parahmen
[=] Masukkan Komposisi Barang : Parachetamol, Glukamolium
[=] Masukkan Anjuran Penggunaan Barang : Diminum sesudah makan
[=] Masukkan Kegunaan Barang : Untuk meredakan sakit kepala
```

```
[i] Anda akan memasukkan barang baru ke server.
```

```
[i] Data berhasil ditambahkan.
```

```
[=] Masukkan Id/name Kategori : 7
```

```
[i] Kategori yang anda pilih adalah Home Appliances
```

```
[=] Masukkan Nama Barang : Sapu Super Kuat
[=] Masukkan Stock Barang : 25
[=] Masukkan Harga Barang : 16000
[=] Masukkan Brand Barang : Super Kuat
[=] Masukkan Kegunaan Barang : Menyapu Lantai
```

```
[i] Anda akan memasukkan barang baru ke server.
```

```
[i] Data berhasil ditambahkan.
```

```
[=] Masukkan Id/name Kategori : 1
```

```
[i] Kategori yang anda pilih adalah Beverage
```

```
[=] Masukkan Nama Barang : qwe
[=] Masukkan Stock Barang : 1
[=] Masukkan Harga Barang : 1
[=] Masukkan Netto/Berat Bersih Barang : qwe
[=] Masukkan Brand Barang : qwe
[=] Masukkan Komposisi Barang : qwe
```

```
[i] Anda akan memasukkan barang baru ke server.
```

```
[i] Data berhasil ditambahkan.
```

```
[~] Pilih Menu: 1
```

```
[1] Daftar Barang
```

Id	Nama Barang	Kategori	Stok	Harga	Tanggal Input	Tanggal Kadaluarsa	Berat Bersih	Brand	Komposisi	Kegunaan	Anjuran Pemakaian
26	New Coke	Beverage	50	5500	2022-05-19	2027-05-19	250g	Coca-Cola	Soda, Gula		
27	Tortilla Snack	Food	50	7500	2022-05-19	2027-05-19	50g	Tortilla	Kentang, Garam		
28	Pemutih Bleach	Cleaner	25	2500	2022-05-19	None	100g	Eloach	Pemutih	Untuk mencuci pakaian/kain	
29	Denuvo Sabun Cair	Personal Care	65	12000	2022-05-19	None	120g	Denuvo	Sabun, Pewangi	Membersihkan tubuh	
30	Platinum Cigar	Other	50	25000	2022-05-19	None	500g	Platinum	Tobako, Kertas	Rokok	
31	Parahmen Sakit Kepala	Medicine	25	2500	2022-05-19	2024-05-19		Parahmen	Parachetamol, Glukamolium	Diminum sesudah makan	Untuk meredakan sakit kepala
32	Sapu Super Kuat	Home Appliances	25	16000	2022-05-19	None		Super Kuat		Menyapu Lantai	
33	qwe	Beverage	1	1	2022-05-19	2027-05-19	qwe	qwe	qwe		

```
[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP
```

```
[~] Pilih Menu: |
```

```
[=] Pilih Menu: 3

[i] Anda akan mengubah beberapa atribut barang.

[i] Daftar Barang

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | Nama Barang | Kategori | Stok | Harga | Tanggal Input | Tanggal Kadaluarsa | Berat Bersih | Brand | Komposisi | Kegunaan | Anjuran Pemakaian |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 26 | New Coke | Beverage | 50 | 5500 | 2022-05-19 | 2027-05-19 | 250g | Coca-Cola | Soda, Gula | | |
| 27 | Tortilla Snack | Food | 50 | 7500 | 2022-05-19 | 2027-05-19 | 50g | Tortilla | Kentang, Garam | | |
| 28 | Pemutih Bleach | Cleaner | 25 | 2500 | 2022-05-19 | None | 100g | Bleach | Pemutih | Untuk mencuci pakaian/kain | |
| 29 | Denuvo Sabun Cair | Personal Care | 65 | 12000 | 2022-05-19 | None | 120g | Denuvo | Sabun, Pewangi | Membersihkan tubuh | |
| 30 | Platinum Cigar | Other | 50 | 25000 | 2022-05-19 | None | 500g | Platinum | Tobako, Kertas | Rokok | |
| 31 | Parahmen Sakit Kepala | Medicine | 25 | 2500 | 2022-05-19 | 2024-05-19 | | Parahmen | Paracetamol, Glukamolium | Diminum sesudah makan | Untuk meredakan sakit kepala |
| 32 | Sapu Super Kuat | Home Appliances | 25 | 16000 | 2022-05-19 | None | | Super Kuat | | Menyapu Lantai | |
| 33 | qwe | Beverage | 1 | 1 | 2022-05-19 | 2027-05-19 | qwe | qwe | qwe | | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

[i] Masukkan ID atau name barang untuk menghapus.
[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.
[=] Masukkan ID atau name barang : 33

[i] Barang yang anda pilih adalah qwe
[i] Apakah anda yakin ingin menghapus data ini?
[=] Y/N : y

[i] Data berhasil dihapus.

[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP

[=] Pilih Menu: 0

[i] Daftar Barang

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | Nama Barang | Kategori | Stok | Harga | Tanggal Input | Tanggal Kadaluarsa | Berat Bersih | Brand | Komposisi | Kegunaan | Anjuran Pemakaian |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 26 | New Coke | Beverage | 50 | 5500 | 2022-05-19 | 2027-05-19 | 250g | Coca-Cola | Soda, Gula | | |
| 27 | Tortilla Snack | Food | 50 | 7500 | 2022-05-19 | 2027-05-19 | 50g | Tortilla | Kentang, Garam | | |
| 28 | Pemutih Bleach | Cleaner | 25 | 2500 | 2022-05-19 | None | 100g | Bleach | Pemutih | Untuk mencuci pakaian/kain | |
| 29 | Denuvo Sabun Cair | Personal Care | 65 | 12000 | 2022-05-19 | None | 120g | Denuvo | Sabun, Pewangi | Membersihkan tubuh | |
| 30 | Platinum Cigar | Other | 50 | 25000 | 2022-05-19 | None | 500g | Platinum | Tobako, Kertas | Rokok | |
| 31 | Parahmen Sakit Kepala | Medicine | 25 | 2500 | 2022-05-19 | 2024-05-19 | | Parahmen | Paracetamol, Glukamolium | Diminum sesudah makan | Untuk meredakan sakit kepala |
| 32 | Sapu Super Kuat | Home Appliances | 25 | 16000 | 2022-05-19 | None | | Super Kuat | | Menyapu Lantai | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] HAPUS DATA ITEM
[4] KELUAR DARI APP

[=] Pilih Menu: 1

[=] Pilih Menu: 4

[i] Good Bye

D:\Univ\Mata Kuliah\SEMESTER 2\PBOB\Pert12-13 Kelompok\tugas>
```

Dokumentasi :



