

Pertemuan 13

Proyek Transaksi Penjualan



CAPAIAN PEMBELAJARAN

1. Mengimplentasikan mysql dan python dalam kasus sehari-hari



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. Sistem Operasi Linux/Window 10
2. Python 3.x
3. Teks Editor
4. XAMPP



DASAR TEORI

Rancangan Struktur Tabel Tabel:

1. Rancangan struktur tabel barang:

No	Nama Field	Type	Null	Key	Keterangan
1	kode	char(5)	NO	PRI	Kode barang
2	nama_barang	varchar(50)	YES		nama barang
3	satuan	varchar(20)	YES		satuan
4	harga	int(11)	YES		harga barang
5	jumlah	int(11)	YES		stok/jumlah

2. Rancangan struktur tabel pelanggan:

No	Nama Field	Type	Null	Key	Keterangan
1	id	int(11)	NO	PRI	id pelanggan
2	nama	varchar(50)	YES		nama pelanggan

3	alamat	varchar(75)	YES		alamat
4	jenis_kel	char(1)	YES		jenis kelamin L=Laki2, P=Perempuan
5	tanggal_lahir	date	YES		tanggal Lahir

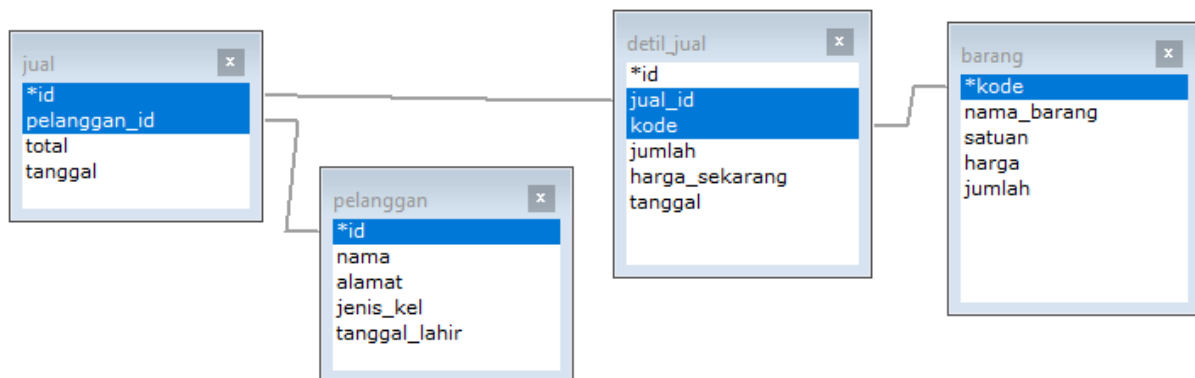
3. Rancangan struktur tabel jual:

No	Nama Field	Type	Null	Key	Keterangan
1	id	int(11)	NO	PRI	no urut/no transaksi (otomatis)
2	pelanggan_id	int(11)	YES		relasi dengan tabel pelanggan
3	total	int(11)	YES		total nilai transaksi
4	tanggal	date	YES		tanggal transaksi

4. Rancangan struktur tabel detail_jual:

No	Nama Field	Type	Null	Key	Keterangan
1	id	int(11)	NO	PRI	no urut otomatis
2	jual_id	int(11)	YES		no urut jual (no transaksi), relasi dengan tabel jual
3	kode	char(5)	YES		kode barang
4	jumlah	int(11)	YES		jumlah yang dijual
5	harga_sekarang	int(11)	YES		harga saat ini
6	tanggal	date	YES		tanggal transaksi

Relasi Tabel



PRAKTIK

1. Buatlah struktur tabel barang (bisa menggunakan modul sebelumnya)

```
CREATE TABLE `barang` (
  `kode` char(5) NOT NULL COMMENT 'Kode barang',
```

```

`nama_barang` varchar(50) DEFAULT NULL COMMENT 'nama barang',
`satuan` varchar(20) DEFAULT NULL COMMENT 'satuan',
`harga` int(11) DEFAULT NULL COMMENT 'harga barang',
`jumlah` int(11) DEFAULT NULL COMMENT 'stok/jumlah',
PRIMARY KEY (`kode`)
);

```

isi rekaman:

```

insert into `barang` (`kode`, `nama_barang`, `satuan`, `harga`, `jumlah`) values
('10001', 'Meja Tulis', 'Unit', 1250000, 1),
('10002', 'Meja Komputer', 'Unit', 1250000, 0),
('10003', 'Meja Belajar', 'Unit', 1200000, 4),
('10004', 'Kursi Lipat', 'Unit', 310000, 2);

```

2. Buatlah struktur tabel pelanggan, dengan perintah skrip sql berikut:

```

CREATE TABLE `pelanggan` (
  `id` int(11) NOT NULL COMMENT 'id pelanggan',
  `nama` varchar(50) DEFAULT NULL COMMENT 'nama pelanggan',
  `alamat` varchar(75) DEFAULT NULL COMMENT 'alamat',
  `jenis_kel` char(1) DEFAULT NULL COMMENT 'jenis kelamin L=Laki2,
P=Perempuan',
  `tanggal_lahir` date DEFAULT NULL COMMENT 'tanggal Lahir',
  PRIMARY KEY (`id`)
);

```

Menambah rekaman pelanggan:

```

insert into `pelanggan` (`id`, `nama`, `alamat`, `jenis_kel`, `tanggal_lahir`)
values
(1, 'Ana Rihana', 'Yogyakarta', 'P', '2001-10-10'),
(2, 'Nugraha', 'Bandung', 'L', '2000-11-10');

```

3. Buatlah struktur tabel jual dengan perintah skrip sql berikut:

```

CREATE TABLE `jual` (
  `id` int(11) NOT NULL AUTO_INCREMENT COMMENT 'no urutan/no transaksi
(otomatis)',
  `pelanggan_id` int(11) DEFAULT NULL COMMENT 'relasi dengan tabel pelanggan',
  `total` int(11) DEFAULT 0 COMMENT 'total nilai transaksi',
  `tanggal` date DEFAULT NULL COMMENT 'tanggal transaksi',
  PRIMARY KEY (`id`)
);

```

4. Buatlah struktur tabel detail_jual, dengan perintah skrip sql berikut:

```

CREATE TABLE `detail_jual` (
  `id` int(11) NOT NULL AUTO_INCREMENT COMMENT 'no urutan otomatis',
  `jual_id` int(11) DEFAULT NULL COMMENT 'no urutan jual, relasi dengan tabel
jual',
  `kode` char(5) DEFAULT NULL COMMENT 'kode barang',
  `jumlah` int(11) DEFAULT NULL COMMENT 'jumlah yang dibeli',
  `harga_sekarang` int(11) DEFAULT NULL COMMENT 'harga saat ini',
  `tanggal` date DEFAULT NULL COMMENT 'tanggal transaksi',

```

```
PRIMARY KEY (`id`)  
);
```

5. Buatlah program Koneksi, untuk user, database, password, disediakan sendiri:

```
*****  
# koneksi.py  
# NIM :.....  
# NAMA :.....  
*****  
import mysql.connector  
from mysql.connector import Error  
  
try:  
    konek = mysql.connector.connect(host='localhost',  
                                    database='kuliah_algo',  
                                    user='root',  
                                    password='')  
  
#cek jika terjadi kesalahan  
except Error as e:  
    print("Tidak bisa koneksi, terjadi kesalahn ", e)
```

6. Fungsi/sub program baca(), fungsi ini dibuat secara general, bisa untuk baca1 rekaman tabel sesuai keinginan:

```
# Sub program baca 1 rekaman dengan hasil list  
# tabel = nama_tabel  
# hasil = nama2 field/kolom yang akan dibaca  
# kunci = nama kolom yang digunakan kunci pencarian (biasanya PRIMARY KEY)  
# nilaiKunci=nilai kunci untuk mencari  
# return /mengembalikan hasil rekaman berbentuk Tuple  
# contoh: hasil=baca("barang","harga","kode",kode)  
# -----  
def baca(tabel,hasil,kunci,nilaiKunci):  
    sql="SELECT "+hasil+" FROM "+tabel+" WHERE "+kunci+"="+nilaiKunci  
    cursor = k.konek.cursor()  
    try:  
        cursor.execute(sql)  
        records = cursor.fetchone()  
        return records  
    except k.konek.Error as e:  
        print("Data tidak ada")
```

7. Input Jual, dengan memasukkan ID pelanggan akan membaca data pelanggan kemudian di rekam di tabel jual. Maka buatlah subprogram inputJual()

```
# mulai transaksi jual  
# membaca pelanggan  
# membuat no transaksi atau id
```

```
# secara otomatis (AUTOINCREMENT)
def inputJual():
    id=input("Masukan ID pelanggan : ")
    nama=baca("pelanggan","nama","id",id)
    if nama==None:
        print("Pelanggan tidak ada !")
        quit()

    print("Nama : ",nama[0])
    cursor = k.konek.cursor()
    sql='''INSERT INTO jual(pelanggan_id,tanggal) VALUES(%s,CURRENT_DATE) '''
    #val=(id)
    cursor.execute(sql, (id,))
    k.konek.commit()
    return cursor.lastrowid
```

`return cursor.lastrowid` menghasilkan `id jual` yang terakhir, nantinya digunakan untuk nomor transaksi pada proses `detil_jual`

8. Berikutnya untuk merekam detail transaksi penjualan maka buatlah Sub Program `inputDetilJual()`, seperti pada kode berikut:

```
# sub program inputDetailJual()
# paramter:
# setiap rekamnn detil_jual
# dalam 1 transaksi akan baca id jual yang terakhir
def inputDetilJual(jual_id):
    kode = input("Kode Barang : ")
    hasil = baca('barang','*', 'kode',kode)

    if hasil==None:
        print("Data Barang tidak ada !")
        quit()

    print("Nama barang :",hasil[1],
          ", Satuan : ",hasil[2],
          ", Harga : ",hasil[3])
    jumlah=input("Jumlah : ")
    #no_trans=10
    harga=hasil[3]
    sql='''INSERT INTO detil_jual
(jual_id,kode,jumlah,harga_sekarang,tanggal)
VALUES (%s,%s,%s,%s,CURRENT_DATE) '''
    val = (jual_id, kode, jumlah, harga)
    cursor = k.konek.cursor()
    cursor.execute(sql, val)
    k.konek.commit()

    #setiap transaksi akan menambah rekaman di tabel detil_jual
    #akan mengurangi stok, dengna memanggil subprogram kurangStok()
    kurangiStok(kode,jumlah)
```

9. Fungsi `kurangiStok(kode,jumlah)`, setiap barang yang dibeli makan akan dikurangi stok/jumlahnya, yang terdapat pada tabel barang

```

#sub program kurangi stok barang
#parameter:
# kode = kode barang yang akan dikurangi
# jml  = jumlah barang yang keluar

def kurangiStok(kode,jml):
    cursor = k.konek.cursor()
    sql="UPDATE barang SET jumlah=jumlah-%s WHERE kode=%s"
    val=(jml,kode)
    cursor.execute(sql,val)
    k.konek.commit()

```

```

# tampilkan struk belanja
# berdasarkan jual_id
def tampilkan(jual_id):
    sql='''SELECT detil_jual.kode, barang.nama_barang, detil_jual.jumlah,
            barang.satuan,detil_jual.harga_sekarang,
            detil_jual.harga_sekarang*detil_jual.jumlah as total
            FROM detil_jual, barang
            WHERE detil_jual.kode = barang.kode AND detil_jual.jual_id=%s'''
    cursor = k.konek.cursor()
    cursor.execute(sql,(jual_id,))
    hasil = cursor.fetchall()
    jual = baca('jual','*', 'id',str(jual_id))
    pelanggan=baca('pelanggan','*', 'id',str(jual[1]))
    print("No Transaksi :",jual_id, ", Tanggal : ",jual[3] )
    print("Nama      :",pelanggan[1])
    print("Alamat :",pelanggan[2],"\n")
    print("NOTA BELANJA")
    print("-----")
    print(" Kode   Nama Barang   Jumlah   Satuan   Harga   Jumlah Harga")
    print("-----")
    jum = 0
    for x in hasil:
        print("{:6s}".format(x[0]),
              "{:15s}".format(x[1]),
              "{:6d}".format(x[2]),
              "{:^8s}".format(x[3]),
              "{:8d}".format(x[4]),
              "{:9d}".format(x[5]))
        jum += x[5]
    print("-----")
    print("{:^46s}".format("Jumlah"),"{:10d}".format(jum))
    print("=====")
    print(">>>>>> Terima kasih atas kunjungan Anda <<<<<<<<")

# mengubah/menyimpan total beanjan pertransaksi
sql="UPDATE jual SET total=%s WHERE id=%s"
val=(jum,jual_id)
cursor = k.konek.cursor()
cursor.execute(sql,val)
k.konek.commit()

```

10. Program Utama

```
lagi="Y"
while lagi=="y" or lagi=="Y":
    os.system('cls')
    jual_id = inputJual()
    detJual="Y"

    while detJual=="Y" or detJual=="y":
        inputDetilJual(jual_id)
        detJual=input("Tambah Lagi? [Y/T]: ")
        if detJual=="t":
            os.system('cls')
            tampilkan(jual_id)
    print("")
    lagi=input("Transaksi Lagi? [Y/T]: ")
```

Pengujian:

```
Masukan ID pelanggan : : 1
Nama : Ana Rihana
Kode Barang : 10003
Nama barang : Meja Belajar , Satuan : Unit , Harga : 1200000
Jumlah : 1
Tambah Lagi? [Y/T]: y
Kode Barang : 10001
Nama barang : Meja Tulis , Satuan : Unit , Harga : 1250000
Jumlah : 1
Tambah Lagi? [Y/T]: y
Kode Barang : 10004
Nama barang : Kursi Lipat , Satuan : Unit , Harga : 310000
Jumlah : 1
Tambah Lagi? [Y/T]: █
```

No Transaksi : 49 , Tanggal : 2021-12-30
Nama : Ana Rihana
Alamat : Yogyakarta

NOTA BELANJA

Kode	Nama Barang	Jumlah	Satuan	Harga	Jumlah Harga
10003	Meja Belajar	1	Unit	1200000	1200000
10001	Meja Tulis	1	Unit	1250000	1250000
10004	Kursi Lipat	1	Unit	310000	310000
Jumlah					2760000

>>>>>> Terima kasih atas kunjungan Anda <<<<<<<<

Transaksi Lagi? [Y/T]: █



LATIHAN



TUGAS



REFERENSI
