

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : ALGORITMA PEMOGRAMAN 3B
Kelas : 2IA11
Praktikum ke- : 3
Tanggal : 4 Januari 2024
Materi : KONSEP DAN KONEKSI BASIS DATA DI GOLANG
NPM : 51422161
Nama : MUHAMMAD TARMIDZI BARIQ
Ketua Asisten : SRI MEINIS
Jumlah Lembar : 3



LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2024

1. system yang dirancang untuk menyimpan, menghapus, mengupdate, mengelola data secara terstruktur aplikasi yang digunakan bisa XAMPP, Firebase, Oracle, dll. Terdapat relasi yang digunakan untuk menghubungkan beberapa table menggunakan primary key dan foreign key

2.

```
ntf("Data berhasil ditambah\n")

// update data
err = db.Exec("update tbl_barang set nama_barang = ? where kode_barang = ?", "kALUNG", 1234)
```

DEBUG CONSOLE COMMENTS

▼ TERMINAL

```
forward a
ally
he

Data berhasil ditambah
1215 | Celana Joger | 40.000 | Pakaian
1231 | Jam Tangan | 150000 | aksesoris
1234 | kALUNG | 500.000 | aksesoris
1235 | Topi | 50000 | aksesoris
12345 | Baju Merah | 20.000 | Perlengkapan
PS C:\CODING\LABTI SEMESTER 3\mtarmidzibariq\m3> go run main.go
```

Ln 34, Col 88 Tab Size: 4 UTF-8 CRLF Go Go Live Analysis Tools Missing

☐ Tampilkan semua Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by key

Extra options

<div>↳ ⌵</div>				kode_barang	nama_barang	harga_barang	jenis_barang
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	1215	Celana Joger	40.000	Pakaian
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	1231	Jam Tangan	150000	aksesoris
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	1234	kALUNG	500.000	aksesoris
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	1235	Topi	50000	aksesoris
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	12345	Baju Merah	20.000	Perlengkapan

☐ Pilih Semua Dengan pilihan: Ubah Salin Hapus Ekspor

☐ Tampilkan semua Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by key

```
package main

import (
    "database/sql"
    "fmt"

    _ "github.com/go-sql-driver/mysql"
)
```

```

type tbl_barang struct {
    kode_barang    int
    nama_barang    string
    harga_barang   string
    jenis_barang   string
}

func connect() (*sql.DB, error) {
    db, err := sql.Open("mysql", "root:@tcp(127.0.0.1:3306)/toko_bariq")
    if err != nil {
        return nil, err
    }
    return db, nil
}

func sqlQuery() {
    db, err := connect()
    if err != nil {
        fmt.Println(err.Error())
        return
    }
    defer db.Close()

    //insert data
    _, err = db.Exec("insert into tbl_barang values (?, ?, ?, ?)", 1231, "Jam
Tangan", "150000", "aksesoris")
    if err != nil {
        fmt.Println(err.Error())
        return
    }
    fmt.Printf("Data berhasil ditambah\n")

    // Update data
    // _, err = db.Exec("update tbl_barang set nama_barang= ? where
kode_barang = ?", "KALUNG", 1234)
    // if err != nil {
    //     fmt.Println(err.Error())
    //     return
    // }
    // fmt.Printf("Data berhasil diubah\n")

    // Hapus Data
    // _, err = db.Exec("delete from tbl_barang where kode_barang=?", 12345)
    // if err != nil {
    //     fmt.Println(err.Error())
    //     return
    // }
}

```

```

// }
// fmt.Printf("Data berhasil dihapus\n")

//select data
rows, err := db.Query("select * from tbl_barang")
if err != nil {
    fmt.Println(err.Error())
    return
}
defer rows.Close()

var result []tbl_barang

for rows.Next() {
    var each = tbl_barang{}
    var err = rows.Scan(&each.kode_barang, &each.nama_barang,
&each.harga_barang, &each.jenis_barang)

    if err != nil {
        fmt.Println(err.Error())
        return
    }

    result = append(result, each)
}
if err = rows.Err(); err != nil {
    fmt.Println(err.Error())
    return
}

for _, each := range result {
    fmt.Print(each.kode_barang)
    fmt.Print(" | ")
    fmt.Print(each.nama_barang)
    fmt.Print(" | ")
    fmt.Print(each.harga_barang)
    fmt.Print(" | ")
    fmt.Print(each.jenis_barang, "\n")
}
}

func main() {
    sqlQuery()
}

```

Import Packages:

database/sql: Paket standar Go untuk interaksi dengan database SQL.

_ "github.com/go-sql-driver/mysql": Mengimpor driver MySQL secara anonim untuk digunakan oleh paket database/sql. Garis bawah (_) mengindikasikan bahwa hanya inisialisasinya yang dibutuhkan, tanpa memerlukan akses langsung ke fungsi atau variabel dalam paket tersebut.

Definisi Struktur Data (tbl_barang):

Struktur tbl_barang mendefinisikan struktur data untuk tabel barang dengan kolom kode_barang, nama_barang, harga_barang, dan jenis_barang.

Fungsi Connect:

connect(): Fungsi ini membuka koneksi ke database MySQL menggunakan kredensial yang disediakan. Alamat database adalah localhost (127.0.0.1) pada port 3306, dengan nama database toko_bariq.

Fungsi sqlQuery:

Fungsi ini bertanggung jawab untuk menjalankan operasi CRUD.

Insert Data: Menambahkan data baru ke dalam tabel tbl_barang.

Update Data: Memperbarui nama_barang di tabel tbl_barang berdasarkan kode_barang.

Delete Data: Menghapus data dari tabel tbl_barang berdasarkan kode_barang.

Select Data: Mengambil semua data dari tabel tbl_barang dan menampilkannya.

Setiap operasi diikuti dengan penanganan error dan pesan konfirmasi jika operasi berhasil.

Fungsi main:

Fungsi main adalah titik masuk program dan hanya memanggil fungsi sqlQuery.

Penanganan Error dan Penutupan Koneksi:

Setiap panggilan ke database diikuti dengan penanganan error untuk memastikan program tidak berhenti jika terjadi kesalahan.

Koneksi ke database ditutup (`defer db.Close()`) setelah operasi selesai untuk menghindari kebocoran sumber daya.

Iterasi dan Tampilan Hasil Query:

Hasil dari query select diiterasi, dan setiap baris hasil diambil datanya, kemudian ditambahkan ke slice result.

Setelah itu, hasil query ditampilkan ke konsol.

Secara keseluruhan, program ini mendemonstrasikan penggunaan dasar dari operasi database dalam Go dengan koneksi ke MySQL, menampilkan penerapan CRUD dalam konteks pengelolaan data barang.