

ACT



Mata Praktikum : GOLANG INTERMEDIET
Kelas : 3IA20
Praktikum ke- : 3
Tanggal : 16 May 2023
Materi : (CRUD MYSQL PADA GOLANG)
NPM : 50420900
Nama : Muhammad Reza Hidayat
Ketua Asisten :
Nama Asisten :
Paraf Asisten :
Jumlah Lembar :



LABORATORIUM TEKNIK INFORMATIKA
UNIVERSITAS GUNADARMA

2022

A. LISTING PROGRAM

Database.go

```
reza_50420900_pert3 - database.go

1 package model
2
3 import (
4     "database/sql"
5     "fmt"
6
7     _ "github.com/go-sql-driver/mysql"
8 )
9
10 type Table interface {
11     Name() string
12     Field() ([]string, []interface{})
13 }
14
15 func Connect(username string, password string, host string, database string) (*sql.DB, error) {
16     conn := fmt.Sprintf("%s:%s@tcp(%s:3306)/%s", username, password, host, database)
17     db, err := sql.Open("mysql", conn)
18     return db, err
19 }
20
21 func CreateDB(db *sql.DB, name string) error {
22     query := fmt.Sprintf("CREATE DATABASE %v", name)
23     _, err := db.Exec(query)
24     return err
25 }
26
27 func CreateTable(db *sql.DB, query string) error {
28     _, err := db.Exec(query)
29     return err
30 }
31
32 func DropDB(db *sql.DB, name string) error {
33     query := fmt.Sprintf("DROP DATABASE %v", name)
34     _, err := db.Exec(query)
35     return err
36 }
37
```

Databse_test.go

```
reza_50420900_pert3 - database_test.go

1 package test
2
3 import (
4     "database/sql"
5     "fmt"
6     "reza_50420900_pert3/model"
7     "testing"
8 )
9
10 var username, password, host, namaDB, defaultDB string
11
12 func init() {
13     username = "root"
14     password = ""
15     host = "localhost"
16     namaDB = "reza_50420900_pert3"
17     defaultDB = "mysql"
18 }
19 func TestDatabase(t *testing.T) {
20
21     t.Run("Testing untuk membuat database", func(t *testing.T) {
22         db, err := model.Connect(username, password, host, defaultDB)
23
24         defer db.Close()
25
26         if err != nil {
27             t.Fatal(err)
28         }
29
30         err = model.CreateDB(db, namaDB)
31
32         if err != nil {
33             t.Fatal(err)
34         }
35     })
36
37     t.Run("Testing untuk memeriksa koneksi database", func(t *testing.T) {
38
39         db, err := model.Connect(username, password, host, defaultDB)
40
41         defer db.Close()
42
43         if err != nil {
44             t.Fatal(err)
45         }
46     })
47 }
48 func initDatabase() (*sql.DB, error) {
49
50     dbInit, err := model.Connect(username, password, host, defaultDB)
51
52     if err != nil {
53         fmt.Println("Gagal melakukan koneksi")
54     }
55
56     if err = model.DropDB(dbInit, namaDB); err != nil {
57         fmt.Println("Gagal menghapus database")
58         return nil, err
59     }
60
61     if err = model.CreateDB(dbInit, namaDB); err != nil {
62         fmt.Println("Gagal membuat database")
63         return nil, err
64     }
65
66     dbInit.Close()
67
68     db, err := model.Connect(username, password, password, namaDB)
69
70     if err != nil {
71         fmt.Println("Gagal melakukan koneksi")
72         return nil, err
73     }
74
75     if err = model.CreateTable(db, model.TabelMahasiswa); err != nil {
76         fmt.Println("Gagal membuat table mahasiswa")
77         return nil, err
78     }
79
80     if err = model.CreateTable(db, model.TabelMatkul); err != nil {
81         fmt.Println("Gagal membuat table matkul")
82         return nil, err
83     }
84
85     if err = model.CreateTable(db, model.TabelNilai); err != nil {
86         fmt.Println("Gagal membuat table nilai")
87         return nil, err
88     }
89
90     return db, nil
91 }
```

Output database_test.go

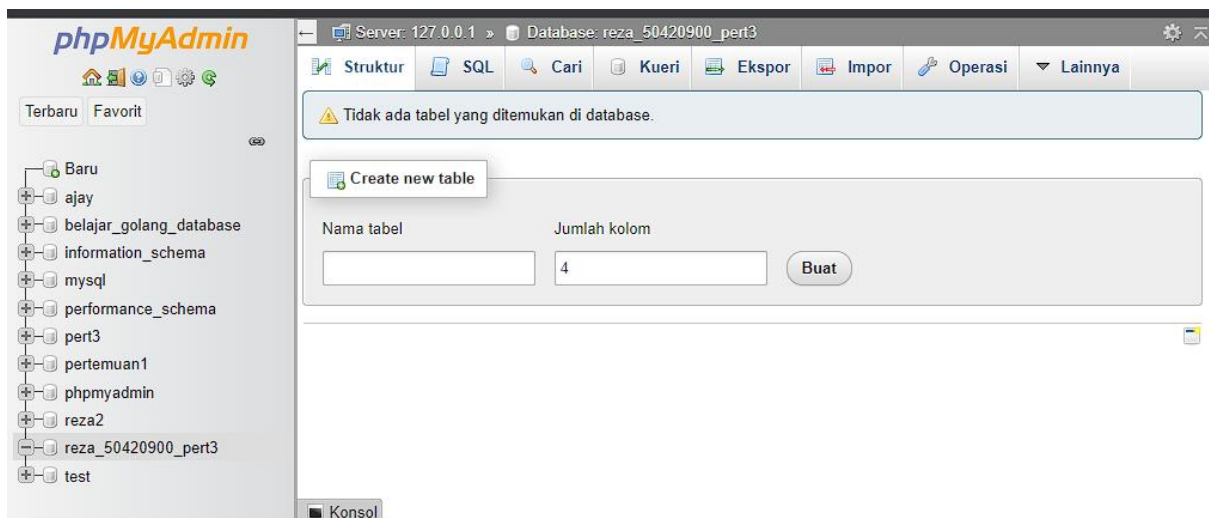
```
reza_50420900_pert3 - database_test.go

> Test run finished at 11/2/2022, 10:57:59 PM <

Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run ^TestDatabase$ reza_50420900_pert3/test

=== RUN   TestDatabase
=== RUN   TestDatabase/Testing_untuk_membuat_database
=== RUN   TestDatabase/Testing_untuk_memeriksa_koneksi_database
--- PASS: TestDatabase (0.01s)
    --- PASS: TestDatabase/Testing_untuk_membuat_database (0.01s)
    --- PASS: TestDatabase/Testing_untuk_memeriksa_koneksi_database (0.00s)
PASS
ok      reza_50420900_pert3/test      0.243s

> Test run finished at 11/2/2022, 10:58:23 PM <
```



mahasiswa.go

```
reza_50420900_part3 - mahasiswa.go

1 package model
2
3 import (
4     "database/sql"
5     "fmt"
6     "strings"
7 )
8
9 var TabelMahasiswa string = `
10 CREATE TABLE mahasiswa(
11     npm VARCHAR(10) PRIMARY KEY,
12     nama VARCHAR(30),
13     kelas VARCHAR(10)
14 );
15 `
16
17 type Mahasiswa struct {
18     NPM    string `json:"NPM"`
19     Nama   string `json:"Nama"`
20     Kelas  string `json:"Kelas"`
21 }
22
23 func (m *Mahasiswa) Fields() ([]string, []interface{}) {
24     fields := []string{"npm", "nama", "kelas"}
25     temp := []interface{}{&m.NPM, &m.Nama, &m.Kelas}
26     return fields, temp
27 }
28
29 func (m *Mahasiswa) Structur() *Mahasiswa {
30     return &Mahasiswa{}
31 }
32
33 func (m *Mahasiswa) Insert(db *sql.DB) error {
34     query := fmt.Sprintf("INSERT INTO %v values(?,?,?)", "mahasiswa")
35     _, err := db.Exec(query, &m.NPM, &m.Nama, &m.Kelas)
36     return err
37 }
38
39 func (m *Mahasiswa) Update(db *sql.DB, data map[string]interface{}) error {
40
41     var kolom = []string{}
42     var args []interface{}
43
44     i := 1
45
46     // Ini loop data untuk dimasukan kedalam set,
47     for key, value := range data {
48         updateData := fmt.Sprintf("%v = ?", strings.ToLower(key))
49         kolom = append(kolom, updateData)
50         args = append(args, value)
51         i++
52     }
53
54     // Ubah array menjadi string dengan pemisah koma
55     dataUpdate := strings.Join(kolom, ",")
56
57     query := fmt.Sprintf("UPDATE %s SET %s WHERE %s = ?", "mahasiswa", dataUpdate, "NPM")
58     args = append(args, m.NPM)
59     // Exec dengan query yang ada
60     _, err := db.Exec(query, args...)
61     return err
62 }
63
64 func (m *Mahasiswa) Delete(db *sql.DB) error {
65     query := fmt.Sprintf("DELETE FROM %s WHERE %s = ?", "mahasiswa", "NPM")
66     // Exec dengan query yang ada
67     _, err := db.Exec(query, m.NPM)
68     return err
69 }
70
71 func GetMahasiswa(db *sql.DB, id string) (*Mahasiswa, error) {
72
73     m := &Mahasiswa{}
74     each := m.Structur()
75     _, dst := each.Fields()
76     query := fmt.Sprintf("SELECT * FROM %s WHERE %s = ?", "mahasiswa", "NPM")
77
78     // isinya akan dimasukan kedalam var dst yang dideklarasikan diatas
79     err := db.QueryRow(query, id).Scan(dst...)
80     if err != nil {
81         fmt.Println(err)
82         return nil, err
83     }
84     return each, nil
85 }
86
87 func GetAllMahasiswa(db *sql.DB) ([]*Mahasiswa, error) {
88     m := &Mahasiswa{}
89     query := fmt.Sprintf("SELECT * FROM %s", "mahasiswa")
90     data, err := db.Query(query)
91     if err != nil {
92         return nil, err
93     }
94     defer data.Close()
95
96     var result []*Mahasiswa
97
98     for data.Next() {
99         each := m.Structur()
100         _, dst := each.Fields()
101
102         err := data.Scan(dst...)
103         if err != nil {
104             return nil, err
105         }
106         fmt.Println(each)
107         result = append(result, each)
108     }
109
110     return result, nil
111 }
112
113 }
114
```

Mahasiswa_test.go

```
reza_50420900_pert3 - mahasiswa_test.go

1 package test
2
3 import (
4     "reza_50420900_pert3/model" //sesuaikan dengan nama folder (case sensitive)
5     "testing"
6 )
7
8 func TestMahasiswa(t *testing.T) {
9
10     var dataInsertMhs = []model.Mahasiswa{
11         model.Mahasiswa{
12             NPM:    "50420900",
13             Nama:    "REZA",
14             Kelas:   "3IA20",
15         },
16         model.Mahasiswa{
17             NPM:    "19283746",
18             Nama:    "Doremi Budi",
19             Kelas:   "4KA20",
20         },
21         model.Mahasiswa{
22             NPM:    "44444444",
23             Nama:    "DoBud",
24             Kelas:   "4KA21",
25         },
26     }
27
28     db, err := initDatabase()
29
30     if err != nil {
31         t.Fatal(err)
32     }
33
34     t.Run("Testing insert mahasiswa", func(t *testing.T) {
35         for _, dataInsert := range dataInsertMhs {
36             err := dataInsert.Insert(db)
37             if err != nil {
38                 t.Fatal(err)
39             }
40         }
41     })
42
43     t.Run("Testing update mahasiswa", func(t *testing.T) {
44         var updateData = map[string]interface{}{
45             "nama": "Abdi Teh Ayeuna",
46         }
47
48         data := dataInsertMhs[0]
49         if err := data.Update(db, updateData); err != nil {
50             t.Fatal(err)
51         }
52     })
53
54     t.Run("Testing Get mahasiswa", func(t *testing.T) {
55         _, err := model.GetMahasiswa(db, "44444444")
56         if err != nil {
57             t.Fatal(err)
58         }
59     })
60
61     t.Run("Testing Get mahasiswa", func(t *testing.T) {
62         _, err := model.GetAllMahasiswa(db)
63         if err != nil {
64             t.Fatal(err)
65         }
66     })
67
68     t.Run("Testing delete mahasiswa", func(t *testing.T) {
69         data := dataInsertMhs[0]
70         if err := data.Delete(db); err != nil {
71             t.Fatal(err)
72         }
73     })
74
75     defer db.Close()
76 }
77
78
```

OUTPUT Mahasiswa_test.go

```

reza_50420900_pert3 - mahasiswa_test.go

Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run ^TestMahasiswa$ reza_50420900_pert3/test

=== RUN   TestMahasiswa
=== RUN   TestMahasiswa/Testing_insert_mahasiswa
=== RUN   TestMahasiswa/Testing_update_mahasiswa
=== RUN   TestMahasiswa/Testing_Get_mahasiswa
=== RUN   TestMahasiswa/Testing_Get_mahasiswa#01
&{19283746 Doremi Budi 4KA20}
&{44444444 DoBud 4KA21}
&{50420900 Abdi Teh Ayeuna 3TA20}
=== RUN   TestMahasiswa/Testing_delete_mahasiswa
--- PASS: TestMahasiswa (0.08s)
    --- PASS: TestMahasiswa/Testing_insert_mahasiswa (0.01s)
    --- PASS: TestMahasiswa/Testing_update_mahasiswa (0.00s)
    --- PASS: TestMahasiswa/Testing_Get_mahasiswa (0.00s)
    --- PASS: TestMahasiswa/Testing_Get_mahasiswa#01 (0.00s)
    --- PASS: TestMahasiswa/Testing_delete_mahasiswa (0.00s)
PASS
ok      reza_50420900_pert3/test      0.334s

> Test run finished at 11/2/2022, 11:00:29 PM <

```

phpMyAdmin

Server: 127.0.0.1 » Database: reza_50420900_pert3 » Tabel: mahasiswa

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Lainnya

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/>	1 npm	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lai
<input type="checkbox"/>	2 nama	varchar(30)	utf8mb4_general_ci		Ya	NULL			Ubah Hapus Lai
<input type="checkbox"/>	3 kelas	varchar(10)	utf8mb4_general_ci		Ya	NULL			Ubah Hapus Lai

Pilih Semua Dengan pilihan: Jelajahi Ubah Hapus Utama Unik Indeks

Spasial Teks penuh Add to central columns Remove from central columns

Cetak Usulkan struktur tabel Lacak tabel Move columns Normalisasi

Tambahkan 1 kolom setelah kelas Kirim

Indeks

Konsol

Nama	Jenis	Unik	Dipadatkan	Kolom	Kardinalitas	Penyortiran	Tak Ternilai	Komentar
------	-------	------	------------	-------	--------------	-------------	--------------	----------

matakuliah.go

```
reza_504209002_pert3 - matakuliah.go

1 package model
2
3 import (
4     "database/sql"
5     "fmt"
6     "strings"
7 )
8
9 var TabelMatkul string = `
10 CREATE TABLE matkul(
11     kd_mk VARCHAR(10) PRIMARY KEY,
12     matakuliah VARCHAR(20)
13 );
14 `
15
16 func (m *Matkul) Structur() *Matkul {
17     return &Matkul{}
18 }
19
20 type Matkul struct {
21     Kd_mk      string `json:"Kd_mk"`
22     Matakuliah string `json:"Matakuliah"`
23 }
24
25 func (m *Matkul) Fields() ([]string, []interface{}) {
26     fields := []string{"kd_mk", "matakuliah"}
27     temp := []interface{}{&m.Kd_mk, &m.Matakuliah}
28     return fields, temp
29 }
30
31 func (m *Matkul) Insert(db *sql.DB) error {
32     query := fmt.Sprintf("INSERT INTO %v values(?,?)", "matkul")
33     _, err := db.Exec(query, &m.Kd_mk, &m.Matakuliah)
34     return err
35 }
36
37 func (m *Matkul) Update(db *sql.DB, data map[string]interface{}) error {
38
39     var kolom = []string{}
40     var args []interface{}
41     i := 1
42
43     // Ini loop data untuk dimasukan kedalam set,
44     for key, value := range data {
45         updateData := fmt.Sprintf("%v = ?", strings.ToLower(key))
46         kolom = append(kolom, updateData)
47         args = append(args, value)
48         i++
49     }
50
51     // Ubah array menjadi string dengan pemisah koma
52     dataUpdate := strings.Join(kolom, ",")
53
54     query := fmt.Sprintf("UPDATE %s SET %s WHERE %s = ?", "matkul", dataUpdate, "kd_mk")
55     args = append(args, m.Kd_mk)
56     // Exec dengan query yang ada
57     _, err := db.Exec(query, args...)
58     return err
59 }
60
61 func (m *Matkul) Delete(db *sql.DB) error {
62     query := fmt.Sprintf("DELETE FROM %s WHERE %s = ?", "matkul", "kd_mk")
63     // Exec dengan query yang ada
64     _, err := db.Exec(query, m.Kd_mk)
65     return err
66 }
67
68 func GetMatkul(db *sql.DB, id string) (*Matkul, error) {
69
70     m := &Matkul{}
71     each := m.Structur()
72     _, dst := each.Fields()
73     query := fmt.Sprintf("SELECT * FROM %s WHERE %s = ?", "matkul", "kd_mk")
74
75     // isinya akan dimasukan kedalam var dst yang dideklarasikan diatas
76     err := db.QueryRow(query, id).Scan(dst...)
77     if err != nil {
78         return nil, err
79     }
80     return each, nil
81 }
82
83 func GetAllMatkul(db *sql.DB) ([]*Matkul, error) {
84     m := &Matkul{}
85     query := fmt.Sprintf("SELECT * FROM %s", "matkul")
86     data, err := db.Query(query)
87     if err != nil {
88         return nil, err
89     }
90
91     defer data.Close()
92
93     var result []*Matkul
94     for data.Next() {
95         each := m.Structur()
96         _, dst := each.Fields()
97         err := data.Scan(dst...)
98
99         if err != nil {
100             return nil, err
101         }
102         result = append(result, each)
103     }
104
105     return result, nil
106 }
107
108 }
```


matakuliah_test.go

```
reza_50420900_pert3 - mahasiswa_test.go

1 package test
2
3 import (
4     "reza_50420900_pert3/model" //sesuaikan dengan nama folder (case sensitive)
5     "testing"
6 )
7
8 func TestMahasiswa(t *testing.T) {
9
10     var dataInsertMhs = []model.Mahasiswa{
11         model.Mahasiswa{
12             NPM:   "50420900",
13             Nama:  "REZA",
14             Kelas: "3IA20",
15         },
16         model.Mahasiswa{
17             NPM:   "19283746",
18             Nama:  "Doremi Budi",
19             Kelas: "4KA20",
20         },
21         model.Mahasiswa{
22             NPM:   "44444444",
23             Nama:  "DoBud",
24             Kelas: "4KA21",
25         },
26     }
27
28     db, err := initDatabase()
29
30     if err != nil {
31         t.Fatal(err)
32     }
33
34     t.Run("Testing insert mahasiswa", func(t *testing.T) {
35         for _, dataInsert := range dataInsertMhs {
36             err := dataInsert.Insert(db)
37             if err != nil {
38                 t.Fatal(err)
39             }
40         }
41     })
42
43     t.Run("Testing update mahasiswa", func(t *testing.T) {
44         var updateData = map[string]interface{}{
45             "nama": "Abdi Teh Ayeuna",
46         }
47
48         data := dataInsertMhs[0]
49         if err := data.Update(db, updateData); err != nil {
50             t.Fatal(err)
51         }
52     })
53
54     t.Run("Testing Get mahasiswa", func(t *testing.T) {
55         _, err := model.GetMahasiswa(db, "44444444")
56         if err != nil {
57             t.Fatal(err)
58         }
59     })
60
61     t.Run("Testing Get mahasiswa", func(t *testing.T) {
62         _, err := model.GetAllMahasiswa(db)
63         if err != nil {
64             t.Fatal(err)
65         }
66     })
67
68     t.Run("Testing delete mahasiswa", func(t *testing.T) {
69         data := dataInsertMhs[0]
70         if err := data.Delete(db); err != nil {
71             t.Fatal(err)
72         }
73     })
74
75     defer db.Close()
76 }
77
78
```

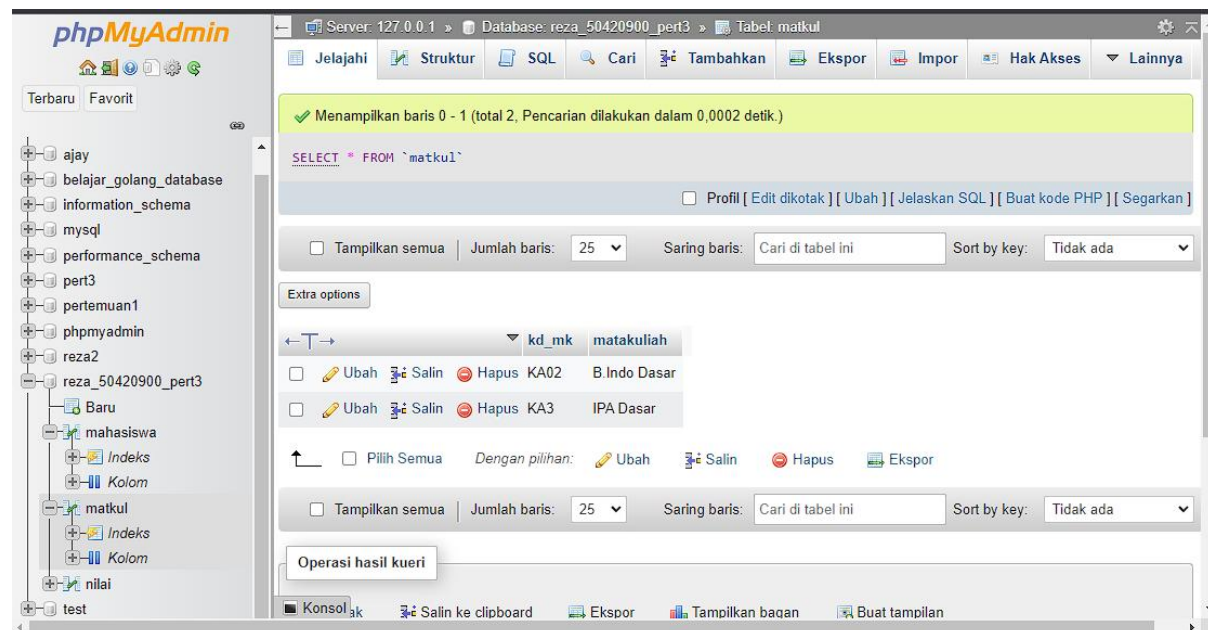
OUTPUT matakuliah_test.go

```
reza_50420900_pert3 - matakuliah_test.go

Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run ^TestMatkul$ reza_50420900_pert3/test

=== RUN   TestMatkul
=== RUN   TestMatkul/Testing_insert_mahasiswwa
=== RUN   TestMatkul/Testing_update_mahasiswwa
=== RUN   TestMatkul/Testing_Get_matkul
=== RUN   TestMatkul/Testing_GetAll_matkul
&{KA01 Algoritma 1}
=== RUN   TestMatkul/Testing_delete_matkul
--- PASS: TestMatkul (0.07s)
    --- PASS: TestMatkul/Testing_insert_mahasiswwa (0.00s)
    --- PASS: TestMatkul/Testing_update_mahasiswwa (0.00s)
    --- PASS: TestMatkul/Testing_Get_matkul (0.00s)
    --- PASS: TestMatkul/Testing_GetAll_matkul (0.00s)
    --- PASS: TestMatkul/Testing_delete_matkul (0.00s)
PASS
ok      reza_50420900_pert3/test      0.313s

> Test run finished at 11/2/2022, 11:02:57 PM <
```



nilai.go

```
1 package model
2
3 import (
4     "database/sql"
5     "fmt"
6     "strings"
7 )
8
9 var TabelNilai string = `
10 CREATE TABLE nilai(
11     id_nilai INT PRIMARY KEY AUTO_INCREMENT,
12     kd_mk VARCHAR(10) ,
13     npm VARCHAR(10) ,
14     uts REAL NOT NULL,
15     uas REAL NOT NULL,
16     total REAL NOT NULL,
17     grade VARCHAR(5) NOT NULL
18 )
19 `
20
21 type Nilai struct {
22     Id_nilai int    `json:"id_nilai"`
23     Kd_mk     string `json:"kd_mk"`
24     NPM       string `json:"npm"`
25     UAS       float64 `json:"uas"`
26     UTS       float64 `json:"uts"`
27     Total     float64 `json:"total"`
28     Grade     string  `json:"grade"`
29 }
30
31 func (m *Nilai) Structur() *Nilai {
32     return &Nilai{}
33 }
34
35 func (m *Nilai) Fields() ([]string, []interface{}) {
36     fields := []string{"id_nilai", "kd_mk", "npm", "uts", "uas", "total", "grade"}
37     temp := []interface{}{&m.Id_nilai, &m.Kd_mk, &m.NPM, &m.UAS, &m.UTS, &m.Total, &m.Grade}
38     return fields, temp
39 }
40
41 func (m *Nilai) Insert(db *sql.DB) error {
42     query := fmt.Sprintf("INSERT INTO %v values(?, ?, ?, ?, ?, ?, ?)", "nilai")
43     _, err := db.Exec(query, &m.Id_nilai, &m.Kd_mk, &m.NPM, &m.UAS, &m.UTS, &m.Total, &m.Grade)
44     return err
45 }
46
47 func (m *Nilai) Update(db *sql.DB, data map[string]interface{}) error {
48
49     var kolom = []string{}
50     var args []interface{}
51
52     i := 1
53
54     // Ini loop data untuk dimasukan kedalam set,
55     for key, value := range data {
56         updateData := fmt.Sprintf("%v = ?", strings.ToLower(key))
57         kolom = append(kolom, updateData)
58         args = append(args, value)
59         i++
60     }
61
62     // Ubah array menjadi string dengan pemisah koma
63     dataUpdate := strings.Join(kolom, ",")
64
65     query := fmt.Sprintf("UPDATE %s SET %s WHERE %s = ?", "nilai", dataUpdate, "npm")
66     args = append(args, m.NPM)
67     // Exec dengan query yang ada
68     _, err := db.Exec(query, args...)
69     return err
70 }
71
72 func (m *Nilai) Delete(db *sql.DB) error {
73     query := fmt.Sprintf("DELETE FROM %s WHERE %s = ?", "nilai", "npm")
74     // Exec dengan query yang ada
75     _, err := db.Exec(query, m.NPM)
76     return err
77 }
78
79 func GetNilai(db *sql.DB, id string) (*Nilai, error) {
80
81     m := &Nilai{}
82     each := m.Structur()
83     _, dst := each.Fields()
84     query := fmt.Sprintf("SELECT * FROM %s WHERE %s = ?", "nilai", "npm")
85
86     // isinya akan dimasukan kedalam var dst yang dideklarasikan diatas
87     err := db.QueryRow(query, id).Scan(dst...)
88     if err != nil {
89         return nil, err
90     }
91     return each, nil
92 }
93
94 func GetAllNilai(db *sql.DB) ([]*Nilai, error) {
95     m := &Nilai{}
96     query := fmt.Sprintf("SELECT * FROM %s", "nilai")
97     data, err := db.Query(query)
98     if err != nil {
99         return nil, err
100     }
101
102     defer data.Close()
103
104     var result []*Nilai
105     for data.Next() {
106         each := m.Structur()
107         _, dst := each.Fields()
108         err := data.Scan(dst...)
109
110         if err != nil {
111             return nil, err
112         }
113         result = append(result, each)
114     }
115
116     return result, nil
117 }
118 }
119
```

Nilai_test.go

```
reza_50420900_pert3 - nilai_test.go

1 package test
2
3 import (
4     "fmt"
5     "reza_50420900_pert3/model"
6     "testing"
7 )
8
9 func TestNilai(t *testing.T) {
10     var dataInsertNilai = []model.Nilai{
11         model.Nilai{
12             Id_nilai: 1,
13             Kd_mk:    "KA02",
14             NPM:      "19283746",
15             UTS:      70,
16             UAS:      70,
17             Total:    70,
18             Grade:    "B",
19         },
20         model.Nilai{
21             Id_nilai: 2,
22             Kd_mk:    "KA03",
23             NPM:      "44444444",
24             UTS:      50,
25             UAS:      50,
26             Total:    50,
27             Grade:    "D",
28         },
29     }
30
31     db, err := initDatabase()
32
33     if err != nil {
34         t.Fatal(err)
35     }
36
37     t.Run("Testing insert nilai", func(t *testing.T) {
38         for _, dataInsert := range dataInsertNilai {
39             err := dataInsert.Insert(db)
40             if err != nil {
41                 t.Fatal(err)
42             }
43         }
44     })
45
46     t.Run("Testing update nilai", func(t *testing.T) {
47         var updateData = map[string]interface{}{
48             "uts": 30,
49             "uas": 30,
50             "total": 30,
51             "grade": "E",
52         }
53         data := dataInsertNilai[0]
54         if err := data.Update(db, updateData); err != nil {
55             t.Fatal(err)
56         }
57     })
58
59     t.Run("Testing GetAll nilai", func(t *testing.T) {
60         // data := dataInsertNilai[0]
61         _, err := model.GetNilai(db, "44444444")
62         if err != nil {
63             t.Fatal(err)
64         }
65         // fmt.Println(result)
66     })
67
68     t.Run("Testing Get nilai", func(t *testing.T) {
69         result, err := model.GetAllNilai(db)
70         if err != nil {
71             t.Fatal(err)
72         }
73
74         fmt.Println(result[0])
75     })
76
77     t.Run("Testing delete nilai", func(t *testing.T) {
78         data := dataInsertNilai[0]
79         if err := data.Delete(db); err != nil {
80             t.Fatal(err)
81         }
82     })
83
84     defer db.Close()
85 }
86
```

OUTPUT Nilai_test.go

```
reza_50420900_pert3 - nilai_test.go

Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run ^TestNilai$ reza_50420900_pert3/test

=== RUN   TestNilai
=== RUN   TestNilai/Testing_insert__nilai
=== RUN   TestNilai/Testing_update__nilai
=== RUN   TestNilai/Testing_GetAll__nilai
=== RUN   TestNilai/Testing_Get__nilai
&{1 KA02 19283746 30 30 30 E}
=== RUN   TestNilai/Testing_delete__nilai
--- PASS: TestNilai (0.07s)
    --- PASS: TestNilai/Testing_insert__nilai (0.00s)
    --- PASS: TestNilai/Testing_update__nilai (0.00s)
    --- PASS: TestNilai/Testing_GetAll__nilai (0.00s)
    --- PASS: TestNilai/Testing_Get__nilai (0.00s)
    --- PASS: TestNilai/Testing_delete__nilai (0.00s)
PASS
ok      reza_50420900_pert3/test      0.318s

> Test run finished at 11/2/2022, 11:07:01 PM <
```

phpMyAdmin

Server: 127.0.0.1 » Database: reza_50420900_pert3 » Tabel: nilai

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Lainnya

Terbaru Favorit

Baru

- ajay
- belajar_golang_database
- information_schema
- mysql
- performance_schema
- pert3
- pertemuan1
- phpmyadmin
- reza2
- reza_50420900_pert3
 - Baru
 - mahasiswa
 - matkul
 - nilai
 - Indeks
 - Kolom
 - test

Menampilkan baris 0 - 0 (total 1, Pencarian dilakukan dalam 0.0002 detik.)

[SELECT * FROM `nilai`](#)

☐ Profil [\[Edit dikotak \]](#) [\[Ubah \]](#) [\[Jelaskan SQL \]](#) [\[Buat kode PHP \]](#) [\[Segarkan \]](#)

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Extra options

	id_nilai	kd_mk	npm	uts	uas	total	grade
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	2	KA03	44444444	50	50	50	D

☐ Pilih Semua Dengan pilihan: ☐ Ubah ☐ Salin ☐ Hapus ☐ Ekspor

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Operasi hasil kueri

☐ Cetak ☐ Salin ke clipboard ☐ Ekspor ☐ Tampilkan bagan ☐ Buat tampilan