

ORM-

Insert/Replace/Save

1. Insert

```
INSERT INTO
```

2. Replace

```
REPLACE INTO
```

3. Save

```
INSERT INTO PostgreSQL, SQL server, Oracle OnConflict
```

```
db.Model(table).Data(g.Map{
    "id": 1,
    "passport": "p1",
    "password": "pw1",
}).OnConflict("id").Save()
```

Replace/Save

Data/

InsertIgnore

```
func (m *Model) InsertIgnore(data ...interface{}) (result sql.Result, err error)
```

InsertAndGetId

ID

```
func (m *Model) InsertAndGetId(data ...interface{}) (lastInsertId int64, err error)
```

OnDuplicate/OnDuplicateEx

OnDuplicate/OnDuplicateExSaveSave/Map

```
OnDuplicate("nickname, age")
OnDuplicate("nickname", "age")
OnDuplicate(g.Map{
    "nickname": gdb.Raw("CONCAT('name_', VALUES(`nickname`))"),
})
OnDuplicate(g.Map{
    "nickname": "passport",
})
```

OnDuplicateEx

Content Menu

- - [Insert/Replace/Save](#)
 - [InsertIgnore](#)
 - [InsertAndGetId](#)
 - [OnDuplicate/OnDuplicateEx](#)
- - [1](#)
 - [2](#)
 - [3](#)
- [RawSQL](#)

1

/DataMap/Struct/Slice

```
// INSERT INTO `user`(`name`) VALUES('john')
g.Model("user").Data(g.Map{"name": "john"}).Insert()

// INSERT IGNORE INTO `user`(`uid`,`name`) VALUES(10000,'john')
g.Model("user").Data(g.Map{"uid": 10000, "name": "john"}).InsertIgnore()

// REPLACE INTO `user`(`uid`,`name`) VALUES(10000,'john')
g.Model("user").Data(g.Map{"uid": 10000, "name": "john"}).Replace()

// INSERT INTO `user`(`uid`,`name`) VALUES(10001,'john') ON DUPLICATE KEY
UPDATE `uid`=VALUES(`uid`),`name`=VALUES(`name`)
g.Model("user").Data(g.Map{"uid": 10001, "name": "john"}).Save()
```

Data/

```
g.Model("user").Insert(g.Map{"name": "john"})
g.Model("user").Replace(g.Map{"uid": 10000, "name": "john"})
g.Model("user").Save(g.Map{"uid": 10001, "name": "john"})
```

struct uid/name/site

```
type User struct {
    Uid  int    `orm:"uid"`
    Name string `orm:"name"`
    Site string `orm:"site"`
}
user := &User{
    Uid: 1,
    Name: "john",
    Site: "https://goframe.org",
}
// INSERT INTO `user`(`uid`,`name`,`site`) VALUES
(1,'john','https://goframe.org')
g.Model("user").Data(user).Insert()
```

2

DataSliceMapStructSQL

```
// INSERT INTO `user`(`name`) VALUES('john_1'),('john_2'),('john_3')
g.Model("user").Data(g.List{
    {"name": "john_1"},
    {"name": "john_2"},
    {"name": "john_3"},
}).Insert()
```

Batch10

```
// INSERT INTO `user`(`name`) VALUES('john_1'),('john_2')
// INSERT INTO `user`(`name`) VALUES('john_3')
g.Model("user").Data(g.List{
    {"name": "john_1"},
    {"name": "john_2"},
    {"name": "john_3"},
}).Batch(2).Insert()
```

3

oracle, dm, mssql

```
// INSERT INTO `user`(`uid`,`name`) VALUES(10000,'john_1'),
(10001,'john_2'),(10002,'john_3')
// ON DUPLICATE KEY UPDATE `uid`=VALUES(`uid`),`name`=VALUES(`name`)
g.Model("user").Data(g.List{
    {"uid":10000, "name": "john_1"},
    {"uid":10001, "name": "john_2"},
    {"uid":10002, "name": "john_3"},
}).Save()
```

RawSQL

`gdb.RawSQL`[SQLORM-RawSQL](#)

```
// INSERT INTO `user`(`id`,`passport`,`password`,`nickname`,`create_time`)
VALUES('id+2','john','123456','now()')
g.Model("user").Data(g.Map{
    "id":          "id+2",
    "passport":    "john",
    "password":    "123456",
    "nickname":    "JohnGuo",
    "create_time": "now()",
}).Insert()
// Error Code: 1136. Column count doesn't match value count at row 1
```

`gdb.Raw`

```
// INSERT INTO `user`(`id`,`passport`,`password`,`nickname`,`create_time`)
VALUES(id+2,'john','123456',now())
g.Model("user").Data(g.Map{
    "id":          gdb.Raw("id+2"),
    "passport":    "john",
    "password":    "123456",
    "nickname":    "JohnGuo",
    "create_time": gdb.Raw("now()"),
}).Insert()
```