

# ORM-



gdb

- null
- :date, datetime, timestampint
- - created\_at
  - updated\_at
  - deleted\_at

CreatedAt, UpdatedAt, DeletedAtTimeMaintainDisabled [ORM](#)



created\_atupdated\_atdeleted\_at

3

created\_at

Insert/InsertIgnore/BatchInsert/BatchInsertIgnore

```
// INSERT INTO `user`(`name`,`created_at`,`updated_at`) VALUES('john',
`2020-06-06 21:00:00`, `2020-06-06 21:00:00`)
g.Model("user").Data(g.Map{"name": "john"}).Insert()

// INSERT IGNORE INTO `user`(`uid`,`name`,`created_at`,`updated_at`) VALUES
(10000,'john', `2020-06-06 21:00:00`, `2020-06-06 21:00:00`)
g.Model("user").Data(g.Map{"uid": 10000, "name": "john"}).InsertIgnore()

// REPLACE INTO `user`(`uid`,`name`,`created_at`,`updated_at`) VALUES
(10000,'john', `2020-06-06 21:00:00`, `2020-06-06 21:00:00`)
g.Model("user").Data(g.Map{"uid": 10000, "name": "john"}).Replace()

// INSERT INTO `user`(`uid`,`name`,`created_at`,`updated_at`) VALUES
(10001,'john', `2020-06-06 21:00:00`, `2020-06-06 21:00:00`) ON DUPLICATE
KEY UPDATE `uid`=VALUES(`uid`),`name`=VALUES(`name`),`updated_at`=VALUES
(`updated_at`)
g.Model("user").Data(g.Map{"uid": 10001, "name": "john"}).Save()
```



Replace

updated\_at

Insert/InsertIgnore/BatchInsert/BatchInsertIgnoreSave/Updateupdated\_atcreated\_at

## Content Menu

- 
- [created\\_at](#)
- [updated\\_at](#)
- [deleted\\_at](#)
  - [Unscoped](#)

```
// UPDATE `user` SET `name`='john guo','updated_at`='2020-06-06 21:00:00'
WHERE name='john'
g.Model("user").Data(g.Map{"name" : "john guo"}).Where("name", "john").
Update()

// UPDATE `user` SET `status`=1,'updated_at`='2020-06-06 21:00:00' ORDER
BY `login_time` asc LIMIT 10
g.Model("user").Data("status", 1).Order("login_time asc").Limit(10).
Update()

// INSERT INTO `user`(`id`,`name`,`update_at`) VALUES(1,'john guo','2020-
12-29 20:16:14') ON DUPLICATE KEY UPDATE `id`=VALUES(`id`,`name`=VALUES
(`name`,`update_at`=VALUES(`update_at`)
g.Model("user").Data(g.Map{"id": 1, "name": "john guo"}).Save()
```



Replace

## deleted\_at

deleted\_at

```
// UPDATE `user` SET `deleted_at`='2020-06-06 21:00:00' WHERE uid=10
g.Model("user").Where("uid", 10).Delete()
```

```
// SELECT * FROM `user` WHERE uid>1 AND `deleted_at` IS NULL
g.Model("user").Where("uid>?", 1).All()
```

deleted\_atdeleted\_at IS NULL

```
// SELECT * FROM `user` AS `u` LEFT JOIN `user_detail` AS `ud` ON (ud.
uid=u.uid) WHERE u.uid=10 AND `u`.`deleted_at` IS NULL AND `ud`.`deleteat`
IS NULL LIMIT 1
g.Model("user", "u").LeftJoin("user_detail", "ud", "ud.uid=u.uid").Where
("u.uid", 10).One()
```

## Unscoped

UnscopedUnscoped

```
// SELECT * FROM `user` WHERE uid>1
g.Model("user").Unscoped().Where("uid>?", 1).All()

// SELECT * FROM `user` AS `u` LEFT JOIN `user_detail` AS `ud` ON (ud.
uid=u.uid) WHERE 1 LIMIT 1
g.Model("user", "u").LeftJoin("user_detail", "ud", "ud.uid=u.uid").Where
("u.uid", 10).Unscoped().One()
```