

ORM-

```
// ("default")
db := g.DB()
```

in

sliceslice?

```
// SELECT * FROM user WHERE uid IN(100,10000,90000)
db.Model("user").Where("uid IN(?,?,?)", 100, 10000, 90000).All()
db.Model("user").Where("uid", g.Slice{100, 10000, 90000}).All()

// SELECT * FROM user WHERE gender=1 AND uid IN(100,10000,90000)
db.Model("user").Where("gender=? AND uid IN(?)", 1, g.Slice{100, 10000, 90000}).All()

// SELECT COUNT(*) FROM user WHERE age in(18,50)
db.Model("user").Where("age IN(?,?)", 18, 50).Count()
db.Model("user").Where("age", g.Slice{18, 50}).Count()
```

map

```
// SELECT * FROM user WHERE gender=1 AND uid IN(100,10000,90000)
db.Model("user").Where(g.Map{
    "gender" : 1,
    "uid"    : g.Slice{100,10000,90000},
}).All()
```

structstruct

```
type User struct {
    Id      []int `orm:"uid"`
    Gender  int    `orm:"gender"`
}
// SELECT * FROM `user` WHERE uid IN(100,10000,90000) AND gender=1
db.Model("user").Where(User{
    "gender" : 1,
    "uid"    : []int{100, 10000, 90000},
}).All()
```

slicenilfalse

```
// SELECT * FROM `user` WHERE 0=1
db.Model("user").Where("uid", g.Slice{}).All()
db.Model("user").Where("uid", nil).All()
```

goframe v1.16goframeORMWhereIn/WhereNotIn/WhereOrIn/WhereOrNotInIn

```
func (m *Model) WhereIn(column string, in interface{}) *Model
func (m *Model) WhereNotIn(column string, in interface{}) *Model
func (m *Model) WhereOrIn(column string, in interface{}) *Model
func (m *Model) WhereOrNotIn(column string, in interface{}) *Model
```

Content Menu

- [in](#)
- [like](#)
- [min/max/avg/sum](#)
- [count](#)
- [distinct](#)
- [between](#)
- [null](#)

```
// SELECT * FROM `user` WHERE `gender`=1 AND `type` IN(1,2,3)
db.Model("user").Where("gender", 1).WhereIn("type", g.Slice{1,2,3}).All()

// SELECT * FROM `user` WHERE `gender`=1 AND `type` NOT IN(1,2,3)
db.Model("user").Where("gender", 1).WhereNotIn("type", g.Slice{1,2,3}).All()

// SELECT * FROM `user` WHERE `gender`=1 OR `type` IN(1,2,3)
db.Model("user").Where("gender", 1).WhereOrIn("type", g.Slice{1,2,3}).All()

// SELECT * FROM `user` WHERE `gender`=1 OR `type` NOT IN(1,2,3)
db.Model("user").Where("gender", 1).WhereOrNotIn("type", g.Slice{1,2,3}).All()
```

like

```
// SELECT * FROM `user` WHERE name like '%john%'
db.Model("user").Where("name like ?", "%john%").All()

// SELECT * FROM `user` WHERE birthday like '1990-%'
db.Model("user").Where("birthday like ?", "1990-%").All()

goframe v1.16goframeORMWhereLike/WhereNotLike/WhereOrLike/WhereOrNotLikeLike
```

```
func (m *Model) WhereLike(column string, like interface{}) *Model
func (m *Model) WhereNotLike(column string, like interface{}) *Model
func (m *Model) WhereOrLike(column string, like interface{}) *Model
func (m *Model) WhereOrNotLike(column string, like interface{}) *Model
```

```
// SELECT * FROM `user` WHERE `gender`=1 AND `name` LIKE 'john%'
db.Model("user").Where("gender", 1).WhereLike("name", "john%").All()

// SELECT * FROM `user` WHERE `gender`=1 AND `name` NOT LIKE 'john%'
db.Model("user").Where("gender", 1).WhereNotLike("name", "john%").All()

// SELECT * FROM `user` WHERE `gender`=1 OR `name` LIKE 'john%'
db.Model("user").Where("gender", 1).WhereOrLike("name", "john%").All()

// SELECT * FROM `user` WHERE `gender`=1 OR `name` NOT LIKE 'john%'
db.Model("user").Where("gender", 1).WhereOrNotLike("name", "john%").All()
```

min/max/avg/sum

Fields

```
// SELECT MIN(score) FROM `user` WHERE `uid`=1
db.Model("user").Fields("MIN(score)").Where("uid", 1).Value()

// SELECT MAX(score) FROM `user` WHERE `uid`=1
db.Model("user").Fields("MAX(score)").Where("uid", 1).Value()

// SELECT AVG(score) FROM `user` WHERE `uid`=1
db.Model("user").Fields("AVG(score)").Where("uid", 1).Value()

// SELECT SUM(score) FROM `user` WHERE `uid`=1
db.Model("user").Fields("SUM(score)").Where("uid", 1).Value()
```

goframe v1.16goframeORMMin/Max/Avg/Sum

```
func (m *Model) Min(column string) (float64, error)
func (m *Model) Max(column string) (float64, error)
func (m *Model) Avg(column string) (float64, error)
func (m *Model) Sum(column string) (float64, error)
```

```
// SELECT MIN(`score`) FROM `user` WHERE `uid`=1
db.Model("user").Where("uid", 1).Min("score")

// SELECT MAX(`score`) FROM `user` WHERE `uid`=1
db.Model("user").Where("uid", 1).Max("score")

// SELECT AVG(`score`) FROM `user` WHERE `uid`=1
db.Model("user").Where("uid", 1).Avg("score")

// SELECT SUM(`score`) FROM `user` WHERE `uid`=1
db.Model("user").Where("uid", 1).Sum("score")
```

count

```
// SELECT COUNT(1) FROM `user` WHERE `birthday`='1990-10-01'
db.Model("user").Where("birthday", "1990-10-01").Count()

// SELECT COUNT(uid) FROM `user` WHERE `birthday`='1990-10-01'
db.Model("user").Fields("uid").Where("birthday", "1990-10-01").Count()
```

goframe v1.16goframeORMCountCountColumn

```
func (m *Model) CountColumn(column string) (int, error)
```

```
db.Model("user").Where("birthday", "1990-10-01").CountColumn("uid")
```

distinct

```
// SELECT DISTINCT uid,name FROM `user`
db.Model("user").Fields("DISTINCT uid,name").All()

// SELECT COUNT(DISTINCT uid,name) FROM `user`
db.Model("user").Fields("DISTINCT uid,name").Count()
```

goframe v1.16goframeORMDistinct

```
func (m *Model) Distinct() *Model
```

```
// SELECT COUNT(DISTINCT `name`) FROM `user`
db.Model("user").Distinct().CountColumn("name")

// SELECT COUNT(DISTINCT uid,name) FROM `user`
db.Model("user").Distinct().CountColumn("uid,name")
```

between

```
// SELECT * FROM `user` WHERE age between 18 and 20
db.Model("user").Where("age between ? and ?", 18, 20).All()
```

```
goframe v1.16goframeORMWhereBetween/WhereNotBetween/WhereOrBetween
/WhereOrNotBetweenBetween
```

```
func (m *Model) WhereBetween(column string, min, max interface{}) *Model
func (m *Model) WhereNotBetween(column string, min, max interface{}) *Model
func (m *Model) WhereOrBetween(column string, min, max interface{}) *Model
func (m *Model) WhereOrNotBetween(column string, min, max interface{})
*Model
```

```
// SELECT * FROM `user` WHERE `gender`=0 AND `age` BETWEEN 16 AND 20
db.Model("user").Where("gender", 0).WhereBetween("age", 16, 20).All()

// SELECT * FROM `user` WHERE `gender`=0 AND `age` NOT BETWEEN 16 AND 20
db.Model("user").Where("gender", 0).WhereNotBetween("age", 16, 20).All()

// SELECT * FROM `user` WHERE `gender`=0 OR `age` BETWEEN 16 AND 20
db.Model("user").Where("gender", 0).WhereOrBetween("age", 16, 20).All()

// SELECT * FROM `user` WHERE `gender`=0 OR `age` NOT BETWEEN 16 AND 20
db.Model("user").Where("gender", 0).WhereOrNotBetween("age", 16, 20).All()
```

null

```
goframe v1.16goframeORMWhereNull/WhereNotNull/WhereOrNull/WhereOrNotNullNull
```

```
func (m *Model) WhereNull(columns ...string) *Model
func (m *Model) WhereNotNull(columns ...string) *Model
func (m *Model) WhereOrNull(columns ...string) *Model
func (m *Model) WhereOrNotNull(columns ...string) *Model
```

```
// SELECT * FROM `user` WHERE `created_at` > '2021-05-01 00:00:00' AND
`inviter` IS NULL
db.Model("user").Where("created_at>", gtime.New("2021-05-01")).WhereNull
("inviter").All()

// SELECT * FROM `user` WHERE `created_at` > '2021-05-01 00:00:00' AND
`inviter` IS NOT NULL
db.Model("user").Where("created_at>", gtime.New("2021-05-01")).
WhereNotNull("inviter").All()

// SELECT * FROM `user` WHERE `created_at` > '2021-05-01 00:00:00' OR
`inviter` IS NULL
db.Model("user").Where("created_at>", gtime.New("2021-05-01")).WhereOrNull
("inviter").All()

// SELECT * FROM `user` WHERE `created_at` > '2021-05-01 00:00:00' OR
`inviter` IS NOT NULL
db.Model("user").Where("created_at>", gtime.New("2021-05-01")).
WhereOrNotNull("inviter").All()
```

```
// SELECT * FROM `user` WHERE `created_at` > '2021-05-01 00:00:00' AND
`inviter` IS NULL AND `creator` IS NULL
db.Model("user").Where("created_at>", gtime.New("2021-05-01")).WhereNull
("inviter", "creator").All()
```

