

# Grafana+Mysql+Go实现监控

---

## 准备阶段

安装Grafana

安装mariadb

Go模拟写入数据

## 配置阶段（grafana）

添加mysql数据源

添加Dashboard

选择保存的Dashboard

查看自定义图表

## 写在最后

## 准备阶段

### 安装Grafana

```
1 # 添加grafana源
2 vim /etc/yum.repos.d/grafana.repo
3 [grafana]
4 name=grafana
5 baseurl=https://packages.grafana.com/enterprise/rpm
6 repo_gpgcheck=1
7 enabled=1
8 gpgcheck=1
9 gpgkey=https://packages.grafana.com/gpg.key
10 sslverify=1
11 sslcacert=/etc/pki/tls/certs/ca-bundle.crt
12
13 # 安装grafana
14 yum install grafana-enterprise.x86_64
15
16 # 启动grafana
17 systemctl daemon-reload
18 systemctl start grafana-server
19 systemctl enable grafana-server
20 systemctl status grafana-server
21
22 # 登录
23 admin/admin
```

## 安装mariadb

略。。。

## Go模拟写入数据

- code

```
1 package main
2
3 import (
4     "database/sql"
5     "fmt"
6     "time"
7
8     _ "github.com/go-sql-driver/mysql"
9 )
10
11 var db *sql.DB
12
13 type user struct {
14     id    int
15     age   int
16     name  string
17 }
18
19 //连接数据库、验证数据库
20 func initDB() (err error) {
21     dsn := "root:root@tcp(127.0.0.1:3306)/sql_test?
22     charset=utf8mb4&parseTime=True"
23     db, err = sql.Open("mysql", dsn)
24     if err != nil {
25         fmt.Printf("open failed err:%v\n", err)
26         return err
27     }
28     err = db.Ping()
29     if err != nil {
30         fmt.Printf("ping failed err:%v\n", err)
31         return err
32     }
33     return nil
34 }
35
36 //插入数据
37 func insertRowDemo(i int) {
38     sqlStr := "insert into grafana(time,value) values(?,?)"
39     ret, err := db.Exec(sqlStr, time.Now().Format("2006-01-02 03:04:05.000"),
40 i)
41     if err != nil {
42         fmt.Printf("insert failed, err:%v\n", err)
43         return
44     }
45     theID, err := ret.LastInsertId()
46     if err != nil {
```

```

47         fmt.Printf("get lastinsert ID failed,err:%v\n", err)
48         return
49     }
50     fmt.Printf("insert success,the id is %d.\n", theID)
51     // wg.Done()
52 }
53
54 // 更新数据
55 func updateRowDemo() {
56     sqlStr := "update grafana set time= ? where id = ?"
57     t1 := time.Now()
58     for i := 1; i < 745; i++ {
59         ret, err := db.Exec(sqlStr, t1.Format("2006-01-02-15:04"), i)
60         if err != nil {
61             fmt.Printf("update failed,err:%v\n", err)
62             return
63         }
64         n, err := ret.RowsAffected()
65         if err != nil {
66             fmt.Printf("get RowsAffected failed,err:%v\n", err)
67             return
68         }
69         fmt.Printf("update success,affected rows:%d\n", n)
70         t1 = t1.Add(time.Minute)
71     }
72 }
73
74 func main() {
75     //连接数据库
76     err := initDB()
77     if err != nil {
78         fmt.Printf("init db failed,err:%v\n", err)
79         return
80     }
81     //模拟插入数据
82     for i := 1; i < 745; i++ {
83         // wg.Add(1)
84         insertRowDemo(i)
85         time.Sleep(10 * time.Millisecond)
86     }
87     //更新插入数据希望格式
88     updateRowDemo()
89 }

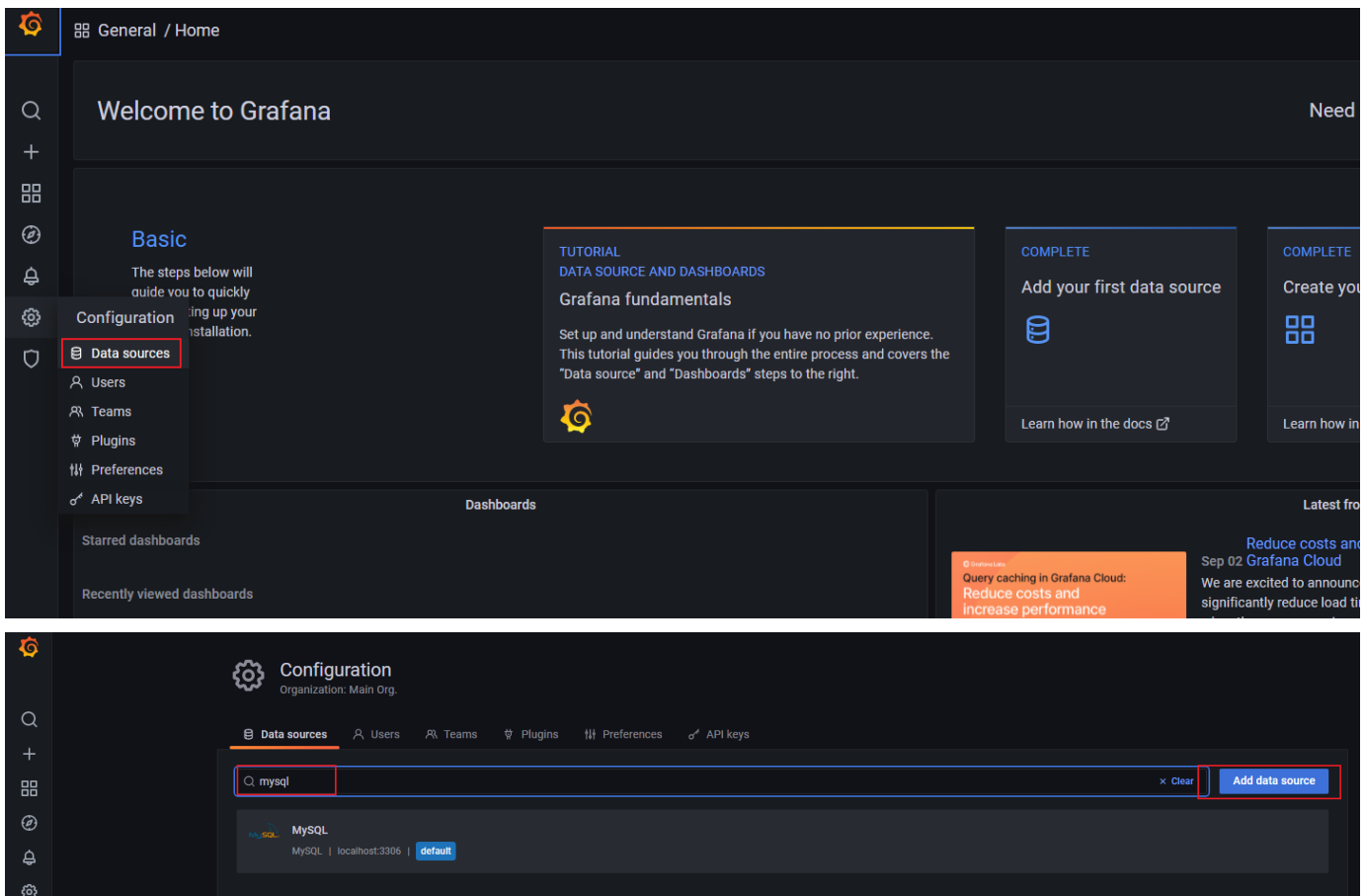
```

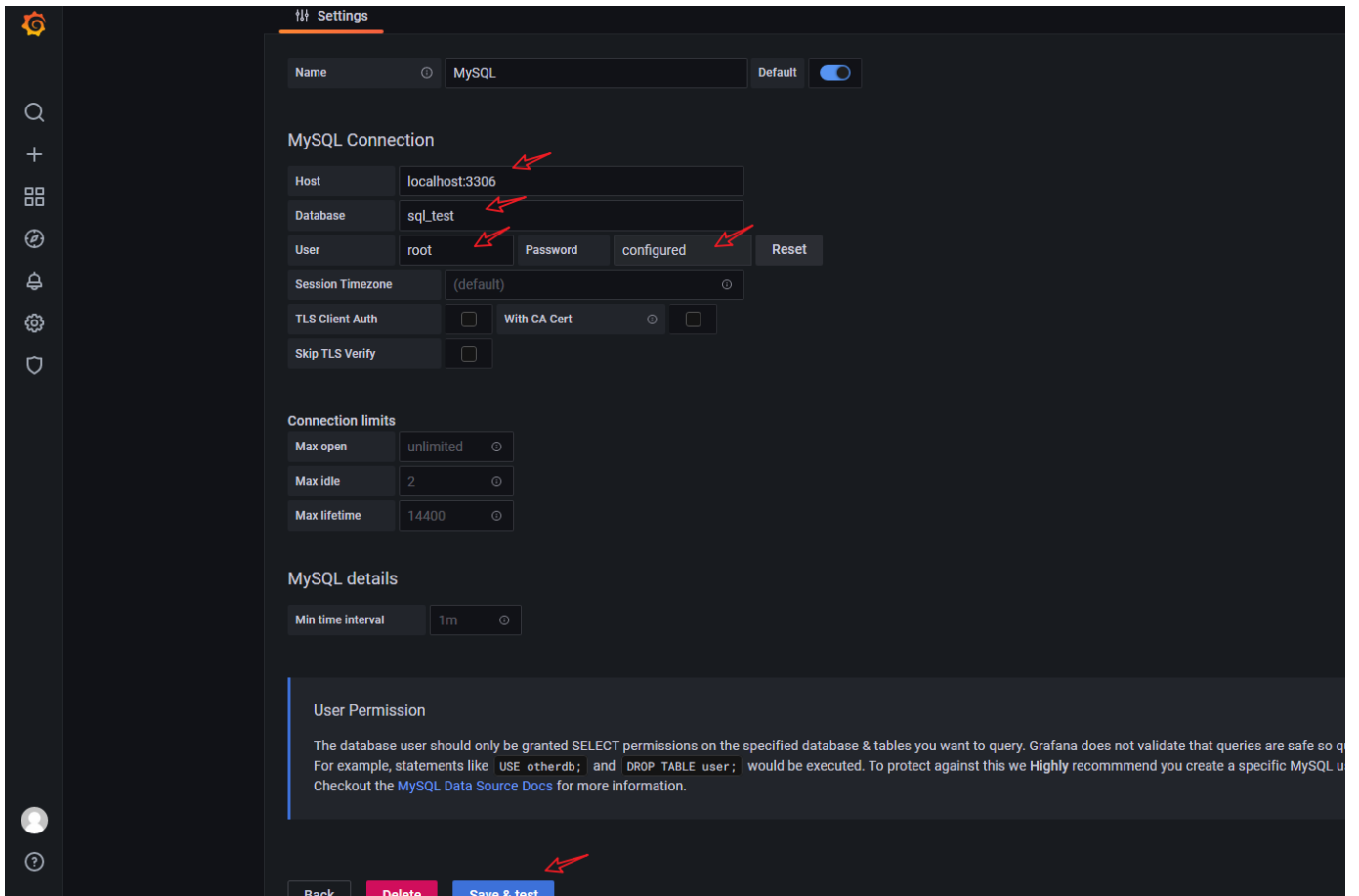
- mariadb

```
1 MariaDB [sql_test]> select * from grafana;
2
3 | id | time | value |
4 |----+-----+-----|
5 | 1 | 2021-09-05-15:04 | 256 |
6 | 2 | 2021-09-05-15:05 | 257 |
7 | 3 | 2021-09-05-15:06 | 258 |
8 | 4 | 2021-09-05-15:07 | 259 |
9 | 5 | 2021-09-05-15:08 | 260 |
10 | 6 | 2021-09-05-15:09 | 261 |
11 | 7 | 2021-09-05-15:10 | 262 |
12 | 8 | 2021-09-05-15:11 | 263 |
13 | 9 | 2021-09-05-15:12 | 264 |
14 | 10 | 2021-09-05-15:13 | 265 |
15 | 11 | 2021-09-05-15:14 | 266 |
16 | 12 | 2021-09-05-15:15 | 267 |
17 | 13 | 2021-09-05-15:16 | 268 |
18 ...
```

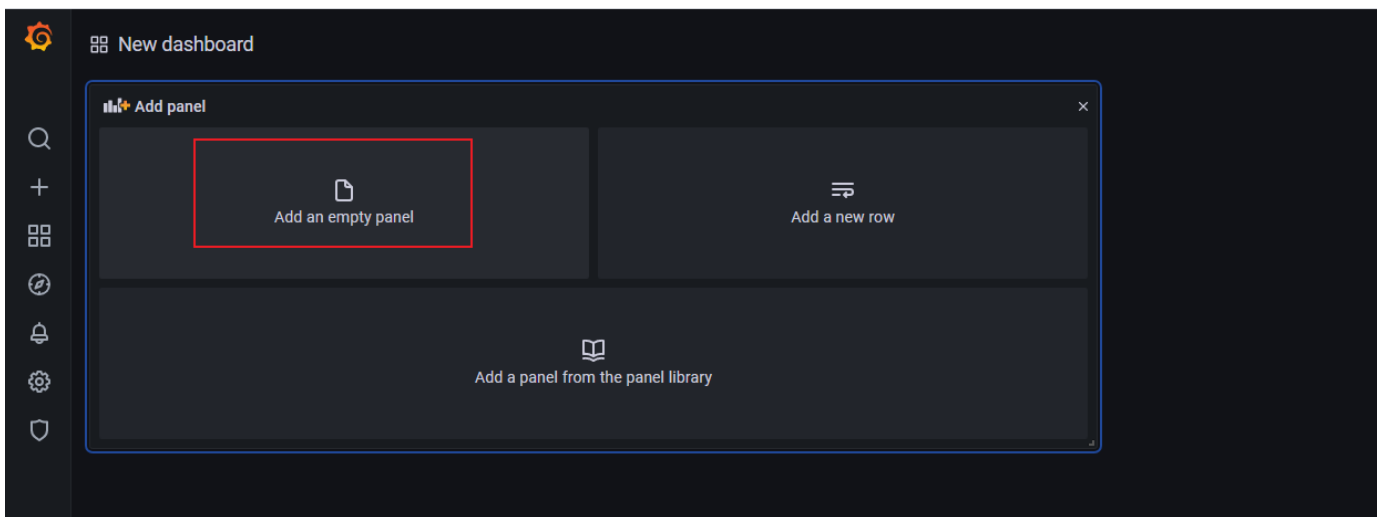
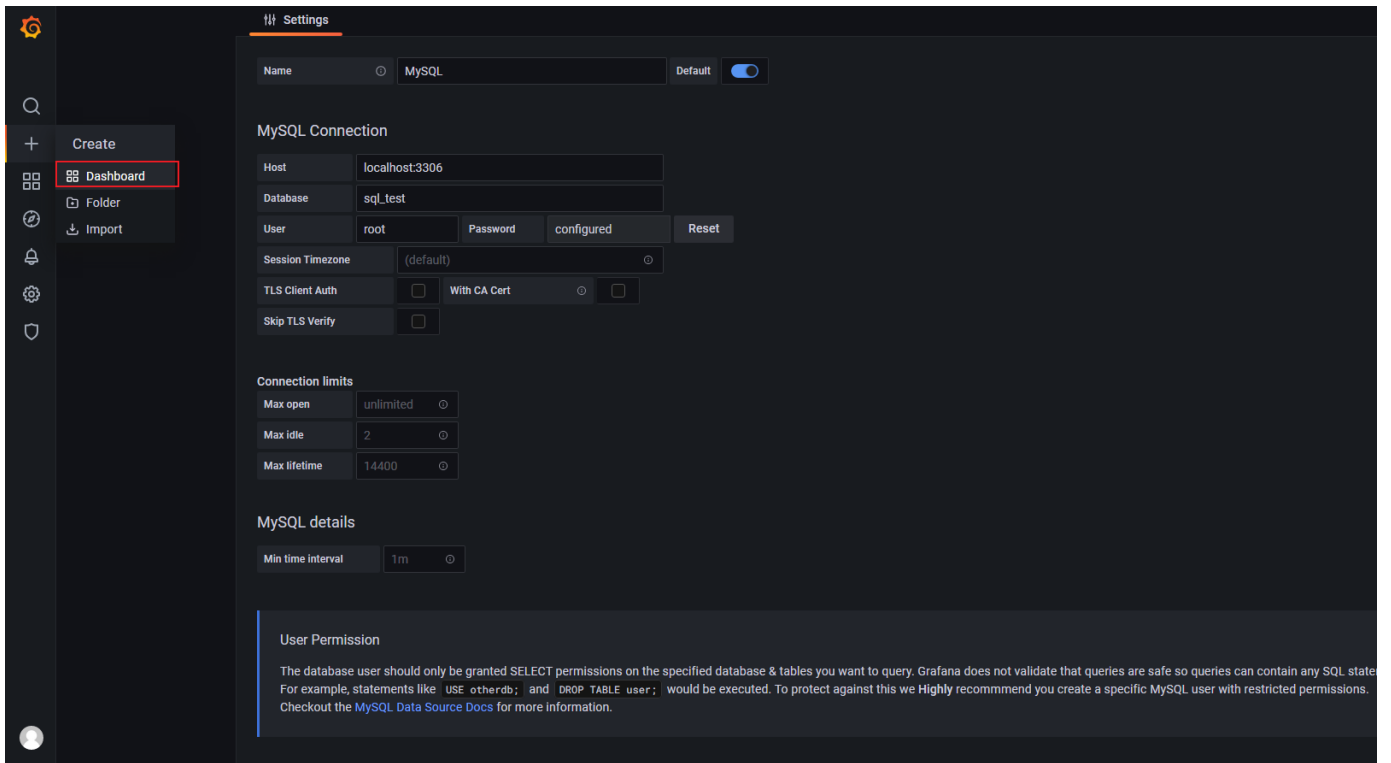
## 配置阶段（grafana）

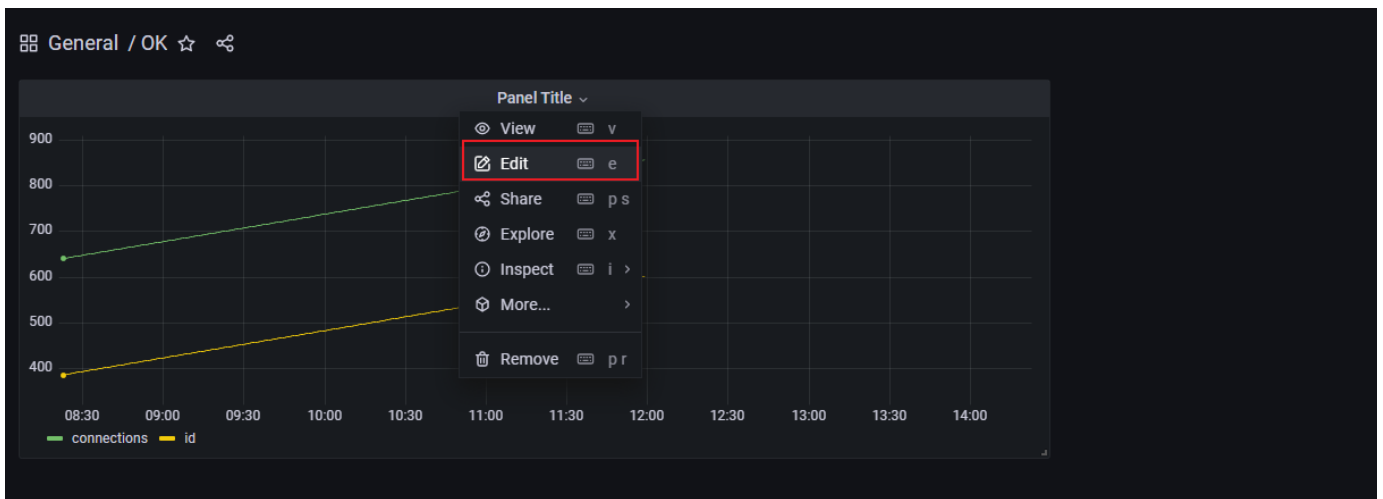
### 添加mysql数据源





添加Dashboard





Query 1 Transform 0 Alert 0

Data source MySQL Query options MD = auto = 910 Interval = 1m Query inspector

A (MySQL)

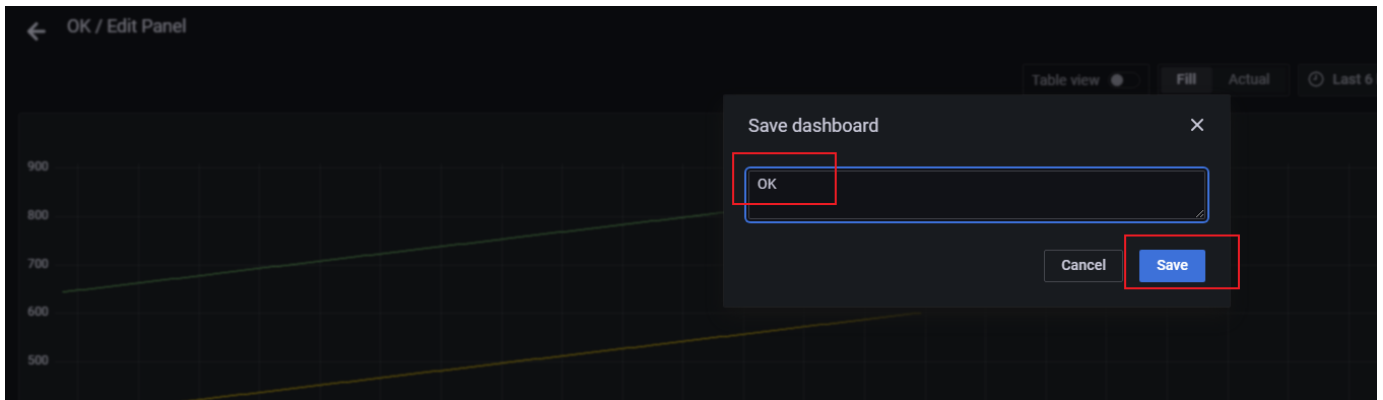
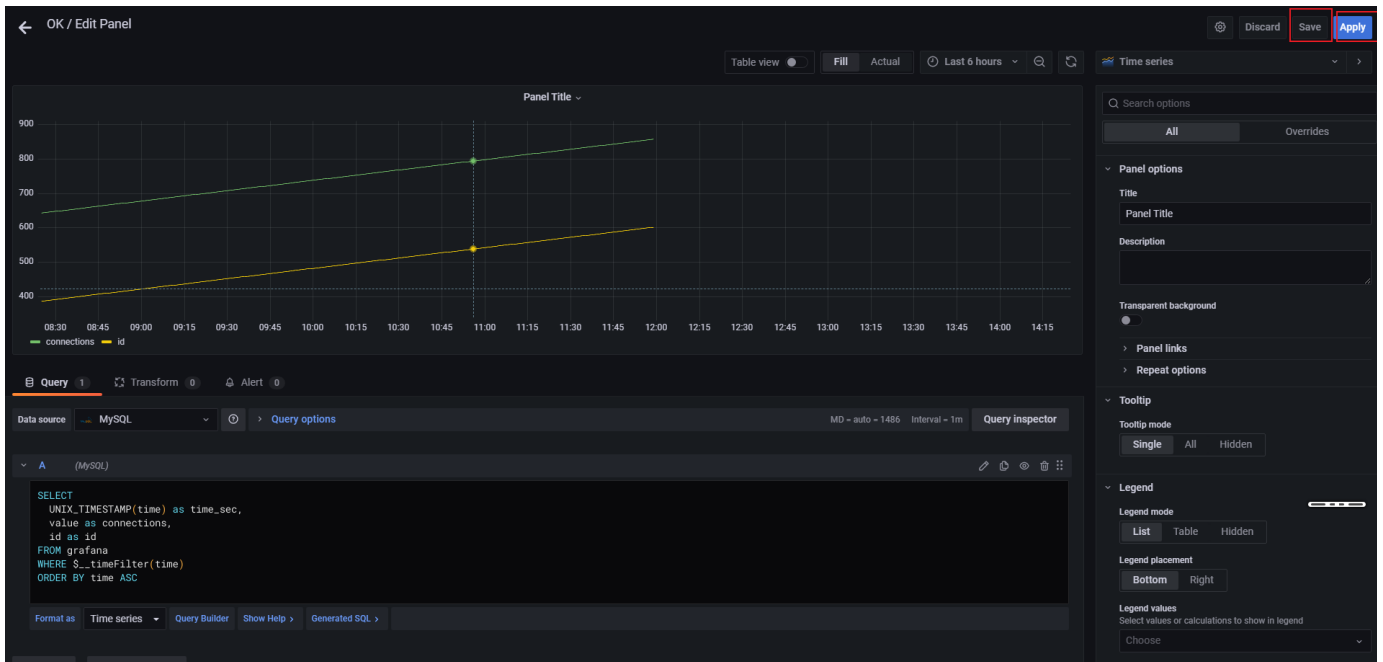
```
SELECT
  UNIX_TIMESTAMP(time) as time_sec,
  value as connections,
  id as id
FROM grafana
WHERE $__timeFilter(time)
ORDER BY time ASC
```

Format as Time series Query Builder Show Help >

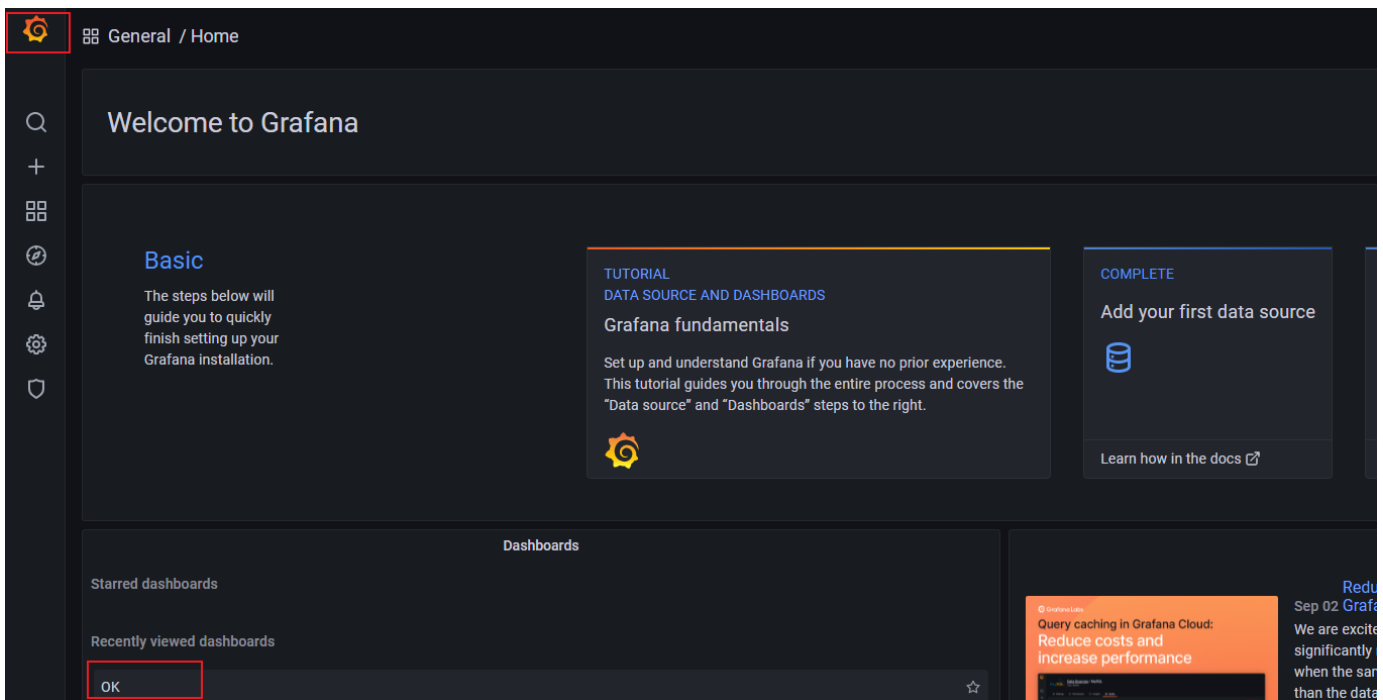
SQL 复制代码

```
1 # 此处根据对应数据库的tables的实际情况，即想获取的字段进行修改
2 SELECT
3     UNIX_TIMESTAMP(time) as time_sec,
4     value as connections,
5     id as id
6 FROM grafana
7 WHERE $__timeFilter(time)
8 ORDER BY time ASC
```

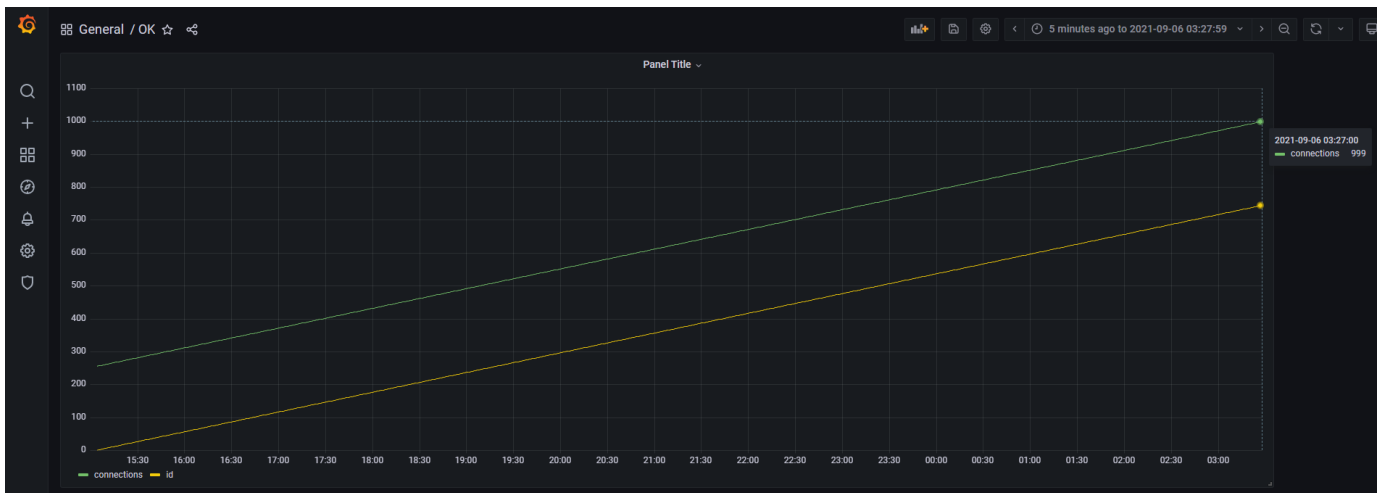




## 选择保存的Dashboard



## 查看自定义图表



## 写在最后

- 这里Go仅仅是模拟了些数据，并写入mysql
- 实际工作中可以使用Go抓取服务器或者网络设备中想要的参数，结合抓取时间一并写入mysql

- 再通过grafana从mysql读取到想要的数据库，并使用图表呈现出来，最终实现监控的目的