

egg.js 中使用 egg-mysql 操作 mysql 数据库 【专题】

主讲教师：（大地）

合作网站：www.itying.com （IT 营）

我的专栏：<https://www.itying.com/category-79-b0.html>

一、egg-mysql 的安装配置.....	1
二、 egg-mysql 的使用.....	2
1、 查找数据的第一种方式.....	2
2、 查找数据的另一种方式.....	2
3、 增加数据.....	3
4、 修改数据的第一种方式：根据主键修改.....	3
5、 修改数据的第二种方式：通过 sql 来修改数据.....	3
6、 删除数据.....	3
7、 执行 sql.....	3
8、 mysql 事务.....	3

一、egg-mysql 的安装配置

1、在 egg 项目中安装 egg-mysql

```
npm i egg-mysql --save
```

2、在 {app_root}/config/plugin.js 中启用 egg-mysql 插件:

```
exports.mysql = {  
  enable: true,
```

```
package: 'egg-mysql',  
};
```

3、在配置文件中配置 mysql 数据库连接地址 {app_root}/config/config.default.js

```
exports.mysql = {  
  // database configuration  
  client: {  
    // host  
    host: 'mysql.com',  
    // port  
    port: '3306',  
    // username  
    user: 'test_user',  
    // password  
    password: 'test_password',  
    // database  
    database: 'test',  
  },  
  // load into app, default is open  
  app: true,  
  // load into agent, default is close  
  agent: false,  
};
```

二、egg-mysql 的使用

1、查找数据的第一种方式

```
const result = await this.app.mysql.get('user', { id: 2 });
```

2、查找数据的另一种方式

```
const result = await this.app.mysql.select('user',{  
  where: { id: '3' },  
  // orders: [['created_at','desc'], ['id','desc']],  
  limit: 10,  
  offset: 0  
});
```

3、增加数据

```
const result = await this.app.mysql.insert('user', { username: '赵四', 'password': '223423' });
```

4、修改数据的第一种方式：根据主键修改

```
const row = {  
  id: 7,  
  username: '赵四'  
};  
const result = await this.app.mysql.update('user', row);
```

5、修改数据的第二种方式：通过 sql 来修改数据

```
const results = await this.app.mysql.query('update user set username = ? where id = ?', [6666, 8]);
```

6、删除数据

```
const result = await this.app.mysql.delete('user', {  
  username: '赵四'  
});
```

7、执行 sql

```
app.mysql.query(sql, values);
```

8、mysql 事务

```
const conn = await this.app.mysql.beginTransaction();  
try {  
  await conn.insert('user', { 'username': 'hahahh', 'password': '223423' });  
  const row = { id: 8, username: '王麻子' };  
  await conn.update('user', row);  
  await conn.commit();  
} catch (err) {  
  await conn.rollback(); // rollback call won't throw err
```



```
throw err;  
}
```