

## 第二章 了解GitHub(4课时)

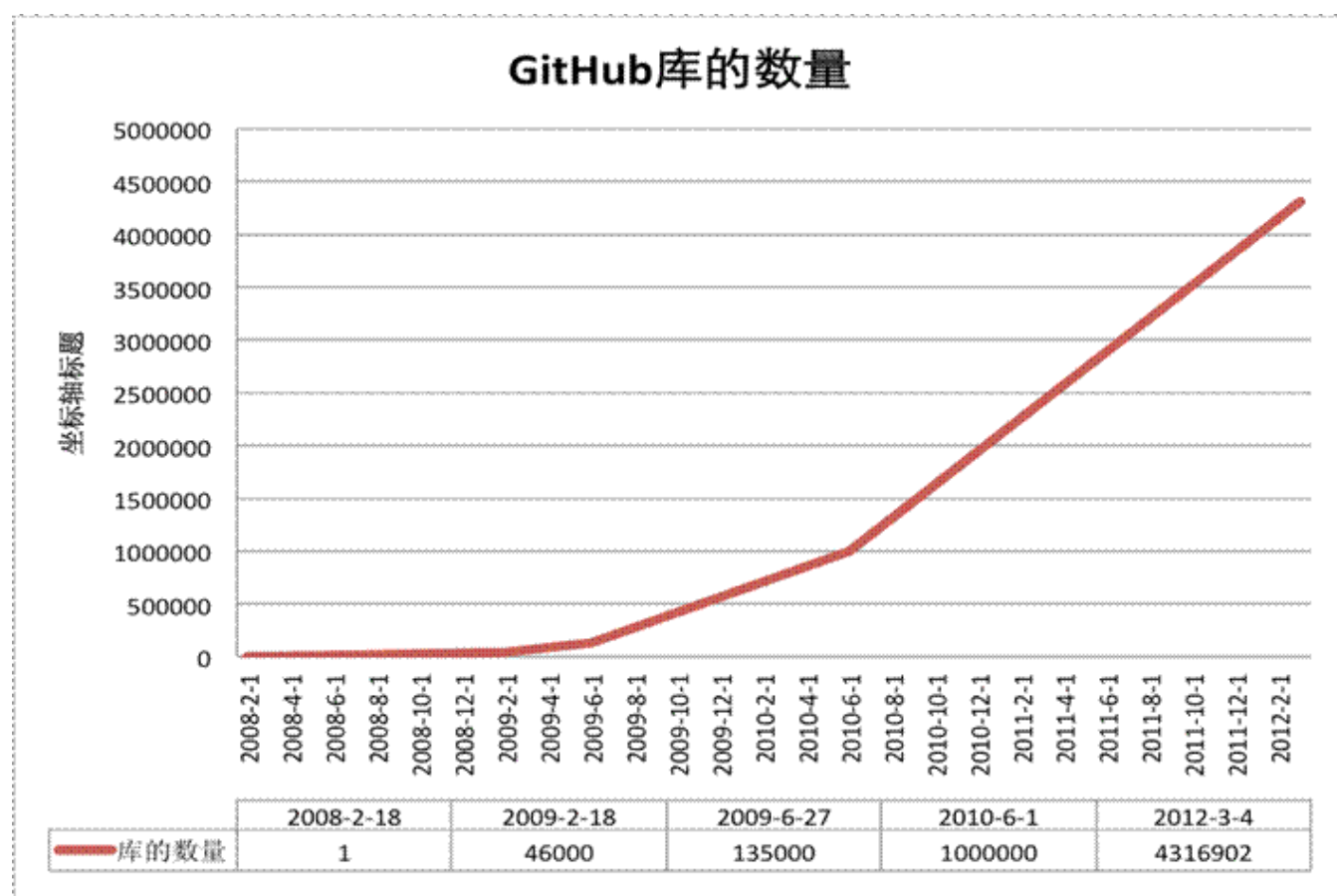
### 1. 了解：GitHub

GitHub是<Git远程库>的一种<产品实现>，它可以托管各种<Git库>，<付费用户>可以构建<私有仓库>，一般的<免费用户>只能使用<公共仓库>，也就是<代码>要公开。

当今的GitHub：

已经是：一个拥有143万开发者的社区。其中不乏Linux发明者[Torvalds](#)这样的顶级黑客，以及Rails创始人[DHH](#)这样的年轻极客。

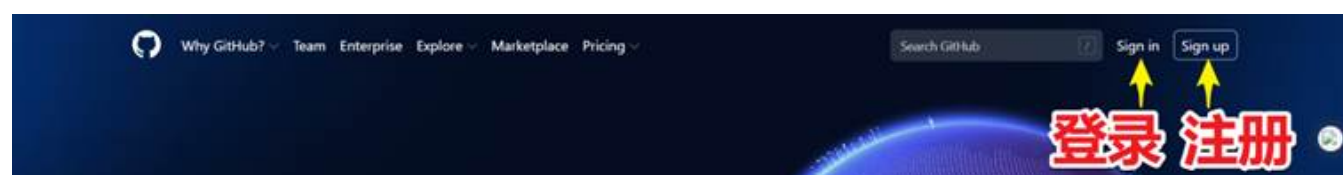
已经是：这个星球上最流行的开源托管服务。目前已托管431万git项目，不仅越来越多知名开源项目迁入GitHub，比如Ruby on Rails、jQuery、Ruby、Erlang/OTP；近三年流行的开源库往往在GitHub首发，例如：[Bootstrap](#)、[Node.js](#)、[CoffeeScript](#)等。



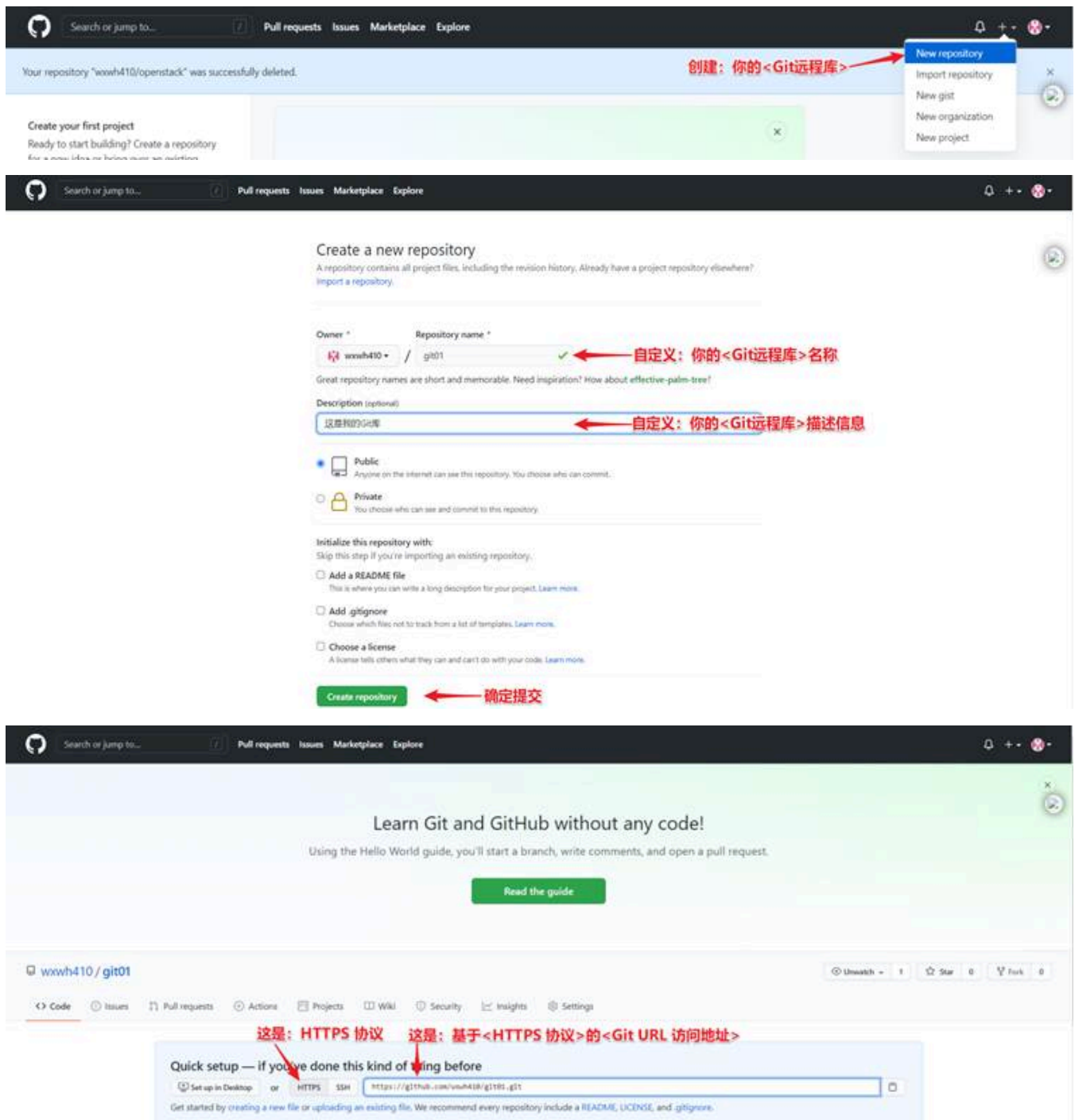
### 2. 注册：GitHub

• GitHub网址：<https://github.com>

• 注册：



• 登录并创建你的<第一个Git远程库>：



## 3. 使用: GitHub

### 3.1. 你的第一次git push推送

```
cd ~
```

```
rm -rf /git01
```

```
mkdir /git01
```

```
cd /git01
```

```
echo "# 这是我的git01版本库" >> README.md
```

```
git init
```

```
git add README.md
```

```
git commit -m "这是<git01版本库>的第一次commit提交"
```

```
git remote add my_public_git01 https://github.com/wxwh410/git01.git
```

```
git push -u my_public_git01 master
```

```
Username for 'https://github.com': wxwh410
Password for 'https://wxwh410@github.com': | ← 输入: 用户名 和 密码
Counting objects: 3, done.
Writing objects: 100% (3/3), 282 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/wxwh410/git01.git
 * [new branch]      master -> master
分支 master 设置为跟踪来自 my_public_git01 的远程分支 master。
```

### 3.2. 别人通过git clone克隆获取: GitHub上<你的Git远程库>

```
cd ~
```

```
rm -rf /git01
```

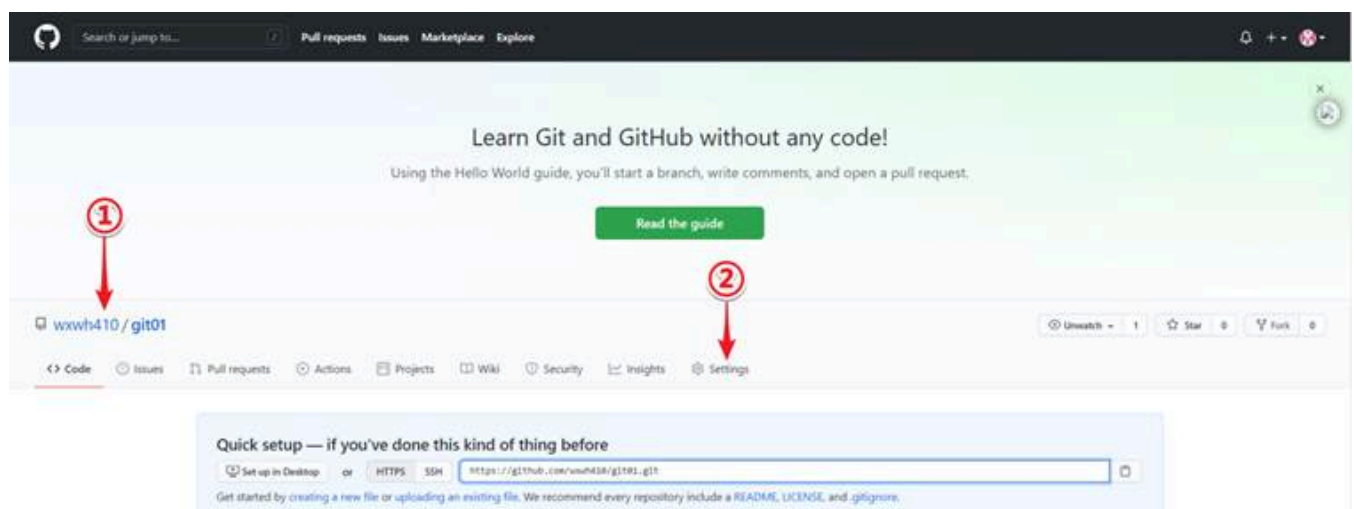
```
git clone -o my_public_git01 https://github.com/wxwh410/git01.git /git01
```

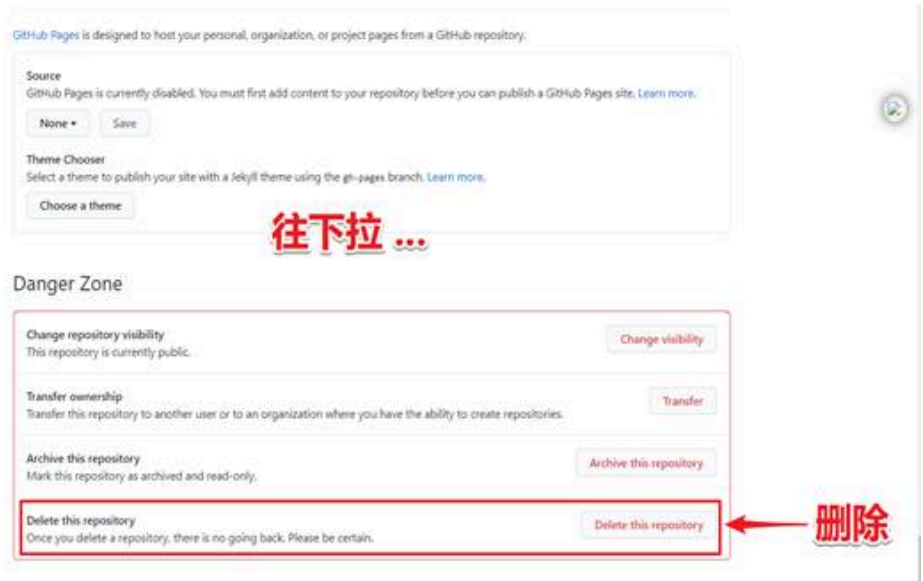
```
cd /git01
```

```
ls -l
```

```
总用量 4
-rw-r--r--. 1 root root 29 2月 14 11:38 README.md 克隆 OK
```

### 3.3. 删除: GitHub上<你的Git远程库>





## 第三章 了解GitLab(4课时)

### 1. 了解: GitLab社区版

- GitLab是<Git远程库>的一种<产品实现>。
- 在GitHub上, 如果不是<公共仓库>, 则须付费构建<私有仓库>, 如果你希望既实现<私有仓库>仅供公司内部使用, 就可以选择使用<GitLab社区版>。

### 2. 构建: GitLab社区版 (IP: 10.0.0.121、内存: 3GB)

cd ~

curl -s <https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.rpm.sh> | sudo bash

yum install -y gitlab-ce.x86\_64

vi /etc/gitlab/gitlab.rb

## 编辑如下



external\_url '<http://10.0.0.121>'

/opt/gitlab/bin/gitlab-ctl reconfigure

```
Running handlers:
Running handlers complete
Chef Infra Client finished, 582/1550 resources updated in 06 minutes 27 seconds

Notes:
It seems you haven't specified an initial root password while configuring the GitLab instance.
On your first visit to your GitLab instance, you will be presented with a screen to set a
password for the default admin account with username `root`.

gitlab Reconfigured!
```

**GitLab 运行成功 OK**

netstat -tunlp

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:9100          0.0.0.0:*               LISTEN      14142/node_exporter
tcp        0      0 127.0.0.1:9229          0.0.0.0:*               LISTEN      14132/gitlab-workho
tcp        0      0 127.0.0.1:9168          0.0.0.0:*               LISTEN      14157/ruby
tcp        0      0 127.0.0.1:8080          0.0.0.0:*               LISTEN      12905/puma 5.1.1 (u
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      13051/nginx: master
tcp        0      0 127.0.0.1:8082          0.0.0.0:*               LISTEN      12949/sidekiq 5.2.9
tcp        0      0 127.0.0.1:9236          0.0.0.0:*               LISTEN      14127/gitaly
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1125/sshd
tcp        0      0 127.0.0.1:3000          0.0.0.0:*               LISTEN      14216/grafana-serve
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      1355/master
tcp        0      0 0.0.0.0:8060            0.0.0.0:*               LISTEN      13051/nginx: master
tcp        0      0 127.0.0.1:9121          0.0.0.0:*               LISTEN      14164/redis_exporte
tcp        0      0 127.0.0.1:9090          0.0.0.0:*               LISTEN      14170/prometheus
tcp        0      0 127.0.0.1:9187          0.0.0.0:*               LISTEN      14210/postgres_expo
tcp        0      0 127.0.0.1:9093          0.0.0.0:*               LISTEN      14200/alertmanager
tcp6       0      0 :::9168                 :::*                   LISTEN      14157/ruby
tcp6       0      0 :::22                   :::*                   LISTEN      1125/sshd
tcp6       0      0 :::1:25                 :::*                   LISTEN      1355/master
tcp6       0      0 :::9094                 :::*                   LISTEN      14200/alertmanager
udp        0      0 127.0.0.1:323          0.0.0.0:*               747/chronyd
udp6       0      0 :::9094                 :::*                   14200/alertmanager
udp6       0      0 :::1:323                 :::*                   747/chronyd
```

firewall-cmd --permanent --zone=public --add-port=80/tcp

firewall-cmd --reload

### 3. 使用：GitLab社区版

#### 3.1. 登录：我的GitLab

<http://10.0.0.121>

Please create a password for your new account.

GitLab

A complete DevOps platform

GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security.

This is a self-managed instance of GitLab.

设置：新密码  
用户名：root

Change your password

New password

Confirm new password

Change your password

Didn't receive a confirmation email? Request a new one

Already have login and password? Sign in

Your password has been changed successfully.

GitLab

A complete DevOps platform

GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security.

This is a self-managed instance of GitLab.

登录

Username or email

root

Password

Remember me

Forgot your password?

Sign in

Don't have an account yet? Register now

#### 3.2. 创建：我的Git远程库

GitLab

Projects

Groups

More

Search or jump to...

Open registration is enabled on your instance.

Learn more about how you can customize / disable registration on your instance.

View setting

Projects

新建项目

New project

Your projects

Starred projects

Explore projects

Filter by name...

Name

All

Personal

M

GitLab Instance / Monitoring

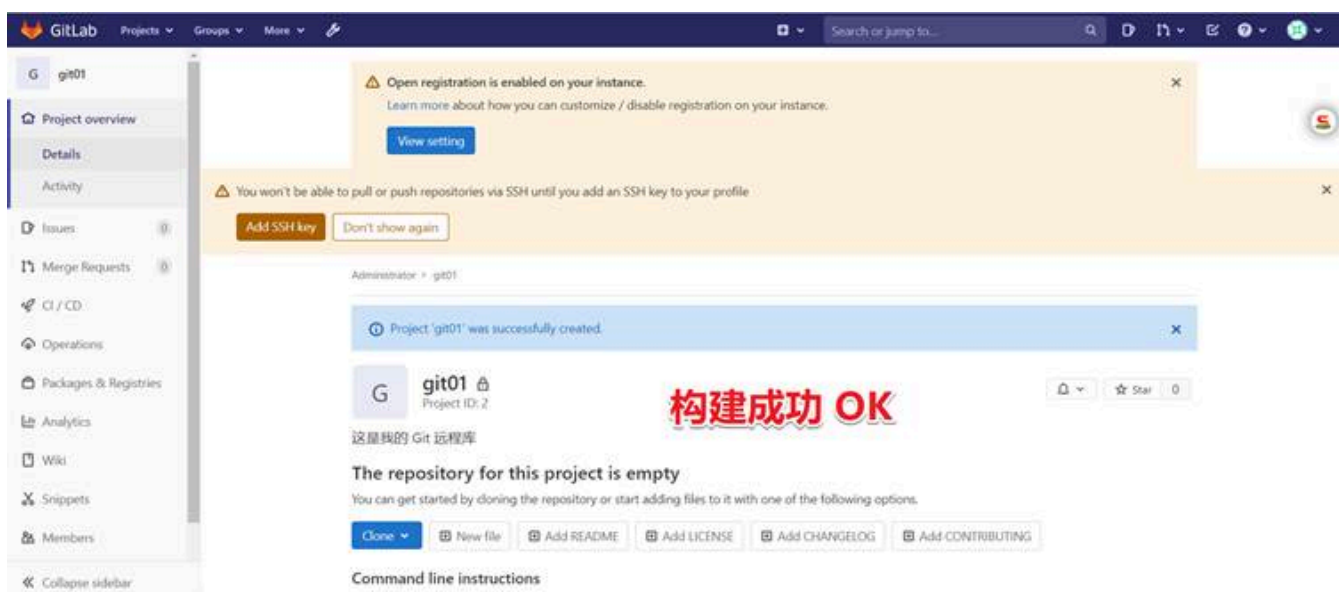
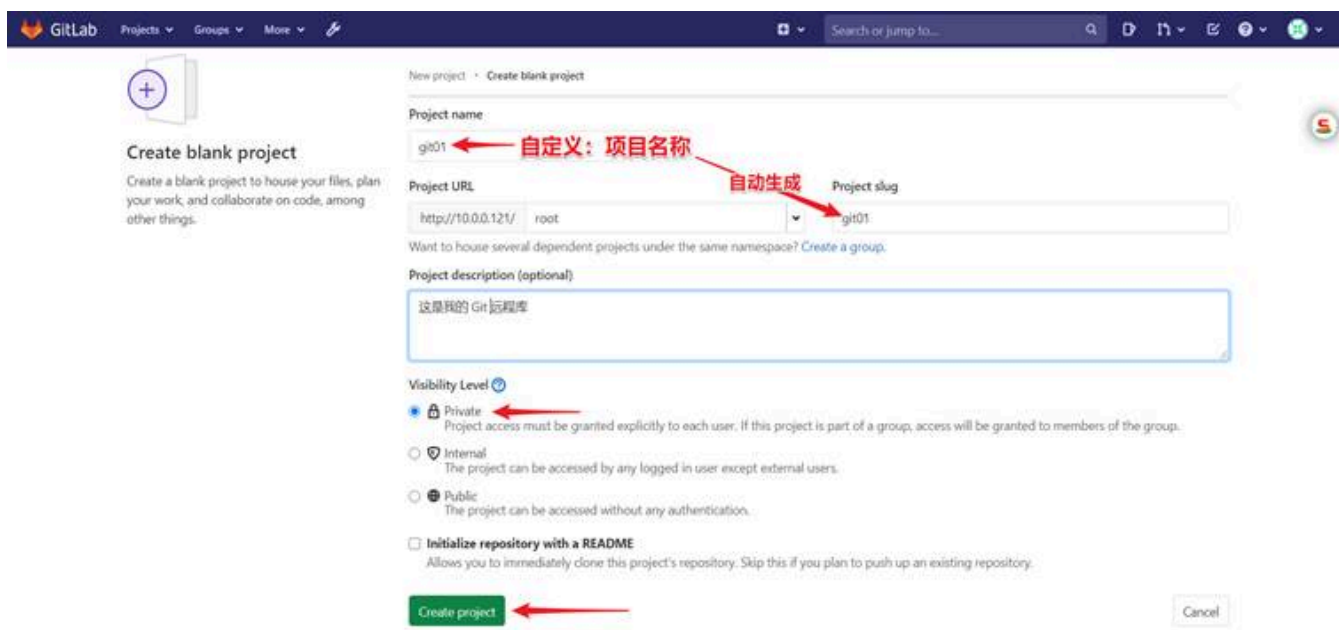
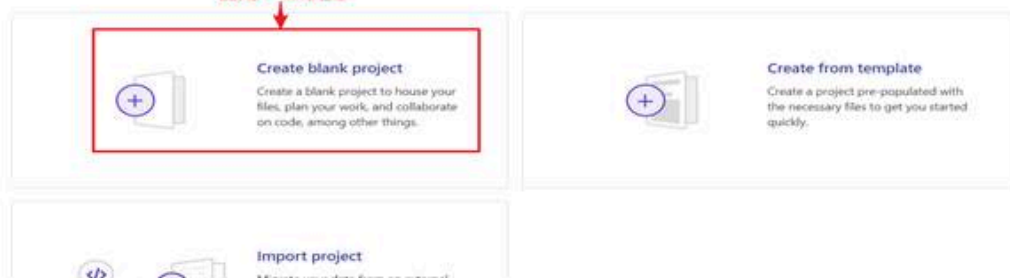
This project is automatically generated and helps monitor this GitLab instance.

Updated 26 minutes ago



## Create new project


创建：空项目



### 3.3. 你的第一次git push推送

```
cd ~  
rm -rf /git01  
mkdir /git01  
cd /git01  
echo "# 这是我的git01版本库" >> README.md  
git init  
git add README.md  
git commit -m "这是<git01版本库>的第一次commit提交"  
git remote add my_public_git01 http://10.0.0.121/root/git01.git  
git push -u my_public_git01 master
```


```
Username for 'http://10.0.0.121': root  
Password for 'http://root@10.0.0.121':  
Counting objects: 3, done.  
Writing objects: 100% (3/3), 282 bytes | 0 bytes/s, done.  
Total 3 (delta 0), reused 0 (delta 0)  
To http://10.0.0.121/root/git01.git  
* [new branch]      master -> master  
分支 master 设置为跟踪来自 my_public_git01 的远程分支 master。
```



### 3.4. 别人通过git clone克隆获取：GitLab上<你的Git远程库>

```
cd ~  
rm -rf /git01  
git clone -o my_public_git01 http://10.0.0.121/root/git01.git /git01
```

```
正克隆到 '/git01'...  
Username for 'http://10.0.0.121': root  
Password for 'http://root@10.0.0.121':  
remote: Enumerating objects: 3, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0  
Unpacking objects: 100% (3/3), done.
```



```
cd /git01
```

```
ls -l
```

```
总用量 4  
-rw-r--r--. 1 root root 29 2月 14 11:38 README.md 克隆 OK
```



## 3.5. GitLab权限管理

### 3.5.1. 了解：GitLab权限管理构成

- 第一类：用户访问等级（Access level）

👉 Regular（一般使用者）

- ◆ 只能<读/写>访问：自己所属的<group组资源>和<project项目库资源>。

👉 Admin（管理者）

- ◆ 可以<读/写>访问：所有的<Group组资源>和<Project项目库资源>，并可操作<管理空间>。

👉 External users（外部使用者）

- ◆ 只能<读/写>访问：<公共库>和<个别被授权的内部库与私有库>。

- ◆ 不能创建<project项目库资源>

- 第二类：组权限（Group）

👉 <User用户>可以创建<Group组>；<Group组>中可以包含<Project项目 和 Member用户成员>。

👉 <Group组>具备三个<配置要素>：

- ◆ <Group组>包含哪些<Member用户成员>？

- ◆ <Group组>包含哪些<Project项目>？

- ◆ <Group组>自身 及其 包含的<Project项目>，他们的<Visibility Level 可见度等级>是什么？

- 第三类：可见度等级（Visibility Level）

👉 Visibility Level可见度等级有三种：

- ◆ public 公共：对<任何人>均可见，无需<用户账户>。

- ◆ internal内部：对<验证用户>或<明确授权用户>可见。

- ◆ private 私有：对<Project项目 或 Group组>的<Member成员>可见。

- 第四类：成员权限（Member Permissions）

👉 <成员权限>用来设定：<User用户>对<Project项目>的具体<权限细则>。

- ◆ 例如：User 是否能建立 branch、能否推送 merge request。

👉 <成员权限>有五种<角色权限集>：

- ◆ Guest 来宾

- ◆ Reporter 记者

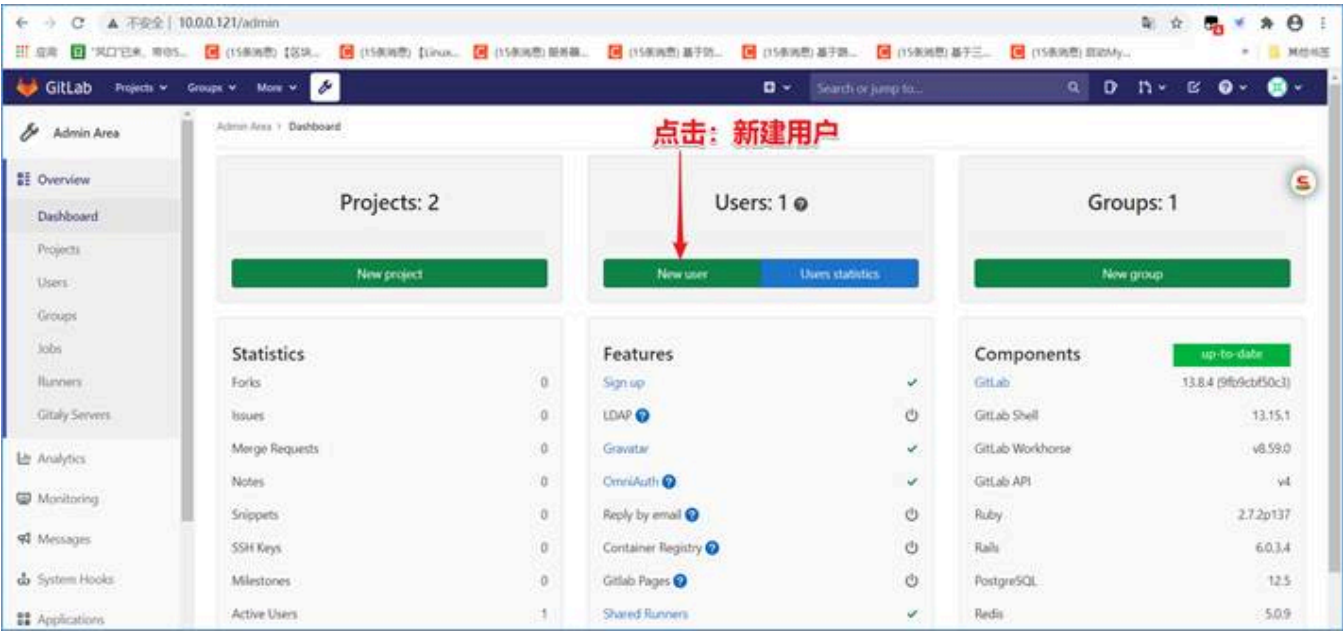
- ◆ Developer 开发者

- ◆ Maintainer 维护管理者

- ◆ Owner 所有者

3.5.2. 创建：<自建用户>，并允许其访问<你的私有仓库>

- 基于工作协作需要，可以创建<用户账户>并授权，让其他同事可以通过<该用户>来访问<我的私有仓库>。
- (1) 创建：<自建用户wx>



Admin Area > New User

New user

Account

Name wx \* required

Username wx \* required

Email 156405304@qq.com \* required

自定义：用户名称

Password 在<用户账户>创建成功之后，再次编辑<用户设置>，就可设置<密码>; <用户>首次登录，须强制更新<密码>

Reset link will be generated and sent to the user. User will be forced to set the password on first sign in.

### Access

Projects limit  **设置：<用户>的<可建项目数量>**

Can create group ☒

Access level ☒ Regular **设置：<用户访问等级>为<Regular 普通>**  
 Regular users have access to their groups and projects.

☐ Admin  
 Administrators have access to all groups, projects and users and can manage all features in this installation

External ☐ External users cannot see internal or private projects unless access is explicitly granted. Also, external users cannot create projects, groups, or personal snippets.  
**设置：是否为<外部用户>，不要勾选，否则就成为了<外部用户>**

### Profile

Avatar  未选择任何文件

Skype

Linkedin

Twitter

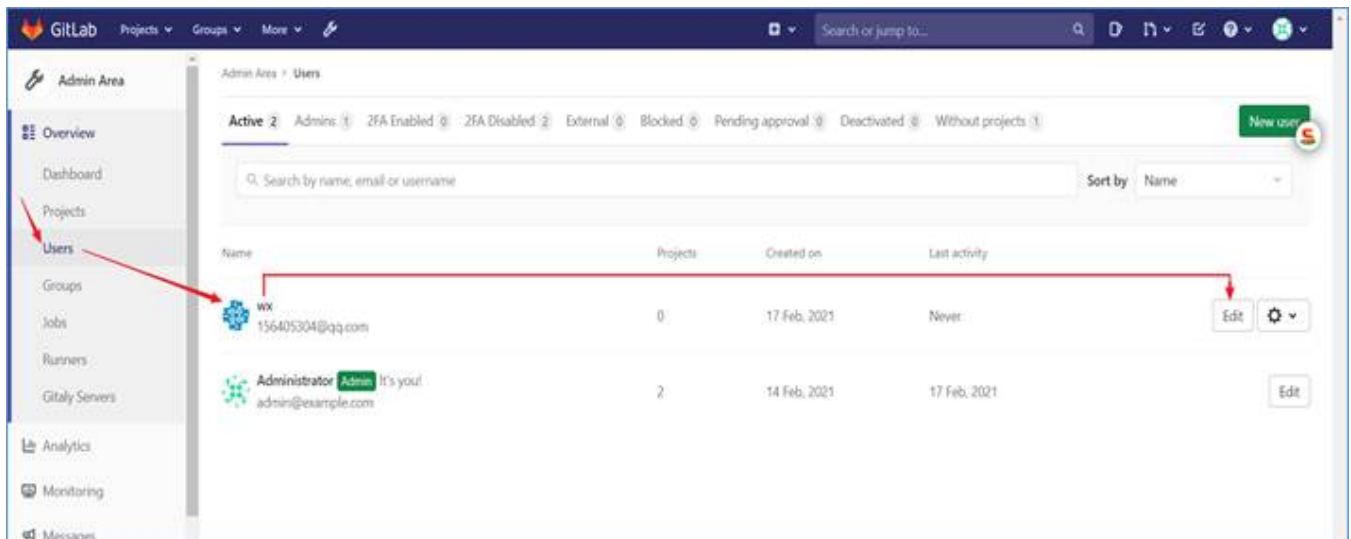
Website url

### Admin notes

Note

**执行：用户创建**

- (2) 设置：<自建用户wx>的<初始密码>



Admin Area > Users

Edit user: wx

### Account

Name: wx  
\* required

Username: wx  
\* required

Email: 156405304@qq.com  
\* required

### Password

Password:

Password confirmation:

设置: <用户>的<初始密码>

- (3) 授权: <自建用户wx>

GitLab Projects

Your projects 2 · Starred projects 0 · Explore projects

Filter by name... Name

All Personal

G	Administrator / git01 这是我的 Git01 远程库	★ 0 Y 0 I 0 D 0	Updated 2 days ago
M	GitLab Instance / Monitoring This project is automatically generated and helps monitor this GitLab instance. Learn more.	★ 0 Y 0 I 0 D 0	Updated 2 days ago

GitLab Projects

git01

You won't be able to pull or push repositories via SSH until you add an SSH key to your profile

Add SSH key Don't show again

Administrator / git01

git01  
Project ID: 3

这是我的 Git01 远程库

The repository for this project is empty

You can get started by cloning the repository or start adding files to it with one of the following options.

Clone New file Add README Add LICENSE Add CHANGELOG Add CONTRIBUTING

Command line instructions

You can also upload existing files from your computer using the instructions below.

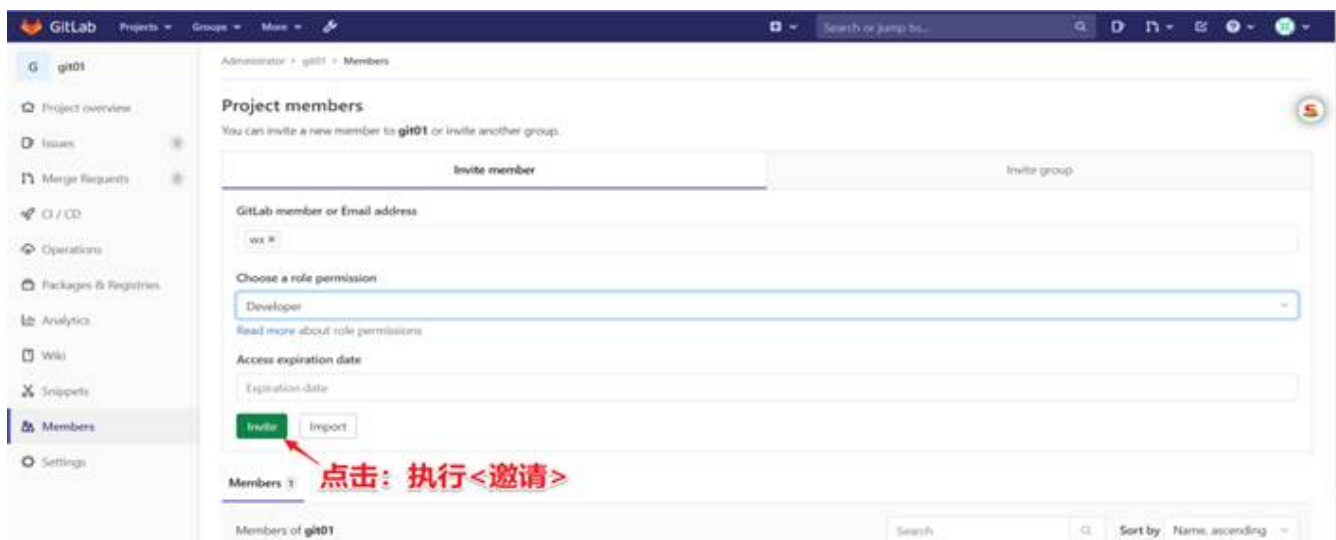
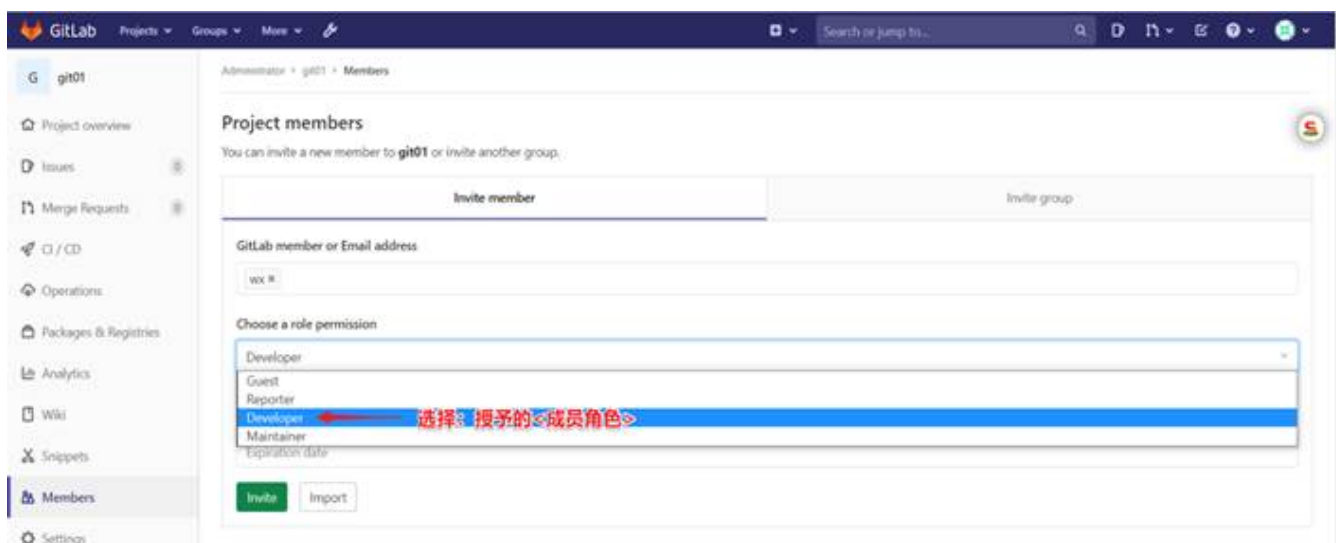
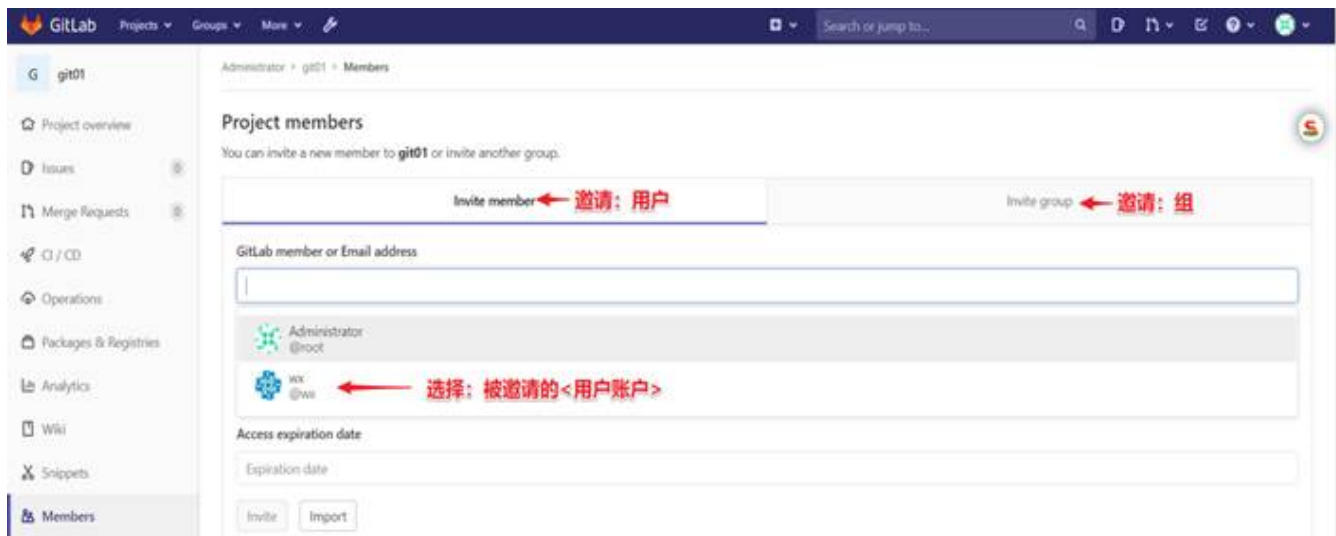
Git global setup

```
git config --global user.name "Administrator"
git config --global user.email "admin@example.com"
```

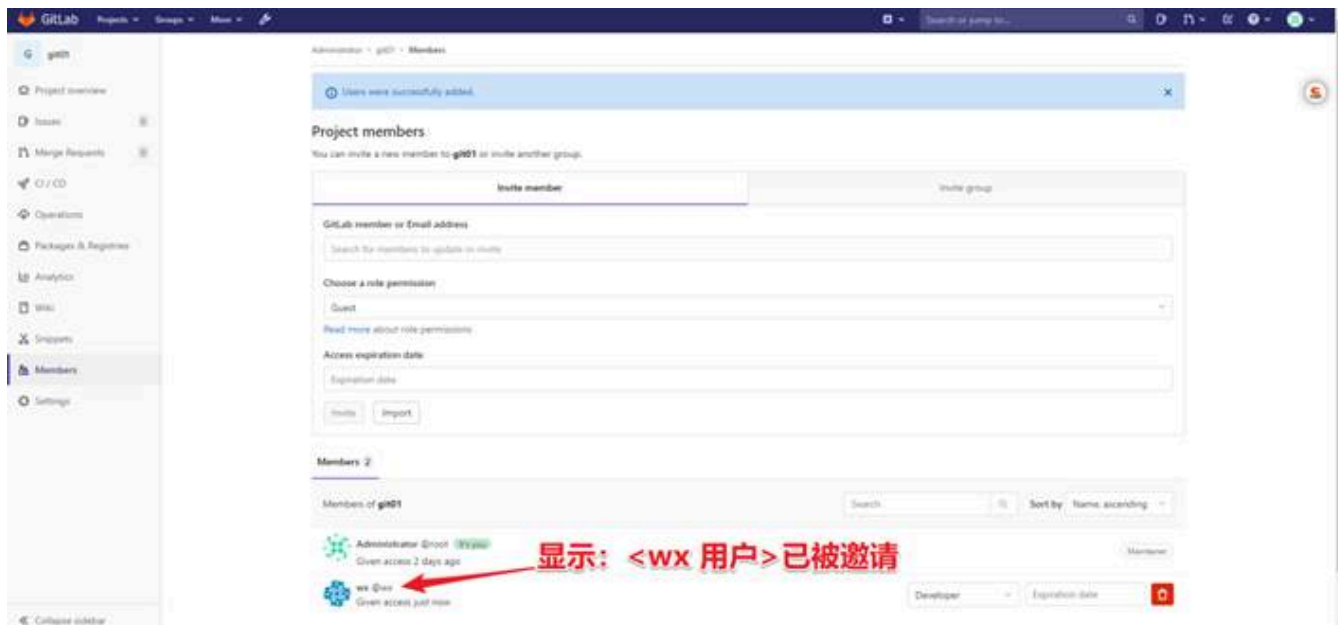
Create a new repository

```
git clone http://10.0.0.121/root/git01.git
cd git01
touch README.md
git add README.md
```

点击进入

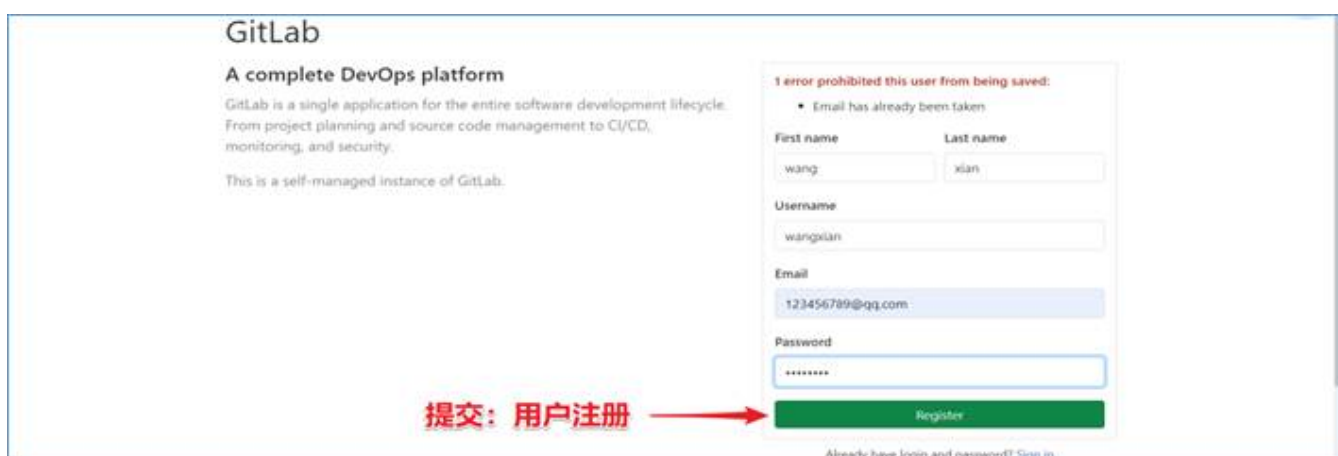
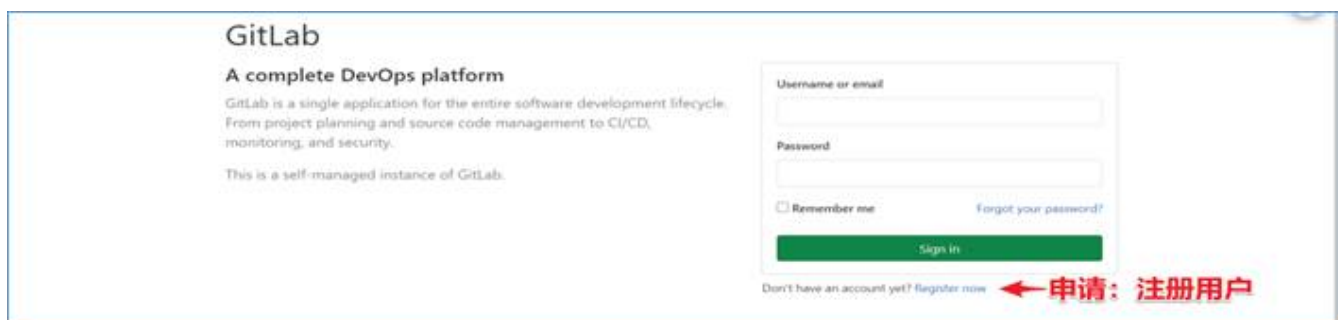


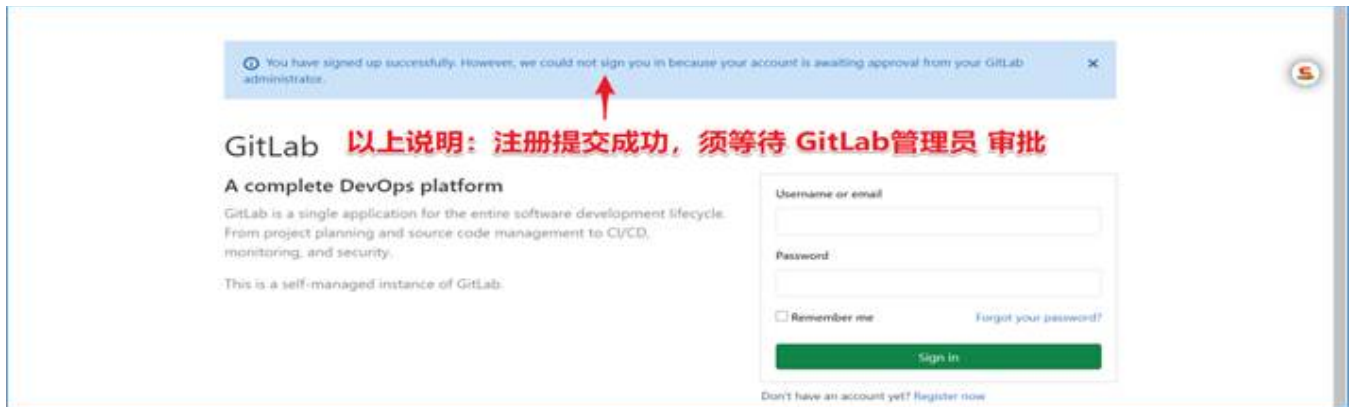




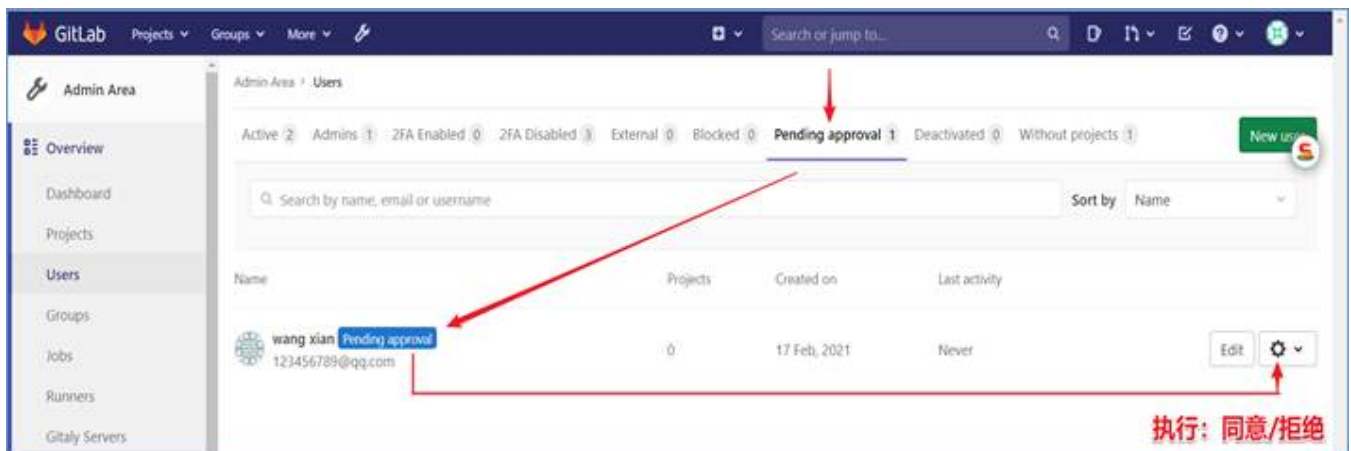
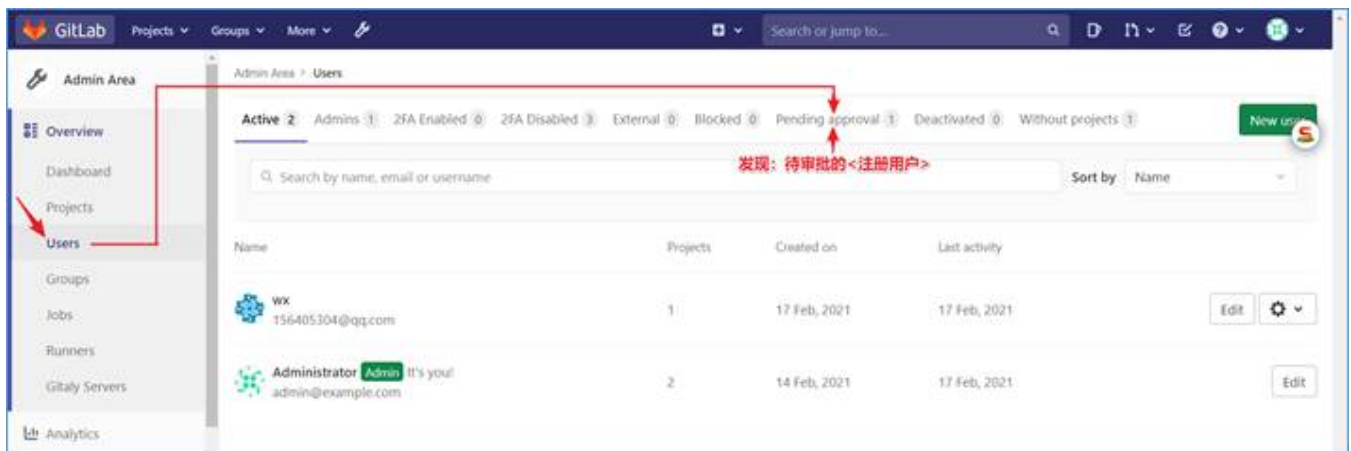
### 3.5.3. 审批: <注册用户>, 并允许其访问<你的私有仓库>

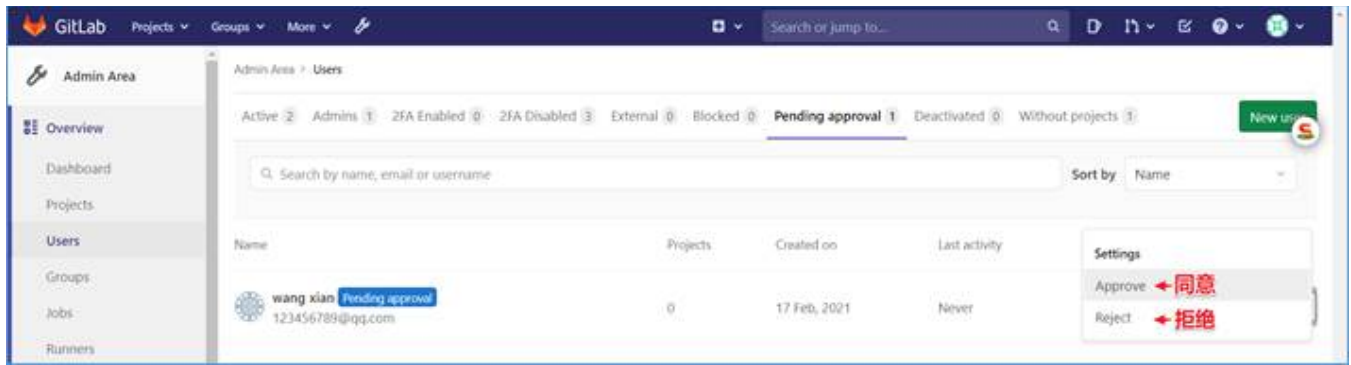
- GitLab允许他人执行注册<用户账户>, 但必须经过<root管理员>手动审批通过并授权访问<我的私有仓库>。
- (1) 提交: <注册用户wangxian>





• (2) 审批：<注册用户wangxian>





- (3) 授权: <注册用户wangxian>

略 ...

### 3.6. 删除: GitLab上<你的Git远程库>

