

本文介绍如何使用GitHub Pages + Hexo搭建个人博客网站，完全免费，所有内容本人亲测，绝对可用。

## 一、准备工作

### 1. GitHub账号

需要有一个 `GitHub` 账号，没有的话到 [官网](#) 申请一个。  
注册很简单，不懂的话可以参考 [GitHub申请账号](#)

### 2. 安装Git

在自己电脑上安装好 `Git`，[hexo](#) 部署到 `GitHub` 时要用。  
网上找篇教程或者参考 [Git安装\(Windows\)](#)。

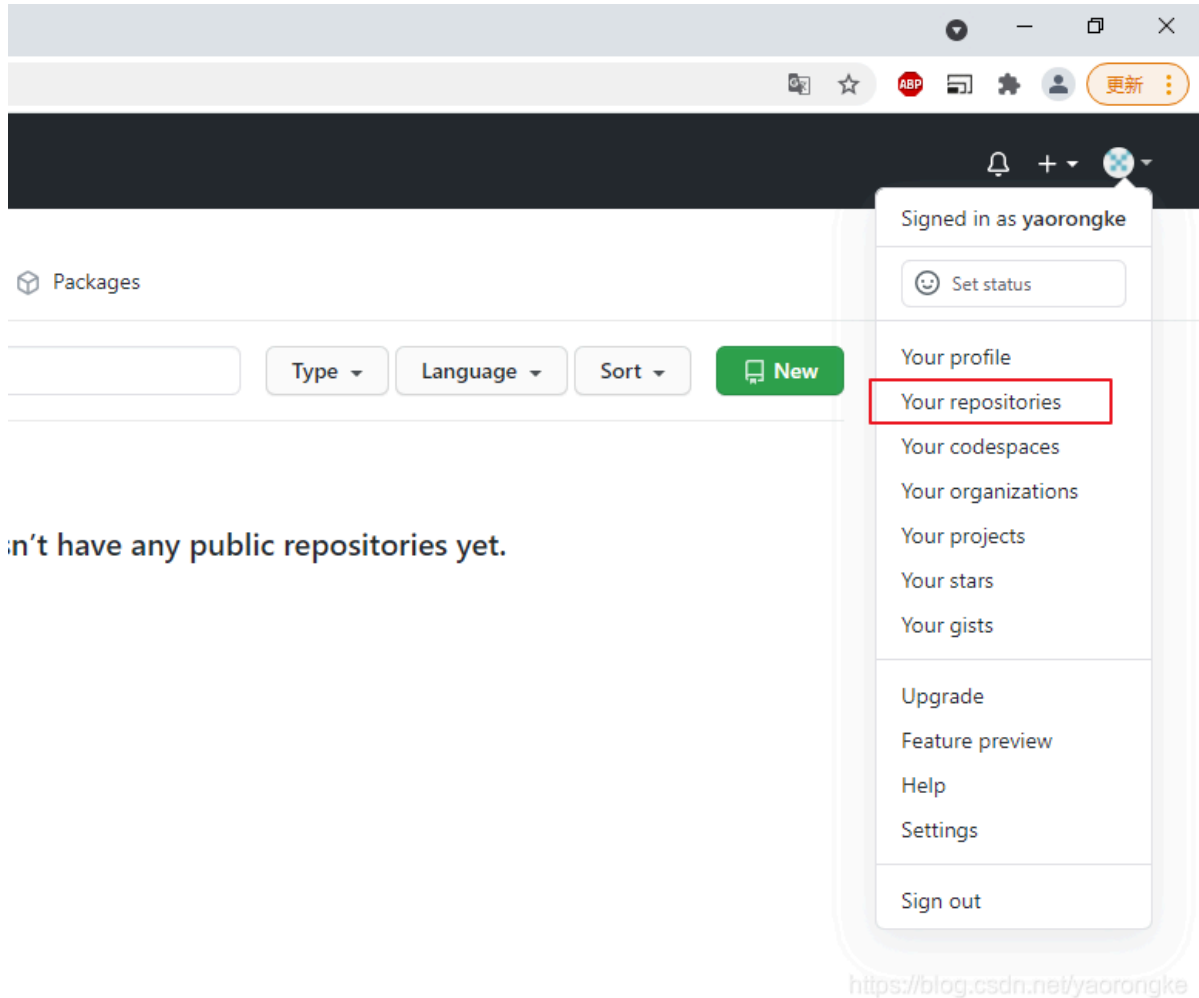
### 3. 安装NodeJS

在自己电脑上安装好 `NodeJS`，`Hexo` 是基于 `NodeJS` 编写的，所以需要安装 `NodeJS` 和 `npm` 工具。  
网上找篇教程或者参考 [NodeJS安装及配置\(Windows\)](#)。

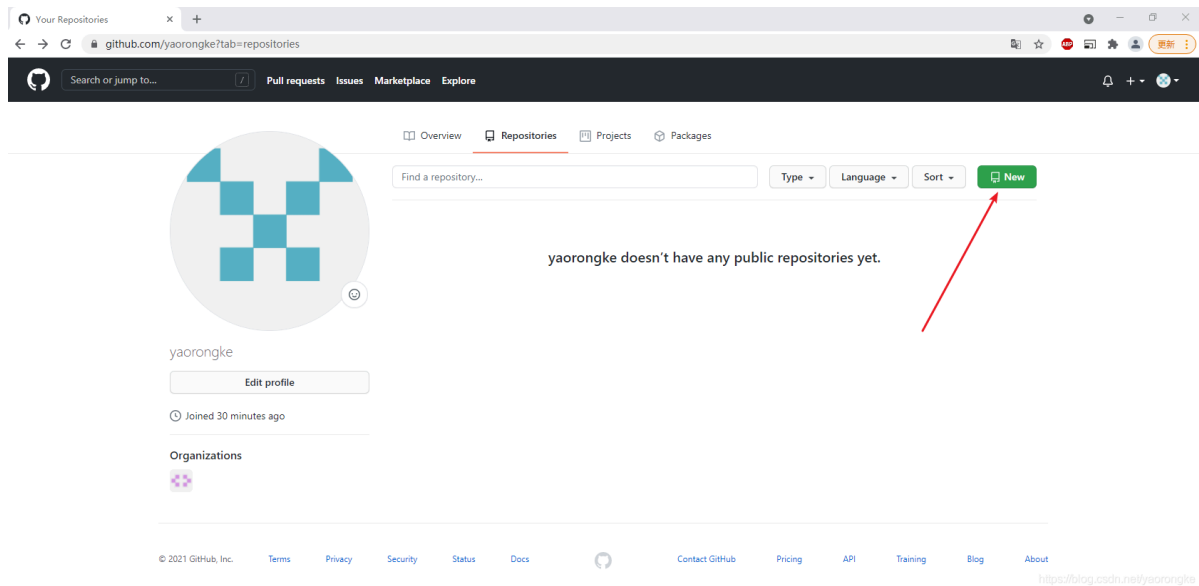
## 二、创建仓库

在 `GitHub` 上创建一个新的代码仓库用于保存我们的网页。

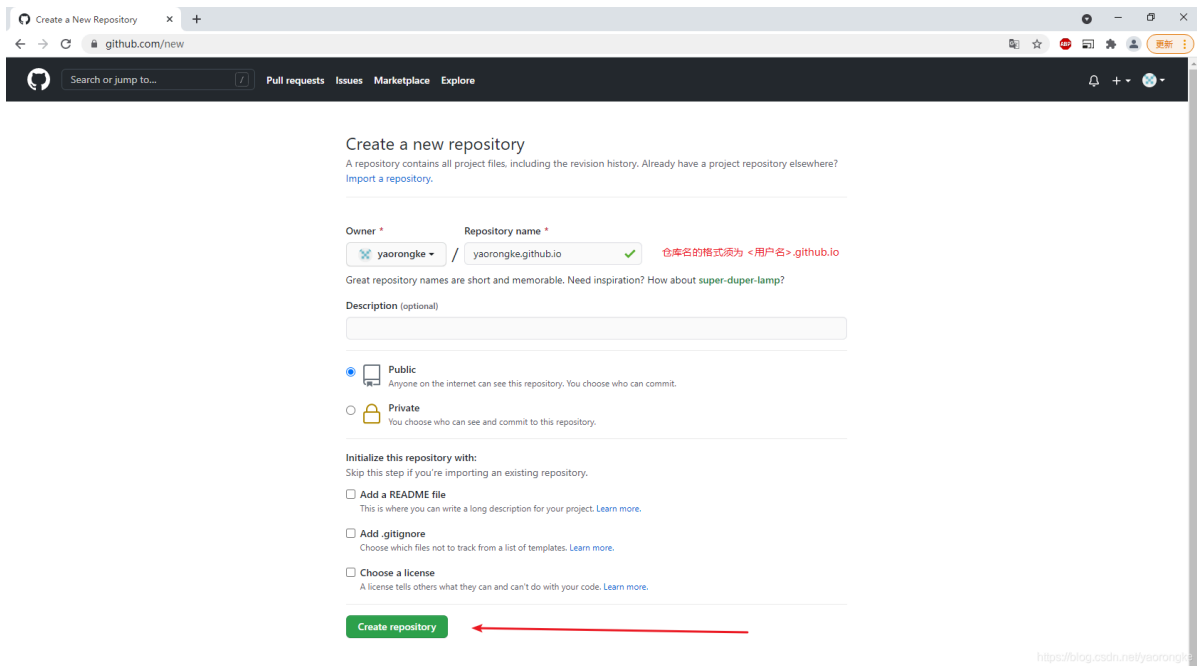
点击 **Your repositories** ，进入仓库页面。



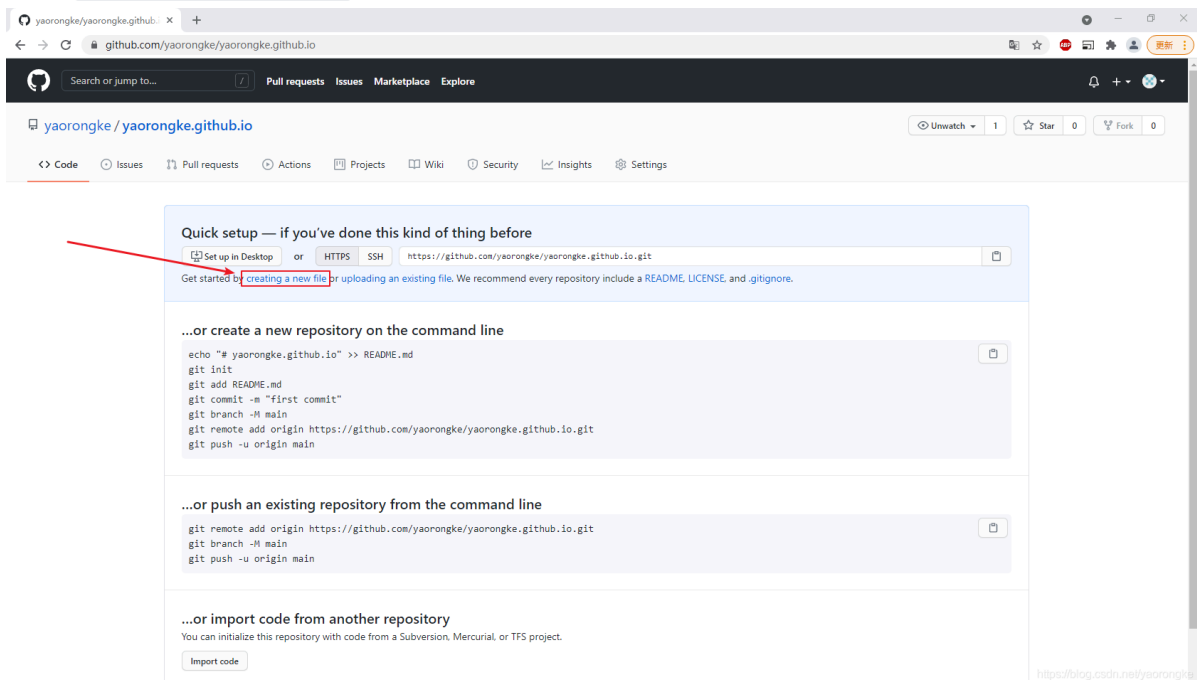
点击 **New** 按钮，进入仓库创建页面。



填写仓库名，格式必须为 **<用户名>.github.io** ，然后点击 **Create repository** 。

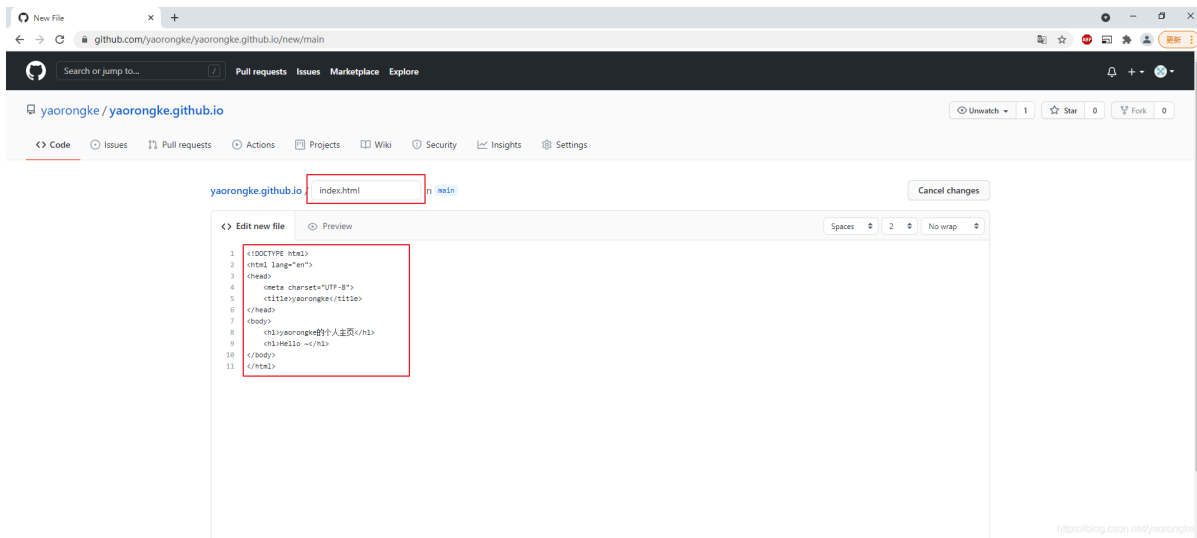


点击 **creating a new file** 创建一个新文件，作为我们网站的主页。

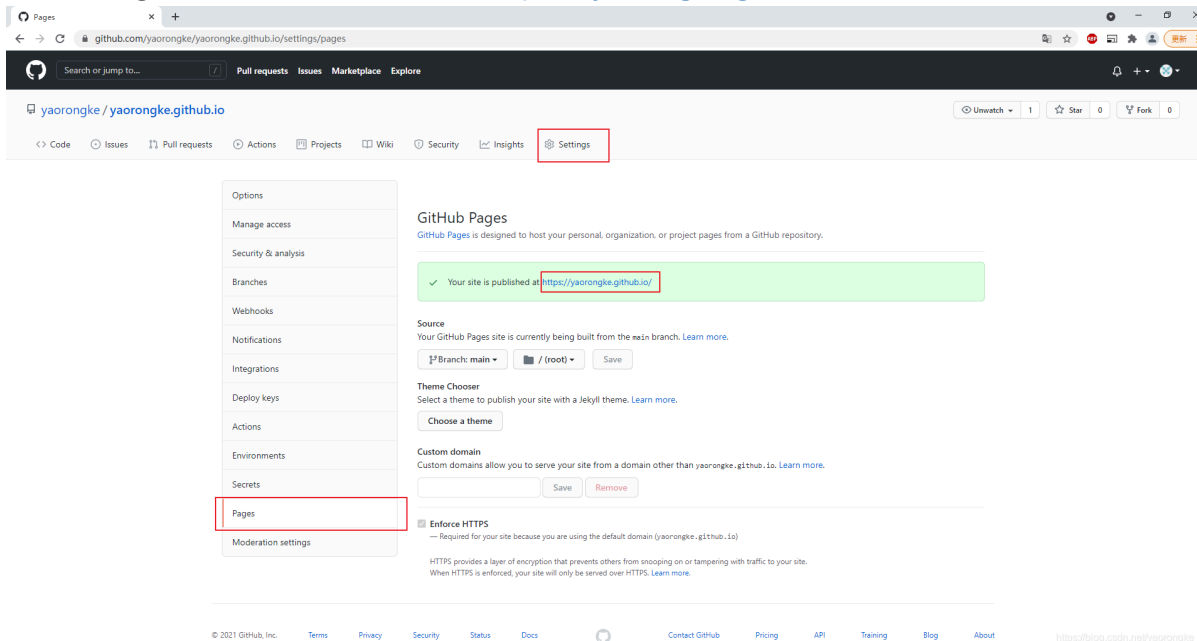


新文件的名字必须为 `index.html`，内容先随便写一个简单的，内容示例如下，填写之后点击 **Commit new file** 提交。

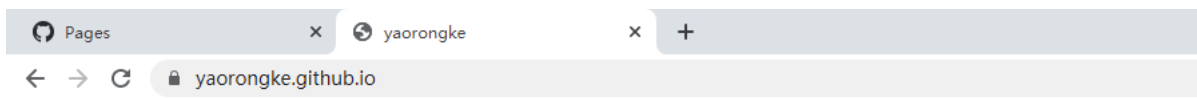
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>yaorongke</title>
6 </head>
7 <body>
8   <h1>yaorongke的个人主页</h1>
9   <h1>Hello ~</h1>
10 </body>
11 </html>
12 html1234567891011
```



GitHub Pages中找到我们主页的地址为 <https://yaorongke.github.io/>



浏览器中访问，展示成功。



## yaorongke的个人主页

Hello ~

这里创建的网页是非常简陋的，只是为了演示下 `GitHub Pages` 的使用方式。

### 三、安装Hexo

我们采用 `Hexo` 来创建我们的博客网站，`Hexo` 是一个基于 `NodeJS` 的静态博客网站生成器，使用 `Hexo` 不需开发，只要进行一些必要的配置即可生成一个个性化的博客网站，非常方便。点击进入 [官网](#)。

安装 `Hexo`

```
1 npm install -g hexo-cli
```

查看版本

```
1 | hexo -v
```

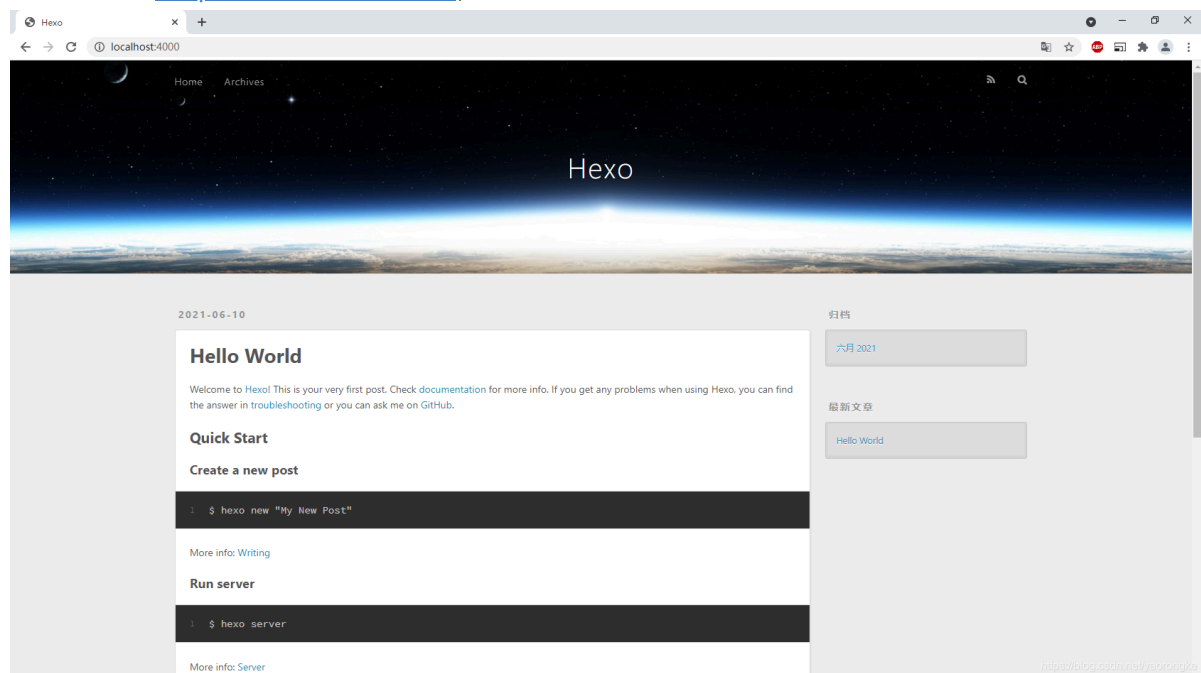
创建一个项目 `hexo-blog` 并初始化

```
1 | hexo init hexo-blog
2 | cd hexo-blog
3 | npm install
```

本地启动

```
1 | hexo g
2 | hexo server
```

浏览器访问 <http://localhost:4000>, 页面默认主图风格如下



## 四、更换主题

`Hexo` 默认的主题不太好看, 不过官方提供了数百种主题供用户选择, 可以根据个人喜好更换, 官网主题点 [这里](#) 查看。这里介绍两个主题的使用方法, `Next` 和 `Fluid`, 个人比较喜欢 `Fluid`, 后面章节的功能也是以 `Fluid` 为基础进行讲解的。

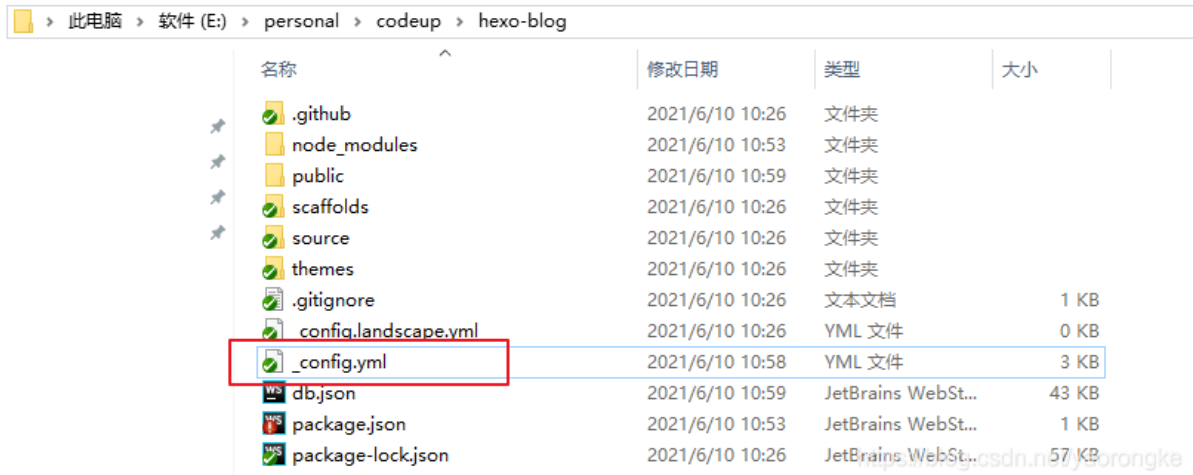
### 1. Next 主题

安装主题

```
1 | cd hexo-blog
2 | git clone https://github.com/iissnan/hexo-theme-next themes/next
```

使用 `Next` 主题

打开 `_config.yml` 文件，该文件为站点配置文件



将主题修改为 `next`

```
1 | theme: next
```

## 本地启动

```
1 | hexo g -d
2 | hexo s
```

## 2. Fluid主题

以下安装步骤摘自 [Fluid官网](#)

### 安装主题

下载 [最新 release 版本](#) 解压到 `themes` 目录，并将解压出的文件夹重命名为 `fluid`。

### 指定主题

如下修改 `Hexo` 博客目录中的 `_config.yml`：

```
1 | theme: fluid # 指定主题
2 | language: zh-CN # 指定语言，会影响主题显示的语言，按需修改
```

### 创建「关于页」

首次使用主题的「关于页」需要手动创建：

```
1 | hexo new page about
```

创建成功后，编辑博客目录下 `/source/about/index.md`，添加 `layout` 属性。

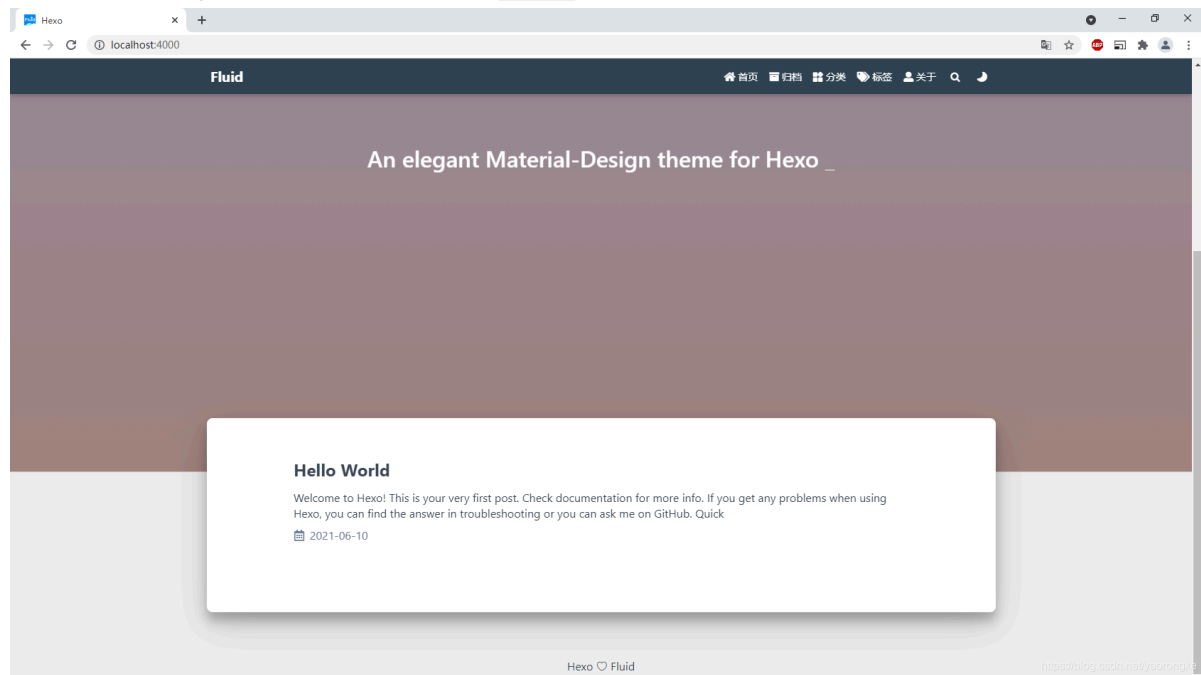
修改后的文件示例如下：

```
1 ---
2 title: about
3 date: 2020-02-23 19:20:33
4 layout: about
5 ---
6
7 这里写关于页的正文，支持 Markdown，HTML
8 1234567
```

## 本地启动

```
1 hexo g -d
2 hexo s
```

浏览器访问 <http://localhost:4000>, Fluid 主题风格页面如下



## 五、创建文章

如下修改 Hexo 博客目录中的 `_config.yml`，打开这个配置是为了在生成文章的时候生成一个同名的资源目录用于存放图片文件。

```
1 post_asset_folder: true
```

执行如下命令创建一篇新文章，名为《测试文章》

```
1 hexo new post 测试文章
```

执行完成后在 `source\_posts` 目录下生成了一个md文件和一个同名的资源目录(用于存放图片)



在资源目录 `测试文章` 中放一张图片 `test.png`



在 `测试文章.md` 中添加内容如下, 演示了图片的三种引用方式。第一种为官方推荐用法, 第二种为markdown语法, 第三种和前两种图片存放位置不一样, 是将图片放在 `\source\images` 目录下。这三种写法在md文件中图片是无法显示的, 但是在页面上能正常显示。

图片的引入方式可参考官方文档 <https://hexo.io/zh-cn/docs/asset-folders.html>, 有详细介绍。

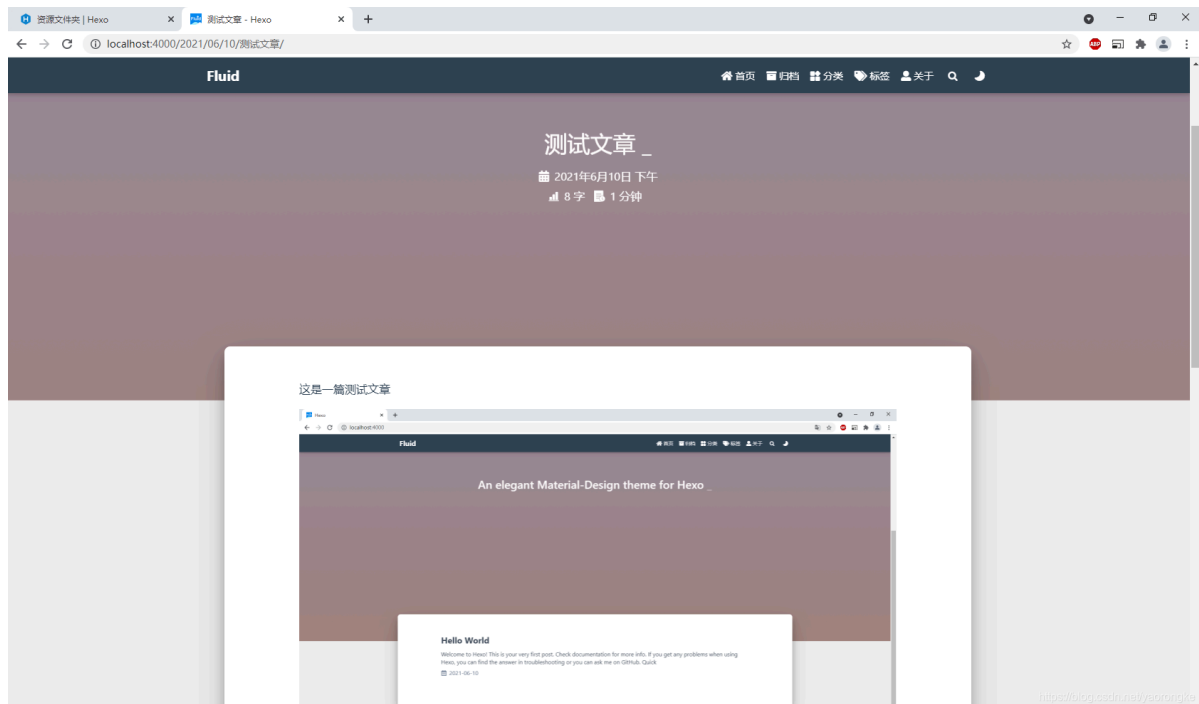
```
1 ---
2 title: 测试文章
3 date: 2021-06-10 16:35:20
4 tags:
5 - 原创
6 categories:
7 - Java
8 ---
9
10 这是一篇测试文章
11
12 {% asset_img test.png 图片引用方法一 %}
13
14 ![图片引用方法二](test.png)
15
16 ![图片引用方法三](/images/test.png)
17 markdown12345678910111213141516
```

## 本地启动

```
1 hexo g -d
2 hexo s
```



浏览器访问 <http://localhost:4000>, 页面如下, 文章添加成功



## 六、个性化页面展示

页面的标题等位置显示默认的文字, 可以改下显示一些个性化的信息。

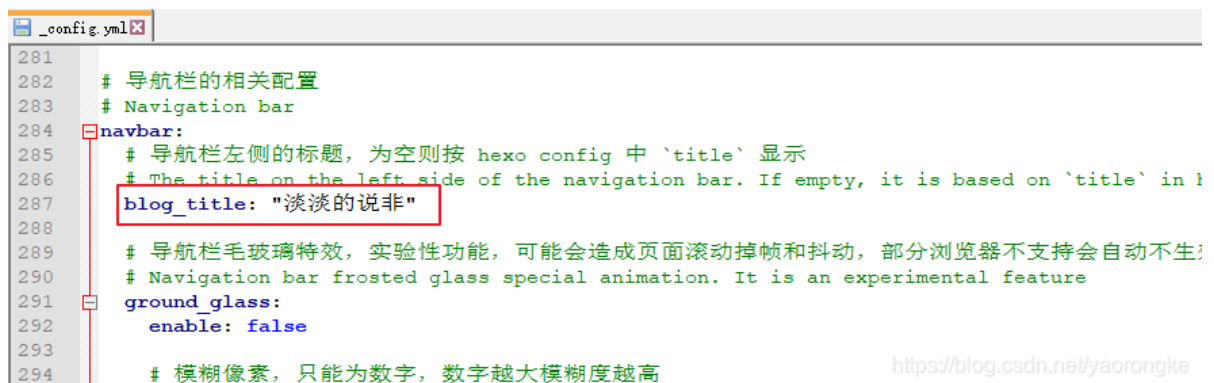
### 1. 浏览器tab页名称

修改根目录下 `_config.yml` 中的 `title` 字段。



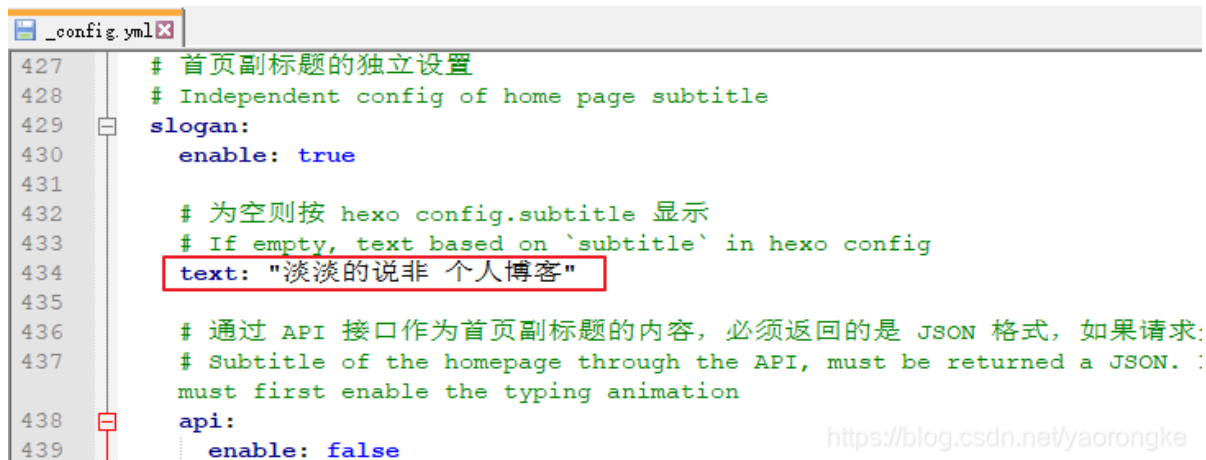
### 2. 博客标题

主题目录 `themes\fluid` 下 `_config.yml` 中的 `blog_title` 字段。



### 3. 主页正中间的文字

主题目录 `themes\fluid` 下 `_config.yml` 中的 `text` 字段。



修改好配置后，页面效果如下，可以看到现在显示的内容变成了我们的个人信息。

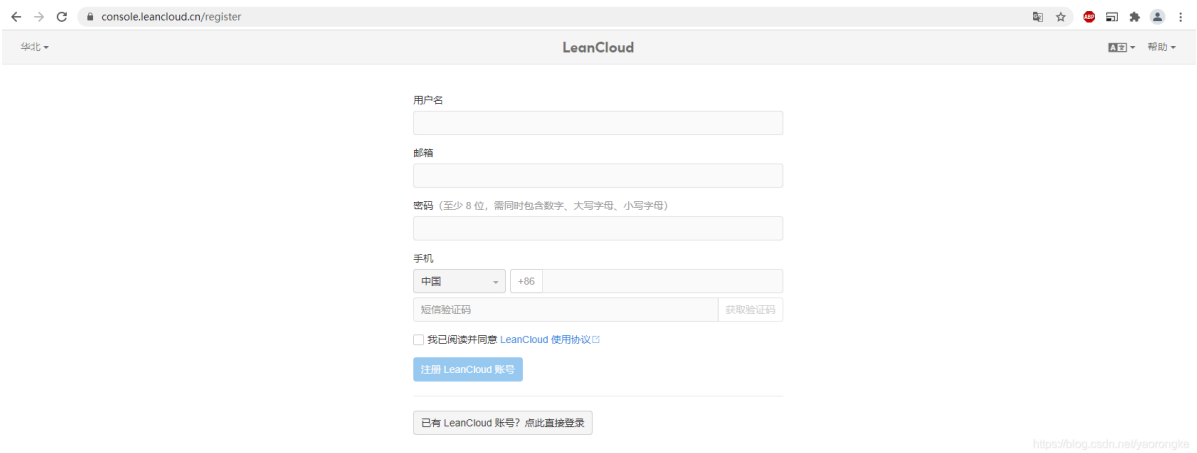


## 七、添加阅读量统计

`Fluid` 主题写好了统计阅读量的代码，但是缺少相应配置所以没有开启，需要借助三方服务来统计阅读量，这里是有 `Leancloud` 的免费服务来进行统计。

1. 申请LeanCloud账号并创建应用

进入 [官网](#) 注册账号



需实名认证，完成后才能使用各项服务



验证邮箱



创建应用，选择 [开发版](#) 即可，免费的



进入该应用的 **设置->应用凭证**，找到 **AppID** 和 **AppKey**，记录下来后面配置要用



## 2. 修改Fluid配置

打开主题目录 `themes\fluid` 下的 `_config.yml` 文件，修改如下配置

## 单篇文章阅读量计数

### 打开统计开关

```
_config.yml
202 # Customize <head> HTML content
203 custom_head: ''
204
205 # 自定义底部 HTML 内容（位于 footer 上方），注意不要和 `post: custom` 配置冲突
206 # Customize the HTML content at the bottom (located above the footer), be careful not to conflict
207 custom_html: ''
208
209 # 网页访问统计
210 # Analysis of website visitors
211 web_analytics: # 网页访问统计
212   enable: true
213
214 # 百度统计的 key, 值需要获取下方链接中 `hm.js?` 后边的字符串
215 # Baidu analytics, get the string behind `hm.js?`
216 # See: https://tongji.baidu.com/sc-web/10000033910/home/site/getjs?siteId=13751376
217 baidu:
218
```

<https://blog.csdn.net/yaorongke>

### 配置 Leancloud 的 app\_id 和 app\_key

```
_config.yml
228
229 # 腾讯统计的 H5 App ID, 开启高级功能才有cid
230 # Tencent analytics, set APP ID
231 # See: https://mta.qq.com/h5/manage/ctr\_app\_manage
232 tencent:
233   sid:
234   cid:
235
236 # 51.la 站点统计 ID
237 # 51.la analytics
238 # See: https://www.51.la/user/site/index
239 woyao1a: # 51.la 站点统计 ID, 参见
240
241 # 友盟/cnzz 站点统计 web_id
242 # cnzz analytics
243 # See: https://web.umeng.com/main.php?c=site&a=show
244 cnzz:
245
246 # LeanCloud 计数统计, 可用于 PV UV 展示, 如果 `web_analytics: enable` 没有开启, PV UV 展示只会查询
247 # LeanCloud count statistics, which can be used for PV UV display. If `web_analytics: enable` is f
248 increase
249 leancloud:
250   app_id: R22mI...jzGzoHsz
251   app_key: BTBFH...FhSD
252   # REST API 服务器地址, 国际版不填
253   # Only the Chinese mainland users need to set
254   server_url:
255
```

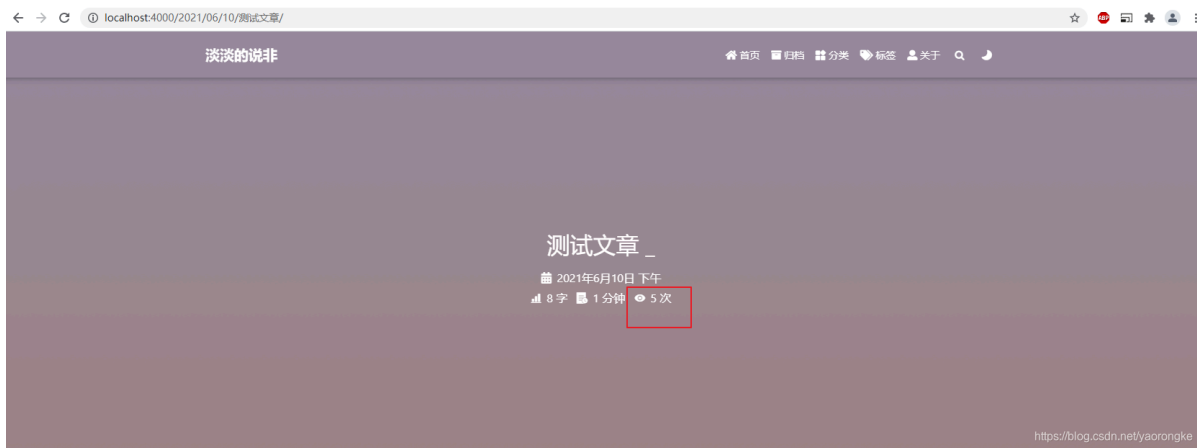
<https://blog.csdn.net/yaorongke>

### 打开计数功能, 统计来源改为 leancloud

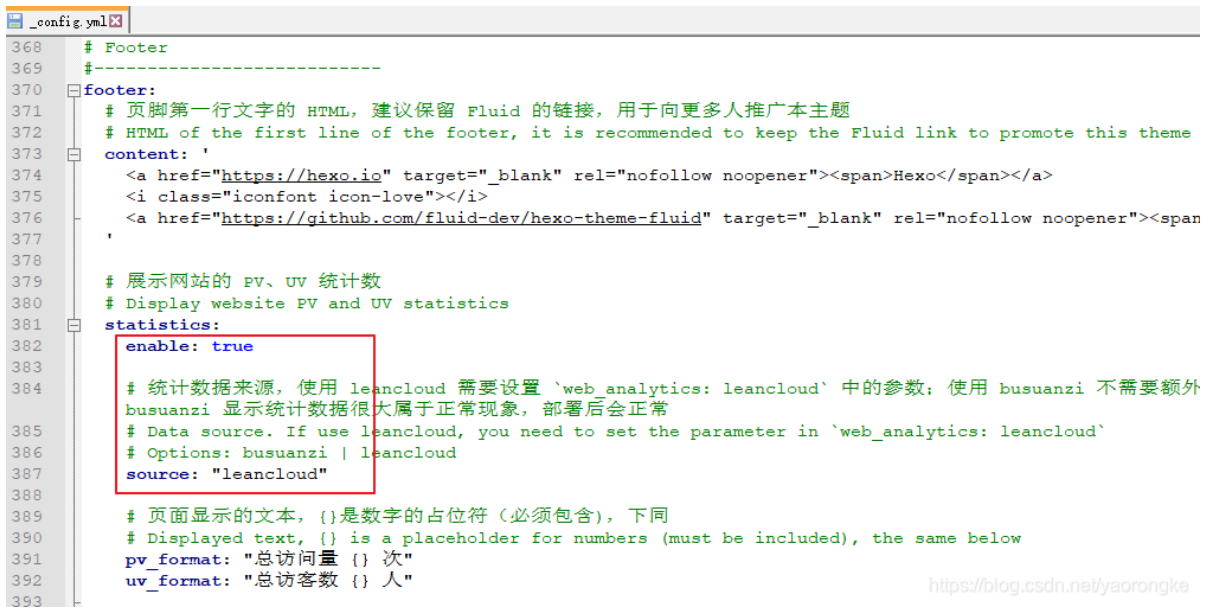
```
_config.yml
516 # 显示的文本, {}是数字的占位符（必须包含）, 下同
517 # Displayed text, {} is a placeholder for numbers (must be included), th
518 format: "{} 字"
519
520 # 估计阅读全文需要的时长
521 # Estimate the time required to read the full text
522 min2read:
523   enable: true
524   format: "{} 分钟"
525   # 每分钟阅读多少字, 如果是技术文章, 建议降低
526   # Words read per minute
527   words: 100
528
529 # 浏览量计数
530 # Number of visits
531 views:
532   enable: true
533   # 统计数据来源
534   # Data Source
535   # Options: busuanzi | leancloud
536   source: "leancloud"
537   format: "{} 次"
538
```

<https://blog.csdn.net/yaorongke>

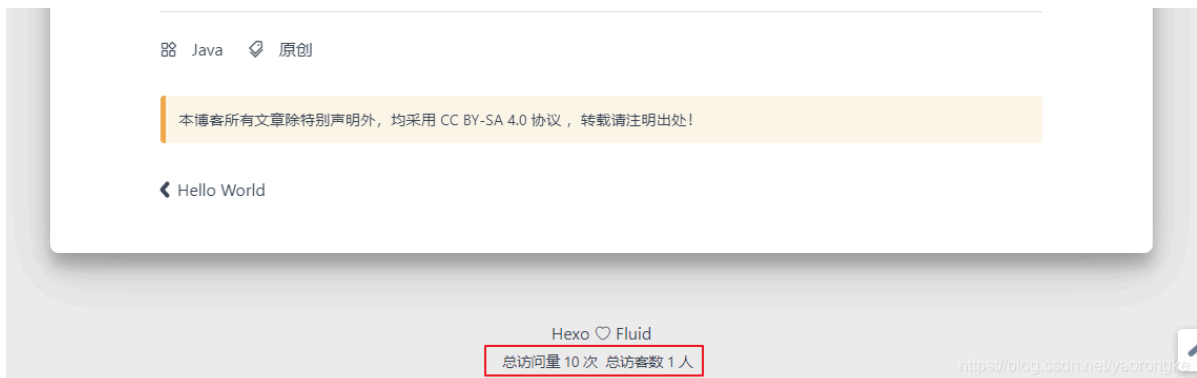
## 页面效果



## 页面底部展示网站的 PV、UV 统计数



## 页面效果



## 八、添加评论功能

评论功能的代码已经写好了，只不过没有开启，需要修改一些配置

打开主题目录 `themes\fluid` 下的 `_config.yml` 文件，修改如下配置

## 启用评论插件

```
_config.yml
620
621 # See: http://mermaid-js.github.io/mermaid/
622 options: { theme: 'default' }
623
624 # 评论插件
625 # Comment plugin
626
627 enable: true
628 # 指定的插件，需要同时设置对应插件的必要参数
629 # The specified plugin needs to set the necessary parameters at the same t
630 # Options: utterances | disqus | gitalk | valine | waline | changyan | liv
631 type: valine
632
```

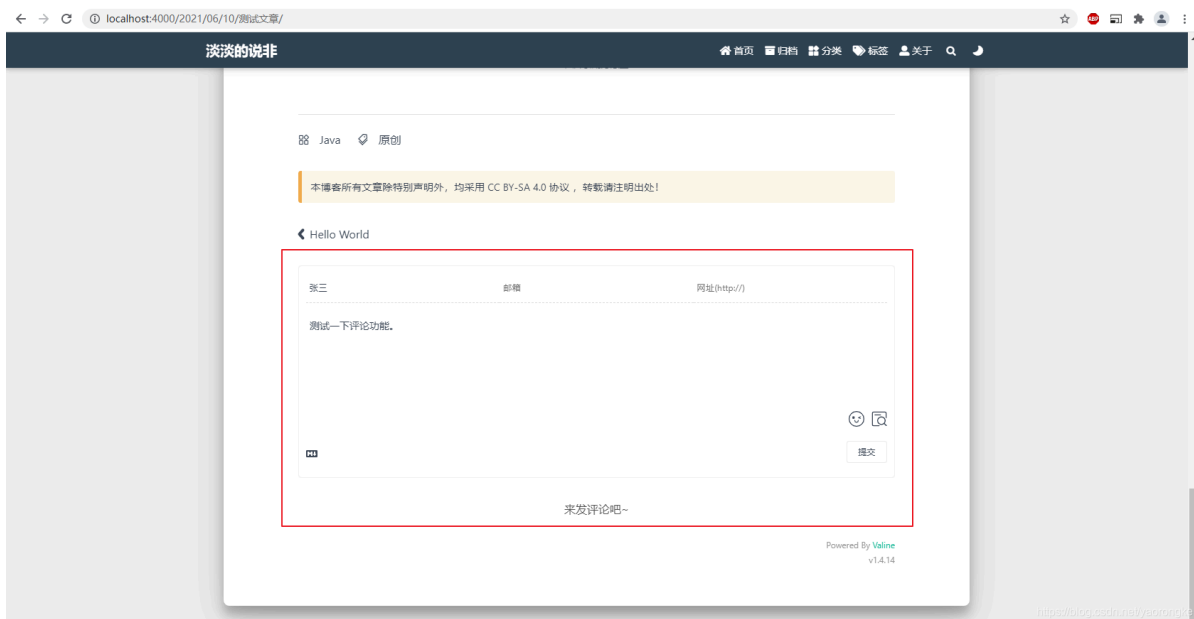
<https://blog.csdn.net/yaorongke>

## 配置 LeanCloud 的 appId 和 appkey

```
_config.yml
688 # Valine
689 # 基于 LeanCloud
690 # Based on LeanCloud
691 # See: https://valine.js.org/
692 valine:
693   appId: R22mI...-gzGzoHsz
694   appKey: BTBF...gFhSD
695   placeholder: 说点什么
696   path: window.location.pathname
697   avatar: retro
698   meta: ['nick', 'mail', 'link']
699   pageSize: 10
700   lang: zh-CN
701   highlight: false
702   recordIP: false
703   serverURLs: ''
704   emojiCDN:
705   emojiMaps:
706   enableQQ: false
707   requiredFields: []
708
```

<https://blog.csdn.net/yaorongke>

## 重新部署后，查看页面效果，评论功能已开启



部署在本地时评论无法提交，会报跨域问题，发布到 [GitHub Pages](#) 上之后可正常提交评论

## 九、发布到GitHub Pages

### 方式一

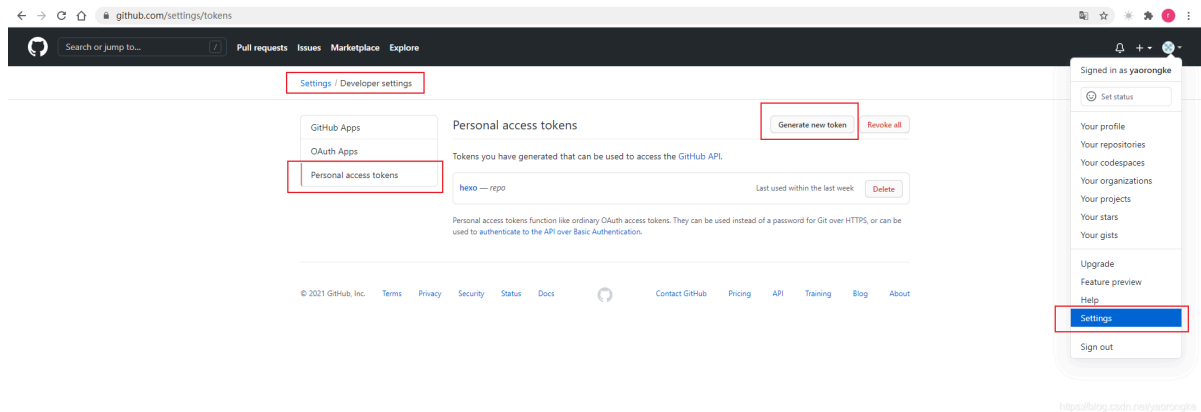
安装hexo-deployer-git

```
1 | npm install hexo-deployer-git --save
```

修改根目录下的 `_config.yml`，配置 [GitHub](#) 相关信息

```
1 deploy:
2   type: git
3   repo: https://github.com/yaorongke/yaorongke.github.io.git
4   branch: main
5   token: ghp_3KakcaPHERunNRyMerofcFd9pb1u282FSbsY
```

其中 `token` 为 `GitHub` 的 `Personal access tokens`，获取方式如下图

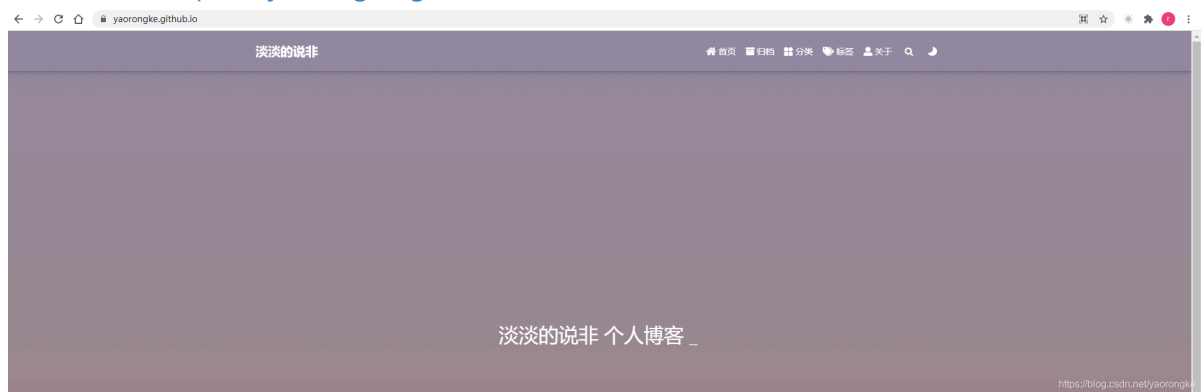


<https://blog.csdn.net/yaorongke>

部署到GitHub

```
1 hexo g -d
```

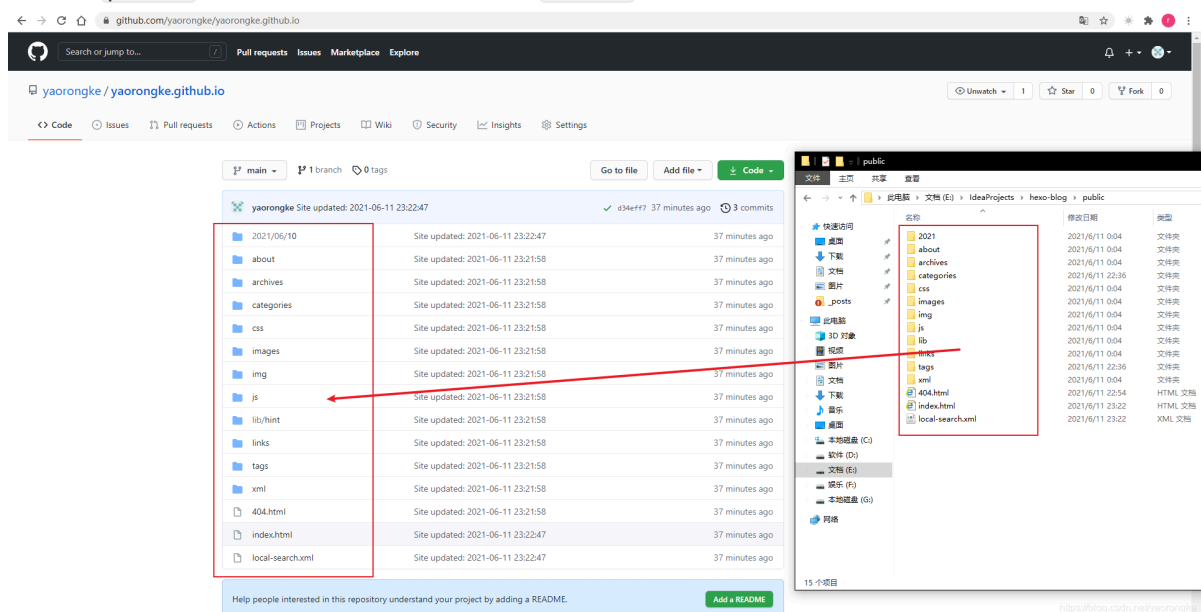
浏览器访问 <https://yaorongke.github.io/>，部署成功



<https://blog.csdn.net/yaorongke>

方式二

直接将 `public` 目录中的文件和目录推送至 `GitHub` 仓库和分支中。



<https://blog.csdn.net/yaorongke>



## 十、发布到自己服务器，Nginx代理

如果自己有服务器的话，也可以不使用 `GitHub Pages`，直接部署到自己的服务器上，通过 `Nginx` 进行代理，我这里有一个阿里云上的 `CentOS 7` 版的 `Linux` 服务器，演示下如何部署，步骤如下。

打开 `hexo-blog` 根目录下的 `_config.yml`，增加如下配置，这是因为把网站存放在子目录中，要和 `Nginx` 配置中的 `location /blog` 路径一致。

```
1 root: /blog
```

`hexo-blog` 根目录下执行打包命令，打包好的文件在 `public` 目录下

```
1 hexo g
```

将 `public` 目录下的文件复制到 `Linux` 服务器上的某个目录下，我的存放目录为

```
1 /opt/rkyao/fronted/hexo-blog
```

修改 `Nginx` 配置文件，我的 `Nginx` 安装路径为 `/usr/local`，大家根据自己实际情况调整

```
1 cd /usr/local/nginx/conf
2 vim nginx.conf
3 # server节点下添加如下配置
4 location /blog {
5     alias /opt/rkyao/fronted/hexo-blog;
6     index index.html index.htm;
7 }
```

重启 `Nginx`

```
1 cd /usr/local/nginx/sbin
2 ./nginx -s reload
```

访问博客

```
1 http://47.96.106.173/blog/
```

## 十一、最终效果展示

可访问如下地址查看

<https://yaorongke.github.io/>