

# 今日知行

## android 大作业

数据保存与python后端，安卓端（前端）的大部分动作通过http请求的方式拉取数据和上传数据

## 首页和帖子

### POST 点赞帖子

```
http://120.27.247.14/like_post?post_id=1&user_id=1
```

请求后用户对帖子完成点赞  
再次发送相同请求可以取消点赞

#### PARAMS

**post\_id**

1

**user\_id**

1

#### BODY formdata

#### Example Request

```
点赞帖子
```

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = new MultipartBody.Builder().setType(MultipartBody.FORM)
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/like_post?post_id=1&user_id=1")
    .method("POST", body)
    .build();
Response response = client.newCall(request).execute();
```

#### Example Response

200 — OK

```
{
  "favorites": 0,
  "img": "1.jpg",
  "postTime": "2019-12-27T07:42:00+00:00",
  "post_id": 1,
  "contend": "哈哈",
  "circle_id": 1,
  "likes": 1,
  "user_id": 1
}
```

## POST 评论帖子

http://120.27.247.14/comment?post\_id=1&user\_id=1&content=hA

#### PARAMS

##### post\_id

1

##### user\_id

1

##### content

hA

#### Example Request

评论帖子

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = RequestBody.create(mediaType, "");
Request request = new Request.Builder()
    .url("http://120.27.247.14/comment?post_id=1&user_id=1&contend=hhh")
    .method("POST", body)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK

```
{
  "commentTime": "2019-12-27T07:43:32+00:00",
  "post_id": 1,
  "contend": "hhh",
  "likes": 0,
  "user_id": 1
}
```

POST 点赞评论

```
http://120.27.247.14/like_comment?comment_id=1&user_id=1
```

PARAMS

**comment\_id**

1

**user\_id**

1

Example Request

点赞评论

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = RequestBody.create(mediaType, "");
Request request = new Request.Builder()
    .url("http://120.27.247.14/like_comment?comment_id=1&user_id=1")
    .method("POST", body)
    .build();
Response response = client.newCall(request).execute();
```

## GET 查看某帖子评论

http://120.27.247.14/comment?post\_id=1

### PARAMS

#### post\_id

1

### Example Request

查看某帖子评论

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/comment?post_id=1")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

### Example Response

200 — OK

```
[
  {
    "commentTime": "2019-12-27T07:43:32+00:00",
    "post_id": 1,
    "contend": "hhh",
    "likes": 0,
    "user_id": 1
  },
  {
    "commentTime": "2019-12-27T07:44:06+00:00",
```

## POST 收藏帖子

```
http://120.27.247.14/favorite?post_id=3&user_id=1
```

### HEADERS

#### Content-Type

application/x-www-form-urlencoded

### PARAMS

#### post\_id

3

#### user\_id

1

### Example Request

#### 收藏帖子

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("application/x-www-form-urlencoded");
RequestBody body = RequestBody.create(mediaType, "");
Request request = new Request.Builder()
    .url("http://120.27.247.14/favorite?post_id=1&user_id=1")
    .method("POST", body)
    .addHeader("Content-Type", "application/x-www-form-urlencoded")
    .build();
Response response = client.newCall(request).execute();
```

Example Response

```
200 — OK

{
  "favorites": 1,
  "img": "1.jpg",
  "postTime": "2019-12-27T07:42:00+00:00",
  "post_id": 1,
  "contend": "哈哈哈",
  "circle_id": 1,
  "likes": 1,
  "user_id": 1
}
```

POST 屏蔽帖子

```
http://120.27.247.14/block_post?post_id=3&user_id=2
```

该帖子不再显示

PARAMS

**post\_id**

3

**user\_id**

2

Example Request

```
屏蔽帖子

OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = RequestBody.create(mediaType, "");
Request request = new Request.Builder()
    .url("http://120.27.247.14/block_post?post_id=3&user_id=2")
    .method("POST", body)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK
"success"

POST 发帖

http://120.27.247.14/post

BODY formdata

user\_id

1

content

hhhab  
文字

circle\_id

1  
属于哪个圈子

img

http://120.27.247.14/static/6.png  
图片链接

Example Request

发帖

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = new MultipartBody.Builder().setType(MultipartBody.FORM)
    .addFormDataPart("user_id", "1")
    .addFormDataPart("content", "hhhab")
    .addFormDataPart("circle_id", "1")
    .addFormDataPart("img", "http://120.27.247.14/static/6.png")
    .build();
Request request = new Request.Builder()
```

## GET 选择圈子（板块）并获取帖子列表

```
http://120.27.247.14/get_by_circle?circle_id=1&user_id=2&start=1&end=5
```

返回的是该圈子按发帖时间（或者推荐系统顺序）排序后的帖子id列表

具体每个帖子的内容要使用获取帖子内容的api里面根据帖子id查询

请求排序顺序下的[start, end]帖子（如果帖子数<end, 则选取[start, 帖子数]）

### PARAMS

#### circle\_id

1

圈子id

#### user\_id

2

用户id

#### start

1

#### end

5

Example Request



选择圈子（板块）并获取帖子列表

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/get_by_circle?circle_id=1&user_id=2&start=1&end=5")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK

```
[
  {
    "circle_id": 1,
    "likes": 0,
    "user_id": 2,
    "favorites": 0,
    "img": "1.jpg",
    "contend": "hhhab",
    "post_id": 4,
    "postTime": "2019-12-27T08:15:26+00:00"
```

GET 获取帖子信息

```
http://120.27.247.14/get_post_info?post_id=2
```

帖子最多9图，视频的相同原理但是后面如果可以再进行迭代开发，通过访问igx\_url可以直接访问到图片并且进行下载

PARAMS

**post\_id**

2

Example Request

获取帖子信息

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/get_post_info?post_id=2")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK

```
{
  "circle_id": 2,
  "likes": 0,
  "user_id": 1,
  "favorites": 0,
  "img": "1.jpg",
  "contend": "hhh",
  "post_id": 2,
  "postTime": "2019-12-27T07:42:28+00:00"
}
```

GET 搜索帖子

http://120.27.247.14/search\_post?key=hh

PARAMS

key

hh

Example Request

搜索帖子

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/search_post?key=hh")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK
<pre>[   {     "circle_id": 2,     "likes": 0,     "user_id": 1,     "favorites": 0,     "img": "1.jpg",     "contend": "hhh",     "post_id": 2,     "postTime": "2019-12-27T07:42:28+00:00"</pre>

GET 获取所有帖子

http://120.27.247.14/get\_posts

HEADERS

BODY formdata

user\_id

1

contend

今天天气很好  
文字

circle\_id

1  
属于哪个圈子

img

1.jpg

图片链接

Example Request

获取所有帖子

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/get_posts")
    .method("GET", null)
    .addHeader("", "")
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK

```
{
  "status": "success",
  "data": [
    {
      "post_id": 1,
      "img": "1.jpg",
      "likes": 1,
      "favorites": 1,
      "circle_id": 1,
      "contend": "2333",
```

# 登陆与注册

## GET 登录验证

http://120.27.247.14/login

# 返回码

- 0 账号密码正确，且不是首次登陆
- 1 账号密码正确，首次登陆（需设置用户信息）
- 2 账号未注册
- 3 账号密码错误

# 有关密码加密

demo可以先不做

后面迭代可以将密码变成加密后再传，md5之类的

## HEADERS

### Content-Type

application/json

## BODY raw

```
{
  "username": "23",
  "password": "1234"
}
```

## Example Request

登陆验证

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/login")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

## Example Response

—

```
{  
  "status_code": 2,  
  "user_id": 1  
}
```

## POST 注册

http://120.27.247.14/register

## 流程

用户提交账号、密码、学生邮箱后（确认密码这一项在前端进行就可以）

给出提示“确认邮件已发送，请查收”，并跳转回登陆界面

用户在点击邮件中的链接后即可完成注册

## 返回码

0 注册成功，确认邮件已发送

1 用户名已存在

2 邮箱已注册

3 邮箱非北交邮箱

## HEADERS

### Content-Type

application/json

## BODY raw

```
{  
  "username": "234",  
  "password": "1234",  
  "email" : "234@bjtu.edu.cn"  
}
```

## Example Request

注册

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("application/json");
RequestBody body = RequestBody.create(mediaType, "{\n\t\"username\": \"hzj\", \n\t\"password\": \"1234\", \n\t\"email\": \"1234567890@qq.com\"}");
Request request = new Request.Builder()
    .url("127.0.0.1:5000/register")
    .method("POST", body)
    .addHeader("Content-Type", "application/json")
    .build();
```

Example Response

201 — CREATED

```
{
  "status_code": 1
}
```

# 消息列表

私聊列表还是得在安卓端用文件的形式储存，实现起来可能的确有点麻烦..不过这个不适合由后端做，不然程序效率会很低

## GET 只看评论

```
http://120.27.247.14/view_my_comment?user_id=1
```

PARAMS

**user\_id**

1

Example Request

只看评论

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/view_my_comment?user_id=1")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

#### Example Response

200 — OK

```
[
  {
    "likes": 0,
    "user_id": 1,
    "commentTime": "2019-12-27T07:43:32+00:00",
    "contend": "hhh",
    "post_id": 1
  },
  {
    "likes": 0,
```

## GET 只看赞

```
http://120.27.247.14/view_my_like?user_id=1
```

为了简洁，只返回点赞的id，具体每个帖子的信息可以调用"获取帖子信息"的api来获取

#### PARAMS

**user\_id**

1

#### Example Request

只看赞



```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/view_my_like?user_id=1")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

#### Example Response

200 — OK

```
[
  {
    "likeTime": "2019-12-27T07:43:10+00:00",
    "post_id": 1,
    "user_id": 1
  }
]
```

## GET 只看收藏

http://120.27.247.14/view\_my\_favourite?user\_id=1

#### PARAMS

**user\_id**

1

#### Example Request

只看收藏

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/view_my_favourite?user_id=1")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

```
200 — OK

[
  {
    "favoriteTime": "2019-12-27T07:44:32+00:00",
    "post_id": 1,
    "user_id": 1
  },
  {
    "favoriteTime": "2019-12-27T09:52:12+00:00",
    "post_id": 3,
    "user_id": 1
  }
]
```

# 用户管理

包括用户基本信息、头像地址 以及圈子都可以在这里改 注：如果加的圈子不足5个，则剩余的圈子值为-1

## GET 获取用户基本资料

```
http://120.27.247.14/view_user_info?user_id=2
```

获取用户昵称、用户简介，用户头像，已加入圈子

PARAMS

**user\_id**

2

Example Request

```
获取用户基本资料
```

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://127.0.0.1:5000/view_user_info?user_id=1")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

#### Example Response

200 — OK

```
{
  "intro": "balabala",
  "circle1": 1,
  "profile": "1.jpg",
  "password": "1234",
  "circle3": 4,
  "circle2": 2,
  "circle4": 10,
  "username": "abc",
  "nick_name": "Master.",
```

## PUT 编辑用户资料

http://120.27.247.14/edit\_user\_info

可以用在两个地方：

1. 首次登陆的话要设置用户资料
2. 修改用户资料

#### HEADERS

##### Content-Type

application/json

#### BODY raw

```
{
  "user_id": 1,
  "nick_name": "Master.",
  "intro": "balabala",
  "profile": "http://120.27.247.14/static/5.png"
}
```

Example Request

编辑用户资料

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("application/json");
RequestBody body = RequestBody.create(mediaType, "{\n\t\"user_id\": 2,\n\t\t\"nick_name\": \"Master.\",\n\t\t\"intro\"");
Request request = new Request.Builder()
    .url("http://120.27.247.14/edit_user_info")
    .method("PUT", body)
    .addHeader("Content-Type", "application/json")
    .build();
```

Example Response

200 — OK

```
{
  "password": "1234",
  "profile": "1.jpg",
  "circle4": -1,
  "circle3": -1,
  "circle2": -1,
  "username": "2",
  "user_id": 2,
  "nick_name": "Master.",
  "circle5": -1,
```

GET 查看历史发帖

```
http://120.27.247.14/history_post?user_id=2
```

PARAMS

user\_id

Example Request

查看历史发帖

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/history_post?user_id=2")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK

```
[
  {
    "user_id": 2,
    "favorites": 0,
    "contend": "hhhab",
    "post_id": 3,
    "likes": 0,
    "postTime": "2019-12-27T08:04:27+00:00",
    "img": "1.jpg",
    "circle_id": 1
```

# 圈子相关

## GET 查看所有可加入圈子

http://120.27.247.14/view\_circles

Example Request

查看所有可加入圈子

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/view_circles")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

200 — OK

```
[
  {
    "circle_name": "魔方",
    "circle_id": 1,
    "people_number": 1
  },
  {
    "circle_name": "学习",
    "circle_id": 2,
    "people_number": 1
  }
]
```

GET 搜索圈子

http://120.27.247.14/search\_circle?key=学

理论上能支持模糊搜索

PARAMS

key  
学

Example Request

搜索圈子

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/search_circle?key=学")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

#### Example Response

200 — OK

```
[
  {
    "circle_name": "学习",
    "circle_id": 2,
    "people_number": 1
  }
]
```

## POST 加入/创建圈子

http://120.27.247.14/join\_circle?circle\_name=魔方&user\_id=2

加入圈子和创建圈子共用一个api，若该圈子未曾存在则自动创建，否则就是直接加入

#### PARAMS

##### circle\_name

魔方

##### user\_id

2

#### Example Request

该用户已经加入过该圈子 ▼

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = RequestBody.create(mediaType, "");
Request request = new Request.Builder()
    .url("http://120.27.247.14/create_circle?circle_name= 学习&user_id=2")
    .method("POST", body)
    .build();
Response response = client.newCall(request).execute();
```

#### Example Response

200 — OK

```
{
  "status": 1,
  "message": "The user has joined this circle!"
}
```

## GET 获取圈子信息

http://120.27.247.14/get\_circle?circle\_id=1

#### PARAMS

##### circle\_id

1

#### Example Request

获取圈子信息

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/get_circle?circle_id=1")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```



Example Response

```
200 — OK

{
  "circle_name": "魔方",
  "circle_id": 1,
  "people_number": 1
}
```

POST 退出圈子

```
http://120.27.247.14/quit_circle?circle_name=学习&user_id=2
```

PARAMS

**circle\_name**

学习

**user\_id**

2

Example Request

```
该用户未加入该圈子 ▼

OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = RequestBody.create(mediaType, "");
Request request = new Request.Builder()
    .url("http://120.27.247.14/quit_circle?circle_name=学习&user_id=2")
    .method("POST", body)
    .build();
Response response = client.newCall(request).execute();
```

Example Response

```
200 — OK
```

```
{
  "status": 1,
  "message": "The user has not joined this circle!"
}
```

## 图片

### GET 根据图片链接下载图片或视频

http://120.27.247.14/static/2.jpg

BODY formdata

#### Example Request

根据图片链接下载图片或视频

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/static/2.jpg")
    .method("GET", null)
    .build();
Response response = client.newCall(request).execute();
```

### POST 上传图片或视频

http://120.27.247.14/image

BODY formdata

**image**

## Example Request

上传图片

```
OkHttpClient client = new OkHttpClient().newBuilder()
    .build();
MediaType mediaType = MediaType.parse("text/plain");
RequestBody body = new MultipartBody.Builder().setType(MultipartBody.FORM)
    .addFormDataPart("image", "Screenshot from 2019-12-11 16-45-40.png",
        RequestBody.create(MediaType.parse("application/octet-stream"),
            new File("/home/huzujun/Pictures/Screenshot from 2019-12-11 16-45-40.png")))
    .build();
Request request = new Request.Builder()
    .url("http://120.27.247.14/image")
```

## Example Response

200 — OK

```
"http://120.27.247.14/static/8.png"
```