

【Android】Retrofit网络请求Service, @Path、@Query、@QueryMap...



作者 带心情去旅行 (/u/769d3d3a9d4b) + 关注

2016.04.22 16:11* 字数 902 阅读 6415 评论 15 喜欢 83

(/u/769d3d3a9d4b)

对Retrofit已经使用了一段时间了，是时候归纳一下各种网络请求的service了。

下面分为GET、POST、DELETE还有PUT的请求，说明@Path、@Query、@QueryMap、@Body、@Field的用法。

初始化Retrofit

```
String BASE_URL = "http://102.10.10.132/api/";
Retrofit retrofit = new Retrofit.Builder()
    .baseUrl(BASE_URL)
    .build();
```

GET

样式1（一个简单的get请求）

http://102.10.10.132/api/News (http://102.10.10.132/api/News)

```
@GET("News")
Call<NewsBean> getItem();
```

样式2（URL中有参数）

http://102.10.10.132/api/News/1 (http://102.10.10.132/api/News/1)

http://102.10.10.132/api/News/{资讯id}

(http://102.10.10.132/api/News/%7B%E8%B5%84%E8%AE%AFid%7D)

```
@GET("News/{newsId}")
Call<NewsBean> getItem(@Path("newsId") String newsId);
```

或

http://102.10.10.132/api/News/1/类型1

(http://102.10.10.132/api/News/1/%E7%B1%BB%E5%9E%8B1)

http://102.10.10.132/api/News/{资讯id}/{类型}

(http://102.10.10.132/api/News/%7B%E8%B5%84%E8%AE%AFid%7D/%7B%E7%B1%BB%E5%9E%8B%7D)

```
@GET("News/{newsId}/{type}")
Call<NewsBean> getItem(@Path("newsId") String newsId, @Path("type") String type)
```

样式3（参数在URL问号之后）

http://102.10.10.132/api/News?newsId=1 (http://102.10.10.132/api/News?newsId=1)

http://102.10.10.132/api/News?newsId={资讯id} (http://102.10.10.132/api/News?

newsId=%7B%E8%B5%84%E8%AE%AFid%7D)



```
@GET("News")
Call<NewsBean> getItem(@Query("newsId") String newsId);
```

或

http://102.10.10.132/api/News?newsId=1&type=类型1 (http://102.10.10.132/api/News?newsId=1&type=%E7%B1%BB%E5%9E%8B1)
 http://102.10.10.132/api/News?newsId={资讯id}&type={类型}
 (http://102.10.10.132/api/News?newsId=%7B%E8%B5%84%E8%AE%AFid%7D&type=%7B%E7%B1%BB%E5%9E%8B%7D)

```
@GET("News")
Call<NewsBean> getItem(@Query("newsId") String newsId, @Query("type") String type);
```

样式4（多个参数在URL问号之后，且个数不确定）

http://102.10.10.132/api/News?newsId=1&type=类型1 (http://102.10.10.132/api/News?newsId=1&type=%E7%B1%BB%E5%9E%8B1)...
 http://102.10.10.132/api/News?newsId={资讯id}&type={类型}
 (http://102.10.10.132/api/News?newsId=%7B%E8%B5%84%E8%AE%AFid%7D&type=%7B%E7%B1%BB%E5%9E%8B%7D)...

```
@GET("News")
Call<NewsBean> getItem(@QueryMap Map<String, String> map);
```

也可以

```
@GET("News")
Call<NewsBean> getItem(
    @Query("newsId") String newsId,
    @QueryMap Map<String, String> map);
```

POST

样式1（需要补全URL，post的数据只有一条reason）

http://102.10.10.132/api/Comments/1 (http://102.10.10.132/api/Comments/1)
 http://102.10.10.132/api/Comments/{newsId}
 (http://102.10.10.132/api/Comments/%7BnewsId%7D)

```
@FormUrlEncoded
@POST("Comments/{newsId}")
Call<Comment> reportComment(
    @Path("newsId") String commentId,
    @Field("reason") String reason);
```

样式2（需要补全URL，问号后加入access_token，post的数据只有一条reason）

http://102.10.10.132/api/Comments/1?access_token=1234123
 (http://102.10.10.132/api/Comments/1?access_token=1234123)
 http://102.10.10.132/api/Comments/{newsId}?access_token={access_token}
 (http://102.10.10.132/api/Comments/%7BnewsId%7D?access_token=%7Baccess_token%7D)



```
@FormUrlEncoded
@POST("Comments/{newsId}")
Call<Comment> reportComment(
    @Path("newsId") String commentId,
    @Query("access_token") String access_token,
    @Field("reason") String reason);
```

样式3（需要补全URL，问号后加入access_token，post一个body（对象））

http://102.10.10.132/api/Comments/1?access_token=1234123
 (http://102.10.10.132/api/Comments/1?access_token=1234123)
 http://102.10.10.132/api/Comments/{newsId}?access_token={access_token}
 (http://102.10.10.132/api/Comments/%7BnewsId%7D?
 access_token=%7Baccess_token%7D)

```
@POST("Comments/{newsId}")
Call<Comment> reportComment(
    @Path("newsId") String commentId,
    @Query("access_token") String access_token,
    @Body CommentBean bean);
```

DELETE

样式1（需要补全URL）

http://102.10.10.132/api/Comments/1 (http://102.10.10.132/api/Comments/1)
 http://102.10.10.132/api/Comments/{newsId}
 (http://102.10.10.132/api/Comments/%7BnewsId%7D)
 {access_token}

```
@DELETE("Comments/{commentId}")
Call<ResponseBody> deleteNewsCommentFromAccount(
    @Path("commentId") String commentId);
```

样式2（需要补全URL，问号后加入access_token）

http://102.10.10.132/api/Comments/1?access_token=1234123
 (http://102.10.10.132/api/Comments/1?access_token=1234123)
 http://102.10.10.132/api/Comments/{newsId}?access_token={access_token}
 (http://102.10.10.132/api/Comments/%7BnewsId%7D?
 access_token=%7Baccess_token%7D)

```
@DELETE("Comments/{commentId}")
Call<ResponseBody> deleteNewsCommentFromAccount(
    @Path("accountId") String accountId,
    @Query("access_token") String access_token);
```

PUT（这个请求很少用到，例子就写一个）

http://102.10.10.132/api/Accounts/1 (http://102.10.10.132/api/Accounts/1)
 http://102.10.10.132/api/Accounts/{accountId}
 (http://102.10.10.132/api/Accounts/%7BaccountId%7D)

```
@PUT("Accounts/{accountId}")
Call<ExtrasBean> updateExtras(
    @Path("accountId") String accountId,
    @Query("access_token") String access_token,
    @Body ExtrasBean bean);
```

总结



@Path: 所有在网址中的参数 (URL的问号前面) , 如:
http://102.10.10.132/api/Accounts/{accountId}
(http://102.10.10.132/api/Accounts/%7BaccountId%7D)
@Query: URL问号后面的参数, 如:
http://102.10.10.132/api/Comments?access_token={access_token}
(http://102.10.10.132/api/Comments?access_token=%7Baccess_token%7D)
@QueryMap: 相当于多个@Query
@Field: 用于POST请求, 提交单个数据
@Body: 相当于多个@Field, 以对象的形式提交

Tips

- Tip1
使用@Field时记得添加@FormUrlEncoded
- Tip2
若需要重新定义接口地址, 可以使用@Url, 将地址以参数的形式传入即可。如

```
@GET
Call<List<Activity>> getActivityList(
    @Url String url,
    @QueryMap Map<String, String> map);

Call<List<Activity>> call = service.getActivityList(
    "http://115.159.198.162:3001/api/ActivitySubjects", map);
```



带心情去旅行 (/u/769d3d3a9d4b)

写了 45360 字, 被 995 人关注, 获得了 1673 个喜欢 (/u/769d3d3a9d4b)

+ 关注

努力做一个喜欢专研的人, 拒绝人云亦云


能帮助到大家就好, 打赏什么的就不用了

赞赏支持

喜欢 | 83




(http://cwb.assets.jianshu.io/notes/images/3679774



写下你的评论...

15条评论 只看作者

按喜欢排序 按时间正序 按时间倒序



laizuling (/u/d068385414ad)
6楼 · 2016.07.25 16:10
(/u/d068385414ad)
@Delete (/users/f92a8ccd26f4)("Comments/{commentId}")
Call<ResponseBody> deleteNewsCommentFromAccount(
@path (/users/b62dd8d8f76b)("accountId") String accountId);

这地方的Path注解是不是应该是commentId呢

+

🔖


🔗

👍 1人赞 💬 回复

带心情去旅行 (/u/769d3d3a9d4b): @laizuling (/users/d068385414ad) 感谢指出，已更新


2016.07.26 12:58 💬 回复

✎ 添加新评论

 sir_奶牛 (/u/aa1e65ef05ce)
2楼 · 2016.04.28 13:40

(/u/aa1e65ef05ce)
之前还一直疑惑path和query有什么不同

👍 赞 💬 回复

 shunxir (/u/5491949519f1)
3楼 · 2016.05.10 06:07


(/u/5491949519f1)
太感谢了，正不造什么时候用query 什么时候用path呢☺

👍 赞 💬 回复

带心情去旅行 (/u/769d3d3a9d4b): @shunxir (/users/5491949519f1) 不客气哈，我一开始也是被这个搞得一头雾水。后来知道怎么用后就记录下来了，希望能帮到被这些用法困扰的人


2016.05.10 07:29 💬 回复

✎ 添加新评论

 yangjianan (/u/25f8193d5561)
4楼 · 2016.05.24 08:44


(/u/25f8193d5561)
最近在学Retrofit,看完这篇文章瞬间清晰了。

👍 赞 💬 回复

 戢鳞X (/u/d8d88bdba392)
5楼 · 2016.05.24 09:35

(/u/d8d88bdba392)
z总结的很好，点个赞👍

👍 赞 💬 回复

 许航MyateJx (/u/5d956b6dcf1f)
7楼 · 2017.02.10 15:12


(/u/5d956b6dcf1f)
多谢！正好需要例子来理解

👍 赞 💬 回复

带心情去旅行 (/u/769d3d3a9d4b): @简洁笔记 (/users/5d956b6dcf1f) 不谢，不谢


2017.02.10 17:30 💬 回复

✎ 添加新评论

 安徒生总是欺骗孩子 (/u/d4b5c0788e11)
8楼 · 2017.02.13 17:47

(/u/d4b5c0788e11)
非常感谢博主分享，最近正在学习

👍 赞 💬 回复

 厉害了我的歌哥 (/u/12e36795b1ea)
9楼 · 2017.03.22 22:32

(/u/12e36795b1ea)

+

🔖

🔗

【Android】 Retrofit网络请求Service, @Path、@Query、@Quer... - 简书

@body (/users/a853c1a1d45e): 相当于多个@Field, 以对象的形式提交,那么 @FiledMap 与@Body区别呢?

👍 赞 💬 回复

带心情去旅行 (/u/769d3d3a9d4b): @厉害了我的歌哥 (/users/12e36795b1ea) body可以直接使用一个对象里面的属性, 不需要一个个写

2017.03.22 23:01 💬 回复

厉害了我的歌哥 (/u/12e36795b1ea): @带心情去旅行 (/users/769d3d3a9d4b) 那在不同条件下, 我需要动态的添加字段,那还是要用@FieldMap,对吧

2017.03.23 17:33 💬 回复

带心情去旅行 (/u/769d3d3a9d4b): @厉害了我的歌哥 (/users/12e36795b1ea) 可以用body, 不需要传的, 使用null就可以了

2017.03.23 18:06 💬 回复

✎ 添加新评论



MonkiRayman (/u/7d1356a53fc0)
10楼 · 2017.03.30 11:09
(/u/7d1356a53fc0)
mark

👍 赞 💬 回复

被以下专题收入, 发现更多相似内容

+

我的专题

Android 开发

Android. ..

Android 知识

Android. ..

网络

Android 学习

Android 日记

android..

Android

Retrofit

+

🔖

🔗