## banner

1. 导入架包

|  |
| --- |
| *//轮播图* implementation **'com.youth.banner:banner:1.4.10'**  *//spanner的自定义图片显示必须要有他* implementation **'com.github.bumptech.glide:glide:4.8.0'** implementation **'com.android.support:support-v4:28.0.0'** |

1. 新建一个HomeBanners类，用来获取网络数据，拿到轮播图的图片和文字资源

|  |
| --- |
| **public class** HomeBanners{  } |

1. 实现HomeBanners的功能

注意：在线程里不能改变UI界面，要改的话要通过Handler改

|  |
| --- |
| **public class** HomeBanners{  **private** List **imageList** = **new** ArrayList();*//存放图片*  **private List titleList = new ArrayList();***//存放标题*  **Map<String, List<String>> map = new HashMap<>();***//用来存放这两个list*  //新建展示的方法 show()  public void show(){  //创建一个Handler  final Handler handler=new Handler(){  //重载handleMessage方法  @Override **public void** handleMessage(Message msg) {  **super**.handleMessage(msg);  **if**(msg.**what**==0){  **map**.put(**"image"**, **imageList**);  **map**.put(**"title"**, **titleList**);  Log.*i*(**"zjc"**, **"里面："** + String.*valueOf*(**map**.size()));  **homeView**.showBanners(**map**);  } }  };  final urls=NET.S\_BANNER;//网址  //新建一个线程  **new** Thread(**new** Runnable() {  @Override  **public void** run() {  OkHttpClient okHttpClient = **new** OkHttpClient();  **final** Request request = **new** Request.Builder().url(url).build();*//访问不同的url在这改* Call call = okHttpClient.newCall(request);  call.enqueue(**new** Callback() {  @Override  **public void** onFailure (Call call, IOException e){  Log.*i*(**"zjc"**, **"网络连接错误"**);  }   @Override  **public void** onResponse (Call call, Response response) **throws** IOException {  String responseData = response.body().string();  **try** {  JSONObject jsonObject = **new** JSONObject(responseData);  JSONArray jsonArray = jsonObject.getJSONArray(**"banners"**);  Log.*i*(**"zjc"**, **"里面返回的数据"** + String.*valueOf*(jsonArray.length()));  **for** (**int** i = 0; i < jsonArray.length(); i++) {  JSONObject jsonObject1 = jsonArray.getJSONObject(i);  **imageList**.add(jsonObject1.get(**"img\_url"**));  **titleList**.add(jsonObject1.get(**"title"**));  }  //这用于给Handler发送消息  handler.sendEmptyMessage(0);  */\* handler.post(new Runnable() {  @Override  public void run() {  map.put("image", imageList);  map.put("title", titleList);  Log.i("zjc", "里面：" + String.valueOf(map.size()));  homeView.showBanners(map);  }   });  \*/* } **catch** (JSONException e) {  e.printStackTrace();  }  }  });  } }).start();  }  }  实现方法showBanners  @Override **public void** showBanners(Map<String, List<String>> zlist) {  Log.*i*(**"zjc"**, **"zhu:"**+String.*valueOf*(zlist.get(**"image"**).size()));  **banner**.setBannerStyle(BannerConfig.***CIRCLE\_INDICATOR\_TITLE\_INSIDE***);  *//设置轮播的动画效果，内含多种特效，可点入方法内查找后内逐一体验* **banner**.setBannerAnimation(Transformer.*Default*);  **banner**.setImages(zlist.get(**"image"**));  **banner**.setBannerTitles(zlist.get(**"title"**));  **banner**.setImageLoader(**new** GlideImageLoader());  **banner**.setDelayTime(3000);  **banner**.isAutoPlay(**true**);  **banner**.start(); } **class** GlideImageLoader **extends** ImageLoader {  @Override  **public void** displayImage(Context context, Object path, ImageView imageView) {  *//具体方法内容自己去选择，次方法是为了减少banner过多的依赖第三方包，所以将这个权限开放给使用者去选择* Glide.*with*(context.getApplicationContext())  .load(path)  .into(imageView);  } } |