# Day06 地图配送

**订单详情页**

1.跳转订单详情页

view.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(mContext, OrderDetailActivity.class);  
 intent.addFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);  
 intent.putExtra("orderId", mOrder.getId());  
 intent.putExtra("type", mOrder.getType());  
 mContext.startActivity(intent);  
 }  
});

2.订单状态初始化

private void processIntent() {  
 if (getIntent() != null) {  
 mOrderId = getIntent().getStringExtra("orderId");  
 mType = getIntent().getStringExtra("type");  
  
 int index = getIndex(mType);  
 ((TextView) mLlOrderDetailTypeContainer.getChildAt(index)).setTextColor(Color.BLUE);  
 ((ImageView) mLlOrderDetailTypePointContainer.getChildAt(index)).setImageResource(R.draw able.order\_time\_node\_disabled);  
 }  
}

3.详情页自动刷新（观察者）

OrderObservable.getInstance().addObserver(this);

@Override  
public void update(Observable o, Object data) {  
 HashMap<String, String> map = (HashMap<String, String>) data;  
 String pushOrderId = map.get("orderId");  
 String pushType = map.get("type");

int newIndex = getIndex(pushType);  
  
 if (pushOrderId.equals(mOrderId)) {  
 //匹配上,选中新的状态  
 ((TextView) mLlOrderDetailTypeContainer.getChildAt(newIndex)).setTextColor(Color.BLUE);  
 ((ImageView) mLlOrderDetailTypePointContainer.getChildAt(newIndex)).setImageResource(R.d rawable.order\_time\_node\_disabled);  
  
}

4.极光推送控制台测试

**地图标注**

1. 商家接单后显示地图

@Override  
protected void onCreate(@Nullable Bundle savedInstanceState) {  
 ...  
 mapView.onCreate(savedInstanceState);// 此方法必须重写  
 init();  
}

private void init() {  
 if (aMap == null) {  
 aMap = mapView.getMap();  
 }  
}

@Override

public void update(Observable o, Object data) {

HashMap<String, String> map = (HashMap<String, String>) data;

String pushOrderId = map.get("orderId");

String pushType = map.get("type");

String lat = map.get("lat");

String lng = map.get("lng");

switch (Integer.parseInt(pushType)){

case 30:

//商家已接单

mapView.setVisibility(View.VISIBLE);

aMap.moveCamera(CameraUpdateFactory.zoomTo(16));

2.标注买家与卖家（marker用法）

//标注卖家位置 丰顺自选快餐 22.5788400000,113.9216330000  
MarkerOptions sellermarkerOptions = new MarkerOptions();  
sellermarkerOptions.title("丰顺自选快餐");  
sellermarkerOptions.position(new LatLng(22.5788400000,113.9216330000));  
sellermarkerOptions.icon(BitmapDescriptorFactory.fromResource(R.drawable.order\_seller\_icon));  
mSellerMarker = aMap.addMarker(sellermarkerOptions);  
  
MarkerOptions buymarkerOptions = new MarkerOptions();  
buymarkerOptions.title("黑马程序员");  
buymarkerOptions.position(new LatLng(22.5765800000,113.9237520000));  
ImageView imageView = new ImageView(this);  
imageView.setImageResource(R.drawable.order\_buyer\_icon);  
buymarkerOptions.icon(BitmapDescriptorFactory.fromView(imageView));  
mBuyerMarker = aMap.addMarker(buymarkerOptions);  
  
aMap.moveCamera(CameraUpdateFactory.changeLatLng(new LatLng(22.5788400000,113.9216330000)));

3.标注骑手（infowindow用法）

private void initRider() {  
 //标记骑手  
 MarkerOptions riderOption = new MarkerOptions();  
// riderOption.title("我是百度骑士");  
 LatLng initPoint = new LatLng(22.5766790000,113.9205490000);  
 riderOption.position(initPoint);  
 ImageView imageView = new ImageView(this);  
 imageView.setImageResource(R.drawable.order\_rider\_icon);  
 riderOption.icon(BitmapDescriptorFactory.fromView(imageView));  
 mRiderMarker = aMap.addMarker(riderOption);  
 mRiderMarker.setSnippet("我是百度骑士！");  
 aMap.moveCamera(CameraUpdateFactory.changeLatLng(new LatLng(22.5766790000,113.9205490000)));  
 mRiderMarker.showInfoWindow();  
 }

4.更新骑手位置

private void updateRider(Double lat, Double lng) {  
 //22.5774220000,113.922475000 产业园B  
 mRiderMarker.hideInfoWindow();  
 LatLng newPosition = new LatLng(lat,lng);  
 mRiderMarker.setPosition(newPosition);  
 aMap.moveCamera(CameraUpdateFactory.changeLatLng(newPosition));  
}

5.绘制骑手配送轨迹

ArrayList<LatLng> mPoints = new ArrayList<>();  
 private void initRider() {  
 //标记骑手  
 MarkerOptions riderOption = new MarkerOptions();  
 LatLng initPoint = new LatLng(22.5766790000,113.9205490000);  
 mPoints.add(initPoint);

//画线  
mPoints.add(newPosition);  
PolylineOptions polylineOptions = new PolylineOptions();  
polylineOptions.width(5);  
polylineOptions.color(Color.RED);  
polylineOptions.add(mPoints.get(mPoints.size() -1),mPoints.get(mPoints.size() -2));  
aMap.addPolyline(polylineOptions);

**首页添加广告（整除算法）**

1.增加广告

public static final int TYPE\_DIVISION = 1; //分割线，即广告

@Override  
public int getItemViewType(int position) {  
 //商家数量比较多，进行分组，每组之间展示广告  
 //TODO: 头布局 （附近0-附近9) 广告 （其他0-其他9) 广告 （其他10-其他19) 广告 （其他20-其他24)  
 if (position == 0) {  
 return TYPE\_TITLE;  
 } else if (position >= (1 + mNearby.size()) && (position - 1 - mNearby.size()) % (GROUP\_SIZE + 1) == 0) {  
 return TYPE\_DIVISION;  
 } else {  
 return TYPE\_SELLER;  
 }  
}

2.广告holder

@Override  
public RecyclerView.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {

View itemView;  
switch (viewType) {

case TYPE\_DIVISION:  
 itemView = View.inflate(mContext, R.layout.item\_division, null);  
 DivisionHolder divisionHolder = new DivisionHolder(itemView);  
 return divisionHolder;

@Override  
public void onBindViewHolder(RecyclerView.ViewHolder holder, int position) {  
 int viewType = getItemViewType(position);  
 switch (viewType) {  
 case TYPE\_DIVISION:  
 DivisionHolder divisionHolder = (DivisionHolder) holder;  
 divisionHolder.setData("----------------我是华丽的分割线--------------------");  
 break;

3.重新计算条目数

@Override  
public int getItemCount() {  
 //TODO: 头布局 （附近0-附近9) 广告 （其他0-其他9) 广告 （其他10-其他19) 广告 （其他20-其他29)  
 if (mNearby.size() == 0 && mOther.size() == 0) {  
 //程序异常数据为空，没有网络，用户旅游进入无人区  
 Log.e("home", "百度外卖破产了，程序异常数据为空，没有网络，用户旅游进入无人区");  
 }  
 int count = 1; //头布局展示促销信息  
 if (mNearby.size() > 0) {  
 count += mNearby.size(); //附近商家  
 } else {  
 Log.e("home", "附近商家为空");  
 }  
  
 if (mOther.size() > 0) {  
 count += 1; //第一个广告条  
 count += mOther.size();  
 //计算广告分割线的个数  
 //如果其他商家是25个，2个广告；如果其他商家是30个，显示2个广告；如果31个，3个广告  
 count += mOther.size() / GROUP\_SIZE;  
 //如果刚好整除，反向减去一个  
 if (mOther.size() % GROUP\_SIZE == 0) {  
 count -= 1;  
 }  
 } else {  
 Log.e("home", "其他商家为空");  
 }  
 return count;  
}

4.重新对应附近商家和其他商家

case TYPE\_SELLER:  
 SellerHolder sellerHolder = (SellerHolder) holder;  
 //区分是附近商家还是其他商家，展示第几个商家index  
 //TODO: 头布局 （附近0-附近9) 广告 （其他0-其他9) 广告 （其他10-其他19) 广告 （其他20-其他24)  
 int newPostion;  
 if (position < mNearby.size() + 1) {  
 newPostion = position - 1;  
 sellerHolder.setData(mNearby.get(newPostion));  
 } else {  
 newPostion = position - 1 - mNearby.size() - 1;  
 newPostion -= newPostion / (GROUP\_SIZE + 1);  
 sellerHolder.setData(mOther.get(newPostion));  
 }  
 break;

## 每日作业

1.会使用地图标记

2.了解绘制地图轨迹

3.首页广告条