

司机端接口说明（Android）

DriverRouteManager

1、构造方法

```
/**
 * 司机端管理类
 *
 * @param context
 * @param aMap AMap对象
 * @param options 配置项，用来自定义轨迹线颜色、起终点图标、InfoWindow样式等属性
 */
public DriverRouteManager(Context context, AMap aMap, RouteOverlayOptions options)
```

2、设置订单信息

```
/**
 *
 * @param orderId 订单ID
 * @param startPosition 起点坐标
 * @param pickUpPosition 上车点坐标
 * @param endPosition 终点坐标
 */
public void setOrderProperty(String orderId, LatLng startPosition, LatLng pickUpPosition, LatLng endPosition)
```

3、更新订单状态

```
/**
 * 更新订单状态，详情参见{@link SCTXConfig}
 * 注：如果在构造时没有传入有效的AMap对象，只上传数据，不绘制routeOverlay
 */
public void setOrderState(int orderState)
```

4、设置地图对象

```
/**
 * 设置AMap对象
 * 注：针对在构造时没有传入有效的AMap直接调用start()，如果希望显示routeOverlay，需要调用此方法
 * @param aMap
 */
public void setMap(AMap aMap){}
```

5、更新InfoWindow信息内容

```

/**
 * 更新InfoWindow信息内容
 * @param remainingDistance 剩余距离
 * @param estimatedTime 预计时间
 * @param cost 费用
 */
public void updateInfoWindow(float remainingDistance,long estimatedTime,float cost)

```

6、设置状态监听

```

/**
 * 设置状态监听
 * @param callback
 */
public void setDriverRouteCallback(DriverRouteCallback callback)

```

7、设置上传司机位置的时间间隔

```

/**
 * 设置上传司机位置的时间间隔
 * @param interval 时间间隔 单位 ms
 */
public void setDriverPositionUploadInterval(int interval)

```

8、是否开启日志

```

/**
 * 是否开启日志
 * @param loggerEnable
 */
public void setLoggerEnable(boolean loggerEnable) {
    this.loggerEnable = loggerEnable;
}

```

9、销毁资源

```

/**
 * 销毁资源
 */
public void destroy()

```

10、设置路线边界值

```

/**
 * 设置路线边界值
 * @param left      左边界
 * @param right     右边界
 * @param top       上边界
 * @param bottom    下边界
 */
public void setNavigationLineMargin(int left, int right, int top, int bottom)

```

11、缩放到路径最佳视野

```

/**
 * 缩放到路径最佳视野
 */
public void zoomToSpan()

```

12、设置导航模式（调试时使用）

```

/**
 * 设置导航模式 调试时使用
 * @param naviType  NaviType.GPS   GPS 导航； NaviType.EMULATOR  模拟导航
 */
public void setNaviType(int naviType)

```

13、设置司机位置

```

/**
 * 设置当前司机位置
 *
 * @param driverPosition 司机位置
 */
public void setDriverPosition(LatLng driverPosition)

```

14、是否开启司机端上传实时位置

```

/**
 * 控制司机端是否上传司机位置
 *
 * @param isUploadDriverPosition
 */
public void setDriverPositionUploadEnable(boolean isUploadDriverPosition)

```

RouteOverlayOptions(同乘客端)

新增：

```

/**
 * 设置上传司机位置间隔
 * @param interval
 * @return
 */
public RouteOverlayOptions intervalForUploadDriverPosition(long interval)

```

DriverRouteCallback

司机端状态监听

```
/**
 * @param distance      已行驶距离
 * @param time          已行驶时间
 * @param remainingDistance  剩余距离
 * @param estimatedTime    预计时间
 */
void onRouteStatusChange(float distance, long time, float remainingDistance, long estimatedTime);

/**
 * 到达上车点
 */
void onArrivePickUpPosition();

/**
 * 到达目的地
 */
void onArriveDestination();

/**
 * 算路失败
 */
void onCalculateRouteFailure();

/**
 * 请求回调错误接口
 * @param errorCode 返回错误码
 * @param message   返回错误信息
 */
void onError(int errorCode, String message);
```

SCTXConfig

订单状态值

SCTX_ORDER_STATUS_UNSPECIFIED = 0; //未知状态
SCTX_ORDER_STATUS_PICKUPPASSENGER = 1; //去接乘客
SCTX_ORDER_STATUS_WAITPASSENGER = 2; //等待乘客上车
SCTX_ORDER_STATUS_PASSENGERONBOARD = 3; //乘客以上车
SCTX_ORDER_STATUS_ORDERCOMPLETE = 4; //订单结束
SCTX_ORDER_STATUS_OFFLINE = 5; //离线状态，只走降级策略获取司机位置，内部不做其他请求

错误码

SCTX_ERROR_REQUEST_FAILED = 1000; //请求失败
SCTX_ERROR_DRIVER_CALCULATE_ROUTE_FAILED = 1001 // 算路失败
SCTX_ERROR_QUERY_TRACK_ROUTE_FAILED = 1002 //获取轨迹失败
SCTX_ERROR_QUERY_TRACK_ROUTE_SUCCESS = 1003 //成功获取轨迹，结束降级策略
SCTX_ERROR_PASSENGER_NO_ORDERID = 2001 //订单不存在 SCTX_ERROR_PASSENGER_INVALID_PARAMS = 2002 // 参数有误
SCTX_ERROR_PASSENGER_ORDERSTATUS_NOT_MATCH = 2003 //订单状态不匹配