

使用方式

一、获取订单列表

Api 方法： `getOrderList(params?: PageOptions & Object): Promise<PageContent<OrderData>>`

参数说明：

- params：获取参数， `PageOptions & Object` 类型，选传，默认 `{}`

返回值说明： `Promise<PageContent<OrderData>>` 类型

示例：

```
const data = await orderTarget.getOrderList({ pageNumber: 1, pageSize: 10 });
console.log('订单列表', data.contents);
```

二、获取订单详情

Api 方法： `getOrderDetail(params: GetOrderDetailParams): Promise<OrderData>`

参数说明：

- params：获取参数， `GetOrderDetailParams` 类型，必传，详细类型说明如下

参数名	说明	类型	必须	默认值
<code>orderNo</code>	订单号	<code>string</code>	是	-
<code>syncPayStatus</code>	是否同步订单最新状态	<code>boolean</code>	是	-

返回值说明： `Promise<OrderData>` 类型

示例：

```
const data = await orderTarget.getOrderDetail({ orderNo: '1234567890' });
console.log('订单详情', data);
```

三、创建订单

Api 方法： `createOrder(params: CreateOrderParams): Promise<CommonResult<CreateOrderFailReason, CreateOrderResponse>>`

参数说明：

- params：创建订单参数， `CreateOrderParams` 类型，必传，详细类型说明如下

参数名	说明	类型	必须	默认值
address	物流信息	AddressBasicData	是	-
products	购买商品	CreateOrderProductItem[]	是	-
payProvider	支付方式	PaymentProvider	是	-

返回值说明： `Promise<CommonResult<CreateOrderFailReason, CreateOrderResponse>>` 类型

示例：

```
const res = await orderTarget.createOrder({ address: { name: '张三', mobile: '13800138000', area: '广东省 广州市 天河区', address: '天河路1号' }, products: [{ productId: 1, quantity: 1 }], payProvider: PaymentProvider.Wechat });
if (res.success) {
  console.log('创建订单成功', res.orderNo);
} else {
  console.log('创建订单失败', res.failReason, res.failMessage);
}
```

四、确认收货

Api 方法： `confirmOrder(orderNo: string): Promise<void>`

参数说明：

- orderNo：订单号， `string` 类型，必传

示例：

```
await orderTarget.confirmOrder('1234567890');
```

五、申请退款

Api 方法： `applyOrderRefund(orderNo: string): Promise<CommonResult<OrderRefundFailReason, ApplyOrderRefundResponse>>`

参数说明：

- orderNo: 订单号, `string` 类型, 必传

返回值说明： `Promise<CommonResult<OrderRefundFailReason, ApplyOrderRefundResponse>>` 类型

示例：

```
const res = await orderTarget.applyOrderRefund('1234567890');
if (res.success) {
  console.log('退款成功', res);
} else {
  console.log('退款失败', res.failReason, res.failMessage);
}
```