刊登商品的 容器结构说明，

用于调用 wish api 的刊登商品封装方法

刊登的商品列表为一个字典结构

Dictionary<string, Nop.Admin.Models.WishMerDevPublish.WishMerchandisePublishContainer>

其中 字典的 key 是店铺 的 clientId,

字典的 Value 是一个 发布商品项

public class WishMerchandisePublishContainer

{

/// <summary>

/// wish 店铺 数据

/// </summary>

public Nop.Core.Domain.WishStores.WishStore WishStoreData { get; set; }

/// <summary>

/// wish 产品 (要刊登的数据)

/// </summary>

public Wl.Wish.Entities.Request.Product WishProductData { get; set; }

/// <summary>

/// 操作标志位回馈

/// </summary>

public bool RetFlag { get; set; }

/// <summary>

/// 操作消息(成功无消息，其他为异常或是出错消息)

/// </summary>

public string Message { get; set; }

/// <summary>

/// 成功刊登后从wish 来的商品id 数据

/// </summary>

public string ProductId { get; set; }

}

刊登商品调用

Nop.Admin.Helpers.WishApiWrapperForController 类中的 ProductCreateWrapper() 方法

调用 此方法前，先创建 Nop.Admin.Helpers.WishApiWrapperForController 类，

然后传入 WishStores 店铺数据，和 安排好 ProductQueueReadyToPub 数据

注意 店铺数据中的token 的有效性，各商品的刊登结果会记录到 ProductQueueReadyToPub 的相应的字段中…