

Класс оборудования Система лояльности (Loyalty system)

Общая информация

Система лояльности — система, включающая комплекс маркетинговых мероприятий для развития повторных продаж существующим клиентам в будущем, продажи им дополнительных товаров и услуг. Типичным примером программы лояльности компании является дисконтная карта, при дальнейших покупках с использованием дисконтной карты могут предоставляться скидки, в том числе по накопительной системе, также могут существовать системы бонусов и подарков. Кроме того, при получении подобных карт, как правило, заполняется анкета, в которой указываются контакты получателя, что даёт организации возможность оповещать покупателя о новых и потенциально интересующих его товарах и услугах.

Алгоритм работы

Драйвер позволяет обмениваться данными с удаленным сервером системы лояльности. Драйвер передает серверу данные о составе заказа. В ответ сервер возвращает данные о предполагаемых скидках и бонусах для данного заказа, вычисленных по алгоритму программ лояльности, заданных на сервере. Эти функции реализуются с помощью метода *CalculateOrder*. Решение о применении скидок возлагается на хост-приложение. При закрытии заказа, драйвер передает серверу данные о примененных скидках из числа предложенных. Это реализуется методом *CloseOrder*. Кроме того, существует возможность отменить заказ (*CancelOrder*) и получить данные клиента по его номер телефона, номеру карты, разовому платежному коду (*GetCustomerData*). Метод *GetCustomerData* может быть вызван перед созданием заказа для того, чтобы определить зарегистрирован ли данный клиент в БД системы лояльности. Зарегистрировать нового клиента (если система позволяет это сделать при помощи API) позволяет метод *RegisterCustomer*. Помимо основных методов, существует ряд вспомогательных необязательных методов, реализация которых зависит от API сервера. Это получение количества доступных товаров *GetAvailableProduct*, запрос информации о доступности применения промокода *VerifyPromocode*, экспорт дерева номенклатуры *ExportGoods*, использующийся старте интеграции и для синхронизации данных при последующих изменениях и верификация клиента с помощью SMS кода *VerifyClient*.

Общие методы

Общие методы реализованы в соответствии с требованиями к разработке драйверов подключаемого оборудования 1С.

Входные и выходнык параметры общих и специфических представляют значения простых типов string, integer, double, boolean. Любой массив - это всегда XML-структура. Некоторые строковые параметры тоже могут быть XML-структурой. Тогда это явно будет указано в описании параметров.

ПолучитьОписание(GetDescription)

Возвращает информацию о драйвере

Синтаксис

```
ПолучитьОписание(DriverDescription: string): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DriverDescription	string	false	true	Структура описания драйвера (XML структура)

Описание XML-структуры DriverDescription

Параметр	Тип	Описание
Name	string	Наименование драйвера
Description	string	Назначение драйвера (Произвольный текст описывающий назначение драйвера)
EquipmentType	string	Строка, определяющая тип оборудования. Возвращаемое значение "СистемаЛояльности"
IntegrationComponent	boolean	Признак интеграционного компонента
MainDriverInstalled	boolean	Для интеграционного компонента возвращает флаг установки основной поставки драйвера
DriverVersion	string	Возвращает номер версии установленной основной поставки драйвера
IntegrationComponentVersion	string	Возвращает номер версии интеграционного компонента

Параметр	Тип	Описание
DownloadURL	string	Возвращает пустую строку или адрес страницы сайта производителя, по которому доступна ссылка для скачивания основной поставки драйвера или иная информация о драйвере
LogIsEnabled	boolean	Возвращает флаг включения лога драйвера
LogPath	string	Возвращает полный путь к файлу лога драйвера

Пример выходного параметра DriverDescription

```
<?xml version="1.0" encoding="UTF-8"?>
  <DriverDescription
    Name="Драйвер Электронных замков"
    Description="Драйвер Электронных замков"
    EquipmentType="ЭлектронныеЗамки"
    IntegrationComponent="false"
    MainDriverInstalled="false"
    DriverVersion="1.1"
    IntegrationComponentVersion="1.1"
    DownloadURL="http://files.1c.ru/file.ZIP"
    LogIsEnabled="true"
    LogPath="C:\Users\drivers\"/>
```

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

Описание остальных общих методов приведено в 1С-документации.

Специфические методы

ПолучитьДанныеКлиента(GetCustomerData)

Получение данных о клиенте по идентификатору (по номеру карты, номеру телефона и т.п.).

Синтаксис

```
ПолучитьДанныеКлиента(DeviceId: string, CustomerId: string, CustomerType: integer, SecurityFlag: integer,
  FullName: string, Phone: string, Email: string, Balance: double, Card: string, Id: string, Gender: integer,
  BirthDate: string, ReferCode: string, ConsentSms: boolean, ConsentAdv: boolean, Comment: string): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
CustomerId	string	true	false	Идентификатор клиента
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
SecurityFlag	integer	true	false	Флаг включения верификации клиента. 0 – не включать; 1- включать. Необязательный
FullName	string	false	true	Полное имя клиента. Необязательный
Phone	string	false	true	Номер телефона. Необязательный
Email	string	false	true	Адрес электронной почты. Необязательный
Balance	double	false	true	Остаток бонусов. Необязательный
Card	string	false	true	Номер карты клиента. Необязательный
Id	string	false	true	Идентификатор клиента. Необязательный
Gender	integer	false	true	Пол: 0 – мужской; 1 – женский. Необязательный
BirthDate	string	false	true	Дата рождения. Дата/время в формате ISO 8601. Необязательный

Параметр	Тип	Входной	Выходной	Описание
ReferCode	string	false	true	Реферальный код. Необязательный
ConsentSms	boolean	false	true	Согласие на рассылку СМС. Необязательный
ConsentAdv	boolean	false	true	Согласие на рассылку рекламы. Необязательный
Comment	string	false	true	Комментарий. Необязательный

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ЗарегистрироватьКлиента(RegisterCustomer)

Создание нового клиента или изменение существующего в БД ПО системы лояльности. Необязательный метод, поскольку не все системы предоставляют такую возможность.

Синтаксис

```
ЗарегистрироватьКлиента(DeviceId: string, CustomerId: string, CustomerIdType: integer, FullName: string,
    Phone: string, Email: string, Gender: integer, BirthDate: string, ModificationMethod: integer,
    CardNumber: string, ReferCode: string, ConsentSms: boolean, ConsentAdv: boolean, Comment: string): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
CustomerId	string	true	false	Идентификатор клиента. При изменении клиента указывается существующий CustomerId и CustomerIdType
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
FullName	string	true	false	Полное имя клиента
Phone	string	true	false	Номер телефона
Email	string	true	false	Адрес электронной почты
Gender	integer	true	false	Пол: 0 – мужской; 1 – женский. Необязательный
BirthDate	string	true	false	Дата рождения. Дата/время в формате ISO 8601. Необязательный
ModificationMethod	integer	true	false	0 – добавление; 1 – изменение. Необязательный
CardNumber	string	true	false	Номер карты
ReferCode	string	true	false	Реферальный код. Необязательный
ConsentSms	boolean	true	false	Согласие на рассылку СМС. Необязательный
ConsentAdv	boolean	true	false	Согласие на рассылку рекламы. Необязательный
Comment	string	true	false	Комментарий. Необязательный

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ПолучитьДоступныеТовары(GetAvailableProduct)

Получение количества доступных товаров. Необязательный метод.

Синтаксис

```
ПолучитьДоступныеТовары(DeviceID: string, Items: string, CashRegId: string, ItemCode: string,
    ItemQuantityNeed: double, ItemQuantityAvailable: double): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
Items	string	true	false	Необходимое количество товара)
CashRegId	string	true	false	Идентификатор кассы. Необязательный
ItemCode	string	false	true	Код номенклатуры товара
ItemQuantityNeed	double	false	true	Необходимое количество товара
ItemQuantityAvailable	double	false	true	Доступное количество товара

Массив Items

Параметр	Тип	Описание
Code	string	Код номенклатуры товара
Quantity	string	Необходимое количество товара

Пример входного параметра Items

```
<?xml version="1.0" encoding="UTF-8"?>
  <Items>
    <Item Code="1" Quantity="1" />
    <Item Code="2" Quantity="3" />
  </Items>
```

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ВерифицироватьКлиента(VerifyClient)

Верификация клиента с помощью SMS кода. Необязательный метод.

Синтаксис

```
ВерифицироватьКлиента(DeviceID: string, CustomerId: string, CustomerType: integer, Code: string,
OrderSum: double, BonusPaymentSum: double, ResponseCode: string): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
CustomerId	string	true	false	Идентификатор клиента
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
Code	string	true	false	Код подтверждения. В случае запроса передавать пустую строку
OrderSum	double	true	false	Сумма покупки
BonusPaymentSum	double	true	false	Количество бонусов для списания
ResponseCode	string	false	true	Код подтверждения, полученный по СМС или иным способом

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ПроверитьПромокод(VerifyPromocode)

Запрос информации о доступности применения промокода. Необязательный метод.

Синтаксис

ПроверитьПромокод(DeviceId: string, Order: string, CustomerId: string, CustomerType: integer, ItemLimits: string, BonusPaymentSum: double, CashRegId: string, Promocode: string): boolean

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
Order	string	true	false	Данные заказа (XML структура)
CustomerId	string	true	false	Идентификатор клиента
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
ItemLimits	string	true	false	Ограничения скидок для номенклатуры. Необязательный. (XML структура)
BonusPaymentSum	double	true	false	Сумма списания бонусов. Необязательный
CashRegId	string	true	false	Идентификатор кассы. Необязательный
Promocode	string	false	true	Промокод

Order (XML-структура)

Параметр	Тип	Описание
Id	string	Идентификатор заказа. Необязательный
Items	string	Состав заказа. Массив строк
Sum	double	Сумма заказа
SumAfterDiscount	double	Сумма заказа с учетом скидок
CashierName	string	Имя кассира
OrderNumber	string	Номер заказа
CashierId	string	Id кассира

Массив Items

Параметр	Тип	Описание
Name	string	Наименование номенклатуры
Code	string	Код номенклатуры
Quantity	double	Количество номенклатуры
Sum	double	Сумма
SumAfterDiscount	double	Сумма с учетом скидки
ProductGroups	string	Родительские группы номенклатуры (через ";")

Пример входного параметра Order

```
<?xml version="1.0" encoding="UTF-8"?>
  <Order Id="1" Sum="489.99" SumAfterDiscount="379.99" CashierName="Иванов" OrderNumber="125" CashierId="12345356">
    <Items>
      <Item Code="1" Name="Вода" Quantity="1" Sum="89.99" SumAfterDiscount="79.99" ProductGroups="Грунна1;Грунна2"/>
      <Item Code="2" Name="Бупреп" Quantity="2" Sum="400.00" SumAfterDiscount="300.00" ProductGroups="Грунна3;Грунна4"/>
    </Items>
  </Order>
```

Массив ItemLimits

Параметр	Тип	Описание
Code	string	Код номенклатуры товара
MaxDiscount	double	Максимальная абсолютная (суммовая) скидка
MaxBonus	double	Максимальная сумма оплаты бонусами

Пример входного параметра ItemLimits

```
<?xml version="1.0" encoding="UTF-8"?>
  <ItemLimits>
    <Item Code="1" MaxDiscount="15.59" MaxBonus="10"/>
    <Item Code="2" MaxDiscount="16.59" MaxBonus="20"/>
  </Items>
```

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ЭкспортироватьНоменклатуру(ExportGoods)

Экспорт дерева номенклатуры. Выгружает дерево товаров и их групп с текущей ценой на момент выгрузки в систему лояльности. Используется на старте интеграции и для синхронизации данных при последующих изменениях. Необязательный метод. Данные о товарах позволяют серверу лояльности рассчитывать расширенные скидки и акции

Синтаксис

```
ЭкспортироватьНоменклатуру(DeviceID: string, Items: string): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
Items	string	true	false	Дерево товаров и их групп(XML структура)

Массив Items

Параметр	Тип	Описание
Code	string	Код товара или группы
ParentCode	string	Код родителя товара или группы
Name	string	Наименование
Price	double	Цена продажи (только для товара)
IsGroup	boolean	Признак группы

Пример входного параметра Items

```
<?xml version="1.0" encoding="UTF-8"?>
  <Items>
    <Item Code="1" ParentCode="0" Name="Вода" IsGroup="true"/>
    <Item Code="2" ParentCode="1" Name="Фанта" Price="90.99" IsGroup="false"/>
  </Items>
```

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

РасчитатьЗаказ(CalculateOrder)

Запрос информации о скидках и количестве бонусных баллов, доступных для заказа. Метод может быть вызван у драйвера произвольное количество раз, по мере изменения заказа. ВАЖНО! Функция CalculateOrder не создает каких-либо новых документов на стороне сервера.

Синтаксис

РассчитатьЗаказ(DeviceID: string, Order: string, CustomerId: string, CustomerType: integer, ItemLimits: string, BonusPaymentSum: double, CashRegId: string, Promocode: string, ShopId: string, CashRegName: string, ResultData: string): boolean

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
Order	string	true	false	Данные заказа(XML структура)
CustomerId	string	true	false	Идентификатор клиента
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
ItemLimits	string	true	false	Ограничения скидок для номенклатуры. Необязательный
BonusPaymentSum	double	true	false	Сумма списания бонусов. Необязательный
CashRegId	string	true	false	Идентификатор кассы. Необязательный
Promocode	string	true	false	Промокод. Необязательный
ShopId	string	true	false	Идентификатор организации. Необязательный
CashRegName	string	true	false	Наименование кассы. Необязательный
ResultData	string	false	true	XML структура

Массив ItemLimits

Параметр	Тип	Описание
Code	string	Код номенклатуры
MaxDiscount	double	Максимальная абсолютная скидка
MaxBonus	double	Максимальная сумма оплаты бонусами

Пример входного параметра ItemLimits

```
<?xml version="1.0" encoding="UTF-8"?>
  <ItemLimits>
    <Item Code="1" MaxDiscount="15.59" MaxBonus="10"/>
    <Item Code="2" MaxDiscount="16.59" MaxBonus="20"/>
  </ItemLimits>
```

Order (XML-структура)

Параметр	Тип	Описание
Id	string	Идентификатор заказа. Необязательный
Items	string	Состав заказа
Sum	double	Сумма заказа
SumAfterDiscount	double	Сумма заказа с учетом скидок
CashierName	string	Имя кассира
OrderNumber	string	Номер заказа
CashierId	string	Id кассира

Массив Items

Параметр	Тип	Описание
----------	-----	----------

Параметр	Тип	Описание
Name	string	Наименование номенклатуры
Code	string	Код номенклатуры
Quantity	double	Количество номенклатуры
Sum	double	Сумма
SumAfterDiscount	double	Сумма с учетом скидки
ProductGroups	string	Родительские группы номенклатуры (через ";")

Пример входного параметра Order

```
<?xml version="1.0" encoding="UTF-8"?>
  <Order Id="1" Sum="489.99" SumAfterDiscount="379.99" CashierName="Иванов" OrderNumber="125" CashierId="12345356">
    <Items>
      <Item Code="1" Name="Вода" Quantity="1" Sum="89.99" SumAfterDiscount="79.99" ProductGroups="Группа1;Группа2"/>
      <Item Code="2" Name="Бургер" Quantity="2" Sum="400.00" SumAfterDiscount="300.00" ProductGroups="Группа3;Группа4"/>
    </Items>
  </Order>
```

ResultData (XML-структура)

Параметр	Тип	Описание
OrderId	string	Идентификатор заказа. Необязательный
BonusPaymentLimit	double	Максимальная сумма оплаты бонусами
LoyaltyResults	string	Предлагаемые скидки. Необязательный
TotalBonus	double	Сумма накопленных бонусов
BillCheque	string	Текст для печати в пречеке. Необязательный
DiscountSum	double	Сумма скидки по заказу

Массив LoyaltyResults

Параметр	Тип	Описание
ProgramId	string	Идентификатор программы лояльности. Необязательный
ProgramName	string	Наименование программы лояльности. Необязательный
Discounts	string	Скидки на товары)

Массив Discounts

Параметр	Тип	Описание
Id	string	Идентификатор скидки
ItemCode	string	Код товара
DiscountSum	double	Сумма скидки
Position	integer	Индекс позиции, соответствующий массиву Items (начало с 0)
BonusPaymentLimit	double	Максимальная сумма оплаты бонусами

Пример выходного параметра ResultData

```
<?xml version="1.0" encoding="UTF-8"?>
  <ResultData OrderId="1" BonusPaymentLimit="130" TotalBonus="100" DiscountSum="30">
    <LoyaltyResults>
      <LoyaltyItem ProgramId="1" ProgramName="Программа дисконт">
        <Discounts>
          <DiscountItem Id="1" ItemCode="1" DiscountSum="10" Position="0" BonusPaymentLimit="60"/>
          <DiscountItem Id="2" ItemCode="2" DiscountSum="20" Position="1" BonusPaymentLimit="70"/>
        </Discounts>
      </LoyaltyItem>
    </LoyaltyResults>
  </ResultData>
```



```
        </LoyaltyItem>
    </LoyaltyResults>
    <BillCheque>
        <TextString Text="Строка 1"/>
        <TextString Text="Строка 2"/>
    </BillCheque>
</ResultData>
```

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ЗакретьЗаказ(CloseOrder)

Создание и закрытие заказа. Перечень операций, выполняемых драйвером: отправка на сервер примененных скидок, оплата бонусами, отправка суммы для начисления бонусов, закрытие заказа, получение текста подвала чека.

Синтаксис

```
ЗакретьЗаказ(DeviceID: string, OrderId: string, Items: string, BonusPaymentSum: double, SumForBonus: double,
    CustomerId: string, CustomerType: integer, OrderType: integer, PurchaseOrderId: string,
    PurchaseOrderDate: string, InputTransactionId: string, OrderNumber: string, CashRegId: string,
    CashierName: string, CashierId: string, Promocode: string, AdditionalData: string, CashRegName: string,
    ResultData: string): boolean
```

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
Deviceld	string	true	false	Идентификатор подключенного экземпляра устройства
OrderId	string	true	false	Идентификатор заказа
Items	string	true	false	Состав заказа и данные о скидках
BonusPaymentSum	double	true	false	Сумма оплаты бонусами
SumForBonus	double	true	false	Сумма для начисления бонусов
CustomerId	string	true	false	Идентификатор клиента
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
OrderType	integer	true	false	Тип заказа. 0 – заказ; 1 – возврат.
PurchaseOrderId	string	true	false	Идентификатор заказа, по которому идет возврат. Обязателен при возврате.
PurchaseOrderDate	string	true	false	Дата заказа, по которому идет возврат. Обязателен при возврате. Дата/время в формате ISO 8601
InputTransactionId	string	true	false	Идентификатор транзакции. Может использоваться как идентификатор чека возврата.
OrderNumber	string	true	false	Номер заказа
CashRegId	string	true	false	Идентификатор кассы. Необязательный
CashierName	string	true	false	Имя кассира
CashierId	string	true	false	Id кассира
Promocode	string	true	false	Промокод
AdditionalData	string	true	false	Именованные дополнительные параметры. Необязательный
CashRegName	string	true	false	Наименование кассы. Необязательный
ResultData	string	false	true	XML структура

Массив Items

Параметр	Тип	Описание
----------	-----	----------

Параметр	Тип	Описание
Id	string	Идентификатор скидки (как в LoyaltyResults)
ProgramId	string	Идентификатор программы лояльности (как в LoyaltyResults)
ItemCode	string	Код товара
ItemName	string	Наименование товара
DiscountSum	double	Сумма скидки
ItemQuantity	double	Количество товара
ItemSum	double	Сумма товара
ItemPrice	double	Цена товара
BonusPayment	double	Оплачено бонусами
MarkingCode	string	Код маркировки
ProductGroups	string	Родительские группы номенклатуры (через ";")

Пример входного параметра Items

```
<?xml version="1.0" encoding="UTF-8"?>
  <Items>
    <Item Id="1" ProgramId="1" ItemCode="111" ItemName="Фанта" DiscountSum="15.00" ItemQuantity="3" ItemSum="270.00"
      ItemPrice="90.00" BonusPayment="10" ProductGroups="Грунна1;Грунна2"
      MarkingCode="010123456789012321rt-MgV9IVz?Fo*91005892y05ioXkPrUCd4Us1RMRH6OL07soBmz4+V1LKt2JbiKiCIfxbWQRN4fQNQbmAJAh7XanRLpuiN4mZD5/B
    </Items>
```

Массив AdditionalData

Параметр	Тип	Описание
Name	string	Имя параметра
Value	string/integer/double/boolean	Значение параметра

Пример входного параметра AdditionalData

```
<?xml version="1.0" encoding="UTF-8"?>
  <AdditionalData>
    <Item Name="Name1" Value="Value1"/>
    <Item Name="Name2" Value="Value2"/>
  </AdditionalData>
```

ResultData (XML-структура)

Параметр	Тип	Описание
TotalBonus	double	Сумма накопленных бонусов после закрытия заказа
Cheque	string	Текст для печати в чеке. Необязательный
OrderDate	string	Дата создания заказа
OutputTransactionId	string	Идентификатор транзакции

Пример выходного параметра ResultData

```
<?xml version="1.0" encoding="UTF-8"?>
  <ResultData TotalBonus="19" OrderDate="2021-09-02T12:59:00" OutputTransactionId="B61C0751-8F75-4B72-B9B8-CEBDAD632B70">
    <Cheque>
      <TextString Text="Строка 1"/>
      <TextString Text="Строка 2"/>
    </Cheque>
  </ResultData>
```

Возвращаемое значение

True – метод выполнен успешно. False – в противном случае.

ОтменитьЗаказ(CancelOrder)

Полная отмена ранее созданного заказа.

Синтаксис

ОтменитьЗаказ(DeviceID: string, OrderId: string, OrderDate: string, CustomerId: string, CustomerType: integer, TransactionId: string, CashRegId: string, CancelSum: double, OrderSum: double, AdditionalData: string, ResultData: string): boolean

Описание параметров

Параметр	Тип	Входной	Выходной	Описание
DeviceId	string	true	false	Идентификатор подключенного экземпляра устройства
OrderId	string	true	false	Идентификатор заказа, полученный в результате выполнения метода CalculateOrder
OrderDate	string	true	false	Дата заказа. Дата/время в формате ISO 8601. Необязательный
CustomerId	string	true	false	Идентификатор клиента
CustomerType	integer	true	false	Тип идентификатора клиента: 0 – номер телефона(ЦЦЦЦЦЦЦЦЦЦЦЦ); 1 – номер карты; 2 – данные дорожки карты; 3 – идентификатор; 4 – платежный код. Система может игнорировать данный параметр
TransactionId	string	true	false	Идентификатор транзакции
CashRegId	string	true	false	Идентификатор кассы. Необязательный
CancelSum	double	true	false	Сумма заказа с вычетом бонусов. Необязательный
OrderSum	double	true	false	Полная сумма заказа
AdditionalData	string	true	false	Массив именованных дополнительных параметров. Необязательный
ResultData	string	false	true	XML структура

Массив AdditionalData

Параметр	Тип	Описание
Name	string	Имя параметра
Value	string/integer/double/boolean	Значение параметра

Пример входного параметра AdditionalData

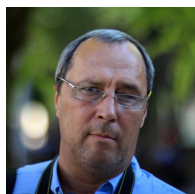
```
<?xml version="1.0" encoding="UTF-8"?>
<AdditionalData>
  <Item Name="Name1" Value="Value1"/>
  <Item Name="Name2" Value="Value2"/>
</AdditionalData>
```

ResultData (XML-структура)

Параметр	Тип	Описание
Cheque	string	Текст для печати в чеке. Необязательный

Пример выходного параметра ResultData

```
<?xml version="1.0" encoding="UTF-8"?>
<ResultData>
  <Cheque>
    <TextString Text="Строка 1"/>
    <TextString Text="Строка 2"/>
  </Cheque>
</ResultData>
```

Стовпец Игорь Александрович

Мужчина, 56 лет, родился 10 октября 1967

+90 (538) 0839471

igor@stovpets.com — предпочитаемый способ связи

Проживает: Москва

Гражданство: Россия, есть разрешение на работу: Россия

Не готов к переезду, не готов к командировкам

Желаемая должность и зарплата

Разработчик Delphi, Android, Qt/C++, Golang (удаленно)

250 000

₽

Специализации:

— Программист, разработчик

Занятость: полная занятость

График работы: полный день

Желательное время в пути до работы: не имеет значения

Опыт работы — 22 года 10 месяцев

Февраль 2001 —
настоящее время
22 года 10
месяцев

1С-Парус

www.rarus.ru

Информационные технологии, системная интеграция, интернет

- Разработка программного обеспечения
- Системная интеграция, автоматизации технологических и бизнес-процессов предприятия, ИТ-консалтинг

Ведущий инженер-программист

Полный цикл разработки программных продуктов - от постановки задачи до разработки, написания документации и поддержки

Образование

Уровень

Среднее образование

Ключевые навыки

Знание языков

Русский — Родной
Английский — A2 — Элементарный
Турецкий — B1 — Средний
Украинский — C1 — Продвинутый

Навыки

Обучаемость Embarcadero Delphi Java Kotlin JSON REST
PostgreSQL CSS Git MS SQL Server ООП Transact-SQL Android SDK
SQLite REST API MVVM Retrofit RxJava SQL
Язык программирования 1С Базы данных Разработка ПО
Разработка мобильных приложений Умение разбираться в чужом коде
Программирование Golang Linux Qt/C++

Дополнительная информация

Обо мне

Знание языка программирования Object Pascal и среды разработки Delphi
Знание языка программирования Golang и среды разработки VSCode
Знание принципов объектно-ориентированного программирования (ООП);
Знание принципов построения сетевого взаимодействия (REST, SOAP HTTP, TCP, UDP);
Знание языка запросов базы данных SQL (Postgres, MS SQL, SQLite);
Опыт разработки ПО для работы с оборудованием, взаимодействующим по RS232, USB, Ethernet
Знание языка программирования Kotlin и Java и среды разработки Android Studio
Знание языка программирования 1C и среды разработки 1C:Предприятие версий 7.7, 8.x
Знание MSBuild скрипт, опыт использования для сборки проектов
Знание языка разметки HTML, CSS, JavaScript, PHP
Опыт использования системы контроля версий Git, продукта GitLab
Опыт разработки и сопровождения проектов для ОС Windows и Linux с нуля