

# Подключение решений

Используется для подключения следующих решений:

- Подмена цен
- Произвольный заказ
- Замена размера товаров прямо в корзине

```
<?php

defined('HOSTCMS') || exit('HostCMS: access denied.');
```

```
/**
 *
 * Версия для HostCMS v.6x
 * @author KAD Systems
 *
 * Копирование и использование файлов модуля
 * в коммерческих целях ЗАПРЕЩЕНО
 *
 */

class Kad_Shop_Cart
{

    /**
     * The singleton instances.
     * @var mixed
     */
    static public $instance = NULL;

    /**
     * Register an existing instance as a singleton.
     * @return object
     */
    static public function instance()
    {
        if (is_null(self::$instance))
        {
            self::$instance = new self();
        }

        return self::$instance;
    }

    public function __construct()
    {
```

```

    }

    /*
    * Решение по смене модификации в зависимости от доп. свойства
    * Пример использования - смена размера товара прямо в корзине
    * Добавляет в xml корзины информацию о списке доступных значений доп. свойств у
    модификаций
    * Производит смену модификации в зависимости от выбора
    */
    static function changeSizeSolution($Shop_Cart_Controller_Show =
Shop_Cart_Controller_Show, $property_id)
    {
        $oShop = $Shop_Cart_Controller_Show->getEntity();

        $oChangeSizeController = new Kad_Cart_Changesize();

        // Название select в котором производится выбор значения свойства
        $oChangeSizeController->input_name = 'mod';

        $oChangeSizeController->checkCartItems($oShop);
        $Shop_Cart_Controller_Show->addEntity($oChangeSizeController->getXmlProperty
ValuesCartItems($oShop, $property_id));

        return true;
    }

    /*
    * Решение по добавлению товара в заказ с произвольно заданной ценой
    */
    static function userPriceSolution()
    {
        Kad::attachObserver("Core_Event::attach('Shop_Item_Controller.onAfterCalcula
tePrice', array('Kad_Shop_Cart_Userprice_Priceobserver',
'onAfterCalculatePrice'))");

        $oUserPriceController = Kad_Shop_Cart_Userprice::instance();
        $oUserPriceController->collectPrice(Core_Array::getRequest('add'),
Core_Array::getRequest('price'));

        return true;
    }

    /*
    * Решение по добавлению товара в заказ с произвольно заданной ценой
    */
    static function customOrderSolution($oShop, $sPropertiesPattern = false,
$exit = true)
    {
        //Kad::attachObserver("Core_Event::attach('Shop_Item_Controller.onAfterCalcula
latePrice', array('Kad_Shop_Cart_Userprice_Priceobserver',
'onAfterCalculatePrice'))");
    }

```

```

    if (!Core_Array::getRequest('property'))
    {
        return false;
    }

    $oCustomOrderController = new Kad_Shop_Cart_Customorder($oShop);

    if ($sPropertiesPattern !== false)
    {
        $oCustomOrderController
            ->propertiesPattern($sPropertiesPattern);
    }
    else
    {
        $oCustomOrderController
            ->propertiesPattern(' {property}: {value},');
    }

    $oCustomOrderController
        ->execute($oShop, $exit);

    // $oCustomOrderController->collectPrice(Core_Array::getRequest('add'),
    Core_Array::getRequest('price'));

    return true;
}
}

```