

1С Bitrix

- Добавление OG разместик
- Преобразование pdf в jpeg
- Шпаргалка по работе с highload-инфоблоками
- Обновление доступного кол-ва у товара
- Обновление стоимости товара
- Информация необходимая для регистрации ключа Bitrix
- Определение города
- Создание типов цен и складов
- Создание превьюшки из PDF и запись картинки в PREVIEW_PICTURE на лету по событию создания элемента
- Получить SEO данные элемента и секции Bitrix
- Оптимизация сайта для PageSpeed Insights
- Пример формирования YML фида
- Водяные знаки на фото
- Настройка редиректов .htaccess
- Битрикс как проверить, что письма были отправлены
- Прикреплять файла к письму (Веб-формы)
- База
 - Вывод всех элементов раздела Bitrix
- Вывод разделов и элементов в bitrix:news.list
- 1С-Bitrix: site under construction – изменяем шаблон
- Шпаргалка Bitrix
- Выставить права на папки и файлы Битрикс SSH
- Добавляем пользовательское поле TEXT/HTML в 1С-Битрикс

Добавление OG разместик

og:title и og:description из meta title и meta description

```
global $APPLICATION;

$APPLICATION->AddHeadString('<meta property="og:title" content="' .
$arResult["IPROPERTY_VALUES"]["ELEMENT_META_TITLE"] . "' />');
$APPLICATION->AddHeadString('<meta property="og:description" content="' .
$arResult["IPROPERTY_VALUES"]["ELEMENT_META_DESCRIPTION"] . "' />');
$APPLICATION->AddHeadString('<meta property="og:image" content="' .
$arResult["PREVIEW_PICTURE"]["SRC"] . "' />');
```

Преобразование pdf в jpeg

```
$file = $_SERVER['DOCUMENT_ROOT'].CFile::GetPath($arResult["PROPERTIES"]["_2D_SCHEME"]["VALUE"][0]);  
$image = new Imagick();  
$image->setResolution(300,300);  
$image->setBackgroundColor('white');  
$image->readImage($file);  
$image->setGravity(Imagick::GRAVITY_CENTER);  
$image->setOption('pdf:fit-to-page',true);  
$image->setImageFormat('jpeg');  
$image->setImageCompression(imagick::COMPRESSION_JPEG);  
$image->setImageCompressionQuality(40);  
$image->scaleImage(1200,1200, true);  
$image->mergeImageLayers(Imagick::LAYERMETHOD_FLATTEN);  
$image->writeImage($_SERVER['DOCUMENT_ROOT'].'/include/'.$arResult["NAME"].'.jpeg');  
$pathImage = str_replace("/var/www/u0876248/data/www/new.allfresco.ru/", "/", $image->getImageFilename());
```

Шпаргалка по работе с highload-инфоблоками

Подготовка к работе

Для работы с highload-ами, нужно подключить подключить модуль highloadblock к сущности highload:

```
// подключаем пространство имен класса HighloadBlockTable и даём ему псевдоним HLBT для удобной
работы
use Bitrix\Highloadblock\HighloadBlockTable as HLBT;

// ID highloadblock'a

const EXP_HL_BLOCK_ID = 1;

//подключаем модуль highloadblock
CModule::IncludeModule('highloadblock');

//Функция получения экземпляра класса:
function GetEntityDataClass($iHlBlockId) {
    if (empty($iHlBlockId) || $iHlBlockId < 1)
    {
        return false;
    }
    $hlblock = HLBT::getById($iHlBlockId)->fetch();
    $entity = HLBT::compileEntity($hlblock);
    $entity_data_class = $entity->getDataClass();
    return $entity_data_class;
}
```

Данная функция получает сущность хайлоадов для дальнейшей работы.

Получение названий полей

```
use Bitrix\Highloadblock\HighloadBlockTable as HLBT;
const EXP_HL_BLOCK_ID = 1;
CModule::IncludeModule('highloadblock');
$hblock = HLBT::getById(EXP_HL_BLOCK_ID)->fetch();
$entity = HLBT::compileEntity($hblock);
var_dump($entity->getFields());
```

Получаем массив, где ключи это поля highload-блока, а значения — объекты, соответствующие типу поля (строка — `Bitrix\Main\Entity\StringField`, число — `Bitrix\Main\Entity\IntegerField`). Тип можно получить следующим образом (ID имя поля):

```
var_dump(get_class($entity->getFields()['ID']));
```

Получить элементы highload-инфоблока

```
const MY_HL_BLOCK_ID = 1;

CModule::IncludeModule('highloadblock');

$entity_data_class = GetEntityDataClass(MY_HL_BLOCK_ID);
$rsData = $entity_data_class::getList(array(
    'select' => array('*')
));

while($el = $rsData->fetch()){
    print_r($el);
}
```

Количество элементов highload-инфоблока

```
const EXP_HL_BLOCK_ID = 1;
CModule::IncludeModule('highloadblock');
$entity_data_class = GetEntityDataClass(EXP_HL_BLOCK_ID);
var_dump($entity_data_class::getCount());
```

Добавить новый элемент в highload-инфоблок

```
const EXP_HL_BLOCK_ID = 1;

CModule::IncludeModule('highloadblock');

$entity_data_class = GetEntityDataClass(EXP_HL_BLOCK_ID);

$result = $entity_data_class::add(array(
    'UF_NAME'      => 'Желтый',
    'UF_CODE'      => 'YELLOW',
    'UF_VALUE'     => '#ffff00',
    'UF_ACTIVE'   => '1'
));
```

Удалить новый элемент в highload-инфоблок

```
const EXP_HL_BLOCK_ID = 1;
```

```
CModule::IncludeModule('highloadblock');

$id = 9;
$entity_data_class = GetEntityDataClass(EXP_HL_BLOCK_ID);
$result = $entity_data_class::delete($id);
```

Обновить новый элемент в highload-инфоблок

```
const EXP_HL_BLOCK_ID = 1;

CModule::IncludeModule('highloadblock');

$id = 10;
$entity_data_class = GetEntityDataClass(MY_HL_BLOCK_ID);
$result = $entity_data_class::update($id, array(
    'UF_NAME'      => 'Фиолетовый',
    'UF_CODE'      => 'PURPLE',
    'UF_VALUE'     => '#5A009D',
    'UF_ACTIVE'   => '1'
));
```

В качестве дополнения, рекомендую ознакомиться с официальной документацией [здесь](#) и [здесь](#)

[Источник](#)

Обновление доступного кол-ва у товара

```
$arFields = Array(  
    "PRODUCT_ID" => $product_id[$key],  
    "STORE_ID" => $data['2'],  
    "AMOUNT" => 100  
);  
CCatalogStoreProduct::UpdateFromForm($arFields);
```

Обновление стоимости товара

```
$product_id = [0, 0, 47, 45, 46, 158, 136, 137, 140, 51, 42];

if (($handle = fopen("price.csv", "r")) !== FALSE) {
    $i = 1;

    while (($data = fgetcsv($handle, 1000, ",")) !== FALSE) {
        foreach ($data as $key => $value) {
            if ($key <= 2){ continue; }
            echo '<b>ID_PRICE:</b> '.$data['0'].' || <b>CURRENT:</b> '.$data['1'].' || <b>ID_PRODUCT:</b>
'.$product_id[$key].' || <b>PRICE:</b> '.$value;
            echo '<br>';
            $arFields = Array( "PRODUCT_ID" => $product_id[$key], "CATALOG_GROUP_ID" => $data['0'], "PRICE"
=> $value, "CURRENCY" => $data['1']);
            $res = CPrice::GetList(
                array(),
                array(
                    "PRODUCT_ID" => $product_id[$key],
                    "CATALOG_GROUP_ID" => $data['0'],
                )
            );
            if ($arr = $res->Fetch()) {
                CPrice::Update($arr["ID"], $arFields);
                echo '<b>Цена обновлена</b>';
            } else {
                CPrice::Add($arFields);
                echo '<b>Цена добавлена</b>';
            }
            echo '<br>';
        }
    }

    $i++;
}
```

```
    echo '<br><br><br>';
  }
}
```

```
$product_id = [0, 0, 76, 77, 78, 79, 80, 129, 130, 114, 115];

if (($handle = fopen("price-1.csv", "r")) !== FALSE) {
    $i = 1;

    while (($data = fgetcsv($handle, 1000, ";")) !== FALSE) {
        foreach ($data as $key => $value) {
            if ($key <= 1){ continue; }
            echo '<b>ID_PRICE:</b> '.$data['0'].' || <b>CURRENT:</b> '.$data['1'].' || <b>ID_PRODUCT:</b>
'.$product_id[$key].' || <b>PRICE:</b> '.$value;
            echo '<br>';

            $arFields = Array( "PRODUCT_ID" => $product_id[$key], "CATALOG_GROUP_ID" => $data['0'], "PRICE"
=> $value, "CURRENCY" => $data['1']);
            $res = CPrice::GetList(
                array(),
                array(
                    "PRODUCT_ID" => $product_id[$key],
                    "CATALOG_GROUP_ID" => $data['0'],
                )
            );
            if ($arr = $res->Fetch()) {
                CPrice::Update($arr["ID"], $arFields);
                echo '<b>Цена обновлена</b>';
            } else {
                CPrice::Add($arFields);
                echo '<b>Цена добавлена</b>';
            }
            echo '<br>';

        }

        $i++;
        echo '<br><br><br>';
    }
}
```


Информация необходимая для регистрации ключа Bitrix

- Полное юридическое название компании-владельца продукта или ФИО частного владельца
- Список адресов, включая тестовые, по которым будет доступна данная копия продукта "1С-Битрикс: Управление сайтом"
- Телефон владельца данной копии продукта (с указанием кода города)
- E-mail владельца для связи по вопросам лицензирования и использования программного продукта
- Контактное лицо, ответственное за использование данной копии программного продукта
- E-mail контактного лица
- Телефон контактного лица
- Прочая контактная информация

Определение города

Определить город по IP

```
static function getCity($ip = ""){
    if($ip){
        return \Bitrix\Main\Service\Geolp\Manager::getCityName($ip,'ru');
    }else{
        return
        \Bitrix\Main\Service\Geolp\Manager::getCityName(\Bitrix\Main\Service\Geolp\Manager::getReallp(),'ru');
    }
}
```

Получаем IP

```
function getUserIp(){
    return \Bitrix\Main\Service\Geolp\Manager::getReallp();
}
```

Геокодирование адреса

```
function askYandex($address){
    $params = array(
        'geocode' => $address, // адрес
        'format' => 'json',           // формат ответа
        'results' => 1,              // количество выводимых результатов
        'apikey'  => 'cd351fdb-0d22-4dd6-a6e5-a400114e552c',           // ваш api key
    );
    $response = json_decode(file_get_contents('https://geocode-maps.yandex.ru/1.x/?' . http_build_query($params,
    ", '&')));
```

```

if ($response->response->GeoObjectCollection->metaDataProperty->GeocoderResponseMetaData->found >
0)
{
    $pos = $response->response->GeoObjectCollection->featureMember[0]->GeoObject->Point->pos;

    $arrCoords = explode(' ', $pos);
    return ['lat' => $arrCoords[1], 'lng' => $arrCoords[0]];
}
else
{
    return 'Ничего не найдено';
}
}

```

Геокодирование адреса

```

function getListAddr($address){
    $params = array(
        'geocode' => $address, // адрес
        'format' => 'json',           // формат ответа
        'results' => 5,              // количество выводимых результатов
        'apikey'   => 'cd351fdb-0d22-4dd6-a6e5-a400114e552c',           // ваш api key
    );
    $response = json_decode(file_get_contents('http://geocode-maps.yandex.ru/1.x/?' . http_build_query($params,
    '', '&')));

    if ($response->response->GeoObjectCollection->metaDataProperty->GeocoderResponseMetaData->found >
0)
    {
        $arResult = [];
        foreach($response->response->GeoObjectCollection->featureMember as $k => $data){
            $arResult[$k]['name'] = $data->GeoObject->name;
            $arResult[$k]['street'] = $data->GeoObject->metaDataProperty->GeocoderMetaData->Address-
>Components[4]->name;
            $arResult[$k]['house'] = $data->GeoObject->metaDataProperty->GeocoderMetaData->Address-
>Components[5]->name;
        }
        return $arResult;
    }
}

```

```
}  
else  
{  
  return 'Ничего не найдено';  
}  
}
```

Создание типов цен и складов

```
if(CModule::IncludeModule("catalog")){

    if (($handle = fopen("partner.csv", "r")) !== FALSE) {
        while (($data = fgetcsv($handle, 1000, ";")) !== FALSE) {
            $rsCity = CIBlockElement::GetList(Array(), Array("IBLOCK_ID" => 14, "NAME" => $data['1']), false,
Array(), Array("ID", "NAME", "DATE_ACTIVE_FROM"));
            while($arCity = $rsCity->GetNextElement()) {
                $arFieldsCity = $arCity->GetFields();
            }

            if ($arFieldsCity['ID']):
                $trans = Cutil::translit($data['2'], "ru", array("replace_space" => "-", "replace_other" => "-"));

                $arFieldsPrice = array(
                    "NAME" => $trans,
                    "SORT" => 100,
                    "USER_GROUP" => array(1, 2),
                    "USER_GROUP_BUY" => array(1,2),
                    "USER_LANG" => array(
                        "ru" => $data['2'],
                        "en" => $data['2']
                    )
                );

                $ID_PRICE = CCatalogGroup::Add($arFieldsPrice);

                if ($ID_PRICE){
                    echo $ID_PRICE.'Тип цены для дилера: '.$data['2'].'<br>';
                }

                $arFieldsWarehouse = Array(
                    "TITLE" => $data['2'],
```

```
"ACTIVE" => "Y",
"ADDRESS" => $data['7'],
);

$ID_WAREHOUSE = CCatalogStore::Add($arFieldsWarehouse);

if($ID_WAREHOUSE) {
    echo 'Склад создан для дилера:'. $data['2'].'<br><br>';
}

CIBlockElement::SetPropertyValueCode($arFieldsCity['ID'], "ID_PRICE", $ID_PRICE);
CIBlockElement::SetPropertyValueCode($arFieldsCity['ID'], "ID_WAREHOUSE", $ID_WAREHOUSE);
endif;
}
}
}
```

Создание превьюшки из PDF и запись картинки в PREVIEW_PICTURE на лету по событию создания элемента

```
// проверить включено ли расширение Imagick (зайти через putty, в меню битрикс env, и там 8 -> 2 -> 7).
AddEventHandler("iblock", "OnAfterIBlockElementAdd", Array("Preview_PDF",
"OnAfterIBlockElementAddHandler")); // вызываем функцию при событии "после добавления элемента в
инфоблок

class Preview_PDF{

    // создаем обработчик события "OnAfterIBlockElementAdd"
    function OnAfterIBlockElementAddHandler(&$arParams){
        if(CModule::IncludeModule("iblock")) // подключаем модуль для работы с инфоблоком
        {
            $arFile = ""; // создаем переменную в которой будет записан путь к файлу
            $res = CIBlockElement::GetProperty($arParams["IBLOCK_ID"], $arParams["ID"], array("sort" => "asc"),
Array("CODE"=>"FILE"));
            if ($ob = $res->GetNext()){
                $arFile = CFile::GetFileArray($ob['VALUE']); // получаем массив данных по файлу
            }
        }
        //генерация превью
        $pdf = $_SERVER["DOCUMENT_ROOT"].$arFile['SRC'];
        $image = new Imagick($pdf.'[0]');
        $image->setImageFormat( "jpg" );
        $image->writeImage($_SERVER["DOCUMENT_ROOT"]. "/include/picture/" . str_replace('pdf', 'jpg',
```

```
$arFile["ORIGINAL_NAME"]);

    // запись превью в массив arFields
    $el = new CIBlockElement;
    $arLoadProductArray = Array(
        "PREVIEW_PICTURE" => CFile::MakeFileArray($_SERVER["DOCUMENT_ROOT"].
"/include/picture/".str_replace('pdf', 'jpg', $arFile["ORIGINAL_NAME"]))
    );
    $res = $el->Update($arFields["ID"], $arLoadProductArray);

}
}
```

Получить SEO данные элемента и секции Bitrix

Задача: Получить данные из вкладки SEO админ панели Битрикс

Не всегда получается решить задачу базовыми компонентами, однако оставить пользователям возможность устанавливать свои мета теги, через стандартный интерфейс необходимо.

Чтобы вытянуть данные из полей, которые находятся во вкладке SEO битрикс, используем следующий код:

Для элементов:

```
$iPropElementValues = new
\\Bitrix\\Iblock\\InheritedProperty\\ElementValues($arParams['IBLOCK_ID'],$element['ID']);
$pageProperties = $iPropElementValues->getValues();

$APPLICATION->SetTitle($pageProperties['ELEMENT_PAGE_TITLE']);
$APPLICATION->SetPageProperty('title',$pageProperties['ELEMENT_META_TITLE']);
$APPLICATION->SetPageProperty('description',$pageProperties['ELEMENT_META_DESCRIPTION']);
```

Для секций:

```
$iPropValues = new \\Bitrix\\Iblock\\InheritedProperty\\SectionValues($arParams['IBLOCK_ID'],
$arParams['SECTION_ID']);
$IPROPERTY = $iPropValues->getValues();
```

Конечно, не забываем про то, что необходимо подключить необходимый класс.

```
use Bitrix\\Iblock\\InheritedProperty\\ElementValues;
```

или

```
Bitrix\\Iblock\\InheritedProperty\\SectionValues
```

В остальном проблем быть не должно.

Оптимизация сайта для PageSpeed Insights

Оптимизация сайта для PageSpeed Insights

Задача

Провести оптимизацию выдачи контента Битрикс, контроль качества оптимизации провести при помощи PageSpeed Insights.

Решение

Инструменты для проверки скорости

- <https://developers.google.com/speed/pagespeed/insights/?hl=ru>
- <https://tools.pingdom.com/https://tools.pingdom.com>

Сервисы оптимизации изображений

- <https://compressor.io/compress>
- <https://tinypng.com/>
- <http://optimizilla.com/ru/>

Общие рекомендации

1. Привести в соответствие изображения, таким образом, что бы браузеру не приходилось их масштабировать
2. Включить сжатие файлов в nginx
3. При помощи настроек битрикс перенести js и css в конец страницы
4. Минифицировать js и css при помощи gulp или иными средствами
5. Исключить служебные скрипты и стили битрикс из формирования пользовательских страниц сайта
6. Минифицировать html

<https://github.com/addyosmani/critical-path-css-tools>

Перенос js и css в конец страницы

JS

Для корректной работы битрикс с файлами стилей и скриптов их нужно подключать при помощи функций Битрикс:

```
$asset = \\Bitrix\\Main\\Page\\Asset::getInstance();

// подключение стилей
$asset->addCss(SITE_TEMPLATE_PATH."/some.css");

// подключение скриптов
$asset->addJs(SITE_TEMPLATE_PATH."/some.js");

// подключение произвольных кусков кода
$asset->addString('<link rel="canonical" href="https://' . $_SERVER['HTTP_HOST'] . $APPLICATION-
>GetCurPage(false) . "' />');
```

Для объединения всех стилей и скриптов в единый файл нужно в настройках Битрикс **Настройки -> Настройки продукта -> Настройки модулей -> Главный модуль** активировать галки **Объединять CSS файлы** и **Объединять JS файлы**

Так же можно проставить галки **Создавать сжатую копию объединенных CSS и JS файлов** и **Переместить весь Javascript в конец страницы**

Программно опцию "Переместить весь Javascript в конец страницы" можно активировать так:

```
$asset->setJsToBody(true);
```

В случае программной активации галочка в админке работать не будет, т.е. скрипты всегда будут помещаться в конец страницы.

Не все скрипты нужно помещать в футер. например tagManager от гугла должен быть вверху. Для исключения скрипта из переносимых нужно добавить атрибут **data-skip-moving=true** в тег script. Пример:

```
<script data-skip-moving="true">
  // your script code
</script>
```

CSS

Для переноса CSS в конец страницы у Битрикса нет стандартных средств, но т.к. за вывод css отвечает отдельный метод, то мы можем вывести его в footer.php самостоятельно.

Для этого нужно вместо:

```
<?php $APPLICATION->ShowHead(); ?>
```

В файле **header.php** добавить:

```
// вместо $APPLICATION->ShowHead();
$bXhtmlStyle = true;
echo '<meta http-equiv="Content-Type" content="text/html; charset="'.LANG_CHARSET.'".($bXhtmlStyle? '
/:"').">!\n";
$APPLICATION->ShowMeta("robots", false, $bXhtmlStyle);
$APPLICATION->ShowMeta("keywords", false, $bXhtmlStyle);
$APPLICATION->ShowMeta("description", false, $bXhtmlStyle);
$APPLICATION->ShowLink("canonical", null, $bXhtmlStyle);
$APPLICATION->ShowHeadStrings();
$APPLICATION->ShowHeadScripts();
```

В файле **footer.php** добавить:

```
$APPLICATION->ShowCSS(true, $bXhtmlStyle);
```

Исключить служебные скрипты и стили битрикс из формирования

ПОЛЬЗОВАТЕЛЬСКИХ СТРАНИЦ САЙТА

Для этого нужно добавить обработку контента страницы перед выдачей его из буфера браузеру.

В `init.php` должно быть добавлено подключение файлов констант, обработчиков и функций, т.к. все действия уже будем совершать с ними.

Пример подключения файлов в `init.php`:

```
if(!defined("B_PROLOG_INCLUDED") || B_PROLOG_INCLUDED!==true) die();

// Подключение констант
if (file_exists($_SERVER["DOCUMENT_ROOT"]."/local/php_interface/include/constants.php")) {
    require_once($_SERVER["DOCUMENT_ROOT"] . "/local/php_interface/include/constants.php");
}

// Подключение обработчиков событий
if (file_exists($_SERVER["DOCUMENT_ROOT"]."/local/php_interface/include/handlers.php")) {
    require_once($_SERVER["DOCUMENT_ROOT"] . "/local/php_interface/include/handlers.php");
}

// Подключение глобальных функций (чаще всего используется для переноса кода сторонних разработчиков
// при получении проекта на доработку)
if (file_exists($_SERVER["DOCUMENT_ROOT"]."/local/php_interface/include/functions.php")) {
    require_once($_SERVER["DOCUMENT_ROOT"] . "/local/php_interface/include/functions.php");
}
```

В файл `functions.php` добавим функции обработчиков событий:

```
<?
// удаляем скрипты ядра при отдаче сайта пользователям
function deleteKernelJs(&$content) {
    global $USER, $APPLICATION;
    if((is_object($USER) && $USER->IsAuthorized()) || strpos($APPLICATION->GetCurDir(), "/bitrix/")!==false)
return;
    if($APPLICATION->GetProperty("save_kernel") == "Y") return;

    $arPatternsToRemove = Array(
        '/<script.+?src=".+?kernel_main\\kernel_main\\.js\\?\\d+ "><\\script\\>/',
```


Источник - <https://gdecider.github.io/articles-bx-site-optimisation-for-google-pagespeed-insights.html>

Пример формирования YML фида

```
require_once($_SERVER['DOCUMENT_ROOT']. "/bitrix/modules/main/include/prolog_before.php");
header("Content-Type: text/xml");
header("Expires: Thu, 19 Feb 1998 13:24:18 GMT");
header("Last-Modified: ".gmdate("D, d M Y H:i:s")." GMT");
header("Cache-Control: no-cache, must-revalidate");
header("Cache-Control: post-check=0,pre-check=0");
header("Cache-Control: max-age=0");
header("Pragma: no-cache");

$IBLOCK_ID = 26;

// получения всех сайтов
$rsSites = CSite::GetList($by="sort", $order="desc", ['ACTIVE' => 'Y']);

// Исключить характеристики
$specificationsArr = array(
    'Рейтинг для микроразметки',
    'Картинки',
    'Похожие товары по фильтру',
    'Аксессуары по фильтру',
    'Товары комплекта',
    'Видео (код из Youtube)',
    'Сертификаты',
    'ID поста блога для комментариев',
    'Регион'
);

// https://site.com/bitrix/admin/cat_group_admin.php?lang=ru
// Розничная цена МСК - 1
// Розничная цена Ростов-на-Дону - 7
// Розничная цена Краснодар - 10
// Розничная цена СПб - 16
```

```

// Розничная цена Новосибирск - 19
// Розничная цена Казань - 13

$text = '<yml_catalog date="" .date('Y-m-d H:i:s').">';
while ($arSite = $rsSites->Fetch()){

    // site.com
    // spb.site.com
    // rostov.site.com
    // krasnodar.site.com

    if (empty($arSite['SITE_URL'])) {
        $arSite['SITE_URL'] = 'https://site.com';
    }

    if ($arSite['SITE_URL'] == 'https://site.com' OR $arSite['SITE_URL'] == 'https://spb.site.com' OR
    $arSite['SITE_URL'] == 'https://rostov.site.com' OR $arSite['SITE_URL'] == 'https://krasnodar.site.com' ) {

        $category_query = CIBlockSection::GetList( Array('name' => 'asc'), Array('IBLOCK_ID' => $IBLOCK_ID, 'ACTIVE'
=> 'Y') );

        $product_query = CIBlockElement::GetList( array("ID" => "ASC"), array("IBLOCK_ID" => $IBLOCK_ID,
"ACTIVE"=>"Y"), false, Array(), array("ID", "IBLOCK_ID", "IBLOCK_SECTION_ID", "NAME", "DETAIL_TEXT",
"DETAIL_PICTURE", "DETAIL_PAGE_URL", "DATE_ACTIVE_FROM", "PROPERTY_*") );

        if ($arSite['SITE_URL'] == 'https://site.com' ) {
            $PRICE_TYPE_ID = 1;
        } elseif ($arSite['SITE_URL'] == 'https://novosibirsk.site.com'){
            $PRICE_TYPE_ID = 19;
        } elseif ($arSite['SITE_URL'] == 'https://rostov.site.com'){
            $PRICE_TYPE_ID = 7;
        } elseif ($arSite['SITE_URL'] == 'https://krasnodar.site.com'){
            $PRICE_TYPE_ID = 10;
        } elseif ($arSite['SITE_URL'] == 'https://kazan.site.com'){
            $PRICE_TYPE_ID = 13;
        } elseif ($arSite['SITE_URL'] == 'https://spb.site.com'){
            $PRICE_TYPE_ID = 16;
        }

    }

    $text .= '<shop>';

```

```

$text .= '<name>'.$arSite['NAME'].'</name>';
$text .= '<company>КНАУФ Инсулейшн</company>';
$text .= '<url>'.$arSite['SITE_URL'].'</url>';
$text .= '<platform>BSM/Yandex/Market</platform>';
$text .= '<version>2.3.4</version>';
$text .= '<сра>1</сра>';
$text .= '<currencies><currency id="RUR" rate="1"/></currencies>';

$text .= '<categories>';
while ($category = $category_query->GetNext()) {
    $text .= '<category id="'. $category['ID']. '">'.$category['NAME'].'</category>';
}
$text .= '</categories>';
$text .= '<enable_auto_discounts>>true</enable_auto_discounts>';
$text .= '<offers>';

while($product = $product_query->GetNextElement()){
    $productArFields = $product->GetFields();
    $productArProps = $product->GetProperties();

    // Определяем цену
    $price_qury = CPrice::GetList(array(), array( "PRODUCT_ID" => $productArFields['ID'], "CATALOG_GROUP_ID"
=> $PRICE_TYPE_ID ));

    if ($price = $price_qury->Fetch()) {
        // Определяем категорию к которой относиться товар
        $category = CIBlockSection::GetByID($productArFields['IBLOCK_SECTION_ID']);
        $category_ID = $category->GetNext();

        $text .= '<offer id="'. $productArFields['ID']. '" available="true">';
        $text .= '<url>'.$arSite['SITE_URL'].$productArFields['DETAIL_PAGE_URL'].'</url>';
        $text .= '<price>'.substr($price["PRICE"], 0, -3).'</price>';
        $text .= '<currencyId>RUR</currencyId>';
        $text .= '<categoryId>'.$category_ID['ID'].'</categoryId>';

        $text .= '<picture>'.$arSite['SITE_URL'].CFile::GetPath($productArFields['DETAIL_PICTURE']).</picture>';

        foreach($productArProps['MORE_PHOTO']['VALUE'] as $picture){
            $text .= '<picture>'.$arSite['SITE_URL'].CFile::GetPath($picture).'</picture>';
        }
    }
}

```

```

$text .= '<name>'.$productArFields['NAME'].'</name>';
$text .= '<description><![CDATA['.$productArFields['DETAIL_TEXT'].']]></description>';

foreach($productArProps as $specifications){
    if ($specifications['VALUE'] AND !in_array($specifications['NAME'], $specificationsArr)){

        if ($specifications['NAME'] == "Материал"){
            $text .= '<param name="'. $specifications['NAME'].'">'.$specifications['VALUE'][0]. '</param>';
        } elseif ($specifications['NAME'] == "Преимущества") {
            $text .= '<param
name="'. $specifications['NAME'].'"><![CDATA['.$specifications['VALUE']['TEXT'].']]></param>';
        } else {
            $text .= '<param name="'. $specifications['NAME'].'">'.$specifications['VALUE']. '</param>';
        }

    }
}

$text .= '</offer>';
}
// break;
}

$text .= '</offers>';
$text .= '</shop>';
// break;
// die();
}
}

$text .= '</yml_catalog>';

echo $text;

```

Водяные знаки на фото

```
$arWaterMark = Array(
    array(
        ["name" => "watermark",
        ["position" => "bottomright", // Положение
        ["type" => "image",
        ["size" => "real",
        ["file" => $_SERVER["DOCUMENT_ROOT"].'/upload/watermark.png', // Путь к картинке
        ["fill" => "exact",
    )
);

$PreviewPict = CFile::ResizeImageGet($arResult["PREVIEW_PICTURE"]["ID"], ['width'=>200, 'height'=>200],
BX_RESIZE_IMAGE_PROPORTIONAL, true, $arWaterMark, false, 80);

echo CFile::ShowImage($PreviewPict["src"], 200, 200, "id='".$arResultIDs['PICT']."' alt='".$arResult["NAME"]."'
itemprop='image' class='catalogLuster' title='".$arResult["NAME"]."' ", "", false);
```

```
$arFilters = array(
    array(
        "name"      => "watermark",
        "position"   => "bc",      // (доступные варианты tl|tc|tr|ml|mc|mr|bl|bc|br или
        topleft|topcenter|topright|centerleft|center|centerright|bottomleft|bottomcenter|bottomright)
        "type"      => "image",    // (доступные варианты "image|text")
        "size"      => "real",     // ( доступные варианты big|medium|small|real; real доступен только для
        type=image )
        "coefficient" => "1",      // ( альтернатива параметру size. В случае текстового варианта - от 1 до
        7, по крайней мере так в исходниках; для картинок 0.1 до 1 )
        "fill"      => "exact",    // ( доступные варианты exact|resize|repeat )
        "file"      => $_SERVER["DOCUMENT_ROOT"]."/watermark.png", // (абсолютный путь до картинки с
        водяным знаком)
        "text"      => "Водяной знак!", // ( текст водяного знака )
        "color"     => "ff0000",   // ( hex формат цвета текста )
        "font"      => $_SERVER["DOCUMENT_ROOT"]."/arial.ttf", // ( путь до файла шрифта в формате
        .ttf )
        "use_copyright" => "Y"    // ( использовать ли знак копирайта перед текстом )
```

)

);

Настройка редиректов .htaccess

```
<IfModule mod_rewrite.c>
[Options +FollowSymLinks
[RewriteEngine On
[]
[# редирект на www
[RewriteCond %{HTTP_HOST} ^site\.ru
[RewriteRule ^(.*)$ https://www.site.ru/$1 [R=302,L]

# редирект с index.php на site.ru
RewriteCond %{THE_REQUEST} ^[A-Z]{3,9}\ /index\.php\ HTTP/
RewriteRule ^index\.php$ https://polymertx.ru/ [R=301,L]

[# редиректы на https
[RewriteCond %{SERVER_PORT} ^80$ [OR]
[RewriteCond %{HTTP} =on
RewriteCond %{HTTP_HOST} ^www.site\.ru
[RewriteRule ^(.*)$ https://www.site.ru/$1 [R=302,L]
[]

[# редирект с /test/index.php на /test/
[RewriteCond %{REQUEST_URI} !^/bitrix/
[RewriteCond %{REQUEST_METHOD} =GET
[RewriteCond %{REQUEST_URI} ^(.*)/index\.php$
[RewriteRule ^(.*)$ %1/ [R=302,L]

[# редирект с /test на /test/
[RewriteCond %{REQUEST_URI} !^/bitrix/
[RewriteCond %{REQUEST_URI} (.*[/^\.]+)(\$|\?)
[RewriteRule .* %1/ [R=302,L]

[# редирект с /test/// на /test/, а также с /test1///test2/ на /test1/test2/
```

```

[]RewriteCond %{REQUEST_URI} !^/bitrix/
[]RewriteCond %{REQUEST_URI} ^(.*)/{2,}?(.*)$
[]RewriteRule .* %1/%2 [R=302,L]
[]RewriteCond %{REQUEST_URI} !^/bitrix/
[]RewriteCond %{THE_REQUEST} //
[]RewriteCond %{QUERY_STRING} !http(s)://
[]RewriteRule .* /$0 [R=302,L]
[]
[]
[]# "родное" правило битрикса
[]RewriteCond %{REQUEST_FILENAME} !-f
[]RewriteCond %{REQUEST_FILENAME} !-l
[]RewriteCond %{REQUEST_FILENAME} !-d
[]RewriteCond %{REQUEST_FILENAME} !/bitrix/urlrewrite.php$
[]RewriteRule ^(.*)$ /bitrix/urlrewrite.php [L]
[]RewriteRule .* - [E=REMOTE_USER:%{HTTP:Authorization}]
# RewriteRule ^robots.txt$ robots.php [L,QSA]
</IfModule>

<IfModule mod_rewrite.c>
[]Options +FollowSymLinks
[]RewriteEngine on
[]
[]# Редирект с www и с http #
[]RewriteCond %{HTTP_HOST} ^www\.(.+)$ [NC]
[]
[]RewriteRule ^(.*)$ https://%1/$1 [L,R=301]
[]RewriteCond %{HTTP:X-Forwarded-Proto} !https [NC]
[]RewriteRule .* https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
[]
[]# Редирект со слеша в конце #
[]RewriteCond %{THE_REQUEST} //
[]RewriteRule .* https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
[]RewriteCond %{REQUEST_FILENAME} !-f
[]RewriteCond %{REQUEST_FILENAME} !-l
[]RewriteCond %{REQUEST_FILENAME} !-d
[]RewriteCond %{REQUEST_FILENAME} !/bitrix/urlrewrite.php$
[]RewriteRule ^(.*)$ /bitrix/urlrewrite.php [L]
[]RewriteRule .* - [E=REMOTE_USER:%{HTTP:Authorization}]
[]

```

Redirect 301 /index.html https://bonolit.ru/

</IfModule>

#####

Стандартный .htaccess для проектов студии Клондайк, версия 2.4

#####

RewriteEngine On # Директива включает редиректы.

RewriteBase / # Без директивы (.*) = /\$1 будет /var/www/site/web/\$1 с директивой = /\$1

Options +FollowSymLinks # Разрешает переход по символическим ссылкам.

#####

Перенаправляем протокол https на http

#####

RewriteCond %{HTTPS} on # Проверяем наличие https в URL.

RewriteRule ^.*\$ http://%{SERVER_NAME}%{REQUEST_URI} # Перенаправляем протокол на http.

#####

Выбор основного зеркала (или с www или без www)

#####

1. Редирект с www на без www. (раскомментировать директивы пункта 1)

#RewriteCond %{HTTP_HOST} ^www\.(.*) [NC]

Проверяем, содержит ли домен www (в начале URL).

#RewriteRule ^(.*)\$ http://%1/\$1 [R=301,L]

Перенаправляем URL на домен без www.

####

2. Редирект без www на www. (раскомментировать директивы пункта 2)

#RewriteCond %{HTTP_HOST} !^www\.(.*) [NC] # Проверяем, не содержит ли домен www (в начале URL).

#RewriteRule ^(.*)\$ http://www.%{HTTP_HOST}/\$1 [R=301,L] # Перенаправляем URL на домен с www.

#####

```
##### Убираем повторяющиеся слешы (/) в URL #####
#####
#####
RewriteCond %{THE_REQUEST} //
    # Проверяем, повторяется ли слеш (//) более двух раз.
RewriteRule .* /$0 [R=301,L]
    # Исключаем все лишние слешы.

#####
#####
##### Убираем слешы в конце URL для статических файлов (содержит точку) #####
#####
#####
RewriteCond %{REQUEST_URI} \.++$
    # Если файл содержит точку.
RewriteCond %{REQUEST_FILENAME} !-d
    # И это не директория.
RewriteCond %{REQUEST_FILENAME} -f
    # Является файлом.
RewriteCond %{REQUEST_URI} ^(.+)/$
    # И в конце URL есть слеш.
RewriteRule ^(.+)/$ /$1 [R=301,L]
    # Исключить слеш.

#####
#####
##### Добавляем слеш(/), если его нет, и это не файл. #####
#####
#####
RewriteCond %{REQUEST_URI} !(\.*/)$
    # Если слеша в конце нет.
RewriteCond %{REQUEST_FILENAME} !-f
    # Не является файлом.
RewriteCond %{REQUEST_URI} !\.++$
    # В URL нет точки (файл).
RewriteRule ^(\.*/)$ $1/ [L,R=301]
    # Добавляем слеш в конце.

#####
#####
```

```
##### Убираем index.php, если он есть в конце URL #####
#####
#####
RewriteCond %{REQUEST_METHOD} =GET
    # Выявляем GET запрос в URL (не POST).
RewriteCond %{REQUEST_URI} ^(.*)/index\.php$
    # URL содержит index.php в конце.
RewriteRule ^(.*)$ %1/ [R=301,L]
    # Удалить index.php из URL.

#####
#####
##### Компрессия статического контента для гугл спид тест #####
#####
#####
<IfModule mod_deflate.c>
    # Compress HTML, CSS, JavaScript, Text, XML and fonts
    AddOutputFilterByType DEFLATE application/rss+xml
    AddOutputFilterByType DEFLATE application/vnd.ms-fontobject
    AddOutputFilterByType DEFLATE application/x-font
    AddOutputFilterByType DEFLATE application/x-font-opentype
    AddOutputFilterByType DEFLATE application/x-font-otf
    AddOutputFilterByType DEFLATE application/x-font-truetype
    AddOutputFilterByType DEFLATE application/x-font-ttf
    AddOutputFilterByType DEFLATE application/x-javascript
    AddOutputFilterByType DEFLATE application/xhtml+xml
    AddOutputFilterByType DEFLATE application/xml
    AddOutputFilterByType DEFLATE font/opentype
    AddOutputFilterByType DEFLATE font/otf
    AddOutputFilterByType DEFLATE font/ttf
    AddOutputFilterByType DEFLATE image/svg+xml
    AddOutputFilterByType DEFLATE image/x-icon
    AddOutputFilterByType DEFLATE text/css
    AddOutputFilterByType DEFLATE text/html
    AddOutputFilterByType DEFLATE text/javascript
    AddOutputFilterByType DEFLATE text/plain
    AddOutputFilterByType DEFLATE text/xml
    AddOutputFilterByType DEFLATE image/svg+xml
</IfModule>
<IfModule mod_expires.c>
```

ExpiresActive on
ExpiresByType image/jpeg "access plus 3 day"
ExpiresByType image/svg "access plus 3 day"
ExpiresByType image/gif "access plus 3 day"
ExpiresByType image/png "access plus 3 day"
ExpiresByType text/javascript "access plus 3 day"
ExpiresByType text/css "access plus 3 day"
ExpiresByType application/javascript "access plus 3 day"

</IfModule>

#####

#####

Конечная часть, далее следуют собственные директивы .htaccess

#####

#####

Битрикс как проверить, что письма были отправлены

Частая проблема письма с сайта не доходят и не всегда понятно, проблема в сайте, хостинге, промежуточных сервисах или даже конечном ящике получателя. В данной заметке расскажу как можно проверить, что письма ушли из Битрикса.

Отправка писем, попадет в журнал событий Битрикса. Хранятся все события в таблице bevent, посмотреть значения можно через Настройки - Производительность - Таблицы (`/bitrix/admin/perfmon_table.php?lang=ru&table_name=b_event`)

Для событий отправки писем нужно смотреть на поле SUCCESS_EXEC Возможные статусы:

- SUCCESS_EXEC = Y — сообщение отправлено.
- SUCCESS_EXEC = N — не обработано событие (а значит не отправлено)
- SUCCESS_EXEC = 0 — не найден шаблон (указан не верный шаблон)
- SUCCESS_EXEC = F — надо смотреть логи отправки сервера или хостинга.

Основное на что стоит обратить внимание, что SUCCESS_EXEC = N не значит, что письмо вообще не отправлено, это текущий статус, и если отправка настроена на cron (по расписанию) или на хиты, то нужно подождать перед тем, как проверять статус!

Прикреплять файла к письму (Веб-формы)

```
$eventManager = \Bitrix\Main\EventManager::getInstance();

$eventManager->addEventHandler("main", "OnBeforeEventAdd", array("MailEventHandler",
"onBeforeEventAddHandler"));
class MailEventHandler{

    static function onBeforeEventAddHandler(&$event, &$lid, &$arFields, &$message_id, &$files){
        // Названия типов почтовых событий и идентификаторы почтовых шаблонов, по которым будет
проходить фильтрация при отлавливании события
        $events = array('FORM_FILLING_resume');
        $messagelds = array('37');

        if (in_array($event, $events) && in_array($message_id, $messagelds)){
            // Определяем массив прикрепляемых к письму идентификаторов файлов, если не задан
            if (!is_array($files)){
                $files = [];
            }
            // Перебираем поля письма
            foreach ($arFields as $field){
                // Если находим ссылку на файл, передаем ее дальше
                if ($link = self::getLinkFromField($field)){
                    // Если получаем идентификатор файла, сохраняем его в массив идентификаторов файлов
письма
                    if ($fileID = self::getFileFromLink($link)){
                        $files[] = $fileID;
                    }
                }
            }
        }
    }

    // Метод возвращает ссылку на файл
```

```
private static function getLinkFromField($field){
    preg_match("/(http\:.*form_show_file.*action\=download)/", $field, $out);
    // Если поле соответствует маске пути к файлу, возвращаем путь
    return ($out[1] ?: false);
}
// Метод возвращает идентификатор файла
private static function getFileFromLink($link){
    // Создаем новый объект, в который записываем ссылку
    $uri = new \Bitrix\Main\Web\Uri($link);
    // Разбираем строку URI и создаем массив из параметров запроса
    parse_str($uri->getQuery(), $query);
    // Метод возвращает массив свойств файла по идентификатору результата веб-формы и хэшу
    $arFile = CFormResult::GetFileByHash($query["rid"], $query["hash"]);
    // Возвращаем идентификатор файла
    return $arFile['FILE_ID'];
}
}
```

База

Вывод всех элементов раздела Bitrix

```
$arSelect = Array("ID", "IBLOCK_ID", "NAME", "PREVIEW_PICTURE", "CODE", "PROPERTY_*");  
$arFilter = Array(  
    "IBLOCK_ID" => 6, // id инфоблока  
    "ACTIVE" => "Y", // активность  
    "SECTION_ID" => 12, // нужная секция  
);  
  
$res = CIBlockElement::GetList(Array("SORT" => "ASC"), $arFilter, false, Array(), $arSelect);  
while($ob = $res->GetNextElement()){  
    $arFields = $ob->GetFields();  
    $arProps = $ob->GetProperties();  
}
```

Вывод разделов и элементов в bitrix:news.list

Часто возникает необходимость вывести разделы инфоблока с элементами на одной странице. Для этой цели подойдет связка catalog.sections + news.list. Или можно обойтись одним bitrix:news.list, немного кастомизировав шаблон компонента. Пойдем вторым путем.

Итак.

1. Размещаем компонент bitrix:news.list на целевой странице.
2. Копируем шаблон компонента в текущий шаблон сайта.
3. Создаем в нем файл result_modifier.php с таким содержимым.

```
// Получаем все активные разделы ИБ
$rsSections = CIBlockSection::GetList(
    Array("SORT" => "ASC"),
    Array(
        "=IBLOCK_ID" => $arParams["IBLOCK_ID"],
        "=ACTIVE"    => "Y"
    )
);

// Собираем разделы в массив
while ($arResult = $rsSections->GetNext())
    $arResult[] = $arResult;

// Фильтруем элементы по принадлежности к разделу
// Получаем итоговый массив со структурой: [SECTION] => [ELEMENTS]
foreach ($arResult as $arResult){
    foreach ($arResult["ITEMS"] as $arResultItem){
        if ($arResultItem["IBLOCK_SECTION_ID"] == $arResult["ID"])
            $arResult["ELEMENTS"][] = $arResultItem;
    }
    $arResultElementGroups[] = $arResult;
}
```

```
$arResult["ITEMS"] = $arResultGroups;
```

В итоге получаем такую структуру:

```
Array
(
    // Разделы
    [0] => Array
        (
            [ID] => 17
            [NAME] => Юридические вопросы
            [CODE] => juridicheskie-voprosy
            [SECTION_PAGE_URL] => /faq/yuridicheskie-voprosy/
            [ELEMENTS] => Array
                (
                    // Элементы
                    [0] => Array
                    [1] => Array
                    [2] => Array
                )
        )
    [1] => Array
    [2] => Array
    ...
)
```

В template.php выводим так:

```
<?foreach ($arResult["ITEMS"] as $arResultSection):?>
    <h4><?=$arResultSection["NAME"]?></h4>
    <?if ($arResultSection["ELEMENTS"]):?>
        <ul>
            <?foreach ($arResultSection["ELEMENTS"] as $arResultItem):?>
                <?
                    $this->AddEditAction($arResultItem['ID'], $arResultItem['EDIT_LINK'], CIBlock::GetArrayByID($arResultItem["IBLOCK_ID"],
"ELEMENT_EDIT"));
                    $this->AddDeleteAction($arResultItem['ID'], $arResultItem['DELETE_LINK'],
CIBlock::GetArrayByID($arResultItem["IBLOCK_ID"], "ELEMENT_DELETE"), array("CONFIRM" =>
GetMessage('CT_BNL_ELEMENT_DELETE_CONFIRM')));
                ?>
            <?>
        <?>
    <?>
</if>
</foreach>
```

```

<li id="<?=$this->GetEditAreaId($arResult['ID']);?>">
    <a href="<?=$arResult["DETAIL_PAGE_URL"]?>"><?=$arResult["NAME"]?></a>
</li>
<?endforeach?>
</ul>
<?endif?>
<?endforeach?>

```

Не нравится мне имя ключа `$arResult["ITEMS"]`, напрашивается "SECTIONS", но пока оставлю, как есть.

Однако есть вариант лучше, который избавит нас от вложенного foreach.

```

$rsSections = CIBlockSection::GetList(
    Array("SORT" => "ASC"),
    Array(
        "IBLOCK_ID" => $arParams['IBLOCK_ID'],
        "ACTIVE" => "Y"
    )
);

// Тут вместо инкрементного индекса, ID раздела
while ($arResult = $rsSections->GetNext())
    $arResult[$arResult['ID']] = $arResult;

// По нему производим неявную фильтрацию
foreach($arResult["ITEMS"] as $arResultItem) {
    $arResult[$arResultItem['IBLOCK_SECTION_ID']]['ITEMS'][] = $arResultItem;
}

$arResult["SECTIONS"] = $arResult;

```

Получаем структуру вида:

```

Array
(
    // Разделы
    [18] => Array
        (
            [ID] => 18
            [NAME] => Вопросы по доставке и оплате

```

```

[CODE] => voprosy-po-dostavka-i-oplate
[SECTION_PAGE_URL] => /faq/voprosy-po-dostavka-i-oplate/
[IBLOCK_TYPE_ID] => references
[ITEMS] => Array
(
    // Элементы
    [0] => Array
        (
            [ID] => 331
            [IBLOCK_SECTION_ID] => 18
            [NAME] => Как изменить номер телефона в личном кабинете?
            [DETAIL_PAGE_URL] => /faq/voprosy-po-dostavka-i-oplate/kak-izmenit-nomer-telefona/
        )
    [1] => Array
    [2] => Array

```

В template.php аналогичная верстка, лишь переименуем ключи \$arResult.

```

<?foreach ($arResult["SECTIONS"] as $arSection):?>
    <h4><?=$arSection["NAME"]?></h4>
    <ul>
        <?foreach ($arSection["ITEMS"] as $arItem):?>
            <li id="<?=$this->GetEditAreaId($arItem['ID']);?>">
                <a href="<?=$arItem["DETAIL_PAGE_URL"]?>"><?=$arItem["NAME"]?></a>
            </li>
        <?endforeach?>
    </ul>
<?endforeach?>

```

1С-Bitrix: site under construction – изменяем шаблон

Способ изменения стандартной надписи Site under construction в 1С-Bitrix при отключения сайта через панель администрирования.

Шаблон “Under construction” находится в следующем файле:

```
/bitrix/modules/main/include/site_closed.php
```

Необходимо скопировать его в

```
/bitrix/php_interface/<language>/
```

или

```
/bitrix/php_interface/include/
```

Шпаргалка Bitrix

Шпаргалка Bitrix

Шаблон

Пролог/подключение языковых файлов (header.php/footer.php)

```
if (!defined("B_PROLOG_INCLUDED") || B_PROLOG_INCLUDED!==true) die();  
IncludeTemplateLangFile(__FILE__);
```

Язык шаблона

```
<html lang="<?=LANGUAGE_ID?>">
```

Заголовки

```
<title><?$APPLICATION->ShowTitle();?></title>  
<h1><?$APPLICATION->ShowTitle(false,false);?></h1>
```

Мета данные для head

```
<?$APPLICATION->ShowHead();?>
```

Панель в <body

```
<?$APPLICATION->ShowPanel();?
```

Подключение CSS и JS

```
<?$APPLICATION->SetAdditionalCSS(SITE_TEMPLATE_PATH.'/css/style.css');?>  
<?$APPLICATION->AddHeadScript(SITE_TEMPLATE_PATH.'/js/script.js');?>
```

Подключение CSS и JS с поддержкой D7

```
use Bitrix\\Main\\Page\\Asset;
Asset::getInstance()->addJs(SITE_TEMPLATE_PATH . "/js/fix.js");
Asset::getInstance()->addCss(SITE_TEMPLATE_PATH . "/styles/fix.css");
Asset::getInstance()->addString("<link href='http://...' rel='stylesheet' type='text/css'>");
```

Путь до шаблона

```
<?=SITE_TEMPLATE_PATH?>
```

Включаемая область

```
$APPLICATION->IncludeComponent(
    "bitrix:main.include",
    "",
    array(
        "AREA_FILE_SHOW" => "sect",
        "AREA_FILE_SUFFIX" => "inc",
        "AREA_FILE_RECURSIVE" => "Y",
        "EDIT_TEMPLATE" => ""
    ),
    false
);
```

Включаемая область из файла

```
$APPLICATION->IncludeComponent(
    "bitrix:main.include",
    "",
    array(
        "AREA_FILE_SHOW" => "file",
        "AREA_FILE_SUFFIX" => "inc",
        "EDIT_TEMPLATE" => "",
        "PATH" => "/include/name.php"
    )
);
```

Проверки URL

```
if ($GLOBALS["APPLICATION"]->GetCurPage() == "/" ) {} // если в корне

if (CSite::InDir('/about/')) {} // раздел и подразделы

if (strstr($GLOBALS["APPLICATION"]->GetCurPage(), "page")) {} // проверка на совпадение
```

Свойства разделов/страницы

```
if ($APPLICATION->GetProperty("FULL_WIDTH") == "Y") {}

$APPLICATION->SetPageProperty("keywords", "...");
```

Фильтр для вывода элементов у которых значение свойства равно определенному значению

```
global $arFilter;

$arFilter = array(
    "IBLOCK_ID" => 9,
    "ACTIVE" => "Y",
    "PROPERTY_NTV_VALUE" => "Да",
);

// в компоненте "FILTER_NAME" => "arFilter",
```

Для свойств типа "список"

```
$enum_list = CIBlockPropertyEnum::GetList(
    array("SORT" => "ASC", "NAME" => "ASC"),
    array("IBLOCK_ID" => 2, "CODE" => "PROPNAME", "VALUE" => "да")
```

```

);

$arEnumIsMain = $enum_list->GetNext();

global $arrFilter;
$arrFilter = array(
    "PROPERTY" => array("PROPNAME" => $arEnumIsMain["ID"])
);

```

Пример фильтра для CIBlockElement::GetList по списку

```

CIBlockElement::GetList(
    array(), // arOrder
    array(
        "IBLOCK_ID" => $arResult["IBLOCK_ID"],
        "ACTIVE" => "Y",
        "PROPERTY_SHOW_IN_POPULAR_VALUE" => array("Да")
    ), // arFilter
    false, // arGroupBy
    false, // arNavStartParams
    array() // arSelectFields
);

```

Редактирование элементов

```

$this->AddEditAction(
    $arResult['ID'],
    $arResult['EDIT_LINK'],
    CIBlock::GetArrayByID($arParams['IBLOCK_ID'], "ELEMENT_EDIT")
);

$this->AddDeleteAction(
    $arResult['ID'],
    $arResult['DELETE_LINK'],
    CIBlock::GetArrayByID($arParams['IBLOCK_ID'], "ELEMENT_DELETE"),
    array("CONFIRM" => GetMessage('CT_BNL_ELEMENT_DELETE_CONFIRM'))
);

```

```
<li id="<?=$this->GetEditAreald($arItem['ID']);?>"></li>
```

Буферизация

Буферизация средствами php

```
ob_start();  
// Код компонента  
$GLOBALS['NAME'] = ob_get_contents();  
ob_end_clean();  
// Вывод $GLOBALS['NAME']
```

Буферизация средствами Bitrix

```
<?  
$this->SetViewTarget('NAME');  
// Код компонента...  
$this->EndViewTarget();  
// Вывод $APPLICATION->ShowViewContent("NAME");  
?>
```

Навигация Следующая - Предыдущая статья.

Код необходимо разместить в файле result_modifier.php

```
<?  
$arResult = array(  
    $arParams["SORT_BY1"] => $arParams["SORT_ORDER1"],  
    $arParams["SORT_BY2"] => $arParams["SORT_ORDER2"],  
);  
$arResult = array("ID", "NAME", "DETAIL_PAGE_URL");  
$arResult = array (  
    "IBLOCK_ID" => $arResult["IBLOCK_ID"],  
    // Можно ограничить секцией
```

```

    // "SECTION_CODE" => $arParams["SECTION_CODE"], или "SECTION_ID" =>
$arResult["IBLOCK_SECTION_ID"],
    "ACTIVE" => "Y",
    "CHECK_PERMISSIONS" => "Y",
);
// выбирать по 1 соседу с каждой стороны от текущего
$arNavParams = array(
    "nPageSize" => 1,
    "nElementID" => $arResult["ID"],
);
$arItems = array();
$rsElement = CIBlockElement::GetList($arResult["SORT"], $arResult["FILTER"], false, $arNavParams, $arResult["SELECT"]);
$rsElement->SetUrlTemplates($arParams["DETAIL_URL"]);

while($obElement = $rsElement->GetNextElement())
    $arItems[] = $obElement->GetFields();

if (count($arItems)==3):
    $arResult["TORIGHT"] = array("NAME" => $arItems[0]["NAME"], "URL" =>
$arItems[0]["DETAIL_PAGE_URL"]);
    $arResult["TOLEFT"] = array("NAME" => $arItems[2]["NAME"], "URL" =>
$arItems[2]["DETAIL_PAGE_URL"]);
elseif (count($arItems)==2):
    if ($arItems[0]["ID"] != $arResult["ID"])
        $arResult["TORIGHT"] = Array("NAME" => $arItems[0]["NAME"], "URL" =>
$arItems[0]["DETAIL_PAGE_URL"]);
    else
        $arResult["TOLEFT"] = Array("NAME" => $arItems[1]["NAME"], "URL" =>
$arItems[1]["DETAIL_PAGE_URL"]);
endif;
// $arResult["TORIGHT"] и $arResult["TOLEFT"] массивы с информацией о соседних элементах
?>

```

Вывод в компоненте

```

<?php if (is_array($arResult["TOLEFT"])) { ?>
    <a id="pre_page" href="<?=$arResult["TOLEFT"]["URL"]?>"> <?=$arResult["TOLEFT"]["NAME"]?></a>
<?php } ?>
<?php if (is_array($arResult["TORIGHT"])) { ?>

```

```
<a id="next_page" href="<?=$arResult["TORIGHT"]["URL"]?>"><?=$arResult["TORIGHT"]["NAME"]?> >
</a>
<?php } ?>
```

Путь к картинке по id

```
<?
$arOnePhoto = CFile::GetFileArray($PHOTO);
// $arOnePhoto['SRC']
?>
```

Если пользователь имеет права администратора

```
<?php if ($USER->Isadmin()) {} ?>
```

Получение количества товаров в наличии

```
<?
$ar_res = CCatalogProduct::GetByID($arelement['ID']);
// $ar_res['QUANTITY']
?>
```

Вывод пользовательских свойств раздела

Вывод пользовательского свойства раздела типа файл

```
<?
$db_list = CIBlockSection::GetList(
    array($by => $order),
    array("IBLOCK_ID" => 8, "ID" => $arSection['ID']),
    true,
    array("UF_NAME_IMG")
);

while($ar_result = $db_list->GetNext()):
    foreach($ar_result["UF_NAME_IMG"] as $PHOTO):
        echo CFile::GetPath($PHOTO);
    endforeach;
endwhile;
?>
```

Свойства элемента по id

```
<?
$db_props = CIBlockElement::GetProperty(IBLOCK_ID, ELEMENT_ID, "sort", "asc", array());
$PROPS = array();
while($ar_props = $db_props->Fetch())
    $PROPS[$ar_props['CODE']] = $ar_props['VALUE'];
?>
```

Замена стандартного выделения слов в поиске

\\www\\bitrix\\modules\\search\\classes\\general\\search.php:317

```
$str_result = str_replace("%/^\%", "</b>", str_replace("%^\%", "<b>", $str_result));
```

Получение элементов инфоблока

```
<?
$arSelect = Array("ID", "NAME", "PREVIEW_PICTURE");
$arFilter = Array("IBLOCK_ID" => 19, "ACTIVE_DATE" => "Y", "ACTIVE" => "Y");
```

```
$res = CIBlockElement::GetList(Array(), $arFilter, false, false, $arSelect);
while($ob = $res->GetNextElement()) {
    $arFields = $ob->GetFields();
    print_r($arFields);
}
?>
```

AJAX подгрузка компонентов

```
<?
define('STOP_STATISTICS', true);
require_once($_SERVER['DOCUMENT_ROOT'].'/bitrix/modules/main/include/prolog_before.php');
$GLOBALS['APPLICATION']->RestartBuffer();
```

[ajax code]

```
?>
```

Дополнительные пункты меню

.left.menu_ext.php

```
<?
if (CModule::IncludeModule("iblock")) {
    global $APPLICATION;
    $aMenuLinksExt = $APPLICATION->IncludeComponent(
        "bitrix:menu.sections",
        "",
        Array(
            "CACHE_TIME" => "3600",
            "CACHE_TYPE" => "A",
            "DEPTH_LEVEL" => "4",
            "DETAIL_PAGE_URL" => "#SECTION_CODE##ELEMENT_CODE#",
            "IBLOCK_ID" => "3",
            "IBLOCK_TYPE" => "content",
            "ID" => $_REQUEST["ID"],
            "IS_SEF" => "Y",
            "SECTION_PAGE_URL" => "#SECTION_CODE#/",
            "SECTION_URL" => "",
```

```
        "SEF_BASE_URL" => "/services/"
    )
);
$aMenuLinks = array_merge($aMenuLinksExt, $aMenuLinks);
}
?>
```

С элементами

```
<?
if (CModule::IncludeModule("iblock")) {

    $aMenuLinksNew = Array();
    $IBLOCK_ID = 3;
    $ctlg_sections = CBlockSection::GetList(
        Array("ID" => "ASC", " ACTIVE" => "Y"),
        Array("IBLOCK_ID" => $IBLOCK_ID),
        true,
        Array('ID', 'NAME', 'CODE', 'DEPTH_LEVEL')
    );

    while ($ar_fields = $ctlg_sections->GetNext()) {

        array_push(
            $aMenuLinksNew,
            array(
                $ar_fields['NAME'],
                $ar_fields['CODE']."/",
                array(),
                array(
                    "FROM_IBLOCK" => "1",
                    "IS_PARENT" => $ar_fields['ELEMENT_CNT'],
                    "DEPTH_LEVEL" => $ar_fields["DEPTH_LEVEL"]
                ),
                ""
            )
        );

        // elements
        $arSelect = array("ID", "NAME", "CODE");
```

```

$arFilter = array(
    "IBLOCK_ID"=> $IBLOCK_ID,
    "SECTION_ID" => $ar_fields['ID'],
    "ACTIVE_DATE"=>"Y",
    "ACTIVE"=>"Y"
);
$res = CIBlockElement::GetList(array(), $arFilter, false, array("nPageSize" => 5000), $arSelect);
while ($ob = $res->GetNextElement()) {
    $arFields = $ob->GetFields();
    array_push(
        $aMenuLinksNew,
        array(
            $arFields['NAME'],
            $ar_fields['CODE']."/".$arFields['CODE']."/",
            array(),
            array(
                "FROM_IBLOCK"=>"1",
                "IS_PARENT"=>"",
                "DEPTH_LEVEL"=>"2"
            ),
            ""
        )
    );
}

}

$aMenuLinks = array_merge($aMenuLinksNew, $aMenuLinks);
}
?>

```

Image resize

```

<?
$arFilters = array(
    array(
        "name" => "watermark",
        "position" => "tr",
        "size" => "small",

```

```

        "file" => $_SERVER["DOCUMENT_ROOT"]."/waterm.png"
    )
); //если нужен водяной знак

// НЕ пропорционально
$renderImage = CFile::ResizImageGet(
    $PHOTO,
    array("width" => 200, "height" => 200),
    $resizeType = BX_RESIZE_IMAGE_EXACT,
    $bInitSizes = false,
    $arFilters
);

// пропорционально
$renderImage = CFile::ResizImageGet(
    $PHOTO,
    array("width" => 1200, "height" => 1000),
    $resizeType = BX_RESIZE_IMAGE_PROPORTIONAL,
    $bInitSizes = false,
    $arFilters
);

// $renderImage['src']
?>

```

Пример news.list с вкладками и бесконечной подгрузкой по кнопке

portfolio.php

```

<?
require_once($_SERVER['DOCUMENT_ROOT'] . "/bitrix/modules/main/include/prolog_before.php");
if (isset($_GET['idtab'])) $idtab = $_GET['idtab']; // id активного раздела (для фильтра)
?>

```

```

<?APPLICATION->IncludeComponent(
    "bitrix:news.list",
    "portfolio",
    Array(
        "ACTIVE_DATE_FORMAT" => "d.m.Y",
        "ADD_SECTIONS_CHAIN" => "N",

```

```
"AJAX_MODE" => "N",
"AJAX_OPTION_ADDITIONAL" => "",
"AJAX_OPTION_HISTORY" => "N",
"AJAX_OPTION_JUMP" => "N",
"AJAX_OPTION_STYLE" => "Y",
"CACHE_FILTER" => "N",
"CACHE_GROUPS" => "Y",
"CACHE_TIME" => "36000000",
"CACHE_TYPE" => "A",
"CHECK_DATES" => "Y",
"COMPONENT_TEMPLATE" => "portfolio",
"DETAIL_URL" => "",
"DISPLAY_BOTTOM_PAGER" => "Y", // для генерации PAGER_
"DISPLAY_DATE" => "Y",
"DISPLAY_NAME" => "Y",
"DISPLAY_PICTURE" => "Y",
"DISPLAY_PREVIEW_TEXT" => "Y",
"DISPLAY_TOP_PAGER" => "N",
"FIELD_CODE" => array(0=>"",1=>"",),
"FILTER_NAME" => "",
"HIDE_LINK_WHEN_NO_DETAIL" => "N",
"IBLOCK_ID" => "8",
"IBLOCK_TYPE" => "design",
"INCLUDE_IBLOCK_INTO_CHAIN" => "N",
"INCLUDE_SUBSECTIONS" => "Y",
"MESSAGE_404" => "",
"NEWS_COUNT" => "6",
"PAGER_BASE_LINK_ENABLE" => "N",
"PAGER_DESC_NUMBERING" => "N",
"PAGER_DESC_NUMBERING_CACHE_TIME" => "36000",
"PAGER_SHOW_ALL" => "N",
"PAGER_SHOW_ALWAYS" => "N",
"PAGER_TEMPLATE" => ".default",
"PAGER_TITLE" => "Новости",
"PARENT_SECTION" => $dtab, // фильтр по конкретному разделу
"PARENT_SECTION_CODE" => "",
"PREVIEW_TRUNCATE_LEN" => "",
"PROPERTY_CODE" => array(0=>"",1=>"",),
"SET_BROWSER_TITLE" => "N",
"SET_LAST_MODIFIED" => "N",
```

```
"SET_META_DESCRIPTION" => "N",
"SET_META_KEYWORDS" => "N",
"SET_STATUS_404" => "N",
"SET_TITLE" => "N",
"SHOW_404" => "N",
"SORT_BY1" => "ACTIVE_FROM",
"SORT_BY2" => "SORT",
"SORT_ORDER1" => "DESC",
"SORT_ORDER2" => "ASC"
)
);?>
```

template.php

```
<?
foreach($arResult["ITEMS"] as $arItem):
    $this->AddEditAction($arItem['ID'], $arItem['EDIT_LINK'], CIBlock::GetArrayByID($arItem["IBLOCK_ID"],
"ELEMENT_EDIT"));
    $this->AddDeleteAction($arItem['ID'], $arItem['DELETE_LINK'], CIBlock::GetArrayByID($arItem["IBLOCK_ID"],
"ELEMENT_DELETE"), array("CONFIRM" => GetMessage('CT_BNL_ELEMENT_DELETE_CONFIRM')));
    ?>
    <div id="<?=$this->GetEditAreaId($arItem['ID']);?>"><?=$arItem["NAME"]?></div>
<?
endforeach;
?>
```

index.php с компонентом

```
<?
CModule::IncludeModule("iblock");
$newsCounter = CIBlockElement::GetList(array(), array("IBLOCK_ID" => 8, "ACTIVE" => "Y"), array());
?>
<div class="ptabs ptabs_active" data-id="" data-count="<?=$newsCounter?>">Bce</div>
<?
$ptabs = CIBlockSection::GetList(array($by=>$order), array('IBLOCK_ID' => 8, 'GLOBAL_ACTIVE' => 'Y'), true);
while ($ar_result = $ptabs->GetNext()) {
    $newsCounter = CIBlockElement::GetList(
        array(), array("IBLOCK_ID" => 8, "ACTIVE" => "Y", "SECTION_ID"=>$ar_result['ID']), array()
    );
?>
```

```
<div class="ptabs" data-id="<?=$ar_result['ID']?>" data-count="<?=$newsCounter?>">
  <?=$ar_result['NAME']?> (id:<?=$ar_result['ID']?>)
</div>
<?
}
?>
```

```
<div>
  <div class="potfolio__list"><?$APPLICATION->IncludeFile("/portfolio.php");?></div>
  <a href="#" class="show-more">Показать еще</a>
</div>
```

script.js

```
<script>
$(document).ready(function(){
  function checkCount(cnt){
    if (currentPage * 6 >= cnt) {
      $('.show-more').hide();
    } else {
      $('.show-more').show();
    }
  }
}
var currentPage = 1;
var path = '/portfolio.php?ajax=Y';
var count = $('.ptabs_active').data('count');
checkCount(count);
$('.ptabs').on('click', function() {
  $('.preloader').fadeIn(0);
  $('.ptabs').removeClass('ptabs_active');
  $(this).addClass('ptabs_active');
  var idtab = $(this).data('id');
  $.get(path, {idtab: idtab}, function(data) {
    $('.potfolio__list').html(data);
    count = $('.ptabs_active').data('count');
    currentPage = 1;
    checkCount(count);
    setTimeout(function() {
      $('.preloader').fadeOut(100);
    }, 500);
```

```

    });
});

$('.show-more').click(function(e) {
    $('.preloader').fadeIn(0);
    var idtab = $('.ptabs_active').data('id');
    $.get(path, {PAGEN_1: ++currentPage, idtab:idtab}, function(data){
        $('.potfolio__list').append(data);
        checkCount(count);
        setTimeout(function(){
            $('.preloader').fadeOut(100);
        }, 500);
    });
    e.preventDefault();
});
}
</script>

```

Если нет возможности узнать общее кол-во (например умный фильтр со scu) проще скрывать по условию:

```

<?
if ($arResult["NAV_RESULT"]->nEndPage > 1 && $arResult["NAV_RESULT"]->NavPageNomer <
$arResult["NAV_RESULT"]->nEndPage):
?>
Показать еще
<?endif;?>

```

Получить поля SEO для элемента

```

<?
CModule::IncludeModule("iblock");
$propValues = new \Bitrix\Iblock\InheritedProperty\ElementValues(
    $arCurElem["IBLOCK_ID"], // ID инфоблока
    $arCurElem["ID"] // ID элемента
);
$arElMetaProp = $propValues->getValues();
echo '<pre>'; print_r ($arElMetaProp); echo '</pre>';
?>

```

Поля SEO для раздела

```
<?
CModule::IncludeModule("iblock");

$rsSection = CIBlockSection::GetList(
    array(),
    array(
        "IBLOCK_ID"=>$arParams['IBLOCK_ID'],
        "ACTIVE"=>"Y",
        "=CODE"=>$arParams["SECTION_CODE"]
    ),
    false
);

if($arSection = $rsSection->GetNext()){

    $ipropValues = new \Bitrix\Iblock\InheritedProperty\SectionValues(
        $arSection["IBLOCK_ID"],
        $arSection["ID"]
    );
    $arSection["IPROPERTY_VALUES"] = $ipropValues->getValues();
    echo "<pre>"; print_r($arSection); echo '</pre>';
}
?>
```

Получение всего списка highload блоков

```
<?
use Bitrix\Highloadblock\HighloadBlockTable as HLBT;
$MY_HL_BLOCK_ID = 3;
CModule::IncludeModule('highloadblock');
function GetEntityDataClass($HIBlockId) {
    if (empty($HIBlockId) || $HIBlockId < 1) return false;
    $hblock = HLBT::getById($HIBlockId)->fetch();
    $entity = HLBT::compileEntity($hblock);
    $entity_data_class = $entity->getDataClass();
    return $entity_data_class;
}
```

```

}
$entity_data_class = GetEntityDataClass($MY_HL_BLOCK_ID);
$rsData = $entity_data_class::getList(array(
    'select' => array('*')
));
while ($el = $rsData->fetch()) {
    print_r($el);
}
?>

```

или

```

<?
if (CModule::IncludeModule('highloadblock')) {
    $Date = date(CDatabase::DateFormatToPHP(FORMAT_DATETIME));
    $arHLBlock = Bitrix\\Highloadblock\\HighloadBlockTable::getByld(3)->fetch();
    $obEntity = Bitrix\\Highloadblock\\HighloadBlockTable::compileEntity($arHLBlock);
    $strEntityDataClass = $obEntity->getDataClass();
    $resData = $strEntityDataClass::getList(array(
        'select' => array('*'),
    ));
    while ($arItem = $resData->Fetch()) {
        echo "<pre>"; print_r($arItem); echo "</pre>";
    }
}
?>

```

Добавление номера страницы в title

каталог/section.php

```

<?
CModule::IncludeModule("iblock");
$arSection = array();
if ($arResult["VARIABLES"]["SECTION_ID"]>0) {
    $arFilter = array('IBLOCK_ID'=>$arParams["IBLOCK_ID"], 'GLOBAL_ACTIVE'=>'Y', "ID" =>
$arResult["VARIABLES"]["SECTION_ID"]);
} elseif(strlen(trim($arResult["VARIABLES"]["SECTION_CODE"]))>0) {
    $arFilter = array('IBLOCK_ID'=>$arParams["IBLOCK_ID"], 'GLOBAL_ACTIVE'=>'Y', "=CODE" =>
$arResult["VARIABLES"]["SECTION_CODE"]);

```

```

}
$db_list = CIBlockSection::GetList(array(), $arFilter, true, array("ID", "NAME", "IBLOCK_ID", "DEPTH_LEVEL",
"IBLOCK_SECTION_ID", "CODE"));

while($section = $db_list->GetNext())
    $arSection = $section;

$arSelect = array("ID");

if ($arParams["LIST_BROWSER_TITLE"]) $arSelect = array_merge($arSelect,
(array)$arParams["LIST_BROWSER_TITLE"]);
if ($arParams["LIST_META_KEYWORDS"]) $arSelect = array_merge($arSelect,
(array)$arParams["LIST_META_KEYWORDS"]);
if ($arParams["LIST_META_DESCRIPTION"]) $arSelect = array_merge($arSelect,
(array)$arParams["LIST_META_DESCRIPTION"]);

$arSectionFull = CIBlockSection::GetList(
    array(),
    array("ID" => $arSection["ID"], "IBLOCK_ID"=> $arSection["IBLOCK_ID"]),
    false,
    $arSelect,
    false
)->GetNext();

if ($_GET["PAGEN_1"]) $page = GetMessage('PAGE').$_GET["PAGEN_1"];

if ($arParams["LIST_BROWSER_TITLE"] && $arSectionFull[$arParams["LIST_BROWSER_TITLE"]])
$APPLICATION->SetPageProperty("title", $arSectionFull[$arParams["LIST_BROWSER_TITLE"]].$page);

// new title
$iPropValues = new \Bitrix\Iblock\InheritedProperty\SectionValues(
    $arSection["IBLOCK_ID"],
    $arSection["ID"]
);

$arSection["IPROPERTY_VALUES"] = $iPropValues->getValues();
$APPLICATION->SetPageProperty("title", $arSection["IPROPERTY_VALUES"]["SECTION_META_TITLE"].$page);
?>

```

Детальная

```
<?
if(CModule::IncludeModule("iblock")){
    $propValues = new \Bitrix\Iblock\InheritedProperty\ElementValues(
        $arParams['IBLOCK_ID'],
        $ElementID
    );

    $arEIMetaProp = $propValues->getValues();

    $title = $arEIMetaProp["ELEMENT_META_TITLE"] ? $arEIMetaProp["ELEMENT_META_TITLE"] : $APPLICATION-
>GetTitle();
    $description = $arEIMetaProp["ELEMENT_META_DESCRIPTION"] ?
$arEIMetaProp["ELEMENT_META_DESCRIPTION"] : $APPLICATION->GetProperty("description");

    if($_GET["PAGEN_2"]){
        $APPLICATION->SetPageProperty('title', $title.". Страница ".$_GET["PAGEN_2"].".");
        $APPLICATION->SetPageProperty('description', $description.". Страница ".$_GET["PAGEN_2"].".");
    }
}
?>
```

HTTPS

```
dbconn.php
$_SERVER["HTTPS"] = "On";
```

CAPTCHA

template.php

```
<?
$code = $APPLICATION->CaptchaGetCode();
?>

<input id="captcha_word" name="captcha_word" type="text" placeholder="Введите слово на картинке">
<input name="captcha_code" value="<?=$code?>" type="hidden">
```

Проверка

```
<?
if (!$APPLICATION->CaptchaCheckCode($_POST["captcha_word"], $_POST["captcha_code"])) {
    // Неправильное значение
} else {
    // Правильное значение
}
?>
```

Пример обновления

```
<div class="captcha__update">обновить картинку</div>
<script>
$('.captcha__update').click(function(){
    $.getJSON('<?=$path?>/reload_captcha.php', function(data) {
        $('.captcha__img').attr('src','/bitrix/tools/captcha.php?captcha_sid='+data);
        $('.captcha__validate').val(data);
    });
});
</script>
```

reload_captcha.php

```
<?
echo json_encode($APPLICATION->CaptchaGetCode());
?>
```

Вывод переменных до их определения

Добавляем ссылку в h1 в шаблоне компонента header.php:

```
<?$APPLICATION->ShowViewContent('news');?><!-- content -->
```

Добавляем в шаблон компонента:

```
<?$this->SetViewTarget('news');?>
<!-- content -->
<?$this->EndViewTarget();?>
```

Генерация url с заменой параметров и сохранением старых

```
<?=$APPLICATION->GetCurPageParam ('sort=price&method=asc', array('sort', 'method'))?>
```

Добавление полей к шаблону письма

```
<?
```

```
AddEventHandler("sale", "OnOrderNewSendEmail", "bxModifySaleMails");
```

```
function bxModifySaleMails($orderId, &$eventName, &$arFields)
```

```
{
```

```
    $arOrder = CSaleOrder::GetByID($orderId);
```

```
    $order_props = CSaleOrderPropsValue::GetOrderProps($orderId);
```

```
    $city_name = "";
```

```
    $address = "";
```

```
    while ($arProps = $order_props->Fetch()){
```

```
        if ($arProps["CODE"] == "PHONE") $phone = htmlspecialchars($arProps["VALUE"]);
```

```
        if ($arProps["CODE"] == "INDEX") $index = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "ADDRESS") $address = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "LOCATION") {
```

```
            $arLocs = CSaleLocation::GetByID($arProps["VALUE"]);
```

```
            $country_name = $arLocs["COUNTRY_NAME_ORIG"];
```

```
            $city_name = $arLocs["CITY_NAME_ORIG"];
```

```
        }
```

```
        /* юр лицо */
```

```
        if ($arProps["CODE"] == "COMPANY") $company = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "COMPANY_ADR") $company_adr = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "INN") $inn = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "KPP") $kpp = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "CONTACT_PERSON") $contact_person = $arProps["VALUE"];
```

```
        if ($arProps["CODE"] == "FAX") $fax = $arProps["VALUE"];
```

```
    }
```

```

// служба доставки
$arDeliv = CSaleDelivery::GetByID($arOrder["DELIVERY_ID"]);
if ($arDeliv) $delivery_name = $arDeliv["NAME"];

// получаем название платежной системы
$arPaySystem = CSalePaySystem::GetByID($arOrder["PAY_SYSTEM_ID"]);
if ($arPaySystem) $pay_system_name = $arPaySystem["NAME"];

// добавляем новые поля в массив результатов
$arFields["ORDER_DESCRIPTION"] = $arOrder["USER_DESCRIPTION"] ? $arOrder["USER_DESCRIPTION"] : "-";
$arFields["PHONE"] = $phone ? $phone : "-";
$arFields["DELIVERY_NAME"] = $delivery_name ? $delivery_name : "-";
$arFields["PAY_SYSTEM_NAME"] = $pay_system_name ? $pay_system_name : "-";
$arFields["FULL_ADDRESS"] = $city_name." ".$address;

$arFields["COMPANY"] = $company ? $company : "-";
$arFields["COMPANY_ADR"] = $company_adr ? $company_adr : "-";
$arFields["INN"] = $inn ? $inn : "-";
$arFields["KPP"] = $kpp ? $kpp : "-";
$arFields["CONTACT_PERSON"] = $contact_person ? $contact_person : "-";
$arFields["FAX"] = $fax ? $fax : "-";

}
?>

```

Размещаем некэшируемую область в кэшируемом компоненте

result_modifier.php

```

<?if(!defined("B_PROLOG_INCLUDED") || B_PROLOG_INCLUDED!==true)die();?>
<?$this->_component->SetResultCacheKeys(array("CACHED_TPL"));?>

```

component_epilog.php

```

<?if(!defined("B_PROLOG_INCLUDED") || B_PROLOG_INCLUDED!==true)die();?>
<?
echo preg_replace_callback(
    "/#VOTE_ID_(\\d+)#/is".BX_UTF_PCRE_MODIFIER,
    create_function('$matches',

```

```

    'ob_start();
    /*тут код который не хотим кэшировать*/
    $retrunStr = @ob_get_contents();
    ob_get_clean();
    return $retrunStr;'
),
$arResult["CACHEД_TPL"]
);
?>

```

(в некэшируемом коде вместо обычного \$APPLICATION пишем \$GLOBALS["APPLICATION"], для видимости объекта внутри временной функции)

template.php

В начало

```
<?ob_start();?>
```

В конец

```
<?
```

```
$this->_component->arResult["CACHEД_TPL"] = @ob_get_contents();
```

```
ob_get_clean();
```

```
?>
```

добавляем маркер

```
<? echo "#VOTE_ID_1#"; ?>
```

(регулярка #VOTE_ID_(\d+)# для примера, удобно использовать \$matches[1] для ID и т.п.)

AJAX сравнение

header.php

```
<script>
```

```
var SITE_DIR = '<?=SITE_DIR?>';
```

```
var CATALOG_COMPARE_LIST = new Object();
```

```
CATALOG_COMPARE_LIST.defaultText = "Сравнить";
```

```
CATALOG_COMPARE_LIST.activeText = "В сравнении";
```

```
</script>
```

```
<?// сравнение
```

```
foreach ($_SESSION["CATALOG_COMPARE_LIST"][2]["ITEMS"] as $value) {
```

```
?>
<script>CATALOG_COMPARE_LIST[<?=$value["ID"]?>] = true;</script>
<?
}
?>
```

template.php

В начало

```
<?ob_start();?>
```

В конец

```
<?
```

```
$this->__component->arResult["CACHED_TPL"] = @ob_get_contents();
```

```
ob_get_clean();
```

```
?>
```

Маркер

```
<? echo "#COMPARE_FAV_ID_".$arResult['ID']."#"; ?>
```

component_epilog.php

```
<?
```

```
echo preg_replace_callback(
```

```
    "/#COMPARE_FAV_ID_(\\d+)#/is".BX_UTF_PCRE_MODIFIER,
```

```
    create_function('$matches',
```

```
        'ob_start();
```

```
    ?>
```

```
    <div class="btn-c">
```

```
        <a href="#" class="add-compare ctrl-radio
```

```
<?if(!empty($_SESSION["CATALOG_COMPARE_LIST"][2]["ITEMS"][$matches[1]]):?> ctrl-
```

```
radio_checked<?endif?>" data-id="<?=$matches[1]?>" >
```

```
    <i class="ctrl-radio__ico"></i>
```

```
    <span class="add-compare__ttl">
```

```
        <?if (!empty($_SESSION["CATALOG_COMPARE_LIST"][2]["ITEMS"][$matches[1]]):?>
```

```
            В сравнении
```

```
        <?else:?>
```

```
            Сравнить
```

```
        <?endif?>
```

```
    </span>
```

```
    </a>
```

```
</div>
```

```
<?
$returnStr = @ob_get_contents();
ob_get_clean();
return $returnStr;
),
$arResult["CACHED_TPL"]);
?>
```

result_modifier.php

```
<?if(!defined("B_PROLOG_INCLUDED") || B_PROLOG_INCLUDED!==true)die();?>
<?$this->__component->SetResultCacheKeys(array("CACHED_TPL"));?>
```

script.js

```
<script>
// compare
$('body').on('click', '.add-compare', function(e){
    e.preventDefault();

    var compareButton = $(this),
        product_id = compareButton.data('id');

    if (CATALOG_COMPARE_LIST[product_id]) {
        compareButton.removeClass('ctrl-radio_checked').find('.add-
compare__ttl').text(CATALOG_COMPARE_LIST.defaultText);
        $.ajax({
            url: SITE_DIR + 'catalog/?action=DELETE_FROM_COMPARE_LIST&id=' + product_id,
            type: 'post',
            success: function (data) {
                $(''.bx_catalog-compare-list').replaceWith(data);
            }
        });
        delete CATALOG_COMPARE_LIST[product_id];
    } else {
        compareButton.addClass('ctrl-radio_checked').find('.add-
compare__ttl').text(CATALOG_COMPARE_LIST.activeText);
        $.ajax({
            url: SITE_DIR + 'catalog/?action=ADD_TO_COMPARE_LIST&id=' + product_id,
            type: 'post',
```

```

        success: function (data) {
            $('bx_catalog-compare-list').replaceWith(data);
        }
    });
    CATALOG_COMPARE_LIST[product_id] = true;
}
});

$('body').on('click', 'bx_catalog_compare_form [data-id]', function() {
    var delId = $(this).data('id');
    delete CATALOG_COMPARE_LIST[delId];
    $('add-compare[data-id='+delId+']').removeClass('ctrl-radio_checked').find('add-
compare__ttl').text(CATALOG_COMPARE_LIST.defaultText);
});
</script>

```

sections.php

```

<?
if ($_REQUEST["action"]=="ADD_TO_COMPARE_LIST" ||
$_REQUEST["action"]=="DELETE_FROM_COMPARE_LIST") {
    $APPLICATION->RestartBuffer();
    if ($arParams["USE_COMPARE"]=="Y") {
        $APPLICATION->IncludeComponent(
            "bitrix:catalog.compare.list",
            "",
            array(
                "IBLOCK_TYPE" => $arParams["IBLOCK_TYPE"],
                "IBLOCK_ID" => $arParams["IBLOCK_ID"],
                "NAME" => $arParams["COMPARE_NAME"],
                "DETAIL_URL" => $arResult["FOLDER"].$arResult["URL_TEMPLATES"]["element"],
                "COMPARE_URL" => $arResult["FOLDER"].$arResult["URL_TEMPLATES"]["compare"],

            ),
            $component
        );
    }
    die();
}
?>

```

AJAX избранное

header.php

```
<script>
    var SITE_DIR = '<?=SITE_DIR?>';
    var DELAY_LIST = new Object();
</script>
```

template.php

В начало

```
<?ob_start();?>
```

В конец

```
<?
```

```
$this->_component->arResult["CACHED_TPL"] = @ob_get_contents();
```

```
ob_get_clean();
```

```
?>
```

Маркер

```
<? echo "#COMPARE_FAV_ID_".$arResult['ID']."#"; ?>
```

component_epilog.php

```
<?
```

```
echo preg_replace_callback(
```

```
    "/#COMPARE_FAV_ID_(\\d+)#/is".BX_UTF_PCRE_MODIFIER,
```

```
    create_function('$matches', 'ob_start();
```

```
?>
```

```
<? global $favourites; ?>
```

```
<a href="#" class="b-popular-tiles__fav btn-fav add-delay<?
```

```
    if (in_array($matches[1], $favourites)):
```

```
        ?> btn-fav_active js-check_checked<?
```

```
    endif;
```

```
?>" data-id="<?=$matches[1]?>">
```

```
    <i class="icon png-icons btn-fav__ico"></i>
```

```
    <span class="add-delay__ttl">
```

```
        <?if (in_array($matches[1], $favourites)):?>
```

```
            В избранном
```

```
        <?else:?>
            В избранное
        <?endif?>
    </span>
</a>
<?
$returnStr = @ob_get_contents();
ob_get_clean();
return $returnStr;'),
$arResult["CACHED_TPL"]);
?>
```

result_modifier.php

```
<?if(!defined("B_PROLOG_INCLUDED") || B_PROLOG_INCLUDED!==true)die();?>
<?$this->_component->SetResultCacheKeys(array("CACHED_TPL"));?>
```

script.js

```
<script>
```

```
$('#body').on('click', '.add-delay', function(e){
    e.preventDefault();

    var delayButton = $(this),
        product_id = delayButton.data('id'),
        productPrice = parseFloat(delayButton.parents('.product-item').find('.cur-price').text().replace(/ /g, ""));

    if (DELAY_LIST[product_id]) {
        delayButton.removeClass('btn-fav_active js-check_checked').find('.add-delay__ttl').html('В избранное');
        $.post(SITE_DIR + 'ajax/addDelay.php?action=DELAY_DELETE&id=' + product_id);
        delete DELAY_LIST[product_id];
    } else {
        delayButton.addClass('btn-fav_active js-check_checked').find('.add-delay__ttl').html('В избранном');
        $.post(SITE_DIR + 'ajax/addDelay.php?action=DELAY_ADD&id=' + product_id + '&PRICE=' +
productPrice);
        DELAY_LIST[product_id] = true;
    }

    $('#head-favourite__info').html(Object.keys(DELAY_LIST).length);
```

```
});

</script>

ajax/addDelay.php

<?
require_once($_SERVER["DOCUMENT_ROOT"] . "/bitrix/modules/main/include/prolog_before.php");

if (CModule::IncludeModule("sale") && CModule::IncludeModule("catalog") && CModule::IncludeModule("iblock"))
{

$productID = htmlspecialchars($_REQUEST["id"]);
$productPrice = htmlspecialchars($_REQUEST["PRICE"]);

if ($_REQUEST['action'] == 'DELAY_ADD') {
    $res = CIBlockElement::GetList(Array(), array("ID" => $productID), false, Array(), array("NAME"));
    if ($ob = $res->Fetch())
        $name = $ob['NAME'];

    // $ar_res = CPrice::GetBasePrice($productID);

    $arFields = array(
        "PRODUCT_ID" => $productID,
        "PRICE" => $productPrice,
        "CURRENCY" => 'RUB',
        // "PRICE" => $ar_res["PRICE"],
        // "CURRENCY" => $ar_res["CURRENCY"],
        "QUANTITY" => 1,
        "LID" => LANG,
        "DELAY" => "Y",
        "NAME" => $name,
    );

    CSaleBasket::Add($arFields);
}

if ($_REQUEST['action'] == 'DELAY_DELETE') {
    $arFilter = array(
        "FUSER_ID" => CSaleBasket::GetBasketUserID(),
```

```

"LID" => SITE_ID,
"ORDER_ID" => "NULL",
"DELAY" => "Y",
"PRODUCT_ID" => $productID,
);

$dbBasketItems = CSaleBasket::GetList(array(), $arFilter, false, false, array("ID"));
if ($arItems = $dbBasketItems->Fetch()) {
    CSaleBasket::Delete($arItems['ID']);
}
}

$cntDelay = array();
$dbBasketItems = CSaleBasket::GetList(array(), array("FUSER_ID" => CSaleBasket::GetBasketUserID(), "LID"
=> SITE_ID, "ORDER_ID" => "NULL", "DELAY" => "Y"), false, false, array("ID"));
while ($arItems = $dbBasketItems->Fetch()) {
    $cntDelay[] = $arItems;
}

echo count($cntDelay);

}
?>

```

Добавление доп свойств к заказу

```

catalog.element script.js
<script>
function getServices(){
    var services = 'получаем свойства';
    return services;
}

window.JCCatalogElement.prototype.SendToBasket = function()
{
    if (!this.canBuy)
        return;

    this.InitBasketUrl();
}

```

```
this.FillBasketProps();
```

```
/* доп свойства */
```

```
this.basketParams.addServices = getServices();
```

```
BX.ajax.loadJSON(
```

```
    this.basketUrl,
```

```
    this.basketParams,
```

```
    BX.proxy(this.BasketResult, this)
```

```
);
```

```
};
```

```
</script>
```

```
init.php
```

```
<?
```

```
AddEventHandler("sale", "OnOrderNewSendEmail", "bxModifySaleMails");
```

```
function bxModifySaleMails($orderId, &$eventName, &$arFields)
```

```
{
```

```
    $additional_information = '';
```

```
    $cntBasketItems = CSaleBasket::GetList(
```

```
        array(
```

```
            "NAME" => "ASC",
```

```
            "ID" => "ASC"
```

```
        ),
```

```
        array(
```

```
            "ORDER_ID" => $orderId
```

```
        ),
```

```
        false,
```

```
        false,
```

```
        array()
```

```
    );
```

```
    $itemIds = array();
```

```
    $itemNames = array();
```

```
    while ($arItems = $cntBasketItems->Fetch()){
```

```

$itemIds[] = $arItems["ID"];
$itemNames[$arItems["ID"]] = $arItems["NAME"];
}

if(!empty($itemIds)) $additional_information .= "Дополнительные услуги:<br>";

$db_res = CSaleBasket::GetPropsList(
    array(),
    array("ORDER_ID" => $orderID, "BASKET_ID"=>$itemIds,"CODE"=>"ST_ADD_SERV")
);

while ($ar_res = $db_res->Fetch()) {

    $additional_information .= $itemNames[$ar_res["BASKET_ID"]].": ".$ar_res["VALUE"]."<br>";

}

$arFields["ADDITIONAL_INFORMATION"] = $additional_information;

$arOrder = CSaleOrder::GetByID($orderID);
$order_props = CSaleOrderPropsValue::GetOrderProps($orderID);

$full_address = "";

while ($arProps = $order_props->Fetch()){
    if ($arProps["CODE"] == "COMPANY") $company = htmlspecialchars($arProps["VALUE"]);

    if ($arProps["CODE"] == "EMAIL") $email = htmlspecialchars($arProps["VALUE"]);

    if ($arProps["CODE"] == "PHONE") $phone = htmlspecialchars($arProps["VALUE"]);

    if ($arProps["CODE"] == "LOCATION") {
        $arLocs = CSaleLocation::GetByID($arProps["VALUE"]);
        if($arLocs["COUNTRY_NAME_ORIG"]) $full_address .= $arLocs["COUNTRY_NAME_ORIG"];
        if($arLocs["REGION_NAME_ORIG"]) $full_address .= ", ".$arLocs["REGION_NAME_ORIG"];
        //if($arLocs["CITY_NAME_ORIG"]) $full_address .= ", ".$arLocs["CITY_NAME_ORIG"];

        $parameters = array();
        $parameters['filter']['=CODE'] = $arLocs["CODE"];
        $parameters['filter']['NAME.LANGUAGE_ID'] = "ru";
    }
}

```

```

$parameters['limit'] = 1;
$parameters['select'] = array('LNAME' => 'NAME.NAME');
$arVal = Bitrix\Sale\Location\LocationTable::getList( $parameters )->fetch();
if($arVal[ 'LNAME' ]) $full_address .= ", ".$arVal[ 'LNAME' ];
}

if ($arParams["CODE"] == "ADDRESS") $address = $arParams["VALUE"];

if ($arParams["CODE"] == "REQUISITES") {
    $requisites = "";
    if (!empty($arParams["VALUE"])) {
        $files = explode(", ", $arParams["VALUE"]);
        foreach ($files as $file) {
            $requisites .= "<a href='".SITE_SERVER_NAME.CFile::GetPath($file)."'>Скачать</a><br>";
        }
    }
}

// тип плательщика
$db_ptype = CSalePersonType::getList(Array("SORT" => "ASC"), Array("LID"=>SITE_ID));
while ($ptype = $db_ptype->Fetch())
    if($ptype["ID"]== $arOrder["PERSON_TYPE_ID"]) $personType = $ptype["NAME"];

// название службы доставки
$arDeliv = CSaleDelivery::GetByID($arOrder["DELIVERY_ID"]);
if ($arDeliv) $delivery_name = $arDeliv["NAME"];

// название платежной системы
$arPaySystem = CSalePaySystem::GetByID($arOrder["PAY_SYSTEM_ID"]);
if ($arPaySystem) $pay_system_name = $arPaySystem["NAME"];

// добавляем новые поля в массив результатов
$arFields["ORDER_DESCRIPTION"] = "Комментарий: ".$arOrder["USER_DESCRIPTION"]."<br>";
$arFields["ORDER_P_TYPE"] = "Тип плательщика: ".$personType."<br>";
$arFields["ORDER_PHONE"] = "Телефон: ".$phone."<br>";
$arFields["ORDER_EMAIL"] = "E-mail: ".$email."<br>";
$arFields["DELIVERY_NAME"] = "Доставка: ".$delivery_name."<br>";
$arFields["PAY_SYSTEM_NAME"] = "Оплата: ".$pay_system_name."<br>";
$arFields["ADDRESS"] = "Адрес доставки: ".$address."<br>";

```

```
$arResult["FULL_ADDRESS"] = "Местоположение: ".$full_address."<br>";
$arResult["COMPANY"] = "Компания: ".$company."<br>";
$arResult["REQUISITES"] = "Реквизиты: ".$requisites."<br>";
}
?>
```

в старых версиях создаем новый компонент

/bitrix/components/site-hit/catalog.element/component.php

```
<?
if ($successfulAdd) {
    if ($_REQUEST["addServices"]) {
        $product_properties[] = array(
            "NAME" => "Дополнительные услуги",
            "CODE" => "ST_ADD_SERV",
            "VALUE" => $_REQUEST["addServices"],
            "SORT" => "1000"
        );
    }
    if (!Add2BasketByProductID($productId, $quantity, $arResult, $product_properties)) {
        if ($ex = $APPLICATION->GetException())
            $strError = $ex->GetString();
        else
            $strError = GetMessage("CATALOG_ERROR2BASKET");
        $successfulAdd = false;
    }
}
?>
```

в новых версиях создаем новый компонент и наследуем от /bitrix/modules/iblock/lib/component/base.php

```
class CatalogElementComponent extends Element
```

```
<?
protected function addProductToBasket($productId, $action)
{
    /** @global \CMain $APPLICATION */
    global $APPLICATION;

    $successfulAdd = true;
```

```

$errorMsg = "";

$quantity = 0;
$productProperties = array();
$iblockId = (int)\CIBlockElement::GetIBlockByID($productId);

if ($iblockId > 0)
{
    $productCatalogInfo = \CCatalogSku::GetInfoByIBlock($iblockId);
    if (!empty($productCatalogInfo) && $productCatalogInfo['CATALOG_TYPE'] ==
\CatalogSku::TYPE_PRODUCT)
    {
        $productCatalogInfo = false;
    }
    if (!empty($productCatalogInfo))
    {
        if ($this->arParams['ADD_PROPERTIES_TO_BASKET'] === 'Y')
        {
            $productIblockId = ($productCatalogInfo['CATALOG_TYPE'] == \CCatalogSku::TYPE_CATALOG
                ? $productCatalogInfo['IBLOCK_ID']
                : $productCatalogInfo['PRODUCT_IBLOCK_ID']
            );
            $iblockParams = $this->storage['IBLOCK_PARAMS'][$productIblockId];
            if ($productCatalogInfo['CATALOG_TYPE'] !== \CCatalogSku::TYPE_OFFERS)
            {
                if (!empty($iblockParams['CART_PROPERTIES']))
                {
                    $productPropsVar = $this->request->get($this->arParams['PRODUCT_PROPS_VARIABLE']);
                    if (is_array($productPropsVar))
                    {
                        $productProperties = \CIBlockPriceTools::CheckProductProperties(
                            $productIblockId,
                            $productId,
                            $iblockParams['CART_PROPERTIES'],
                            $productPropsVar,
                            $this->arParams['PARTIAL_PRODUCT_PROPERTIES'] === 'Y'
                        );
                        if (!is_array($productProperties))
                        {
                            $errorMsg = Loc::getMessage('CATALOG_PARTIAL_BASKET_PROPERTIES_ERROR');
                        }
                    }
                }
            }
        }
    }
}

```

```

        $successfulAdd = false;
    }
}
else
{
    $errorMsg = Loc::getMessage('CATALOG_EMPTY_BASKET_PROPERTIES_ERROR');
    $successfulAdd = false;
}
}
}
else
{
    $skuAddProps = $this->request->get('basket_props') ?: '';
    if (!empty($iblockParams['OFFERS_CART_PROPERTIES']) || !empty($skuAddProps))
    {
        $productProperties = \\CIBlockPriceTools::GetOfferProperties(
            $productId,
            $productIblockId,
            $iblockParams['OFFERS_CART_PROPERTIES'],
            $skuAddProps
        );
    }
}
}
else
{
    $errorMsg = Loc::getMessage('CATALOG_PRODUCT_NOT_FOUND');
    $successfulAdd = false;
}

if ($this->arParams['USE_PRODUCT_QUANTITY'])
{
    $quantity = (float)$this->request->get($this->arParams['PRODUCT_QUANTITY_VARIABLE']);
}

if ($quantity <= 0)
{
    $ratioIterator = \\CCatalogMeasureRatio::getList(
        array(),

```

```

        array('PRODUCT_ID' => $productId),
        false,
        false,
        array('PRODUCT_ID', 'RATIO')
    );
    if ($ratio = $ratioIterator->Fetch())
    {
        $intRatio = (int)$ratio['RATIO'];
        $floatRatio = (float)$ratio['RATIO'];
        $quantity = $floatRatio > $intRatio ? $floatRatio : $intRatio;
    }
}

if ($quantity <= 0)
{
    $quantity = 1;
}
else
{
    $errorMsg = Loc::getMessage('CATALOG_PRODUCT_NOT_FOUND');
    $successfulAdd = false;
}

$rewriteFields = $this->getRewriteFields($action);

if ($successfulAdd)
{
    /**/
    if($_REQUEST["optionsprice"]){
        $productProperties[] = array(
            "NAME" => "Сумма дополнительных опций (за ед.товара)",
            "CODE" => "PICKED_OPTIONS_PRICE",
            "VALUE" => $_REQUEST["optionsprice"]." руб.",
            "SORT" => "1000"
        );
    }
    if($_REQUEST["edge"] && $_REQUEST["edgeprice"]){
        $productProperties[] = array(
            "NAME" => "Обработка края",

```

```

        "CODE" => "PICKED_EDGE",
        "VALUE" => $_REQUEST["edge"],
        "SORT" => "1100"
    );
}
if($_REQUEST["edges"] && $_REQUEST["edgesprice"]){
    $productProperties[] = array(
        "NAME" => "Отстрочка края",
        "CODE" => "PICKED_EDGES",
        "VALUE" => $_REQUEST["edges"],
        "SORT" => "1200"
    );
}
if($_REQUEST["bracing"] && $_REQUEST["bracingprice"]){
    $productProperties[] = array(
        "NAME" => "Крепления",
        "CODE" => "PICKED_BRACING",
        "VALUE" => $_REQUEST["bracing"],
        "SORT" => "1300"
    );
}
if($_REQUEST["logo"] && $_REQUEST["logoprice"]){
    $productProperties[] = array(
        "NAME" => "Логотип",
        "CODE" => "PICKED_LOGO",
        "VALUE" => $_REQUEST["logo"],
        "SORT" => "1400"
    );
}
if($_REQUEST["extra"] && $_REQUEST["extraprice"]){
    $productProperties[] = array(
        "NAME" => "Дополнительно",
        "CODE" => "PICKED_EXTRA",
        "VALUE" => $_REQUEST["extra"],
        "SORT" => "1500"
    );
}
/* */

if (!Add2BasketByProductID($productId, $quantity, $rewriteFields, $productProperties))

```

```

    {
        if ($ex = $APPLICATION->GetException())
        {
            $errorMsg = $ex->GetString();
        }
        else
        {
            $errorMsg = Loc::getMessage('CATALOG_ERROR2BASKET');
        }

        $successfulAdd = false;
    }
}

return array($successfulAdd, $errorMsg);
}
?>

```

Добавление наценки в корзине (пример с добавлением суммы из выбранных свойств товара)

```

<?
\\Bitrix\\Main\\EventManager::getInstance()->addEventHandler(
    'sale',
    'OnBeforeSaleBasketItemSetField',
    'changePrice'
);

function changePrice(\\Bitrix\\Main\\Event $event)
{
    $name = $event->getParameter('NAME');
    $value = $event->getParameter('VALUE');
    $item = $event->getParameter('ENTITY');

    if ($name === 'PRICE') {
        $basketPropertyCollection = \\Bitrix\\Sale\\BasketPropertiesCollection::load($item);
        $basketId = $basketPropertyCollection->getBasketId();
        $db_res = CSaleBasket::GetPropsList(
            array(
                "SORT" => "ASC",

```

```
        "NAME" => "ASC"
    ),
    array("BASKET_ID" => $basketId)
);
while ($ar_res = $db_res->Fetch()) {
    if ($ar_res["CODE"] == "PICKED_OPTIONS_PRICE") $value += floatval($ar_res["VALUE"]);
}
$event->addResult(
    new \\Bitrix\\Main\\EventResult(
        \\Bitrix\\Main\\EventResult::SUCCESS, array('VALUE' => $value)
    )
);
}
}
?>
```

PHP 7

- 1) Меняем версию на хостинге
- 2) Меняем mod_php5.c в .htaccess
- 3) /bitrix/php_interface/dbconn.php

```
define("BX_USE_MYSQLI", true);
```

- 4) /bitrix/.settings.php

```
className: 'className' => '\\\\Bitrix\\\\Main\\\\DB\\\\MysqliConnection',
```

```
...
```

```
'handled_errors_types' => 4437,
```

```
'exception_errors_types' => 4437,
```

- 5) на сервере отключить вывод E_DEPRECATED

Логи

```
dbconn.php
```

```
define("LOG_FILENAME", $_SERVER["DOCUMENT_ROOT"]."/log.txt");
```

```
AddMessage2Log("Сообщение");
```

Список сработавших почтовых событий

(Настройки → Инструменты → SQL-запрос)

```
select * from b_event  
where event_name like '%form%'  
order by date_insert desc
```

Выставить права на папки и файлы Битрикс SSH

```
// Переходим в нужную директорию
cd ~/www

// Установка прав для папок
find . -type d -exec chmod 775 {} \;

// Установка прав для файлов
find . -type f -exec chmod 664 {} \;

// Если необходимо изменить владельца группы
find . -type d -exec chown bitrix:bitrix {} \;
find . -type f -exec chown bitrix:bitrix {} \;
```

Добавляем пользовательское поле ТЕХТ/HTML в 1С-Битрикс

До сих пор в Битриксе нет пользовательского свойства с HTML кодом (визуальным редактором). Устраняем это досадное недоразумение.

На просторах интернета быстро нашлось решение, правда сильно устаревшее, немного пришлось допилить напильником, и теперь всё работает.

Данный код необходимо вставить в `init.php` вашего сайта на Битриксе. Решение поддерживает PHP8.

```
//выводим пользовательское HTML поле в свойствах разделов
AddEventHandler('main', 'OnUserTypeBuildList', array('CUserTypeSectionsHtmlField', 'GetUserTypeDescription'),
5000);

class CUserTypeSectionsHtmlField {

    public static function GetUserTypeDescription() {
        return array(
            // уникальный идентификатор
            'USER_TYPE_ID' => 'sections_html_field',
            // имя класса, методы которого формируют поведение типа
            'CLASS_NAME' => 'CUserTypeSectionsHtmlField',
            // название для показа в списке типов пользовательских свойств
            'DESCRIPTION' => 'HTML/text',
            // базовый тип на котором будут основаны операции фильтра
            'BASE_TYPE' => 'string',
        );
    }

    public static function GetDBColumnType($arUserField) {
        switch (strtolower($GLOBALS['DB']->type)) {
```

```

        case 'mysql':
            return 'text';
            break;
    }
}

public static function GetSettingsHTML($arUserField = false, $arHtmlControl, $bVarsFromForm) {
    $result = "";

    return $result;
}

public static function CheckFields($arUserField, $value) {
    $aMsg = array();
    return $aMsg;
}

public static function GetEditFormHTML($arUserField, $arHtmlControl) {
    if ($arUserField["ENTITY_VALUE_ID"] < 1 && strlen($arUserField["SETTINGS"]["DEFAULT_VALUE"]) > 0)
        $arHtmlControl["VALUE"] = htmlspecialchars($arUserField["SETTINGS"]["DEFAULT_VALUE"]);
    ob_start();
    CFileMan::AddHTMLFrame($arHtmlControl["NAME"], $arHtmlControl["VALUE"], "html", "html", 200,
"N", 0, "", "", "s1");
    $b = ob_get_clean();
    return $b;
}

public static function GetEditFormHTMLMulty($arUserField, $arHtmlControl) {
    $html = 'Поле не может быть множественным!';
    return $html;
}

public static function GetFilterHTML($arUserField, $arHtmlControl) {
    $sVal = intval($arHtmlControl['VALUE']);
    $sVal = $sVal > 0 ? $sVal : "";

    return CUserTypeSectionsHtmlField::GetEditFormHTML($arUserField, $arHtmlControl);
}

public static function GetAdminListViewHTML($arUserField, $arHtmlControl) {

```

```
        return "";
    }

    public static function GetAdminListViewHTMLMulty($arUserField, $arHtmlControl) {
        return "";
    }

    public static function GetAdminListEditHTML($arUserField, $arHtmlControl) {
        return "";
    }

    public static function GetAdminListEditHTMLMulty($arUserField, $arHtmlControl) {
        return "";
    }

    public static function onsearchIndex($arUserField) {
        return "";
    }

    public static function OnBeforeSave($arUserField, $value) {
        return $value;
    }
}
```

Теперь в пользовательских типах у вас появится Text/HTML