
Bitrix Notes Documentation

Release 0.2.0

Rodion Abdurakhimov

Feb 14, 2020

1	Recipes for common tasks	3
1.1	Loop for all elements in iblock	3
1.2	Include section	3
1.3	Menu _ext example	4
1.4	Delete offers without photos	4
1.5	Update all prices	5
1.6	Detect mobile regex	5
1.7	Build table with all products	6
1.8	Read large CSV file	6
1.9	Write CSV file	7
1.10	Caching example	7
1.11	Get Iblock id by it's code	8
1.12	Admin section recipes	9
1.13	Set price extra for all products	13
1.14	Send instant message to users	13
2	Recipes for work with orders	15
2.1	Get order props	15
2.2	Copy order	16
2.3	Get basket props	18
3	Recipes for SEO tasks	19
3.1	301 Redirect from old URL	19
3.2	Find and fix sections duplicates	21
3.3	.htaccess SEO redirects	21
3.4	Fix pictures URL changing	22
3.5	Change elements code separator from '_' to '-'	22
3.6	Change sections code separator from '_' to '-'	23
4	Common event handlers	25
4.1	Minimal and maximal price props	25
4.2	Before mail sent	28
4.3	Restrict 1C from updating element fields	28
5	D7	29
5.1	ORM	29
5.2	Bitrix\Main\Application	31

5.3	Bitrix\Main\Context	31
5.4	Bitrix\Main\Request	31
5.5	Bitrix\Main\Server	32
5.6	Basket	32
5.7	Order	34
5.8	Files and directories operations	35
5.9	HTTP	37
6	Tasks with cron utility	39
6.1	Agents on cron	39
7	Useful functions	41
7.1	Dump for admin	41
7.2	Dump for admin limited by IP	42
7.3	Export array with short array syntax	42
8	Example of using PDO	43
9	Other	45
9.1	Dates on russian	45
9.2	Phinx config	45
9.3	Errors in bitrix core	46
9.4	Examples of DateTime class	46
9.5	Logging examples	47
9.6	Update 1C exchange module error	47
9.7	Reset module files	47
10	Indices and tables	49
	Index	51

Contents:

single: Recipes for common tasks

CHAPTER 1

Recipes for common tasks

This section contains scripts for common tasks often found in work with Bitrix

1.1 Loop for all elements in iblock

Getlist sorted by TIMESTAMP_X asc and limited by 1000 elements.

```
<?php
CModule::IncludeModule('iblock');
$arSelect = array('ID', 'NAME', 'CODE');
$arFilter = array('IBLOCK_ID' => 10, 'ACTIVE' => 'Y');
$res = CIBlockElement::GetList(
    array('TIMESTAMP_X' => 'ASC'),
    $arFilter,
    false,
    array('nPageSize' => 1000),
    $arSelect
);
while ($ob = $res->GetNextElement()) {
    $arFields = $ob->GetFields();
}
```

1.2 Include section

```
<?APPLICATION->IncludeFile(
    APPLICATION->GetTemplatePath("/include/logo.php"),
    Array(),
    Array("MODE"=>"php")
);?>
```

(continues on next page)

(continued from previous page)

```

<?APPLICATION->IncludeComponent (
    "bitrix:main.include",
    "sidebar",
    Array(
        "AREA_FILE_SHOW" => "sect",
        "AREA_FILE_SUFFIX" => "inc",
        "AREA_FILE_RECURSIVE" => "N",
        "EDIT_MODE" => "html",
    ),
    false,
    Array('HIDE_ICONS' => 'Y')
);?>

```

1.3 Menu _ext example

```

<?php
// .left.menu_ext.php

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

global $APPLICATION;

$aMenuLinksExt = $APPLICATION->IncludeComponent (
    "bitrix:menu.sections",
    "",
    array(
        "ID" => $_REQUEST["ELEMENT_ID"],
        "IBLOCK_TYPE" => "books",
        "IBLOCK_ID" => "30",
        "SECTION_URL" => "/e-store/books/index.php?SECTION_ID=#ID#",
        "CACHE_TIME" => "3600"
    )
);

$aMenuLinks = array_merge($aMenuLinks, $aMenuLinksExt);

```

1.4 Delete offers without photos

```

<?php
CModule::IncludeModule('iblock');
$arSelect = array('ID', 'DETAIL_PICTURE', 'PREVIEW_PICTURE');
$arFilter = array('IBLOCK_ID' => 9, 'ACTIVE' => 'Y');
$res = CIBlockElement::GetList(
    array('TIMESTAMP_X' => 'ASC'),
    $arFilter,
    false,
    array('nPageSize' => 1000),
    $arSelect
);
while ($ob = $res->GetNextElement()) {

```

(continues on next page)

(continued from previous page)

```

$arResult = $obj->GetFields();
$bNoPics = false;

if (empty($arResult['PREVIEW_PICTURE']) && empty($arResult['DETAIL_PICTURE'])) {
    $bNoPics = true;
}

if ($bNoPics) {
    $arResult = array('ID', 'PROPERTY_CML2_LINK');
    $arResult = array('IBLOCK_ID' => 13, 'PROPERTY_CML2_LINK' => $arResult['ID
    => ']);
    $res = CIBlockElement::GetList(
        array('TIMESTAMP_X' => 'ASC'),
        $arResult,
        false,
        array('nPageSize' => 1),
        $arResult
    );
    while ($obj = $res->GetNextElement()) {
        $arResult = $obj->GetFields();
        CIBlockElement::Delete($arResult['ID']);
    }
}
}

```

1.5 Update all prices

```

<?php
CModule::IncludeModule('catalog');

$db_res = CPrice::GetList(
    array(),
    array(
        'CURRENCY' => 'RUB'
    )
);
while ($ar_res = $db_res->Fetch()) {
    CPrice::Update($ar_res['ID'], ['CURRENCY' => 'UAH']);
}

```

1.6 Detect mobile regex

```

<?php
$useragent=$_SERVER['HTTP_USER_AGENT'];
if(preg_match('/android.+mobile|avantgo|bada\
    =>|blackberry|blazer|compal|elaine|fennec|hiptop|iemobile|ip(hone|od)|iris|kindle|lge_
    =>|maemo|midp|mmp|opera m(ob|in)i|palm( os)?|phone|p(ixi|re)\
    =>|plucker|pocket|psp|symbian|treo|up\.(browser|link)|vodafone|wap|windows_
    =>|(ce|phone)|xda|xiino/i',$useragent)||preg_match('/1207|6310|6590|3gso|4thp|50[1-
    =>6]|i770s|802s|a wa|abac|ac(er|oo|s\
    =>)|ai(ko|rn)|al(av|ca|co)|amoi|an(ex|ny|yw)|aptu|ar(ch|go)|as(te|us)|attw|au(di|\
    =>m|r |s )|avan|be(ck|ll|nq)|bi(lb|rd)|bl(ac|az)|br(e|v)w|bumb|bw\-(n|u)|c55\
    =>|capi|ccwa|cdm\-(cell|chtm|clde|cmd\-(co(mp|nd)|craw|da(it|ll|ng)|dbte|dc\
    =>|s|l|si|si|si|do(c|p)|ods(12|1\d)|el(49|ai)|em(l2|ul)|er(ic|k0)|esl8|ez([4-
    =>7]|o|os|wa|ze)|fetc|fly(\-|_)|g1 u|g560|gene|gf\5|g\mo|go\
    =>w|od)|gr(ad|un)|haie|hcit|hd\-(m|p|t)|hei\-(hi|pt|ta)|hp( i|ip)|hs\-c|ht(c\-\|_|
    =>|a|g|p|s|t)|tp|h(u|aw|tc)|i\-(20|go|ma)|i230|iac( |-\|\\
    =>)|libro|idea|iq01|ikom|imlk|inno|ipaq|iris|ja(t|v)a|jbro|jemu|jigs|kddi|keji|kqt( |\\

```

(continues on next page)

(continued from previous page)

```
header('Location: http://m.site.ru '.$APPLICATION->GetCurPageParam());
}
```

1.7 Build table with all products

```
<?php
require($_SERVER["DOCUMENT_ROOT"] . '/bitrix/modules/main/include/prolog_before.php');
\Bitrix\Main\Loader::includeModule('iblock');

$arSelect = array('ID', 'NAME', 'CODE', 'ACTIVE');
$arFilter = array('IBLOCK_ID' => CATALOG_IBLOCK, 'ACTIVE' => 'Y');
$res = CIBlockElement::GetList(
    array(),
    $arFilter,
    false,
    false,
    $arSelect
);
?>
<table>
    <tr>
        <td>ID</td>
        <td></td>
        <td> </td>
        <td></td>
    </tr>
    <?php while ($arFields = $res->GetNext()) {
        ?>
        <tr>
            <td><?= $arFields['ID'] ?></td>
            <td><?= $arFields['NAME'] ?></td>
            <td><?= $arFields['CODE'] ?></td>
            <td><?= $arFields['ACTIVE'] ?></td>
        </tr>
    <?php
    }
    ?>
</table>
```

1.8 Read large CSV file

```
$ composer require league/csv:^8.0
```

```
<?php
require_once '/vendor/autoload.php';
use League\Csv\Reader;

set_time_limit(0);
ini_set('display_errors', 1);

$importStep = 1000;
```

(continues on next page)

(continued from previous page)

```

$importStart = 0;

$csv = Reader::createFromPath(__DIR__.'/your.csv');
$csv->setDelimiter('|')
    ->setOffset($importStart)
    ->setLimit($importStep);

$csv->each(function ($row, $rowOffset) {
    var_dump($row);

    return true;
});

```

1.9 Write CSV file

```
$ composer require league/csv:^8.0
```

```

<?php
use Bitrix\Main\Application;
use League\Csv\Writer;

$root = Application::getDocumentRoot();

$writer = Writer::createFromPath($root.'/upload/your.csv', 'w');
$writer->setDelimiter(';');
$writer->insertOne(['col1', 'col2']);

foreach ($data as $item) {
    $writer->insertOne([$item['field1'], $item['field2']]);
}

```

1.10 Caching example

1.10.1 D7 style

```

<?php
use \Bitrix\Main\Data\Cache;

$cache = Cache::createInstance();
if ($cache->initCache(7200, "cache_key")) {
    $vars = $cache->getVars();
}
elseif ($cache->startDataCache()) {
    $cache->endDataCache(array("key" => "value"));
}

//public function initCache($TTL, $uniqueString, $initDir = false, $baseDir = "cache")

```

Managed cache

```
<?php
$cache = \Bitrix\Main\Application::getInstance()->getManagedCache();

if ($cache->read($cacheTtl, $cacheId)) {
    $vars = $cache->get($cacheId);
} else {
    $cache->set($cacheId, array("key" => $value));
}

//clear by tag
$cache->clean($cacheId);
```

1.10.2 Old style

```
<?php
$cache = new CPHPCache;
$cache_time = 3600;
$cache_id = 'cache_id';
if ($cache->InitCache($cache_time, $cache_id, '/' . SITE_ID . '/cache/path/')) {
    $arVars = $cache->GetVars();
    return $arVars['DATA'];
} else {
    $cache->StartDataCache($cache_time, $cache_id);

    //get your data
    $data = 'some_text';

    if ($data) {
        $cache->EndDataCache(['DATA' => $data]);
        return $data;
    } else {
        return false;
    }
}
```

1.11 Get Iblock id by it's code

```
<?php
/**
 * Returns Iblock ID by it's code
 *
 * @param $iblockCode
 * @return bool
 */
function getIblockIdByCode($iblockCode)
{
    $cache = new CPHPCache;
    $cache_time = 3600;
    $cache_id = 'get_id_'.$iblockCode;
    if ($cache->InitCache($cache_time, $cache_id, '/' . SITE_ID . '/iblock/helper/')) {
        $arVars = $cache->GetVars();
        return $arVars['DATA']['ID'];
    }
```

(continues on next page)

(continued from previous page)

```

    } else {
        $cache->StartDataCache($cache_time, $cache_id);

        $arIblock = IblockTable::getList([
            'filter' => ['CODE' => $iblockCode],
            'select' => ['ID']
        ]->fetch());

        if ($arIblock) {
            $cache->EndDataCache(['DATA' => $arIblock]);
            return $arIblock['ID'];
        } else {
            return false;
        }
    }
}

```

1.12 Admin section recipes

1.12.1 Add custom page in admin section

Add new folder under your local folder

```
$ mkdir local/bitrix.admin
```

Update `urlrewrite.php` file - add new rule

```

<?
$arUrlRewrite = array(
    // ...
    array(
        'CONDITION' => '#^/bitrix/admin/(.*)#',
        'RULE' => '/local/bitrix.admin/$1',
        'ID' => '',
        'PATH' => '',
    ),
    // ...
);

```

Admin page example. Place it in `local/bitrix.admin` folder

```

<?php
require_once $_SERVER['DOCUMENT_ROOT'].'/bitrix/modules/main/include/prolog_admin_
before.php';
require_once $_SERVER['DOCUMENT_ROOT'].'/bitrix/modules/sale/include.php';

/** @var CMain $APPLICATION */
$saleModulePermissions = $APPLICATION->GetGroupRight('sale');

/** @var CUser $USER */
if (!$USER->IsAdmin()) {
    $APPLICATION->AuthForm(GetMessage('ACCESS_DENIED'));
}

```

(continues on next page)

(continued from previous page)

```

IncludeModuleLangFile(__FILE__);

$APPLICATION->SetTitle(' eSputnik');

require $_SERVER['DOCUMENT_ROOT'].'/bitrix/modules/main/include/prolog_admin_after.php
↪';
?>
<?php
$moduleName = 'esputnik';
$settings = array(
    array(
        'LABEL' => '',
        'FORM_NAME' => 'esp-login',
        'OPTION_NAME' => 'LOGIN',
    ),
    array(
        'LABEL' => '',
        'FORM_NAME' => 'esp-password',
        'TYPE' => 'password',
        'OPTION_NAME' => 'PASSWORD',
    ),
);
if (isset($_REQUEST['save']) && strlen($_REQUEST['save']) > 0) {
    foreach ($settings as $setting) {
        if (isset($_REQUEST[$setting['FORM_NAME']])) {
            COption::SetOptionString($moduleName, $setting['OPTION_NAME'], $_REQUEST[
↪$setting['FORM_NAME']]);
        }
    }
}
?>
<style>
    .form-group {
        display: block;
        margin: 3px;
    }
    .form-group > label {
        display: inline-block;
        min-width: 200px;
    }
</style>
<form method="POST" action="<?=$APPLICATION->GetCurUri() ?>">
    <?php foreach ($settings as $setting) :?>
        <div class="form-group" title="<?=$setting['DESCRIPTION'] ?>">
            <label for="<?=$setting['FORM_NAME'] ?>"><?=$setting['LABEL'] ?>:</
↪label>

            <input type="<?=$setting['TYPE']?:'text' ?>"
                name="<?=$setting['FORM_NAME'] ?>" id="<?=$setting['FORM_NAME'] ?
↪>"

                value="<?=COption::getOptionString($moduleName, $setting[
↪'OPTION_NAME'] ?>" ?>"
                placeholder="<?=$setting['DESCRIPTION'] ?>"

                >
            </div>
    <?php endforeach; ?>
<div class="form-group">

```

(continues on next page)

(continued from previous page)

```

        <input type="submit" class="adm-btn" name="save" title="" value="">
    </div>
</form>
<?php
require $_SERVER['DOCUMENT_ROOT'].'/bitrix/modules/main/include/epilog_admin.php';

```

Add new admin menu item

```

AddEventHandler('main', 'OnBuildGlobalMenu', 'addMenuItem');

function OnBuildGlobalMenuHandler(&$adminMenu, &$moduleMenu)
{
    global $USER;
    if($USER->IsAdmin())
    {
        $moduleMenu[] = array(
            'parent_menu' => 'global_menu_store',
            'sort' => 10,
            'url' => 'your_new_page.php?lang='.LANG,
            'text' => 'your_new_page',
            'title' => 'your_new_page',
            'icon' => 'fav_menu_icon',
            'page_icon' => 'fav_menu_icon',
            'items_id' => 'menu_order',
        );
    }
}

```

1.12.2 Custom order page

Info bar

```

<?php
$eventManager = \Bitrix\Main\EventManager::getInstance();
$eventManager->addEventHandler('sale', 'onSaleAdminOrderInfoBlockShow',
    => 'onSaleAdminOrderInfoBlockShow');

function onSaleAdminOrderInfoBlockShow(\Bitrix\Main\Event $event)
{
    $order = $event->getParameter("ORDER");
    //$basket = $event->getParameter("ORDER_BASKET");

    $propertyCollection = $order->getPropertyCollection();
    $location = \Bitrix\Sale\Location\Admin\LocationHelper::getLocationStringByCode(
        => $propertyCollection->getDeliveryLocation()->getValue());

    return new \Bitrix\Main\EventResult(
        \Bitrix\Main\EventResult::SUCCESS,
        array(
            array(
                'TITLE' => ':',
                'VALUE' => $location,
                'ID' => 'location'
            ),
        )
    );
}

```

(continues on next page)

(continued from previous page)

```
);  
}
```

Tabs

```
<?php  
$eventManager = \Bitrix\Main\EventManager::getInstance();  
$eventManager->addEventHandler("main", "OnAdminSaleOrderView", array(  
    ↪ "MyAdminOrderFormTabs", "onInit"));  
  
class MyAdminOrderFormTabs  
{  
    function onInit()  
    {  
        return array(  
            "TABSET" => "MyTabs",  
            "GetTabs" => array("MyAdminOrderFormTabs", "getTabs"),  
            "ShowTab" => array("MyAdminOrderFormTabs", "showTabs"),  
            "Action" => array("MyAdminOrderFormTabs", "onSave"),  
            "Check" => array("MyAdminOrderFormTabs", "onBeforeSave"),  
        );  
    }  
  
    function getTabs($args)  
    {  
        return array(  
            array(  
                "DIV" => "myTab1",  
                "TAB" => "New Tab",  
                "TITLE" => "New Tab Title",  
                "SORT" => 1  
            )  
        );  
    }  
  
    function showTabs($tabName, $args, $varsFromForm)  
    {  
        if ($tabName == "myTab1") {  
            echo "Tab Content";  
        }  
    }  
  
    function onBeforeSave($args)  
    {  
        return true;  
    }  
  
    function onSave($args)  
    {  
        return true;  
    }  
}
```


1.13 Set price extra for all products

```
<?php
function setGlobalPriceExtra()
{
    $arSelect = ['ID', 'NAME'];
    $arFilter = ['IBLOCK_ID' => OFFERS_IBLOCK_ID,];
    $resOffer = CIBlockElement::GetList(
        [],
        $arFilter,
        false,
        ['nPageSize' => 5000],
        $arSelect
    );
    $el = new CIBlockElement;
    global $USER;

    \Bitrix\Main\Diag\Debug::dump($resOffer->SelectedRowCount());
    \Bitrix\Main\Diag\Debug::dump('=====');

    while ($arFieldsOffer = $resOffer->GetNext()) {
        $arFields = array(
            "PRODUCT_ID" => $arFieldsOffer['ID'],
            "CATALOG_GROUP_ID" => 4,
            "EXTRA_ID" => 1,
            "CURRENCY" => "RUB",
        );

        $res = CPrice::GetList(
            array(),
            array(
                "PRODUCT_ID" => $arFieldsOffer['ID'],
                "CATALOG_GROUP_ID" => 4
            )
        );

        if ($arr = $res->Fetch()) {
            \Bitrix\Main\Diag\Debug::dump(CPrice::Update($arr["ID"], $arFields));
        } else {
            \Bitrix\Main\Diag\Debug::dump(CPrice::Add($arFields));
        }

        $arLoadProductArray = array(
            "MODIFIED_BY" => $USER->GetID(),
            "TIMESTAMP_X" => ConvertTimeStamp(time(), 'FULL')
        );
        $el->Update($arFieldsOffer['ID'], $arLoadProductArray);
    }
}
```

1.14 Send instant message to users

```
<?php
CModule::IncludeModule("socialnetwork");
```

(continues on next page)

(continued from previous page)

```
$arFields = array(
    "FROM_USER_ID" => 2,
    "TO_USER_ID" => 1,
    "MESSAGE" => "You message",
    "=DATE_CREATE" => $GLOBALS["DB"]->CurrentTimeFunction(),
    "MESSAGE_TYPE" => "S",
);
CSocNetMessages::Add($arFields);
```

Recipes for work with orders

This section contains scripts for common tasks often found in work with orders

2.1 Get order props

```
<?php
\Bitrix\Main\Loader::includeModule('sale');

//get needed order properties
$db_vals = CSaleOrderPropsValue::GetList(
    array('SORT' => 'ASC'),
    array(
        'ORDER_ID' => $ORDER_ID,
        'CODE' => array(
            'PROP_CODE',
            'PROP_CODE_1',
        ),
    ),
    false,
    false,
    array(
        'VALUE',
        'CODE'
    )
);
while ($arVals = $db_vals->Fetch()) {
}
```

2.2 Copy order

This function shows how you can copy order by it's ID. This function do NOT copies basket properties

```
<?php
/**
 * Make copy of order with $orderId. Increments value of COPIED_ORDERS_COUNT order_
 * property
 * @param $orderId
 */
function doCopyOrder($orderId)
{
    \Bitrix\Main\Loader::includeModule('sale');

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

    $newOrderPrice = $arOrder['PRICE'];

    $arNewOrderFields = array(
        'LID' => $arOrder['LID'],
        'PERSON_TYPE_ID' => $arOrder['PERSON_TYPE_ID'],
        'PAYED' => 'N',
        'CANCELED' => 'N',
        'STATUS_ID' => 'N',
        'EMP_STATUS_ID' => 1,
        'PRICE_DELIVERY' => $arOrder['PRICE_DELIVERY'],
        'ALLOW_DELIVERY' => 'N',
        'PRICE' => round($newOrderPrice, 0, PHP_ROUND_HALF_DOWN),
        'CURRENCY' => $arOrder['CURRENCY'],
        'DISCOUNT_VALUE' => $arOrder['DISCOUNT_VALUE'],
        'USER_ID' => $arOrder['USER_ID'],
        'PAY_SYSTEM_ID' => $arOrder['PAY_SYSTEM_ID'],
        'DELIVERY_ID' => $arOrder['DELIVERY_ID'],
        'USER_DESCRIPTION' => $arOrder['USER_DESCRIPTION'],
        'ADDITIONAL_INFO' => $arOrder['ADDITIONAL_INFO'],
        'COMMENTS' => 'Order copy',
        'TAX_VALUE' => $arOrder['TAX_VALUE'],
        'AFFILIATE_ID' => $arOrder['AFFILIATE_ID'],
    );

    $newOrderId = CSaleOrder::Add($arNewOrderFields);

    if ($newOrderId) {
        //copy needed order properties
        $db_vals = CSaleOrderPropsValue::GetList(
            array('SORT' => 'ASC'),
            array('ORDER_ID' => $orderId),
            false,
            false,
            array()
        );
        while ($arVals = $db_vals->Fetch()) {
            unset($arVals['ID']);
            $arVals['ORDER_ID'] = $newOrderId;

            CSaleOrderPropsValue::Add($arVals);
        }
    }
}
```

(continues on next page)

(continued from previous page)

```

//copy order basket items
$dbBasket = CSaleBasket::GetList(
    array('ID' => 'ASC'),
    array('ORDER_ID' => $orderId),
    false,
    false,
    array(
        'SET_PARENT_ID', 'TYPE', 'ID',
        'PRODUCT_ID', 'PRODUCT_PRICE_ID', 'PRICE', 'CURRENCY', 'WEIGHT',
        'QUANTITY', 'LID',
        'NAME', 'CALLBACK_FUNC', 'MODULE', 'NOTES', 'PRODUCT_PROVIDER_CLASS',
        'CANCEL_CALLBACK_FUNC',
        'ORDER_CALLBACK_FUNC', 'PAY_CALLBACK_FUNC', 'DETAIL_PAGE_URL',
        'CATALOG_XML_ID', 'PRODUCT_XML_ID',
        'VAT_RATE'
    )
);

$item = new \CSaleBasket;
while ($arBasket = $dbBasket->Fetch()) {
    if (\CSaleBasketHelper::isSetItem($arBasket)) {
        continue;
    }

    $arFields = array(
        'ORDER_ID' => $newOrderId,
        'PRODUCT_ID' => $arBasket['PRODUCT_ID'],
        'PRODUCT_PRICE_ID' => $arBasket['PRODUCT_PRICE_ID'],
        'PRICE' => $arBasket['PRICE'],
        'CURRENCY' => $arBasket['CURRENCY'],
        'WEIGHT' => $arBasket['WEIGHT'],
        'QUANTITY' => $arBasket['QUANTITY'],
        'LID' => $arBasket['LID'],
        'NAME' => $arBasket['NAME'],
        'CALLBACK_FUNC' => $arBasket['CALLBACK_FUNC'],
        'MODULE' => $arBasket['MODULE'],
        'NOTES' => $arBasket['NOTES'],
        'PRODUCT_PROVIDER_CLASS' => $arBasket['PRODUCT_PROVIDER_CLASS'],
        'CANCEL_CALLBACK_FUNC' => $arBasket['CANCEL_CALLBACK_FUNC'],
        'ORDER_CALLBACK_FUNC' => $arBasket['ORDER_CALLBACK_FUNC'],
        'PAY_CALLBACK_FUNC' => $arBasket['PAY_CALLBACK_FUNC'],
        'DETAIL_PAGE_URL' => $arBasket['DETAIL_PAGE_URL'],
        'CATALOG_XML_ID' => $arBasket['CATALOG_XML_ID'],
        'PRODUCT_XML_ID' => $arBasket['PRODUCT_XML_ID'],
        'VAT_RATE' => $arBasket['VAT_RATE'],
        'PROPS' => array(),
        'TYPE' => $arBasket['TYPE']
    );

    $item->Add($arFields);
}
}

```

2.3 Get basket props

```
<?php
/**
 * Function obtains all properties of a basket item
 * @param int $id Basket item Id to search for
 * @return mixed[] List of basket item properties
 */
protected function getBasketItemProps($id)
{
    \Bitrix\Main\Loader::includeModule('sale');

    $arProps = array();
    $dbBasketProps = CSaleBasket::GetPropsList(
        array("SORT" => "ASC"),
        array("BASKET_ID" => $id),
        false,
        false,
        array("ID", "BASKET_ID", "NAME", "VALUE", "CODE", "SORT")
    );

    if ($arBasketProps = $dbBasketProps->Fetch())
    {
        do
        {
            $arProps[] = array(
                "NAME" => $arBasketProps["NAME"],
                "CODE" => $arBasketProps["CODE"],
                "VALUE" => $arBasketProps["VALUE"]
            );
        }
        while ($arBasketProps = $dbBasketProps->Fetch());
    }

    return $arProps;
}
```

Recipes for SEO tasks

This section contains scripts for common SEO tasks

3.1 301 Redirect from old URL

Save current URL into property

```
<?php
CModule::IncludeModule('iblock');
$arSelect = array('ID', 'NAME', 'CODE', 'DETAIL_PAGE_URL', 'ACTIVE');
$arFilter = array('IBLOCK_ID' => 4);
$res = CIBlockElement::GetList(
    array('TIMESTAMP_X' => 'ASC'),
    $arFilter,
    false,
    array('nPageSize' => 1000),
    $arSelect
);
while ($ob = $res->GetNextElement()) {
    $arFields = $ob->GetFields();
    CIBlockElement::SetPropertyValuesEx(
        $arFields['ID'],
        false,
        array('OLD_URL' => $arFields['DETAIL_PAGE_URL'])
    );
}
```

Release redirect logic in `init.php`

```
<?php
AddEventHandler('main', 'OnBeforeProlog', 'MyOnBeforePrologHandler');

function MyOnBeforePrologHandler()
```

(continues on next page)

(continued from previous page)

```

{
    global $APPLICATION;
    $curPage = 'http://example.com' . $APPLICATION->GetCurPage();
    if ($newUrl = getNewUrlFromOld($curPage, 2)) {
        LocalRedirect($newUrl, false, '301 Moved permanently');
    }
    if ($newUrl = getNewUrlFromOld($curPage, 1)) {
        LocalRedirect($newUrl, false, '301 Moved permanently');
    }
}

/**
 * returns a new url from an old stored in IBLOCK element/section property 'OLD_URL'/
 * → 'UF_OLD_URL'
 * or false if no such element/section in IBLOCK
 * @param string $oldUrl
 * @param integer $iBlockId
 * @param string $elementFieldName (optional)
 * @param string $sectionFieldName (optional)
 * @return mixed
 */
function getNewUrlFromOld($oldUrl, $iBlockId, $elementFieldName = 'OLD_URL',
    → $sectionFieldName = 'UF_OLD_URL')
{
    if (CModule::IncludeModule('iblock')) {
        $elementList = CIBlockElement::GetList(
            array('SORT' => 'ASC'),
            array(
                'IBLOCK_ID' => $iBlockId,
                'PROPERTY_' . $elementFieldName => $oldUrl
            ),
            false,
            false,
            array(
                'DETAIL_PAGE_URL'
            )
        );
        if ($arElement = $elementList->GetNext()) {
            return $arElement['DETAIL_PAGE_URL'];
        }
        $sectionList = CIBlockSection::GetList(
            array('SORT' => 'ASC'),
            array(
                'IBLOCK_ID' => $iBlockId,
                $sectionFieldName => $oldUrl
            ),
            false,
            array(
                'SECTION_PAGE_URL'
            )
        );
        if ($arSection = $sectionList->GetNext()) {
            return $arSection['SECTION_PAGE_URL'];
        }
    }
}

```

(continues on next page)

(continued from previous page)

```

    return false;
}

```

3.2 Find and fix sections duplicates

For all section codes duplicates concat section code of it's parent

```

<?php
CModule::IncludeModule('iblock');

$arFilter = array('IBLOCK_ID' => 4);
$by = 'ID';
$order = 'ASC';
$db_list = CIBlockSection::GetList(array($by => $order), $arFilter, false);
while ($ar_result = $db_list->GetNext()) {
    $arSect[] = $ar_result;
    $arCode[] = $ar_result['CODE'];
}

$arCodes = array_count_values($arCode);

foreach ($arCodes as $key => $arCd) {
    if ($arCd > 1) {
        $arEndCode[] = $key;
    }
}

foreach ($arSect as $arSection) {
    if (in_array($arSection['CODE'], $arEndCode)) {
        $resa = CIBlockSection::GetByID($arSection['IBLOCK_SECTION_ID']);
        if ($ar_resf = $resa->GetNext()) {
            $name = $ar_resf['CODE'];

            $code = $name . '-' . $arSection['CODE'];

            $bs = new CIBlockSection;

            $arFields = array(
                'CODE' => $code,
            );

            $res = $bs->Update($arSection['ID'], $arFields);
        }
    }
}

```

3.3 .htaccess SEO redirects

```

#remove index.(php|html|htm)
RewriteRule ^(.*)\./index\.(php|html?)$ /$1/ [R=301,NC,L]
RewriteRule ^index\.(php|html?)$ / [R=301,NC,L]

```

(continues on next page)

(continued from previous page)

```
#www. to no www
RewriteCond %{SERVER_PORT}s ^((443(s)|[0-9]+s))$
RewriteRule ^(.*)$ - [env=askapache:%2]
RewriteCond %{HTTP_HOST} ^www\.(.*)$ [NC]
RewriteRule ^(.*)$ http%{ENV:askapache}://%1/$1 [R=301,L]

#add trailing slash
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_URI} !(\.*)?(?:\//|\.html?|\.php)$
RewriteRule ^(.*)$ %{REQUEST_URI}/ [R=301,L]
```

3.4 Fix pictures URL changing

```
<?php

AddEventHandler('iblock', 'OnBeforeIBlockElementUpdate', 'checkImagesSize');

/**
 * Restrict image update if same image size given
 * @param $arParams
 */
function checkImagesSize(&$arParams)
{
    $arParamsImageInfoDetail = \CFile::MakeFileArray($arParams['DETAIL_PICTURE']['old_file']
    => '');
    if ($arParams['DETAIL_PICTURE']['size'] == $arParamsImageInfoDetail['size']) {
        unset($arParams['DETAIL_PICTURE']);
    }

    $arParamsImageInfoPreview = \CFile::MakeFileArray($arParams['PREVIEW_PICTURE']['old_file']
    => '');
    if ($arParams['PREVIEW_PICTURE']['size'] == $arParamsImageInfoPreview['size']) {
        unset($arParams['PREVIEW_PICTURE']);
    }
}
```

3.5 Change elements code separator from ‘_’ to ‘-‘

```
<?php
CModule::IncludeModule('iblock');
$arSelect = array('ID', 'NAME', 'CODE');
$arFilter = array('IBLOCK_ID' => 10, 'ACTIVE' => 'Y');
$res = CIBlockElement::GetList(
    array('TIMESTAMP_X' => 'ASC'),
    $arFilter,
    false,
    array('nPageSize' => 1000),
    $arSelect
);
while ($obj = $res->GetNextElement()) {
```

(continues on next page)

(continued from previous page)

```

$arFields = $ob->GetFields();
$newCode = str_replace('_', '-', $arFields['CODE']);

$el = new CIBlockElement;

$arLoadProductArray = array(
    'MODIFIED_BY' => $USER->GetID(),
    'CODE' => $newCode,
);

$res2 = $el->Update($arFields['ID'], $arLoadProductArray);
}

```

3.6 Change sections code separator from ‘_’ to ‘-’

```

<?php
CModule::IncludeModule("iblock");
$arFilter = array("IBLOCK_ID" => 11);
$db_list = CIBlockSection::GetList(array(), $arFilter, true);
while ($ar_result = $db_list->GetNext()) {
    $newCode = str_replace("_", "-", $ar_result["CODE"]);

    $bs = new CIBlockSection;
    $arFields = array(
        "CODE" => $newCode,
    );

    $res = $bs->Update($ar_result["ID"], $arFields);
}

```

Common event handlers

This section contains common Bitrix event handlers

4.1 Minimal and maximal price props

```
<?php

AddEventHandler('iblock', 'OnAfterIBlockElementAdd', 'DoIBlockAfterSave');
AddEventHandler('iblock', 'OnAfterIBlockElementUpdate', 'DoIBlockAfterSave');
AddEventHandler('catalog', 'OnPriceAdd', 'DoIBlockAfterSave');
AddEventHandler('catalog', 'OnPriceUpdate', 'DoIBlockAfterSave');

/**
 * Runs on events:
 * - OnAfterIBlockElementAdd
 * - OnAfterIBlockElementUpdate
 * - OnPriceAdd
 * - OnPriceUpdate
 *
 * Calculates min and max prices and saves them into props
 * @param      $arg1
 * @param bool $arg2
 */
function DoIBlockAfterSave($arg1, $arg2 = false)
{
    $ELEMENT_ID = false;
    $IBLOCK_ID = false;
    $OFFERS_IBLOCK_ID = false;
    $OFFERS_PROPERTY_ID = false;

    //Check for catalog event
    if (is_array($arg2) && $arg2["PRODUCT_ID"] > 0) {
        //Get iblock element
```

(continues on next page)

(continued from previous page)

```

    $rsPriceElement = CIBlockElement::GetList(
        array(),
        array(
            "ID" => $arg2["PRODUCT_ID"],
        ),
        false,
        false,
        array("ID", "IBLOCK_ID")
    );
    if ($arPriceElement = $rsPriceElement->Fetch()) {
        $arCatalog = CCatalog::GetByID($arPriceElement["IBLOCK_ID"]);
        if (is_array($arCatalog)) {
            //Check if it is offers iblock
            if ($arCatalog["OFFERS"] == "Y") {
                //Find product element
                $rsElement = CIBlockElement::GetProperty(
                    $arPriceElement["IBLOCK_ID"],
                    $arPriceElement["ID"],
                    "sort",
                    "asc",
                    array("ID" => $arCatalog["SKU_PROPERTY_ID"])
                );
                $arElement = $rsElement->Fetch();
                if ($arElement && $arElement["VALUE"] > 0) {
                    $ELEMENT_ID = $arElement["VALUE"];
                    $IBLOCK_ID = $arCatalog["PRODUCT_IBLOCK_ID"];
                    $OFFERS_IBLOCK_ID = $arCatalog["IBLOCK_ID"];
                    $OFFERS_PROPERTY_ID = $arCatalog["SKU_PROPERTY_ID"];
                }
            } //or iblock wich has offers
            elseif ($arCatalog["OFFERS_IBLOCK_ID"] > 0) {
                $ELEMENT_ID = $arPriceElement["ID"];
                $IBLOCK_ID = $arPriceElement["IBLOCK_ID"];
                $OFFERS_IBLOCK_ID = $arCatalog["OFFERS_IBLOCK_ID"];
                $OFFERS_PROPERTY_ID = $arCatalog["OFFERS_PROPERTY_ID"];
            } //or it's regular catalog
            else {
                $ELEMENT_ID = $arPriceElement["ID"];
                $IBLOCK_ID = $arPriceElement["IBLOCK_ID"];
                $OFFERS_IBLOCK_ID = false;
                $OFFERS_PROPERTY_ID = false;
            }
        }
    }
} //Check for iblock event
elseif (is_array($arg1) && $arg1["ID"] > 0 && $arg1["IBLOCK_ID"] > 0) {
    //Check if iblock has offers
    $arOffers = CIBlockPriceTools::GetOffersIBlock($arg1["IBLOCK_ID"]);
    if (is_array($arOffers)) {
        $ELEMENT_ID = $arg1["ID"];
        $IBLOCK_ID = $arg1["IBLOCK_ID"];
        $OFFERS_IBLOCK_ID = $arOffers["OFFERS_IBLOCK_ID"];
        $OFFERS_PROPERTY_ID = $arOffers["OFFERS_PROPERTY_ID"];
    }
}

if ($ELEMENT_ID) {

```

(continues on next page)

(continued from previous page)

```

static $arPropCache = array();
if (!array_key_exists($IBLOCK_ID, $arPropCache)) {
    //Check for MINIMAL_PRICE property
    $rsProperty = CIBlockProperty::GetByID("MINIMUM_PRICE", $IBLOCK_ID);
    $arProperty = $rsProperty->Fetch();
    if ($arProperty)
        $arPropCache[$IBLOCK_ID] = $arProperty["ID"];
    else
        $arPropCache[$IBLOCK_ID] = false;
}

if ($arPropCache[$IBLOCK_ID]) {
    //Compose elements filter
    $arProductID = array($ELEMENT_ID);
    if ($OFFERS_IBLOCK_ID) {
        $rsOffers = CIBlockElement::GetList(
            array(),
            array(
                "IBLOCK_ID" => $OFFERS_IBLOCK_ID,
                "PROPERTY_" . $OFFERS_PROPERTY_ID => $ELEMENT_ID,
                "ACTIVE_DATE" => "Y",
                "ACTIVE" => "Y",
                "<CATALOG_QUANTITY" => AVAILIABLE_VAL,
                ">CATALOG_QUANTITY" => 0
            ),
            false,
            false,
            array("ID")
        );
        while ($arOffer = $rsOffers->Fetch())
            $arProductID[] = $arOffer["ID"];
    }

    $minPrice = false;
    $maxPrice = false;
    //Get prices
    $rsPrices = CPrice::GetList(
        array(),
        array(
            "BASE" => "Y",
            "PRODUCT_ID" => $arProductID,
        )
    );
    while ($arPrice = $rsPrices->Fetch()) {
        $PRICE = $arPrice["PRICE"];

        if ($minPrice === false || $minPrice > $PRICE)
            $minPrice = $PRICE;

        if ($maxPrice === false || $maxPrice < $PRICE)
            $maxPrice = $PRICE;
    }

    //Save found minimal price into property
    if ($minPrice !== false) {
        CIBlockElement::SetPropertyValuesEx(
            $ELEMENT_ID,

```

(continues on next page)

(continued from previous page)

```

        $IBLOCK_ID,
        array(
            "MINIMUM_PRICE" => $minPrice,
            "MAXIMUM_PRICE" => $maxPrice,
        )
    );
}
}
}
}
}

```

4.2 Before mail sent

```

<?php

AddEventHandler('sale', 'OnOrderNewSendEmail', 'processOrderMailTemplate');

function processOrderMailTemplate($ID, &$eventName, &$arFields)
{
    if ($eventName == 'SALE_NEW_ORDER') {

    }
}

```

4.3 Restrict 1C from updating element fields

```

AddEventHandler(
    'sale',
    'OnBeforeIBlockElementUpdate',
    array('OnBeforeIBlockElementUpdate', 'restrictFieldsUpdateFrom1C')
);

class OnBeforeIBlockElementUpdate
{
    public static $group1cId = 23;

    public static function restrictFieldsUpdateFrom1C(&$arFields)
    {
        global $USER;
        if(in_array(self::$group1cId, $USER->GetUserGroupArray()))
        {
            unset($arFields["PREVIEW_TEXT"]);
            unset($arFields["DETAIL_TEXT"]);
            unset($arFields["CODE"]);
        }
    }
}

```


This section contains scripts with example of using Bitrix D7 core

5.1 ORM

5.1.1 Select with Query object

```
<?php
use \Bitrix\Main\Loader;
use \Bitrix\Main\Entity\Query;
use \Bitrix\Crm\DealTable;

Loader::includeModule('crm');

$query = new Query(DealTable::getEntity());
$query->setFilter(array(
    'STAGE_ID' => 'WON',
));

$query->setSelect(array(
    'ID',
    'LEAD_DATE_CREATE' => 'LEAD_BY.DATE_CREATE',
    'CONTACT_DATE_CREATE' => 'CONTACT_BY.DATE_CREATE',
    'USER_ASSIGNED' => 'CONTACT_BY.ASSIGNED_BY_ID',
    'CLOSEDATE' => 'CLOSEDATE',
    'MONEY' => 'OPPORTUNITY',
    'CURRENCY' => 'CURRENCY_ID'
));

$result = $query->exec();

$deals = array();
```

(continues on next page)

(continued from previous page)

```
while ($arDeal = $result->fetch()) {
}
```

5.1.2 Select with Entity object

```
<?php
use \Bitrix\Main\Loader;
use \Bitrix\Crm\DealTable;

Loader::includeModule('crm');

$deals = DealTable::getList([
    'filter' => [
        'ID' => $dealId
    ]
])->fetch();
```

5.1.3 Join with ORM

```
<?php

use \Bitrix\Main\Loader;
use \Bitrix\Main\Entity\Query;
use \Bitrix\Crm\DealTable;

Loader::includeModule('crm');

$arResult = array();
$query = new Query(DealTable::getEntity());
$query
    //JOIN for b_crm_status table
    ->registerRuntimeField(
        'STATUSES',
        array(
            'data_type' => '\\Bitrix\\Crm\\StatusTable',
            'reference' => array(
                '=this.STATUS_ID' => 'ref.STATUS_ID',
                '=ref.ENTITY_ID' => new SqlExpression('"STATUS"')
            ),
        )
    )
    ->setSelect(array('STATUS_NAME' => 'STATUSES.NAME'))
    ->setFilter(
        array(
            'ID' => $dealId,
        )
    )
    ->setOrder(array('DATE_CREATE' => 'ASC'));

$result = $query->exec();
```

(continues on next page)

(continued from previous page)

```
while ($arLead = $result->fetch()) {
    $arResult[] = $arLead;
}
```

5.2 Bitrix\Main\Application

```
<?php
use Bitrix\Main\Application;

$application = Application::getInstance();

$docRoot = Application::getDocumentRoot();
$connection = Application::getConnection();
```

5.3 Bitrix\Main\Context

```
<?php
use Bitrix\Main\Application;
use Bitrix\Main\Context;

$context = Application::getInstance()->getContext();
// $context = Context::getCurrent();

$request = $context->getRequest();
$server = $context->getServer();
$siteId = $context->getSite();
$langId = $context->getLanguage();
```

5.4 Bitrix\Main\Request

```
<?php
use Bitrix\Main\Context;
use Bitrix\Main\Request;

$context = Context::getCurrent();

$request = $context->getRequest();
// $request = Context::getCurrent()->getRequest();

$value = $request->get("param");
$value = $request["param"];
$value = $request->getQuery("param");
$values = $request->getQueryList();
$value = $request->getPost("param");
$values = $request->getPostList();
$value = $request->getFile("param");
$values = $request->getFileList();
$value = $request->getCookie("param");
```

(continues on next page)

(continued from previous page)

```
$values = $request->getCookieList();

$method = $request->getRequestMethod();
$flag = $request->isGet();
$flag = $request->isPost();
$flag = $request->isAjaxRequest();
$flag = $request->isHttps();

$flag = $request->isAdminSection();
$requestUri = $request->getRequestUri();
$requestPage = $request->getRequestedPage();
$rDir = $request->getRequestedPageDirectory();
```

5.5 Bitrix\Main\Server

```
<?php
use Bitrix\Main\Context;
use Bitrix\Main\Server;

$context = Context::getCurrent();

$server = $context->getServer();
// $server = Context::getCurrent()->getServer();

$server->getDocumentRoot();
$server->getPersonalRoot();
$server->getHttpHost();
$server->getServerName();
$server->getServerAddr();
$server->getServerPort();
$server->getRequestUri();
$server->getRequestMethod();
$server->getPhpSelf();
$server->getScriptName();
$server->get('HTTP_ACCEPT');
```

5.6 Basket

5.6.1 Bitrix\Sale\Basket

```
<?php
use Bitrix\Sale;

// cur user basket
$basket = Sale\Basket::loadItemsForFUser(Sale\Fuser::getId(),
    Bitrix\Main\Context::getCurrent()->getSite());

// order's basket
$basket = Sale\Order::load($orderId)->getBasket();
```

(continues on next page)

(continued from previous page)

```
$price = $basket->getPrice();
$fullPrice = $basket->getBasePrice();
$weight = $basket->getWeight();

//add to basket
if ($item = $basket->getExistsItem('catalog', $productId)) {
    $item->setField('QUANTITY', $item->getQuantity() + $quantity);
} else {
    $item = $basket->createItem('catalog', $productId);
    $item->setFields(array(
        'QUANTITY' => $quantity,
        'CURRENCY' => \Bitrix\Currency\CurrencyManager::getBaseCurrency(),
        'LID' => \Bitrix\Main\Context::getCurrent()->getSite(),
        'PRODUCT_PROVIDER_CLASS' => 'CCatalogProductProvider',
    ));
}
$basket->save();

//delete basket item
$basket->getItemById($id)->delete();
$basket->save();

//CAN_BUY=Y items
$orderBasket = $basket->getOrderableItems();
```

5.6.2 Bitrix\Sale\BasketItem

```
<?php
use Bitrix\Sale;

$basketItems = $basket->getBasketItems();

foreach ($basket as $basketItem) {
    echo $basketItem->getField('NAME') . ' - ' . $basketItem->getQuantity() . '<br />
    ↪';
}

$item = $basketItems[0];
$item->getId();
$item->getProductId();
$item->getPrice();
$item->getQuantity();
$item->getFinalPrice();
$item->getWeight();
$item->getField('NAME');
$item->canBuy();
$item->isDelay();

$item->getPropertyCollection();
$item->getCollection();

//operations
$item->setField('QUANTITY', $quantity);
$item->setFields(array(
    'QUANTITY' => $quantity,
```

(continues on next page)

(continued from previous page)

```
'CUSTOM_PRICE' => $customPrice,
));

$item->delete();
$item->save(); //or $basket->save();
```

5.7 Order

5.7.1 Bitrix\Sale\Order

```
<?php
use Bitrix\Sale;

$order = Sale\Order::load($orderId);

//fields
$order->getId();
$order->getSiteId();
$order->getDateInsert();
$order->getPersonTypeId();
$order->getUserId();

$order->getPrice();
$order->getDiscountPrice();
$order->getDeliveryPrice();
$order->getSumPaid();
$order->getCurrency();

$order->isPaid();
$order->isAllowDelivery();
$order->isShipped();
$order->isCanceled();

$fields = $order->getAvailableFields();
$order->getField("ORDER_WEIGHT");
$order->getField('PRICE');

//change
$order->setField('USER_DESCRIPTION', 'Comment');
$order->save();

//payments and deliveries
$paymentIds = $order->getPaymentSystemId();
$deliveryIds = $order->getDeliverySystemId();

//discounts
$discountData = $order->getDiscount()->getApplyResult();

//basket
$order->setBasket($basket);
$basket = $order->getBasket();
```

5.7.2 Bitrix\Sale\PropertyValue

```
<?php
$propertyCollection = $order->getPropertyCollection();

$props = $propertyCollection->getArray();
$propGroups = $propertyCollection->getGroups();
$propsFromGroup = $propertyCollection->getGroupProperties($groupId);

$emailPropValue = $propertyCollection->getUserEmail();
$namePropValue = $propertyCollection->getPayerName();
$locPropValue = $propertyCollection->getDeliveryLocation();
$taxLocPropValue = $propertyCollection->getTaxLocation();
$profNamePropVal = $propertyCollection->getProfileName();
$zipPropValue = $propertyCollection->getDeliveryLocationZip();
$phonePropValue = $propertyCollection->getPhone();
$addressPropValue = $propertyCollection->getAddress();

$somePropValue = $propertyCollection->getItemByOrderPropertyId($orderPropertyId);
$somePropValue->getValue();
$somePropValue->getViewHtml();

$arProp = $somePropValue->getProperty();
$propId = $somePropValue->getPropertyId();
$propName = $somePropValue->getName();
$isRequired = $somePropValue->isRequired();
$propPerson = $somePropValue->getPersonTypeId();
$propGroup = $somePropValue->getGroupId();

//update
$somePropValue->setValue("value");
$order->save();
```

5.8 Files and directories operations

5.8.1 Bitrix\Main\IO\File

```
<?php
use Bitrix\Main\IO;
use Bitrix\Main\Application;

$file = new IO\File(Application::getDocumentRoot() . "/file.txt");

//file info
$isExist = $file->isExists();

$dir = $file->getDirectory();
$dir = $file->getDirectoryName();

$fileName = $file->getName();
$fileExt = $file->getExtension();
$fileSize = $file->getSize();
$contentType = $file->getContentType();
```

(continues on next page)

(continued from previous page)

```

$createdAt = $file->getCreationTime();
$accessAt = $file->getLastAccessTime();
$modifiedAt = $file->getModificationTime();

$perms = $file->getPermissions();
$perms = substr(sprintf('%o', $file->getPermissions()), -3);

//file operations
$content = $file->getContents();
$file->putContents("data");
$file->putContents("data", IO\File::APPEND);
$file->readFile();

$file->rename(Application::getDocumentRoot() . "/new_file.txt");
$file->delete();

```

5.8.2 Bitrix\Main\IO\Directory

```

<?php
use Bitrix\Main\IO;
use Bitrix\Main\Application;

$dir = new IO\Directory(Application::getDocumentRoot() . "/test/");

$dir->create();

//dir info
$isExist = $dir->isExists();

$createdAt = $dir->getCreationTime();
$accessAt = $dir->getLastAccessTime();
$modifiedAt = $dir->getModificationTime();

$perms = $dir->getPermissions();
$perms = substr(sprintf('%o', $dir->getPermissions()), -3);

//dir operations
$childDir = $dir->createSubdirectory("child");
$dir->rename(Application::getDocumentRoot() . "/another_path/");
$dir->delete();

$files = $dir->getChildren();

```

5.8.3 Bitrix\Main\IO\Path

```

<?php
use Bitrix\Main\IO;
use Bitrix\Main\Application;

$path = Application::getDocumentRoot() . "/some_dir/some_file.ext";
$fileExt = IO\Path::getExtension($path);
$fileName = IO\Path::getName($path);
$fileDir = IO\Path::getDirectory($path);

```


5.9 HTTP

```
<?php
use \Bitrix\Main\Application;
use \Bitrix\Main\Web\Uri;
use \Bitrix\Main\Web\HttpClient;

//URI
$uri = new Uri("http://username:password@example.com/some/path/?param1=value#comments
→");

$uri->getLocator();
$uri->getUri();
$uri->getHost();
$uri->getUser();
$uri->getPass();
$uri->getPath();
$uri->getPathQuery();
$uri->getPort();
$uri->getQuery();
$uri->getScheme();
$uri->deleteParams(array("param1"));
$uri->getUri();
$uri->addParams(array("param2" => "value"));
$uri->getUri();

//HTTP Client
$options = array(
    "redirect" => true,
    "redirectMax" => 5,
    "waitResponse" => true,
    "socketTimeout" => 30,
    "streamTimeout" => 60,
    "version" => HttpClient::HTTP_1_0, // HttpClient::HTTP_1_0 or HttpClient::HTTP_1_1
    "proxyHost" => "",
    "proxyPort" => "",
    "proxyUser" => "",
    "proxyPassword" => "",
    "compress" => false, // Accept-Encoding: gzip
    "charset" => "", //request body charset
    "disableSslVerification" => false,
);
$httpclient = new HttpClient($options);

//headers
$name = "User-Agent";
$value = "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_3) AppleWebKit/537.75.14 (KHTML,
→ like Gecko) Version/7.0.3 Safari/7046A194A";
$httpclient->setHeader($name, $value, true);

//cookies
$cookies = array(
    "name" => "value",
);
$httpclient->setCookies($cookies);

//basic auth
```

(continues on next page)

(continued from previous page)

```
$httpClient->setAuthorization($user, $pass);

//send request
//methods - HttpClient::HTTP_GET, HttpClient::HTTP_POST, HttpClient::HTTP_PUT,
↳ HttpClient::HTTP_HEAD HttpClient::HTTP_PATCH.
$httpClient->query($method, $url, $entityBody = null);

//result
$httpClient->getResult();
$httpClient->getStatus();
$httpClient->getContentType();
$httpClient->getEffectiveUrl();
$httpClient->getCookies();
$httpClient->getHeaders();
$httpClient->getError();

//download file
$httpClient->download('http://example.com/file.pdf', Application::getDocumentRoot() .
↳ '/upload/example/document.pdf');
```

Tasks with cron utility

This section contains recipes for using cron

6.1 Agents on cron

Example of file that runs agents and events on cron. Place it in local/php_interface folder.

```
<?php
$_SERVER['DOCUMENT_ROOT'] = realpath(dirname(__FILE__) . '/../..');
$DOCUMENT_ROOT = $_SERVER['DOCUMENT_ROOT'];

define('NO_KEEP_STATISTIC', true);
define('NOT_CHECK_PERMISSIONS', true);
define('BX_NO_ACCELERATOR_RESET', true);
define('BX_CRONTAB_SUPPORT', false);

require($_SERVER['DOCUMENT_ROOT'] . '/bitrix/modules/main/include/prolog_before.php');
if ('N' != COption::GetOptionString('main', 'agents_use_crontab')) {
    COption::SetOptionString('main', 'agents_use_crontab', 'N');
}
if ('N' != COption::GetOptionString('main', 'check_agents')) {
    COption::SetOptionString('main', 'check_agents', 'N');
}

@set_time_limit(0);

CAgent::CheckAgents();
CEvent::CheckEvents();

if (CModule::IncludeModule('subscribe')) {
    $cPosting = new CPosting;
    $cPosting->AutoSend();
}
```


This section contains useful functions

7.1 Dump for admin

```
<?php
function pre($array, $exit = false, $hide = false)
{
    global $USER;
    if ($USER->IsAdmin()) {
        if ($hide) {
            echo "<!--";
        }

        echo "<pre>";
        var_dump($array);
        echo "</pre>";

        if ($hide) {
            echo "-->";
        }

        if ($exit) {
            exit();
        }
    }
}
```

7.2 Dump for admin limited by IP

```
<?php
function pre($array, $exit = false, $hide = false)
{
    if ($_SERVER["REMOTE_ADDR"] == "95.67.105.122") {
        if ($hide) {
            echo "<!--";
        }

        echo "<pre>";
        var_dump($array);
        echo "</pre>";

        if ($hide) {
            echo "-->";
        }

        if ($exit) {
            exit();
        }
    }
}
```

7.3 Export array with short array syntax

```
<?php
/**
 * Var expor array as PHP 5.4 notation
 * @param      $var
 * @param string $indent
 * @return mixed|string
 */
function var_export54($var, $indent = "")
{
    switch (gettype($var)) {
        case "string":
            return "'" . addslashes($var, "\\$\"\\r\\n\\t\\v\\f") . "'";
        case "array":
            $indexed = array_keys($var) === range(0, count($var) - 1);
            $r = [];
            foreach ($var as $key => $value) {
                $r[] = "$indent"
                    . ($indexed ? "" : var_export54($key) . " => ")
                    . var_export54($value, "$indent");
            }
            return "[\n" . implode(",\n", $r) . "\n" . $indent . "]";
        case "boolean":
            return $var ? "TRUE" : "FALSE";
        default:
            return var_export($var, TRUE);
    }
}
```

Example of using PDO

This section contains scripts with PDO examples

```
<?php

set_time_limit(0);
ini_set('display_errors', 1);

$time_start = microtime(true);

$username = 'username';
$password = 'password';
$dsn = 'mysql:dbname=db_name;host=localhost';

try {
    $conn = new PDO($dsn, $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    //prepare example
    $insert = $conn->prepare("INSERT INTO table (col1, col2) VALUES (:col1, :col1)");
    $insert->bindParam(':col1', 'foo');
    $insert->bindParam(':col2', 'bar');
    $insert->execute();

    //last insert ID
    $conn->lastInsertId();

    //select query
    $select = $conn->query('SELECT * FROM table_1');
    $select->execute();
    $select->setFetchMode(PDO::FETCH_ASSOC);
    $result = $select->fetchAll();
} catch (PDOException $e) {
```

(continues on next page)

(continued from previous page)

```
} catch (Exception $e) {  
  
}  
  
$conn = null;
```


This section contains various recipes that do not fit to other categories

9.1 Dates on russian

```
<?php
setlocale(LC_ALL, 'ru_RU.UTF-8');
strftime('%d %B %Y', strtotime("+1 day"))
```

9.2 Phinx config

```
<?php
define("NOT_CHECK_PERMISSIONS", true);
define("NO_AGENT_CHECK", true);
$GLOBALS["DBType"] = 'mysql';
$_SERVER["DOCUMENT_ROOT"] = realpath(__DIR__ . '/../');
include($_SERVER["DOCUMENT_ROOT"] . "/bitrix/modules/main/include/prolog_before.php");
// manual saving of DB resource
global $DB;
$app = \Bitrix\Main\Application::getInstance();
$con = $app->getConnection();
$DB->db_Conn = $con->getResource();
// "authorizing" as admin
$_SESSION["SESS_AUTH"]["USER_ID"] = 1;

$config = include realpath(__DIR__ . '/../bitrix/.settings.php');

return array(
    "paths" => array(
```

(continues on next page)

(continued from previous page)

```

        "migrations" => realpath(__DIR__ . '/migrations/')
    ),
    "environments" => array(
        "default_migration_table" => "phinxlog",
        "default_database" => "dev",
        "dev" => array(
            "adapter" => "mysql",
            "host" => $config['connections']['value']['default']['host'],
            "name" => $config['connections']['value']['default']['database'],
            "user" => $config['connections']['value']['default']['login'],
            "pass" => $config['connections']['value']['default']['password']
        )
    )
);

```

9.3 Errors in bitrix core

Get last bitrix error

```

<?php
$error = $APPLICATION->GetException();
echo $error->GetString();

```

9.4 Examples of DateTime class

```

<?php
$curTime = time();
//get current day of week
$curDayOfWeek = date("l", $curTime);

//date calculations
$intervalStartDate = date("Y-m-d", strtotime('+2 days'));
$end = new DateTime();
$end = $end->modify('+6 weeks');

//date intervals
$begin = new DateTime($intervalStartDate);

//set interval for 1 day in step
$interval = new DateInterval('P1D');
$dateRange = new DatePeriod($begin, $interval, $end);

//loop on days intervals
foreach($dateRange as $date)
{
    $dayTimestamp = $date->getTimestamp();

    //formatting
    $printDate = $date->format("j.n.Y");
    $dayOfWeek = $date->format('l');
}

```

9.5 Logging examples

9.5.1 Using file_put_contents

```
<?php
$logFile = $_SERVER['DOCUMENT_ROOT'].'/local/logs/log-'.strftime("%Y-%m-%d").'.log';

$line = strftime("%d.%m.%Y | %H:%M:%S")." | LOG MESSAGE\n";
file_put_contents($logFile, $line, FILE_APPEND);
```

9.5.2 Using Monolog

```
$ composer require monolog/monolog
```

```
<?php
use Monolog\Logger;
use Monolog\Handler\StreamHandler;
use Monolog\Formatter\LineFormatter;

$stream = new StreamHandler('logs/solutions.log');
$formatter = new LineFormatter("[%datetime%] %message%\n");
$stream->setFormatter($formatter);

$log = new Logger('logger_name');
$log->pushHandler($stream);

$log->addDebug('LOG MESSAGE');
```

9.6 Update 1C exchange module error

```
COption::SetOptionString("catalog", "DEFAULT_SKIP_SOURCE_CHECK", "Y");
COption::SetOptionString("sale", "secure_1c_exchange", "N");
```

9.7 Reset module files

Head to `/bitrix/admin/update_system.php?lang=ru&BX_SUPPORT_MODE=Y` page and find ‘System area’ block

CHAPTER 10

Indices and tables

- `genindex`
- `modindex`
- `search`

C

Common event handlers, [23](#)

E

Example of using PDO, [42](#)

Examples of Bitrix ORM using, [28](#)

O

Other, [44](#)

R

Recipes for SEO tasks, [18](#)

Recipes for work with orders, [14](#)

T

Tasks with cron utility, [38](#)

U

Useful functions, [39](#)