

## Задания для подготовки

### Экзамен «№1» Интеграция дизайна и настройка публичной части по созданию проектов на базе Bitrix Framework

---

Обновите документ, актуальная версия по ссылке: <https://academy.1c-bitrix.ru/~ex11train>

#### Материалы

Материалы к заданиям <https://academy.1c-bitrix.ru/~ex11materials> (10 Мб, zip)

#### Общие условия для решения задания:

##### 1) Установка «1С-Битрикс: Управление сайтом»

Во время экзамена вам будет доступна установка «1С-Битрикс: Управление сайтом» редакции «Стандарт», с установленным решением «Корпоративный сайт производственной компании».

Рекомендуем развернуть аналогичную установку для подготовки.

##### 2) Соблюдайте требования

Задания на экзамене необходимо будет выполнить, соблюдая требования к решениям, иначе они не засчитываются.

Вы сможете использовать список требований <https://academy.1c-bitrix.ru/~ex11desc> во время экзамена.

##### 3) Документация

Во время экзамена можно пользоваться только этими ресурсами:

- Справочником по PHP размещенным на рабочем столе экзаменационного сервера.
- Справочниками API на сайте: [https://dev.1c-bitrix.ru/api\\_d7/](https://dev.1c-bitrix.ru/api_d7/) и [https://dev.1c-bitrix.ru/api\\_help/](https://dev.1c-bitrix.ru/api_help/)
- Документацией по платформе: [https://dev.1c-bitrix.ru/learning/course/index.php?COURSE\\_ID=43](https://dev.1c-bitrix.ru/learning/course/index.php?COURSE_ID=43)

##### 4) Среда разработки

Во время экзамена вы будете решать задания, подключившись к удаленному рабочему столу, изменить его программное окружение вы не сможете.

Чтобы непривычная среда не отвлекала во время экзамена, ознакомьтесь заранее с настройками окружения <https://academy.1c-bitrix.ru/certification/programs/exam-server.php> и используйте образ демо-сервера <https://academy.1c-bitrix.ru/certification/programs/demo-server.php> во время подготовки.

## [ex1-100] Установка и первичная настройка системы

Для журнала событий установите опции:

- «Сколько дней хранить события» - 60 дней.
- Должны быть отмечены все события для сохранения, если нет – активируйте.
- «Сохранять историю изменения полей профиля пользователя» – активно, историю хранить 60 дней.
- «Сохранять историю входов с устройств пользователя» – активно.

## [ex1-200] Основы устройства Bitrix Framework

В файле конфигурации Bitrix Framework включите вывод PHP ошибок на страницах сайта.

## [ex1-300] Управление публичной структурой сайта

### Структура публичных разделов

- Задайте в корне сайта публичные разделы и страницы (страницы создавайте пустыми) по таблице ниже.
- При выполнении задания сохраните разделы search, login, они будут полезны при решении следующих заданий, остальные дефолтные контентные страницы удалите.
- Таблица структуры публичной части:

В файловой системе	Название на сайте
/index.php	Главная
/about/index.php	О компании
/about/history/index.php	История
/about/team/index.php	Команда
/about/join/index.php	Присоединиться
/about/join/careers/index.php	Карьера
/about/join/benefits/index.php	Преимущества
/about/join/vacancies/index.php	Вакансии
/services/index.php	Услуги
/portfolio/index.php	Портфолио
/articles/index.php	Статьи
/contacts/index.php	Контакты

### Настройте меню

- Используйте типы меню из установленного решения (top, left), выполните настройки:

- Верхнее меню отображает несколько уровней вложенности. Первый уровень строится на основании меню типа top, в нем такие пункты:

Название	Ссылка
Главная	/
О компании	about/
Услуги	services/
Портфолио	portfolio/
Статьи	articles/
Контакты	contacts/

- Вложенные пункты верхнего меню, а также меню в левой колонке сайта, строятся на основании меню типа left. В левой колонке сайта меню должны выводить пункты только одного уровня, без вложенности.

В шаблоне сайта /bitrix/templates/furniture\_blue/header.php в вызове компонента для левого меню (menu) в теге <div id="sidebar"> добавьте параметр "CACHE\_SELECTED\_ITEMS" => "Y".

Это необходимо для корректного отображения разных пунктов меню в одной области шаблона при включенном кэшировании.

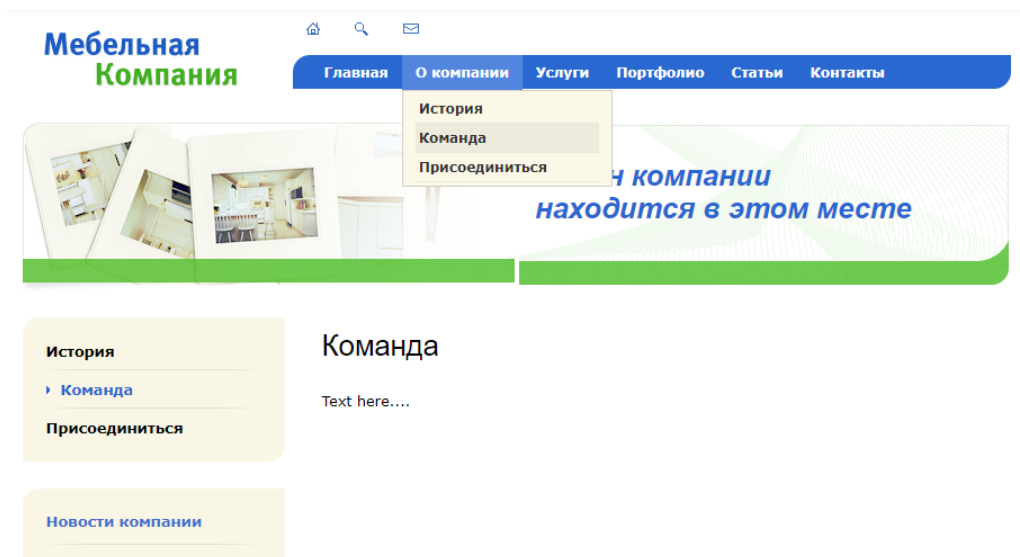
В разделе /about/ пункты левого меню какие:

Название	Ссылка
История	history/
Команда	team/
Присоединиться	join/

В разделе /about/join/ пункты левого меню такие:

Название	Ссылка
Карьера	careers/
Преимущества	benefits/
Вакансии	vacancies/

Должно получиться так, например, находясь на странице /about/team/



### Задайте метаданные страниц

- Для всех разделов и страниц сайта:
  - Ключевые слова (keywords) «Проминжиниринг, Инженерные услуги».
  - Описание (description) «Добро пожаловать на сайт компании, лидера в области проминжиниринга».
- Для страницы История:
  - Ключевые слова (keywords) «Проминжиниринг, История».
  - Описание (description) «История компании и путь к успеху».

### Настройте права доступа

- **Создайте новую группу** пользователей **Управление контентом**. Пользователям этой группы должно быть доступно изменение и создание страниц и разделов в разделе **/about/**, контент остальных разделов сайта они не должны иметь возможность менять.
- Создайте пользователя с логином **user1**, именем **Контент**, почтой **user1@academy.1c-bitrix.ru** и паролем **123456**, включите его в группу **Управление контентом**.
- Проверьте. Авторизуйтесь под созданным пользователем на странице **/login/** (если вы ее удалили – можно взять из материалов **demo/login/**) и убедитесь, что настройки доступа для группы выполнены верно, создайте раздел **/about/test/** с индексной страницей, назовите «Тест» и добавьте ее в левое меню.

### [ex1-400] Интеграция HTML-шаблона

Создайте шаблон сайта на основе HTML-верстки, предоставленной в архиве с материалами.

- В этом задании создать только шаблон сайта, **без внедрения в него компонентов**.
- Исходный HTML-шаблон в материалах в папке **html\_type1**.

### HTML-шаблон

- Представленный HTML-шаблон имеет два макета, которые едины по стилю, у них подключается один набор ресурсов (CSS, JS), но есть различия в отображаемых блоках на страницах разного типа. Необходимо внедрить верстку сохранив эти отличия.
- Оцените верстку макетов `template-content.html`, `template-dev.html`, определите корректное разделение на хедер, футер и контентную часть, а также в чём состоит отличие:

- До тега `<main>` и после его закрытия `</main>` макеты одинаковы.
- `template-content.html` – это классический шаблон для текстовых страниц, которые напрямую создают и меняют контент-редакторы.

Шаблон включает две колонки в центральной части. Левая входит в хедер, содержит боковое меню и другие блоки. Правая колонка — это основная часть страницы, включает заголовок, цепочку навигации и контентную часть.

Содержимым контентной части и должен управлять специалист через визуальный редактор, это непосредственный контент файлов публичной части, остальное – это шаблон сайта.

- `template-dev.html` — содержит одну колонку в центральной части. Это макет для страниц с контентом, который выводится компонентом, либо для страниц со сложной версткой, не предполагающих редактирование в визуальном редакторе контент-менеджерами.

Контентная часть страниц в этом макете строится с помощью тега `<section>`. Их может быть несколько, расположенных последовательно, как показано на странице `dev-index.html`. Следовательно, тег `<section>` не должен входить в хедер и футер.

Этот макет применяется с особенностью для главной страницы сайта: на ней не выводится блок с заголовком, текстом под ним и цепочкой навигации. Верстка главной страницы — `dev-index.html`.

### Создайте шаблоны сайта

- Создайте два шаблона сайта, в настройках сайта укажите условия применения согласно таблице:

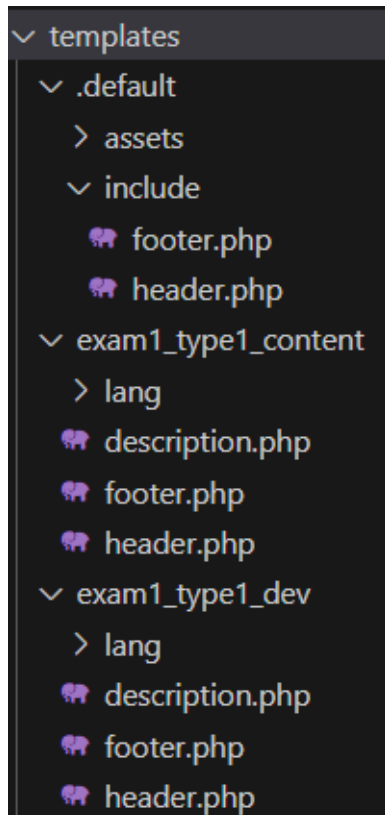
Исходный HTML-макет	Задать директорию шаблона	Задать название для админки	Использовать для
<code>template-content.html</code>	<code>exam1_type1_content</code>	«Экзамен N1 - TYPE1 - content»	Для раздела <code>/about/</code> , всех вложенных разделов.
<code>template-dev.html</code>	<code>exam1_type1_dev</code>	«Экзамен N1 - TYPE1 - dev»	Для всех остальных страниц сайта.

- Все ресурсы и шаблоны компонентов размещайте в рамках шаблона `.default`.

Общие части HTML-верстки для шаблонов вынесите в /local/templates/.default/include/ в файлы header.php и footer.php. Подключайте их в шаблонах:

```
require_once $_SERVER["DOCUMENT_ROOT"] . "/local/templates/.default/include/ ...
```

Получится такая структура:



- Для удобства определите константу для подключения ресурсов.

```
define("DEFAULT_TEMPLATE_PATH", '/local/templates/.default');
```

Создайте файл boot.php в /local/templates/.default/include, определите в нем константу и подключите его в /local/templates/.default/include/header.php.

- Для главной страницы в шаблоне реализуйте условие - не показывать блок с заголовком, строкой текста и цепочкой навигации. Определить раздел страницы можно с помощью API:

```
use Bitrix\Main\Application;
Application::getInstance()->getContext()->getRequest()->getRequestedPageDirectory();
```

Проверьте, какое именно значение возвращает это API на вашей установке для главной страницы, чтобы корректно задать условие скрытия блока на главной.

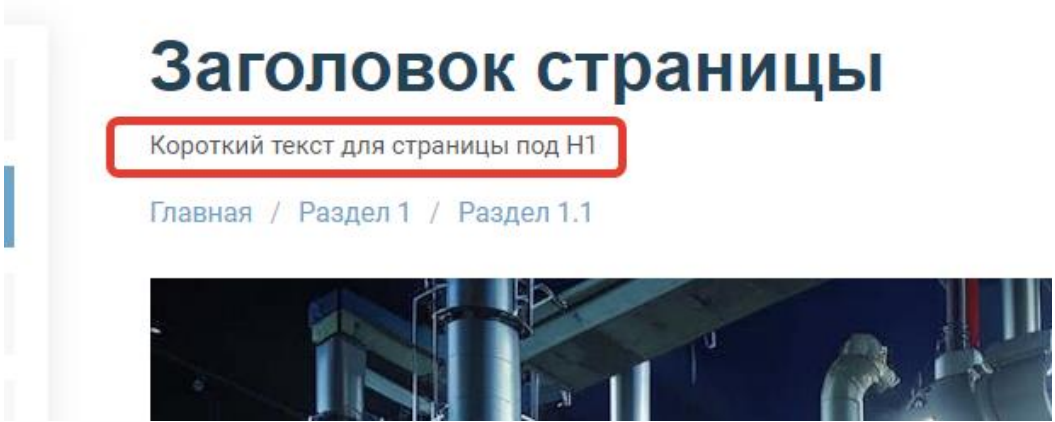
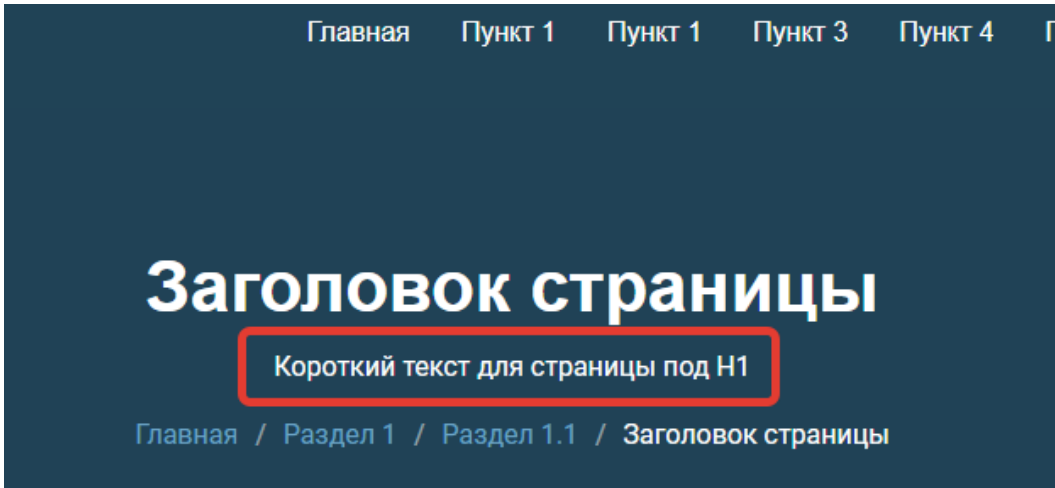
- Сразу после открытия тега body подключите панель управления в теге div:

```
<div id="panel"><?APPLICATION->ShowPanel();?></div>
```

- Включите объединение, минифицирование и сжатие CSS и JS. Не забудьте про настройки платформы: вы можете указать в какой момент должны подключаться файлы JS.
- В языковой файл вынесите только текст «Компания»

Свойства страниц

- При анализе верстки вы наверняка заметили, что под заголовком H1 в шаблонах `template-dev.html` и `template-content.html` выводится короткая фраза.



- Создайте свойство страницы, в котором контент-менеджеры смогут задавать этот текст, код свойства укажите **page\_text\_under\_title**, замените текстовую заглушку в шаблонах выводом значения этого свойства. Если значение не задано – пусть выводится пустой тег.
- Задайте значения для следующих страниц:

В файловой системе	Текст под заголовком H1
/about/history/index.php	Наш путь к успеху — это история инноваций, роста и достижений.
/about/join/benefits/index.php	Работайте с нами и раскройте свой потенциал в комфортной и поддерживающей среде.
/articles/index.php	Погрузитесь в инновации и передовые идеи с нашими статьями.
/contacts/index.php	Свяжитесь с нами — мы всегда готовы помочь и ответить на ваши вопросы.

## Демо-контент

Разместите на этих страницах сайта демо контент из HTML-страниц шаблона. Проставьте пути для изображений (папка `/assets/img/` в шаблоне). Здесь только размещение HTML, без внедрения компонентов.

В итоге должно получиться так:

В файловой системе	Файл с контентом	Заголовок	Текст под заголовком H1
<code>/index.php</code>	<code>dev-index.html</code>	Главная	<i>Нет в шаблоне</i>
<code>/about/history/index.php</code>	<code>content-history.html</code>	История	Наш путь к успеху — это история инноваций, роста и достижений.
<code>/about/join/benefits/index.php</code>	<code>content-benefits.html</code>	Преимущества	Работайте с нами и раскройте свой потенциал в комфортной и поддерживающей среде.
<code>/articles/index.php</code>	<code>dev-articles.html</code>	Статьи	Погрузитесь в инновации и передовые идеи с нашими статьями.
<code>/contacts/index.php</code>	<code>dev-contact.html</code>	Контакты	Свяжитесь с нами — мы всегда готовы помочь и ответить на ваши вопросы.

---

Получилось? Отлично!

Проверьте: шаблоны подключаются по условиям, заголовок и текст под ним выводится, все подключаемые ресурсы на месте, нет 404 в консоли браузера?

Поехали дальше!

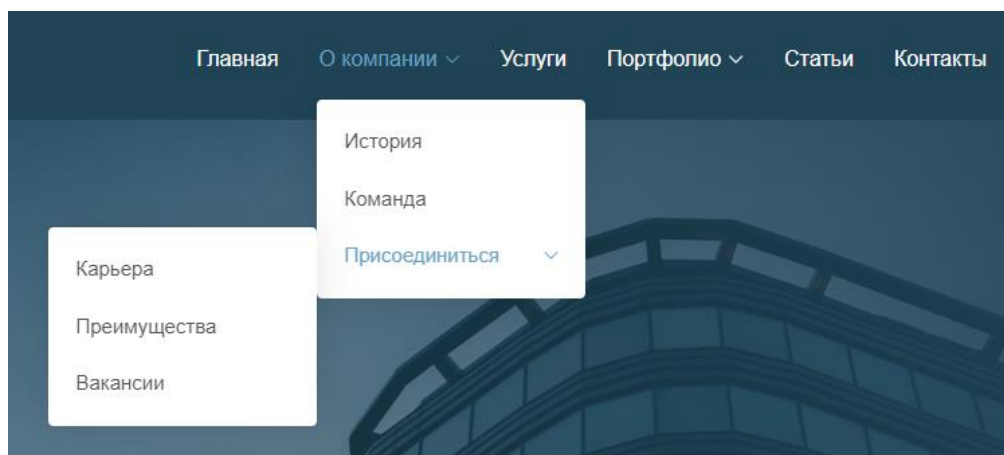
## [ex1-500] Базовые компоненты

Реализуйте на сайте ряд задач с помощью компонентов. Создайте шаблоны компонентов на основе внедряемого HTML-шаблона.

### Меню

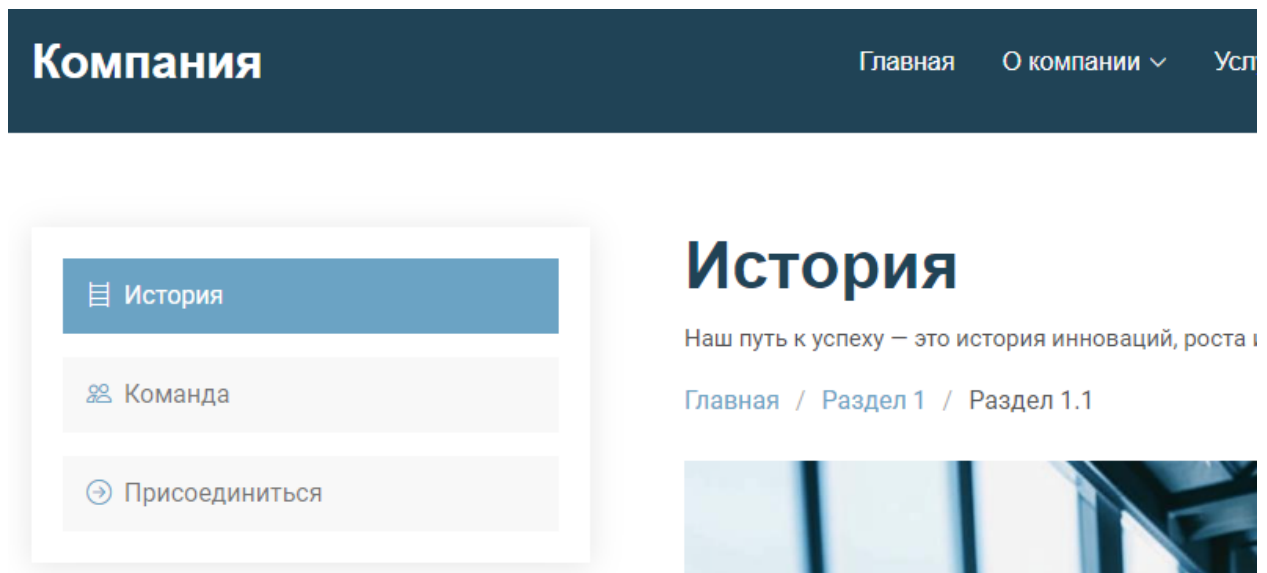
Реализуйте верхнее и левое меню сайта с помощью компонента «Меню» (**menu**). Меню должно строиться на основании соответствующих системных файлов.

- Верхнее меню





- Многоуровневое, поддерживает до трех уровней вложенностей. Как пример – есть системный шаблон компонента меню «horizontal\_multilevel» поддерживающий 3 уровня вложенности.
  - Строится на основании типа меню «top» для первого уровня и типа «left» для остальных.
  - Кеширование в компоненте должно быть включено.
- Левое меню

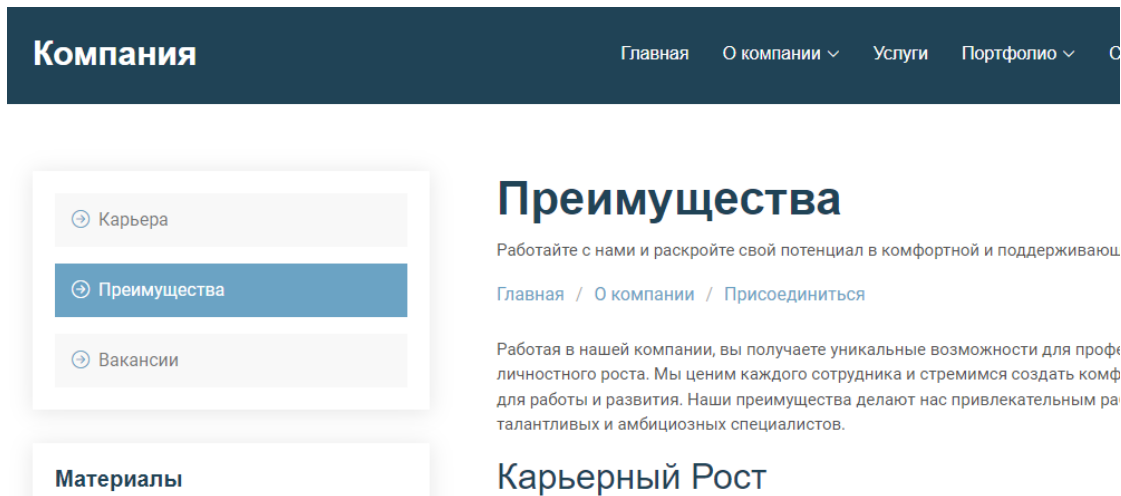


- В параметрах вызова компонента добавьте "CACHE\_SELECTED\_ITEMS" => "Y". Это необходимо для корректного отображения разных пунктов меню в одной области шаблона при включенном кэшировании.
- Кеширование в компоненте должно быть включено.
- Строится на основании меню типа left.
- Выводит только один уровень.
- В каждом пункте меню, перед текстом, выводится иконка. В HTML-шаблоне эта иконка задается CSS классом, файл стилей с иконками assets/vendor/bootstrap-icons/bootstrap-icons.css, он уже должен быть подключен у вас в шаблоне сайта.
- В файле меню задайте дополнительное свойство для пунктов, в котором вы будете указывать CSS класс иконки. Код свойства укажите как **menu\_ico**. В шаблоне подставляйте значение из этого свойства.
- Если значение не задано, то по умолчанию используйте в шаблоне компонента класс для иконки **bi-arrow-right-circle**, он задан в исходном HTML-шаблоне у всех пунктов.
- Для примера задайте значения иконки для страниц:
  - История – **bi-bookshelf**
  - Команда - **bi-people**

## Цепочка навигации

Цепочка навигации выводится в двух шаблонах и отличается по формату, реализуйте ее с помощью компонента «Навигационная цепочка» (breadcrumb). Создайте два шаблона компонента.

- Для контентных страниц template-content.htm:



- Формат: отображается иерархия разделов от корня сайта до текущей страницы, на каждый можно кликнуть. Текущая страница в цепочку не добавляется.
- По условиям заданий и требованиям экзамена – для каждой страницы есть свой раздел. Поэтому чтобы реализовать такую цепочку навигации можно применить такой подход:
  - И у раздела для страницы и самой страницы указывается заголовок. Например:

Раздел	Страница	Заголовок
/about/join/	/about/join/index.php	Присоединиться

Заголовок страницы используется для вывода в виде title браузера или H1 страницы. А заголовок раздела - в цепочке навигации для вложенных разделов, и в административной части.

- В шаблоне компонента «Навигационная цепочка» последний пункт не выводится, так как он будет отражать текущую страницу.
- Для макета template-dev.html: отображается иерархия разделов для текущей страницы, последний пунктом отображается заголовок текущей страницы, он без ссылки и в другом стиле.

# Контакты

Свяжитесь с нами — мы всегда готовы помочь и ответить на ваши вопросы.

[Главная](#) / [Контакты](#)



- Формат: отображается иерархия разделов от корня сайта до текущей страницы, на каждый можно кликнуть. Текущая страница добавляется в цепочку, но выводится другим цветом и не кликабельна. Т.е. для статических страниц последний пункт цепочки навигации и заголовок страницы будут совпадать.
- По условиям заданий и требованиям экзамена – для каждой страницы есть свой раздел. При создании и у раздела, и у страницы указывается один заголовок. Поэтому в цепочке навигации, для статических страниц, последний пункт уже совпадает с заголовком страницы, осталось выводить его в шаблоне компонента без ссылки и в другом стиле. Для динамических страниц набор пунктов цепочки навигации каждый компонент формирует по-своему, в этом задании не учитываем такие разделы.

## Включаемая область

- Контент главной страницы состоит из трех блоков section: слайдер (hero), возможности (featured-services), о компании (about).
- Контент каждого блока вынесите в отдельный файл, названия файлов задайте соответствующие: hero.php, featured-services.php, about.php
- Файлы разместите внутри шаблона, для этого создайте директорию /local/templates/.default/include/main/
- Подключайте этот контент на главной странице с помощью вызовов компонента «Включаемая область» (main.include).

## [ex1-600] Информационные блоки

- Создайте тип информационного блока Контент (**Content**). Укажите ID равным **content**, укажите опцию использования древовидного классификатора элементов по разделам.
- Импортируйте в него через XML информационный блок. Данные в материалах – раздел **portfolio\_export**.

- Контент инфоблока должен быть доступен всем посетителям сайта.

[ex1-800] Раздел сайта с динамической информацией и ЧПУ

Замените страницу `/portfolio/index.php` на раздел с динамической информацией с помощью комплексного компонента **catalog** и информационного блока «Портфолио». Выведите список разделов с переходом на список элементов и детальную страницу элемента.

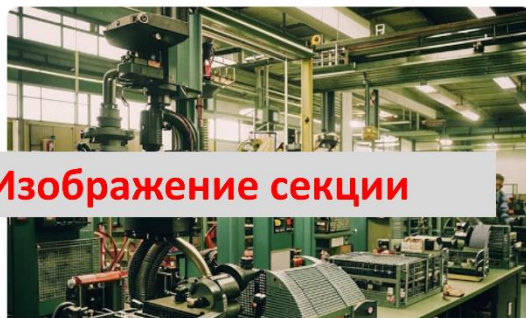
- Шаблон комплексного компонента назовите **portfolio**.
  - **Поддержку «эрмитажа», редактирование элементов над сайтом, реализовывать в этом задании не нужно.**
  - Кеширование в компонентах и настройках платформы должно быть включено.
  - В этом задании комплексный компонент будет использоваться в очень простом виде для вывода портфолио, поэтому большая часть стандартного функционала не понадобится. Реализуйте так:
    - На странице списка разделов (`sections.php`) – нужен компонент `bitrix:catalog.section.list`
    - На странице списка элементов (`section.php`) – нужен компонент `bitrix:catalog.section`
    - На странице списка элементов (`element.php`) – нужен компонент `bitrix:catalog.element`
- Все остальное из стандартного шаблона комплексного компонента использовать не нужно.
- Создайте шаблон комплексного компонента, используйте исходный HTML-шаблон:

- Для списка разделов dev-portfolio-sections.html



## Проектирование и разработка

Примеры инженерных решений и разработанных проектов.



## Изображение секции

## Название секции, ссылка список элементов

Установка и обслуживание

Кейсы по установке оборудования и последующему обслуживанию.

## Описание секции



## Продажа оборудования

Каталог реализованных проектов по поставке оборудования.



## Консультационные услуги

Примеры успешных консультаций и внедренных решений.

- Для списка элементов dev-portfolio-list.html

Компания

ГлавнаяО компании ▾Портфолио ▾Контакты

# Продажа оборудования


Ознакомьтесь с нашими успешными проектами и реализованными решениями.

Главная / Портфолио

Детальное изображение элемента


Название элемента и ссылка на детальку

Описание элемента для анонса



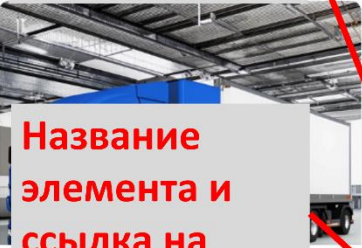
**Поставка строительной техники**

Поставка и обслуживание грузовой техники для возведения нового жилого комплекса.




**Поставка упаковочного оборудования**

Поставка и установка оборудования для автоматизации упаковки на пищевом производстве.




**Поставка реакторов**

Поставка и установка реакторных систем для нового химического



**Поставка станков с ЧПУ**

Поставка и установка станков с ЧПУ для изготовления деталей

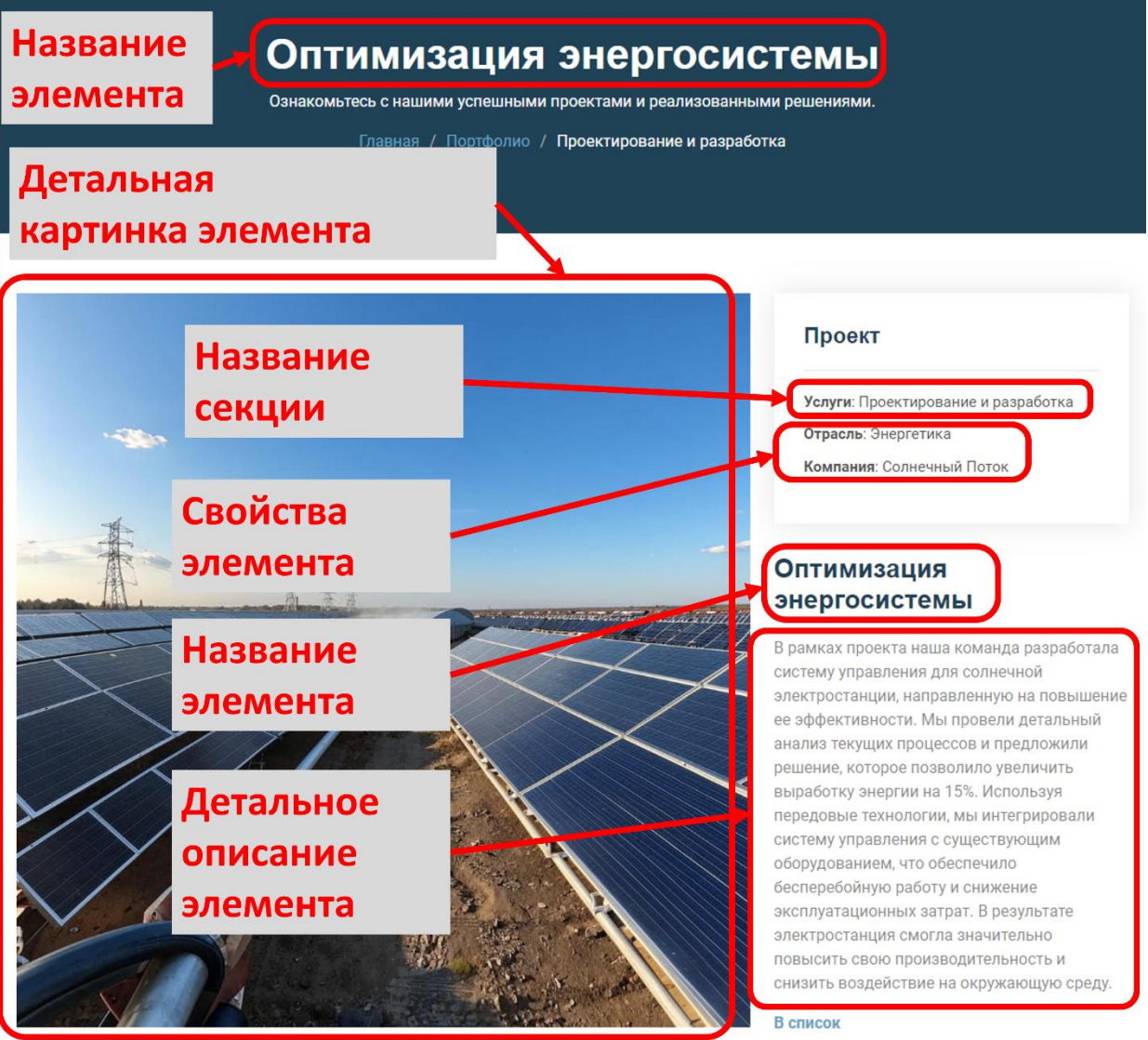


**Поставка солнечных панелей**

Комплексная поставка солнечных панелей для новой солнечной



○ Для детальной страницы dev-portfolio-details.html



- Для упрощения во время экзамена: внутри шаблонов простых компонентов не учитывайте никакие опции компонента, выводите безусловно тот контент, что предусмотрен внедряемой html-версткой.
- Для списка разделов и элементов в шаблоне компонента используйте `ResizeImageGet` чтобы уменьшить детальную картинку в соответствии с размером изображения в HTML-шаблоне.
- В шаблоне детальной станицы вынесите в языковые фразы надписи: «Проект», «Услуги», «Отрасль», «Компания».
- Настройте компонент на работу в ЧПУ режиме
  - Раздел – по ID секции `#SECTION_ID#`/
  - Детальный просмотр по ID секции и ID элемента `#SECTION_ID#/#ELEMENT_ID#`/
- Используя настройки инфоблока (SEO), установите метаданные динамических страниц по шаблонам.
  - Для страницы раздела со списком элементов:
    - Заголовок страницы (Заголовок раздела): «[Название раздела]»
    - Заголовок браузера (META TITLE): «Портфолио – [Название раздела]»
    - Описание (META DESCRIPTION): «[Описание раздела]»
  - Для детальной страницы элемента портфолио:
    - Заголовок страницы (Заголовок товара): «[Название элемента]»
    - Заголовок браузера (META TITLE): «Портфолио – [Название элемента]»
    - Описание (META DESCRIPTION): «[Текст анонса элемента]»

Где в квадратных скобках указаны поля или свойства элемента инфоблока и разделов.

- Меню на основе разделов информационного блока.
  - В верхнем меню, в разделе «Портфолио» должно появиться выпадающее меню в соответствии с разделами инфоблока.
  - Для реализации используется файл расширения меню вида `.<type>.menu_ext.php`. В нем вызывается компонент `menu.sections`, который возвращает массив данных для построения пунктов меню.
  - Пример такого файла из типового решения «Корпоративный сайт производственной компании» в материалах (`demo/.left.menu_ext.php`), используйте как пример, необходимо в компоненте указать корректные настройки.

## [ex1-900] Поиск

Реализуйте поиск по сайту. Выполните настройки внедрите верстку.



- Используйте стандартный компонент «Стандартная страница поиска» (search.page), он размещен в типовом разделе /search/ (если удалили - раздел можно взять из материалов demo/search/ или разместить и настроить компонент самостоятельно).
  - Компонент «Стандартная страница поиска» поддерживает большое количество опций, используем его в самом простом варианте: без выбора зоны поиска, без быстрых результатов поиска. Только список результатов с постраничной навигацией.
  - Создайте шаблон компонента на основании верстки dev-seach.html
  - Настройте на вывод по 6 элементов, используйте для постраничной навигации стандартный шаблон «modern».
- Настройте информационный блок «Портфолио».
  - Поиск должен идти **по всем его полям и свойствам**.
  - Настройте шаблоны ссылок для перехода с результатов поиска на элементы информационного блока.
- Переиндексируйте контент сайта.
- В футере, с помощью компонента «Форма поиска» (search.form), реализуйте работу строки поиска, переход должен быть на созданный раздел поиска, без указания «index.php» в адресе страницы поиска. Надпись на кнопке и заголовок «Поиск» – вынесите в языковой файл.
- Проверьте, все работает? Если ввести «проект» в строке поиска в футере, должен произойти переход на страницу поиска, найдены элементы. Должен быть корректный переход на просмотр элемента. По слову «СтройИнвест» должен находиться один элемент.

---

Первая часть готова! Выполнили - молодец 😊!

Проверьте себя, все ли требования к решению заданий соблюдены? Не забыли выключить автокеширование, если отключали на время выполнения заданий с динамической информацией?

### Структура публичных разделов

- Создайте раздел **Статистика**, название директории укажите **statistic\_na**.
- Добавьте ссылку на него в верхнее меню сайта последним пунктом.
- Создайте разделы и страницы внутри раздела **Статистика** (страницы создавайте пустыми):

В файловой системе	Название на сайте
/statistic_na/index.php	Статистика
/statistic_na/dashboard/index.php	Дашборд
/statistic_na/main/index.php	Основные
/statistic_na/main/info/index.php	Отчеты
/statistic_na/main/table/index.php	Данные
/statistic_na/extra/index.php	Дополнительные
/statistic_na/extra/base/index.php	Базы
/statistic_na/extra/info/index.php	Информация
/statistic_na/profile/index.php	Профиль

### Редирект

- Со страницы /statistic\_na/index.php с помощью метода LocalRedirect (или аналогичным на D7) реализуйте переадресацию на страницу /statistic\_na/dashboard/.

### Настройте меню

- Для созданного раздела /statistic\_na/ настройте свое меню.
- Создайте два типа меню **st\_first** и **st\_second**, они будут нужны для построения меню сайта двух уровней в разделе **statistic\_na**.
- Первый уровень меню строится на основании типа **st\_first**, в нем такие пункты:

Название	Ссылка
Дашборд	statistic_na/dashboard/
Основные	statistic_na/main/
Дополнительные	statistic_na/extra/
Профиль	statistic_na/profile/

- Второй уровень строится на основании типа **st\_second**. Создайте меню этого типа для разделов «Основные» и «Дополнительные», внесите соответствующие пункты.

### Задайте метаданные страниц

- Для всех разделов и страниц раздела **Статистика**:
  - Ключевые слова (keywords) «аналитика, данные».
  - Описание (description) «анализ данных».

#### Настройте права доступа

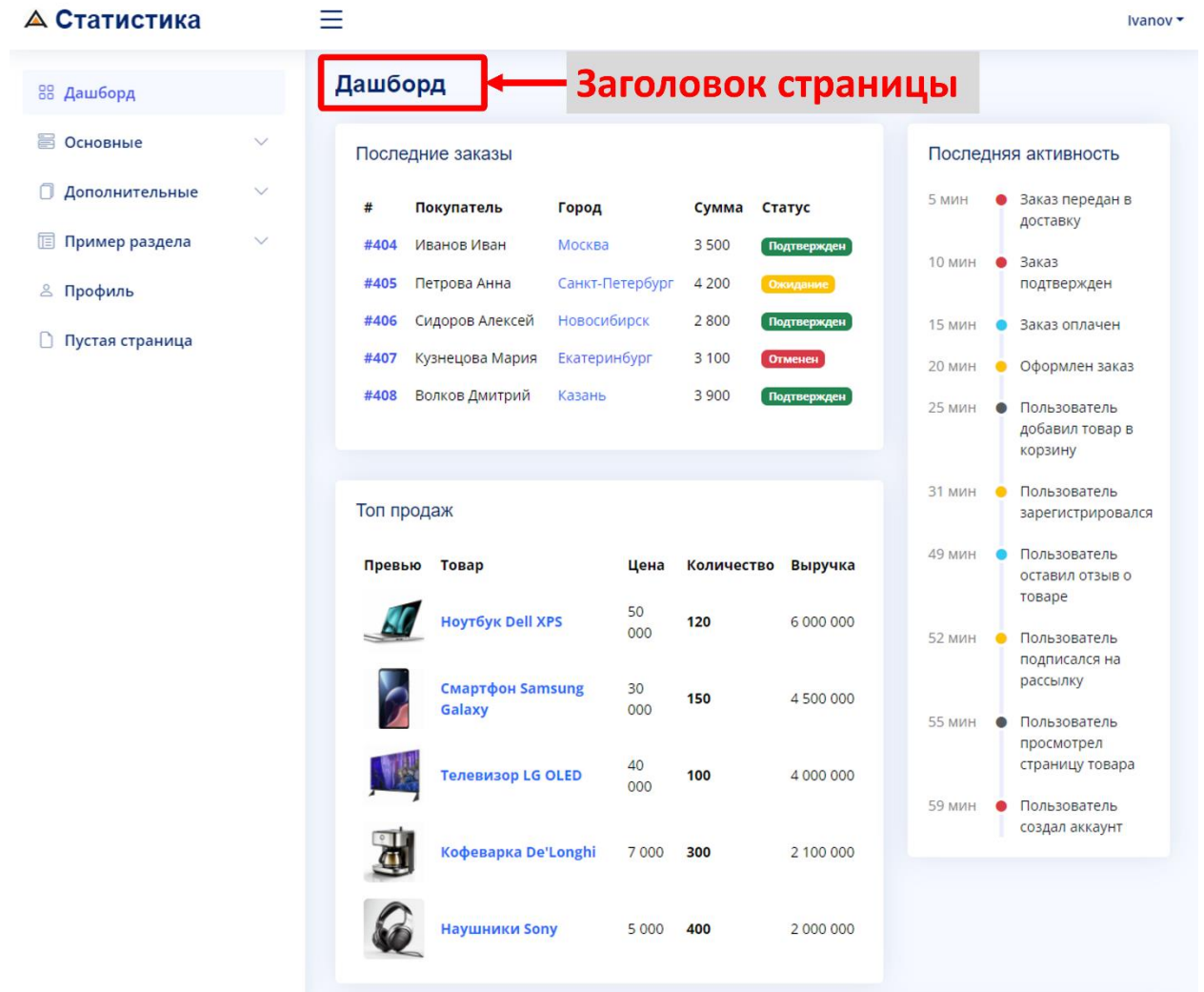
- Создайте группу пользователей **Аналитики**. Только администраторы сайта и пользователи этой группы должны иметь возможность просматривать страницы раздела **Статистика**. Пользователи типовой группы **Контент-редакторы (ID 5)** не должны иметь возможность просматривать и редактировать страницы в этом разделе.
- Создайте пользователя с логином **user2**, именем **Аналитик**, почтой **user2@academy.1c-bitrix.ru** и паролем **123456**, включите его в группу **Аналитики**.
- Авторизуйтесь под этим пользователем в разделе /login/ и проверьте, доступен ли ему раздел **Статистика**.

#### [ex1-410] Интеграция HTML-шаблона

Создайте шаблон сайта на основе HTML-верстки, предоставленной в архиве с материалами.

- В этом задании создать только шаблон сайта, **без внедрения в него компонентов**.
- Рассматривайте его как отдельный шаблон, ресурсы не нужно объединять с предыдущими внедренными шаблонами, даже если есть одинаковые библиотеки.
- Исходный HTML-шаблон в материалах в папке **html\_type2**.

## HTML-шаблон



- Оцените все страницы, предоставленные в рамках HTML-шаблона, чтобы определить корректное разделение на хедер, футер и контентную часть.

В этом шаблоне все что внутри тега `<section>` - контентная область, сам тег является частью общего «каркаса» и должен быть в шаблоне. В контентной части должно быть только содержание конкретной страницы.

### Создайте шаблон сайта

- Директорию шаблона задать `exam1_type2`, название шаблона для админки: **Экзамен N1 – TYPE2**.
- Созданный шаблон примените только для ранее созданного раздела `statistic_na`.
- Ресурсы (`/assets/`) разместите в директории шаблона.
- Административную панель управления не нужно подключать.
- Включите объединение, минифицирование и сжатие CSS и JS. Не забудьте про настройки платформы: вы можете указать в какой момент должны подключаться файлы JS.
- В языковые файлы вынесите только одну фразу в подвале сайта «Дашборд: место, где данные обретают смысл.»

**Конкурс: выиграй сертификат на 5000 руб.**

Участвуйте и получите шанс на покупку мечты в нашем магазине!...

Дашборд: место, где данные обретают смысл.

Designed by BootstrapMade



### Свойства страниц

- При анализе верстки вы заметите что тег `<section>`, для некоторых страниц, обладает специфичным классом CSS стилей. Например, для страницы **dashboard.html** это класс **dashboard**.
- Создайте свойство страницы, которое позволит указать CSS класс, код свойства укажите **page\_css\_class**.
- С помощью отложенной функции дополняйте CSS классы тега `<section>` в шаблоне сайта значением созданного свойства страницы.
- Заполните значение для страницы «Дашборд» и «Профиль» значениями из исходных HTML-страниц.

### Демо-контент

Разместите на этих страницах сайта демо контент из HTML-страниц шаблона. Проставьте пути для изображений (папка `/assets/img/` в шаблоне). Здесь только размещение HTML, без внедрения компонентов.

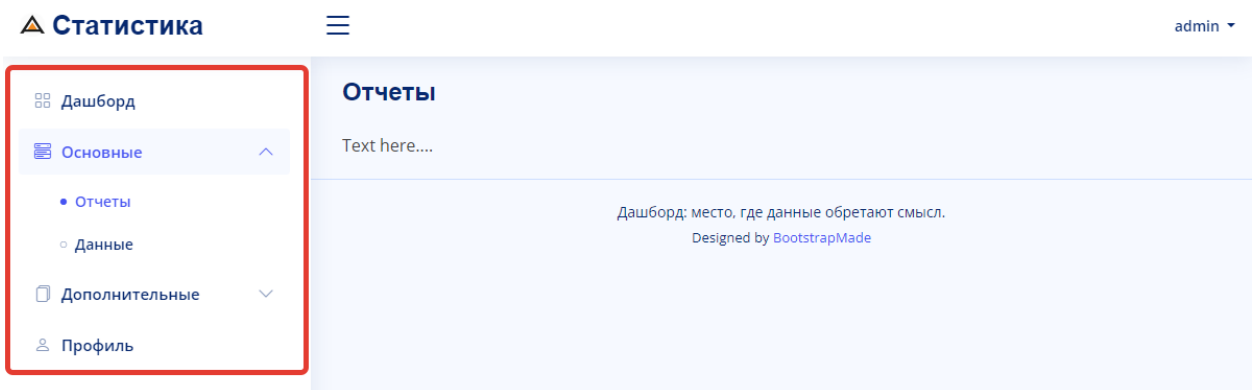
В файловой системе	Название на сайте	Контент
<code>/statistic_na/dashboard/index.php</code>	Дашборд	dashboard.html
<code>/statistic_na/profile/index.php</code>	Профиль	profile.html

### [ex1-510] Базовые компоненты

Реализуйте на сайте ряд задач с помощью компонентов. Создайте шаблоны компонентов на основе внедряемого HTML-шаблона.

#### Меню

Реализуйте навигацию по разделу статистики с помощью компонента «Меню» (menu). Меню должно быть двух уровней, строиться на основании ранее созданных типов меню **st\_first** для первого уровня и **st\_second** для второго.



- Кеширование в компоненте должно быть включено.
- Обратите внимание на исходную верстку, реализованное меню должно работать аналогично:
  - Пункты первого уровня с вложенными пунктами по клику показывают подменю, переход возможен только на подпункты.
  - Когда открыта страница второго уровня, то выделяется цветом и она и ее родительский пункт, список страниц второго уровня отображается раскрытым при загрузке страницы (пример в tables.html).

Для выделения сразу нескольких пунктов у компонента меню представлена настройка:

`"ALLOW_MULTI_SELECT" => "Y"`

- Посмотрите на исходные html файлы: для каждого пункта меню первого уровня используется иконка.
- Иконки задаются классами CSS стилей для определенного тега в шаблоне меню, файл стилей с иконками `assets\vendor\bootstrap-icons\bootstrap-icons.css`, он уже должен быть подключен у вас в шаблоне сайта.
- В файле меню задайте дополнительное свойство для пунктов меню, в котором можно указать CSS класс иконки. Код свойства укажите как `menu_ico`, задайте значения для пунктов меню в соответствии с исходными HTML-страницами. В шаблоне компонента меню подставляйте CSS класс иконки из созданного свойства.

#### [ex1-610] Информационные блоки

- Создайте тип информационного блока Статистика (**Statistic**), с ID - **statistic**.
- Создайте информационный блок Данные с символьным кодом **date**
- Создайте свойства:

Название	Тип	Код
Товар	Строка	PRODUCT
Категория товара	Строка	PRODUCT_CATEGORY
Город	Строка	CITY
Количество	Число	QUANTITY

- Импортируйте данные из файла `tables_export/items.csv` в материалах, в нем первая колонка – названия (просто числа), остальные колонки – свойства.
- Контент инфоблока должен быть доступен только пользователям группы **Аналитики**.

#### [ex1-810] Раздел сайта с динамической информацией и ЧПУ

Замените страницу `/statistic_na/main/table/` на раздел с динамической информацией с помощью комплексного компонента **news**, выведите список элементов из информационного блока «Данные» с переходом на детальную страницу элемента.

#### Общие требования:

- Поддержку «эрмитажа», редактирование элементов над сайтом, реализовывать в этом задании не нужно.
- Кеширование в компонентах и настройках платформы должно быть включено.
- В шаблонах компонентов текст выносите в языковые файлы.
- Для упрощения во время экзамена: внутри шаблонов простых компонентов не учитывайте никакие опции компонента, выводите безусловно тот контент, что предусмотрен внедряемой HTML-версткой.

#### Комплексный компонент:

- Шаблон комплексного компонента назовите **table**, разместите его в созданном шаблоне сайта **exam1\_type2**.

Обратите внимание, такой шаблон не появится в списке шаблонов в визуальных настройках компонента, его нужно указать в коде, каждый раз при сохранении настроек. Классический момент – редактировать файл с вызовом компонента и в редакторе кода и через визуальный редактор – обязательно перетрете изменения и не поймете сразу, выберите что-то одно.

- Настройте компонент на работу в ЧПУ режиме, адреса страниц:
  - Список сразу в разделе (/)
  - Детальный просмотр по ID (/#ELEMENT\_ID#/)
- В этом задании комплексный компонент будет использоваться в очень простом виде, большая часть стандартного функционала не понадобится. Реализуйте так:
  - На странице списка (`news.php`) нужен только компонент `bitrix:news.list`
  - На странице элемента (`detail.php`) нужен только компонент `bitrix:news.detail`

Все остальное из стандартного шаблона комплексного компонента использовать не нужно.

- Создайте шаблоны простых компонентов в составе комплексного.

#### Список

- Для списка элементов шаблон создайте на основе верстки страницы `html_type2/tables.html`

**Свойства**

**Данные**

№	Товар	Категория	Город	Количество
1	Очки	Аксессуары	Иваново	17
2	Проектор	Электроника	Мурманск	12
3	Шарф	Одежда и обувь	Тамбов	15

**Название**

Дашборд: место, где данные обретают смысл.  
Designed by BootstrapMade

- Текст выносите в языковые файлы.
- Включите постраничную навигацию, используйте для нее штатный шаблон **bootstrap\_v4**, в шаблоне выведите ее вместо <!--Pagination here-->

#### Страница детального просмотра

- Для детальной страницы шаблон создайте на основе верстки страницы tables-detail.html

**Данные - 104**

ID	Товар
[ID]	Стиральная машина
	Бытовая техника
	Калуга
	6

**Свойства**

Назад к списку

- Используя настройки инфоблока (SEO), установите метаданные для страницы просмотра элемента по шаблонам:
  - Заголовок браузера (META TITLE): «[Поле «Название»] :: Просмотр элемента»



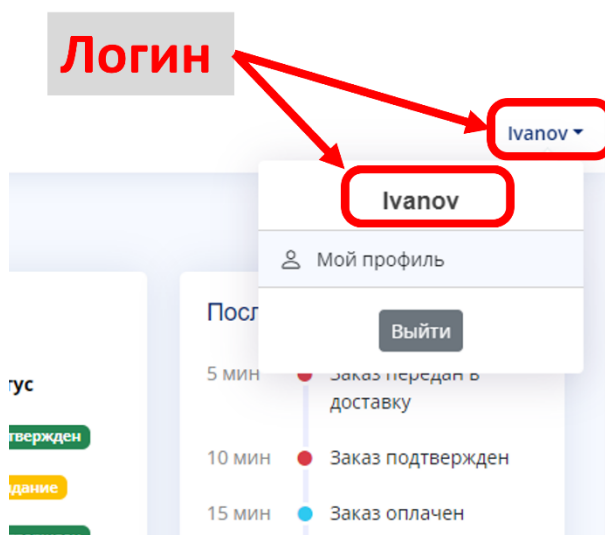
- Заголовок страницы (Заголовок товара): «Данные по товару :: [Поле «Название»] :: [Свойство «Товар»]»

Где в квадратных скобках указаны поля или свойства элемента инфоблока.

#### [ex1-1100] Доступ к сайту

Реализуйте сценарии доступа к сайту.

- В шаблонах компонентов текст выносите в языковые файлы.
- В настройках платформы:
  - Отключите возможность пользователям самостоятельно регистрироваться.
  - Отключите авторизацию через внешние сервисы.
  - Включите CAPTCHA при восстановлении пароля.
- В header.php реализуйте выпадающий блок с логином, ссылкой на профиль и кнопкой выхода:



- Для решения используйте компонент `system.auth.form`, внедрите в него верстку из HTML-шаблона. Для неавторизованного пользователя компонент ничего не выводит.
- Ссылку на профиль укажите `/statistic_na/profile/`, она должна строиться на основе данных компонента, задается параметром "PROFILE\_URL".
- Реализуйте сценарий выхода по кнопке «Выйти».
- Реализуйте стандартные сценарии.
  - При открытии страниц в разделе **statistic\_na** для не авторизованных пользователей платформа покажет формы: авторизации, восстановления и сброса пароля на основании соответствующих параметров в URL.
  - Адаптируйте внешний вид этих форм, внедрите верстку в соответствии с шаблоном раздела **statistic\_na**, используйте для этого компоненты `system.auth.*` и верстку для форм на странице `auth-form.html`
  - Все фразы из шаблона выносите в языковые файлы.

- Настройте стандартный почтовый шаблон для события сброса пароля (Запрос на смену пароля [USER\_PASS\_REQUEST]): измените в нем ссылку, стандартная ведет на раздел **/auth/**, измените его на раздел **/statistic\_na/**.
- Проверьте себя - проиграйте сценарий сброса пароля для пользователя **user2** с помощью настроенных форм и письма (если вы используете демо сервер для подготовки, то отправленные письма в директорию /home/bitrix/mail в виде файлов, на рабочем столе есть ярлычок mailbox).

При сбросе пароля должно появиться письмо, с ссылкой на раздел **/statistic\_na/**, при переходе по ссылке должна появиться форма для установки нового пароля, пароль должен смениться. Форма авторизации, восстановления и смены пароля должны быть адаптированы.

- Доработайте ранее внедренный шаблон меню.

Определяйте, если страницы или раздел пункта меню не доступны текущему пользователю, то не выводите этот пункт меню.

Нужные данные для решения этой задачи стандартный компонент меню уже передает в шаблон компонента в \$arResult.

Должно получиться так:

The screenshot shows a web application interface. On the left, there is a navigation menu with a blue header containing a triangle icon and the text "Статистика". To the right of the header is a hamburger menu icon. The main content area has a light blue background and is titled "Авторизация" in bold blue text. In the center of this area is a white login form with a blue border. The form contains the following elements:
 

- A heading: "Пожалуйста, авторизуйтесь:"
- A label "Логин:" followed by a text input field containing the value "demo".
- A label "Пароль:" followed by a password input field with a blue border and a vertical cursor.
- A checkbox labeled "Запомнить меня".
- A blue button with the text "Войти".
- A link "Забыли свой пароль?" followed by a blue link "Восстановить".

 At the bottom of the main content area, there is a footer with the text "Дашборд: место, где данные обретают смысл." and "Designed by BootstrapMade".

## Запрос пароля

Если вы забыли пароль, введите  
логин или E-Mail.  
Контрольная строка для смены  
пароля будет выслана вам по E-  
Mail.

Логин

E-Mail

Выслать

[Вернуться к авторизации](#)

Дашборд: место, где данные обретают смысл.  
Designed by BootstrapMade

## Изменение пароля

Логин

Контрольная строка

Новый пароль

Подтверждение пароля

Изменить пароль

Пароль должен быть не менее 6  
символов длиной.

[Авторизация](#)

Дашборд: место, где данные обретают смысл.  
Designed by BootstrapMade

На этом все! Вы молодец 😊

Проверьте себя, все ли требования к решению заданий соблюдены? **Не забыли включить кеширование в настройках компонента или продукта, если отключали на время выполнения заданий с динамической информацией?**