# Задание 1 (html)

Пример персонального сайта: <https://yanakhodkina.com/>

1.1. Создать макет html - страницы для персонального сайта (сохраните как example.html).

* навигация (меню):
  + логотип(является ссылкой на главную страницу),
  + о себе(главная),
  + компетенции (ссылка на блок 2),
  + резюме,
  + портфолио (мои работы),
  + контакты;
* шапка (заголовок: веб - технологии или любой другой);
* основное содержимое (заполнить случайным текстом);
* подвал:
  + сайт разработан ФИ, Copyright © 2021,
  + иконки социальных сетей.

В Браузере страница должна выглядеть как на рисунке 1.1:



рисунок 1.1 (задание 1.1)

1.2. На основе макета создать страницы:

главная (лендинг) содержащий разделы:

* блок 1 (изображение, заголовок, текст), рисунок 1.2.



рисунок 1.2 (задание 1.2)

* блок 2 (Заголовок, 3 или более изображений, под каждым заголовок), рисунок 1.3.



рисунок 1.3 (задание 1.2)

* блок 3 (2 списка), рисунок 1.4

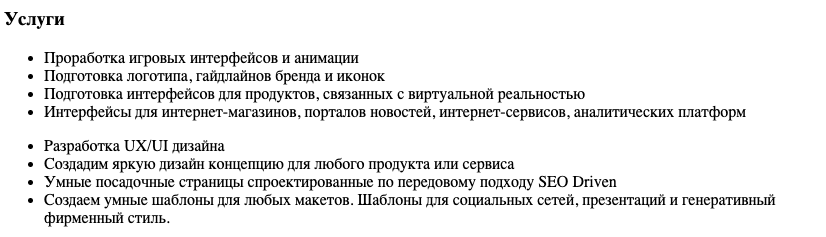


рисунок 1.4 (задание 1.2)

* блок 4 (заголовок, случайный текст, кнопка), рисунок 1.5

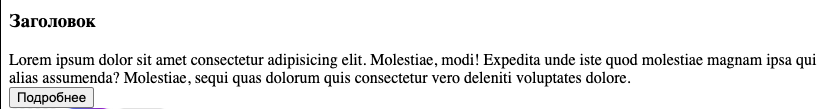


рисунок 1.5 (задание 1.2)

резюме (оформлено в виде таблицы), рисунок 1.6:

в таблице перечислены даты и место работы или учебы, должностные обязанности (для наполнения страницы можно использовать готовое резюме);

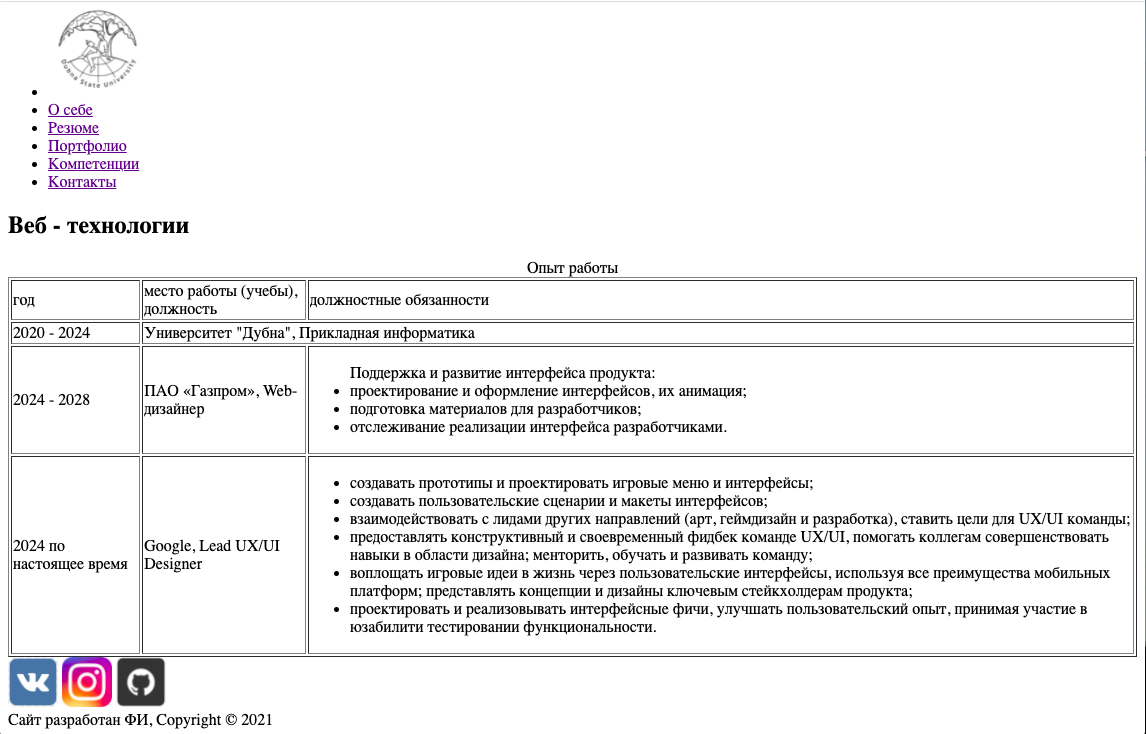


рисунок 1.6 (задание 1.2)

портфолио, рисунок 1.7

несколько блоков, каждый из которых содержит: заголовок, текст, изображение;

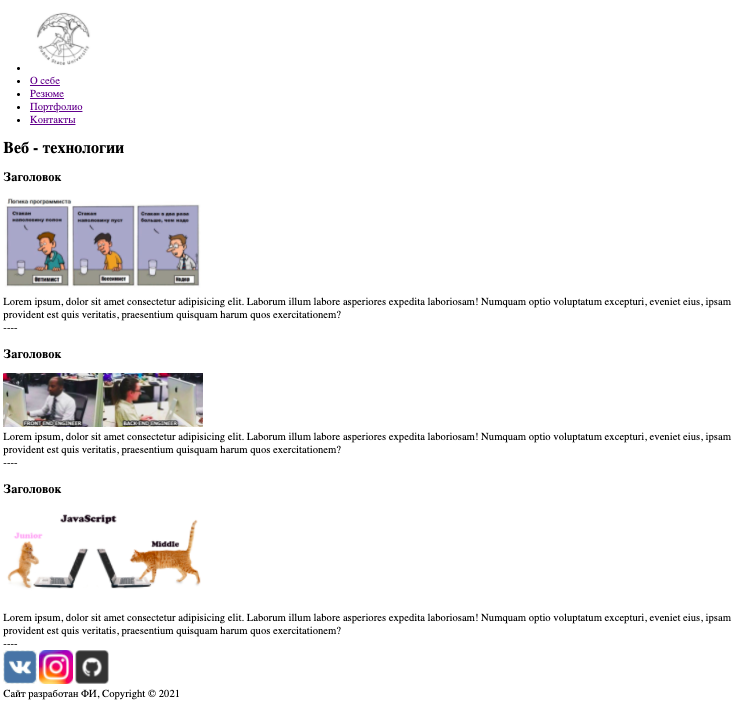
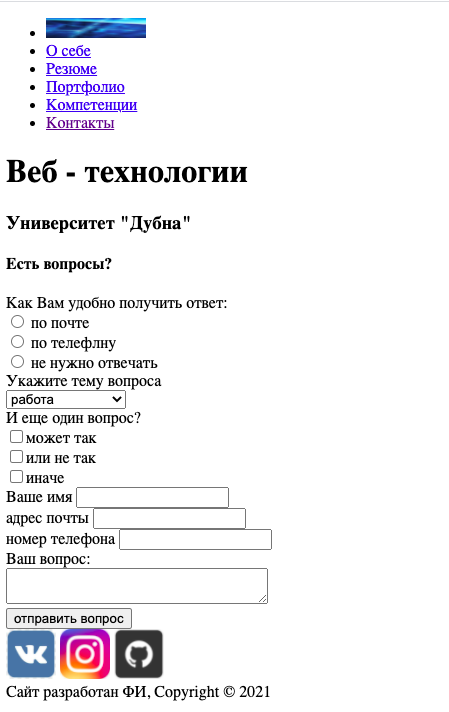


рисунок 1.7 (задание 1.2)

контакты

иконки социальных сетей, адрес электронной почты, телефон.

форма обратной связи.



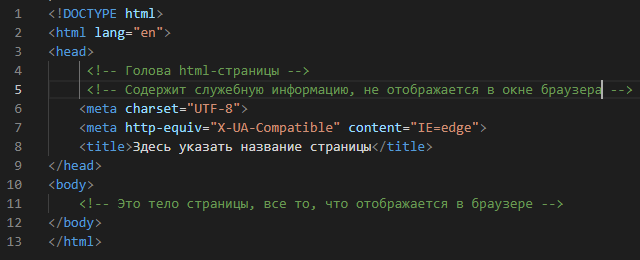
# Справочные материалы

Сайт должен иметь следующую структуру:

* index.htm - стартовая страница сайта
* images - каталог с изображениями
* pages - каталог со страницами

### Использование плагина Emmet

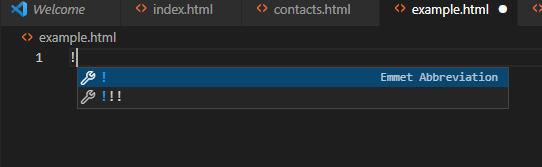
Структура html-страницы:



<meta name="viewport" content="width=device-width, initial-scale=1.0"> - используется только в том случае, если применяется адаптивная верстка.

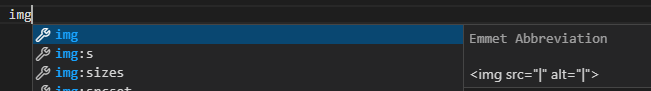
Для создания html-страниц удобно использовать плагин emmet (https://emmet.io/).

Для создания стандартной html-страницы достаточно набрать символ ! и нажать enter:



Для добавления на страницу тега, достаточно написать название и нажать на enter, в результате будет добавлен открывающий и закрывающий тег и необходимы атрибуты.

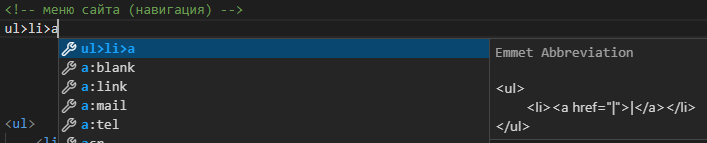
Например, если набрать img:



то будет добавлен тег img и необходимые атрибуты:



Можно создавать вложенные теги, например, если набрать ul>li>a:



то будет добавлена конструкция:

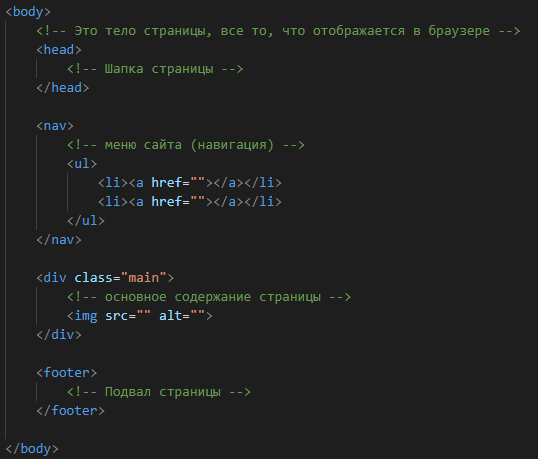


Для генерации случайного текста используется сокращение lorem (lorem\*5 выведет текст 5 раз).

Плагин Emmet по умолчанию содержится в VS Code, также его можно установить и для других редакторов. Подробнее про сокращений для ускорения верстки можно почитать по ссылке: <https://webdesign-master.ru/blog/html-css/2.html> (Справочник EMMET).

### Тело документа

Тело html-страницы имеет следующую структуру:



Справочник по html-тегам: <https://html5book.ru/html-html5/>

Добавление изображения

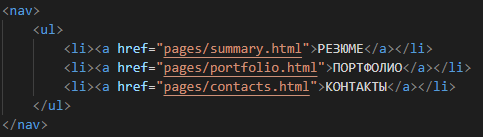
<img> добавляет изображение,   
атрибут src задает абсолютный или относительный путь к изображению;  
атрибут alt задает текст, который будет выведен, если изображение не доступно;

Элемент <img> является строчным, поэтому рекомендуется располагать его внутри блочного элемента <div>.



Меню сайта

Меню сайта рекомендуется оформлять с помощью списка:



Форма обратной связи

Для создания формы обратной связи используются html-формы (<https://html5book.ru/html5-forms/>)



Добавление иконок https://fontawesome.com/

Подключить библиотеку можно указав в голове документа link:

<link href="https://use.fontawesome.com/releases/v5.0.13/css/all.css" rel="stylesheet">

Теперь можно добавлять на страницы иконки, используя код:

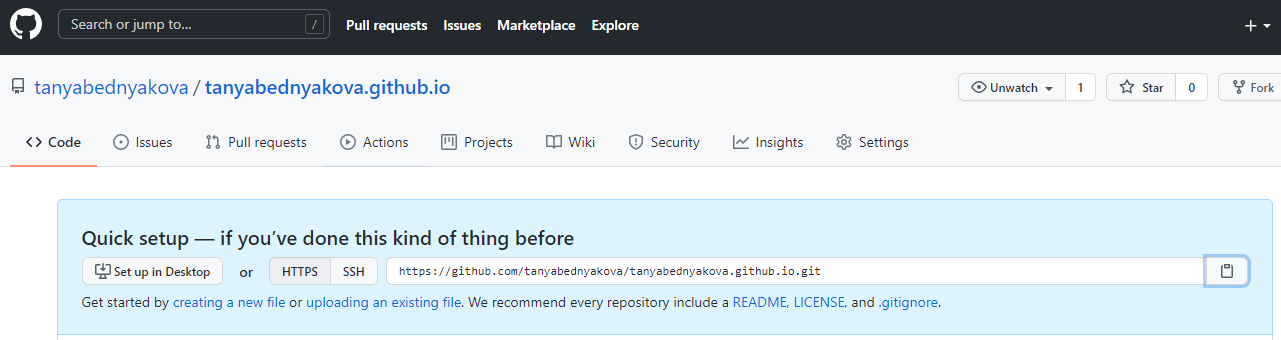
Со

Создание репозитория на GitHub

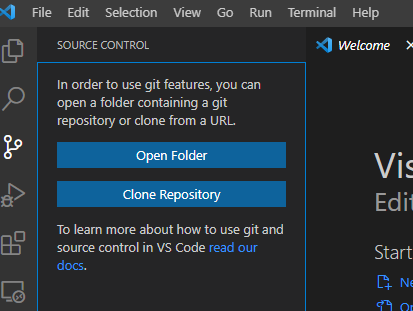
Авторизоваться (зарестрироваться) на <https://github.com/>

Создайте репозиторий на GitHub с именем логин\_на\_gitHub.github.io (это позволит использовать репозиторий как хостинг)

Запустите VS Code и клонируйте репозиторий:



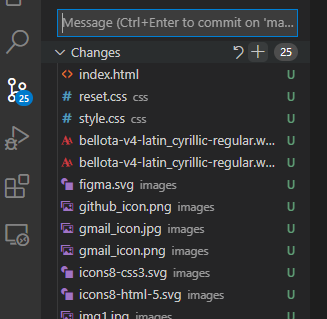
* скопируйте строку с адресом репозитория;
* в VS Code откройте вкладку с Git и нажмите “клонировать репозиторий”



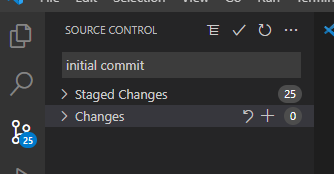
или откройте терминал и наберите команду: 

и откройте в VS Code папку с репозиторием.

* Перенесите в каталог репозитория уже созданные файлы, если они есть.
* На вкладке Git нажмите Stage all changes (символ: +)



в текстовое поле введите название и сделайте коммит (нажать на ):



Для загрузки изменений в репозиторий на GitHub (команда push) необходимо связать локальный и удаленные репозитории и выполнить начальную настройку Git, задать имя пользователя.

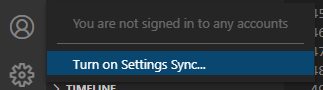
С помощью командной строки:



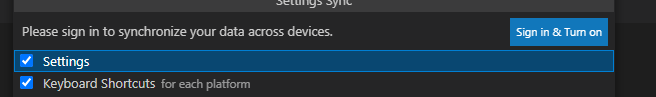
опцию global можно не использовать, тогда настройка будет только для данного репозитория.

или непосредственно через VS Code:

в нижнем левом углу:



вверху окна появится окно setting sync, здесь нужно нажать sign in & Turn on



и выбрать Sign in with GitHub



после чего будет открыт браузер на странице подтверждения синхронизации GitHub и VS Code.

Для просмотра сайта введите в адресную строку браузера: логин\_на\_gitHub.github.io

(эту же строку нужно загрузить в lms)

# Задание 2 (CSS)

## Задание 2.1

Добавить каталог css.

Добавить файл reset.css для сбрасывания стилей.

Добавить файл style.css

Подключить файлы со стилями ко всем страницам:

<link rel="stylesheet" href="css/style.css">

Выбрать и скачать шрифт: <https://google-webfonts-helper.herokuapp.com/fonts>

(для поиска шрифтов, поддерживающих кириллицу в поле поиска нужно написать: cyrillic)

Распаковать шрифт в каталог fonts и подключить шрифты, добавив в style.css необходимы код (код можно скопировать на странице со шрифтом).



## Задание 2.2

Оформите горизонтальное меню, как показано на рисунке 1 или 2:



рисунок 1

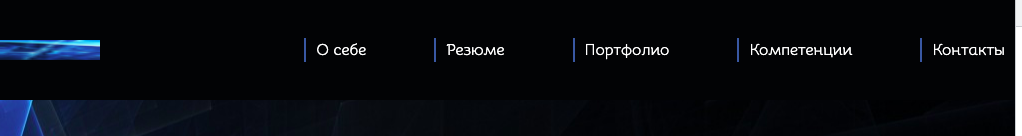


рисунок 2

Оформление меню можно разделить на 3 этапа:

1. расположить по горизонтали, выровнять по правому краю, логотип сместить к левому краю;

Для этого удобно использовать значение flex для свойства display,

Чтобы логотип занимал больше места, чем остальные пункты меню можно использовать свойство flex-grow, для пунктов меню flex-grow: 1; для логотипа flex-grow: 5;

или свойство flex-basic:14% для пунктов меню и flex-basic:30% для логотипа (важно, что 14\*5 + 30 = 100)

(подробнее про flex: <https://tproger.ru/translations/how-css-flexbox-works/>)

1. для каждого пункта добавить левую рамку, отступ, для последнего пункта уменьшить размер и отступ;

Чтобы задать специальные свойства для последнего пункта можно использовать псевдокласс last-child.

1. добавить подчеркивание при наведении на пункт меню и для активного пункта меню.

Цвет фона лучше задать в отдельном классе, так как это свойство будет использоваться и для других элементов.

Имена классов в css должны содержать название элемента, для которого задаются свойства и названия самих свойств (по возможноси). Для примера посмотрите названия классов на сайте github



Каждый класс должен решать одну задачу. Например, неправильно если класс одноверменно задает для элемента flex-разметку и цвет, размер, шрифт текста. Это затруднит дальнейшее использование такого класса.

Для оформения меню можно использовать классы:  
.nav-flex для задания flex-разметки

.nav-flex-item для задания стилей для каждого пункта меню

.nav-flex-item-5 для задания стилей для логотипа

Для задания стилей для ссылки можно использовать вложенные селекторы:

nav a - стили для ссылки

Класс .background-color-dark задает цвет фона и текста на темном фоне и будет использоваться не только для оформления меню.

## Задание 2.3.

Оформите шапку, как показано на рисунке:

## 

пример стилей:

header

{

text-align: center;

background-image: url(../images/header-bg\_big.jpeg);

height: 400px;

background-repeat: no-repeat;

background-position: left;

background-color: #020305;

background-attachment: fixed;

background-size: cover;

padding-left: 250px;

padding-top: 100px;

}

<header>

<!-- Шапка страницы -->

<!-- использование тегов h1 и h3 для поисковых машин -->

<h1 class="header-title">Веб - технологии</h1>

<h3 class="header-title">Университет "Дубна"</h3>

</header>

## Задание 2.4.

Оформите подвал, как показано на рисунке:

## 

## Для разметки использовать два класса:

## flex-row-div flex-разметка, располагает элементы в строку

## flex-row-div-between-center задает выравнивание элементов для flex-элементов

## 

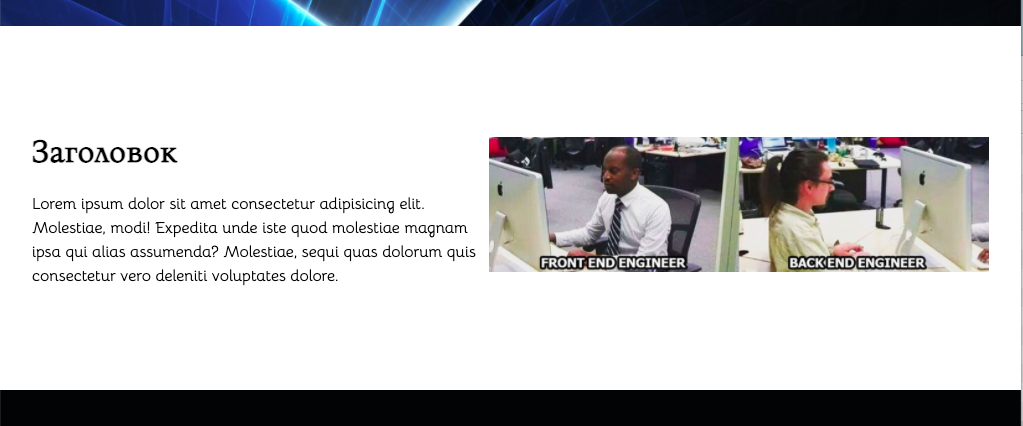
## Задание 2.5.

Оформите блоки страницы “О себе” ().

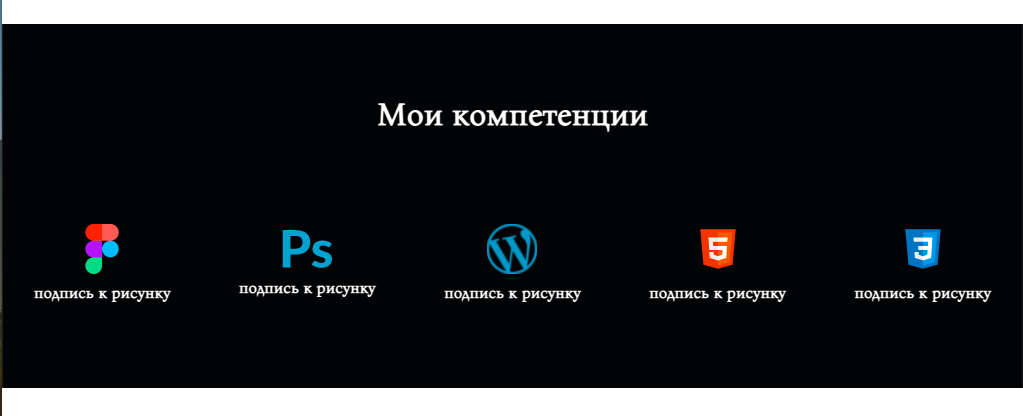
Высота каждого блока не менее 250px

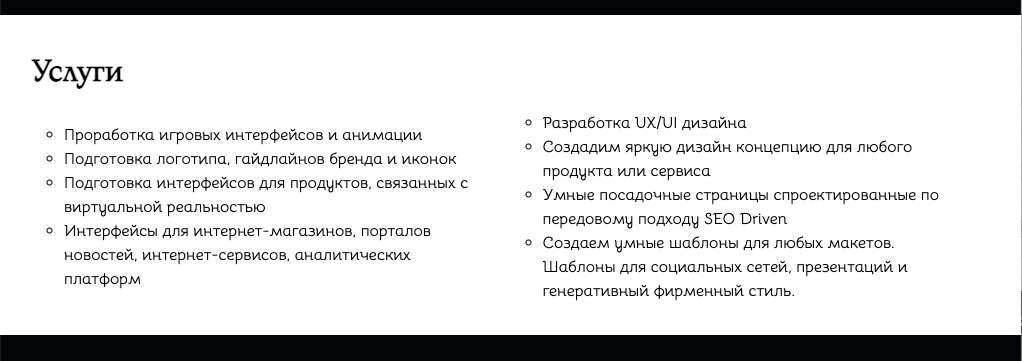
Фон занимает всю ширину страницы, а содержимое не более чем 1200px. Если ширина менее чем 1200px, то добавляется отступ от содержимого до края страницы 2 rem.

Блок 1



Блок 2



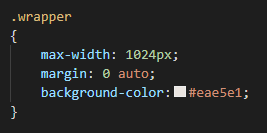
Блок 3

## Блок 4

## 

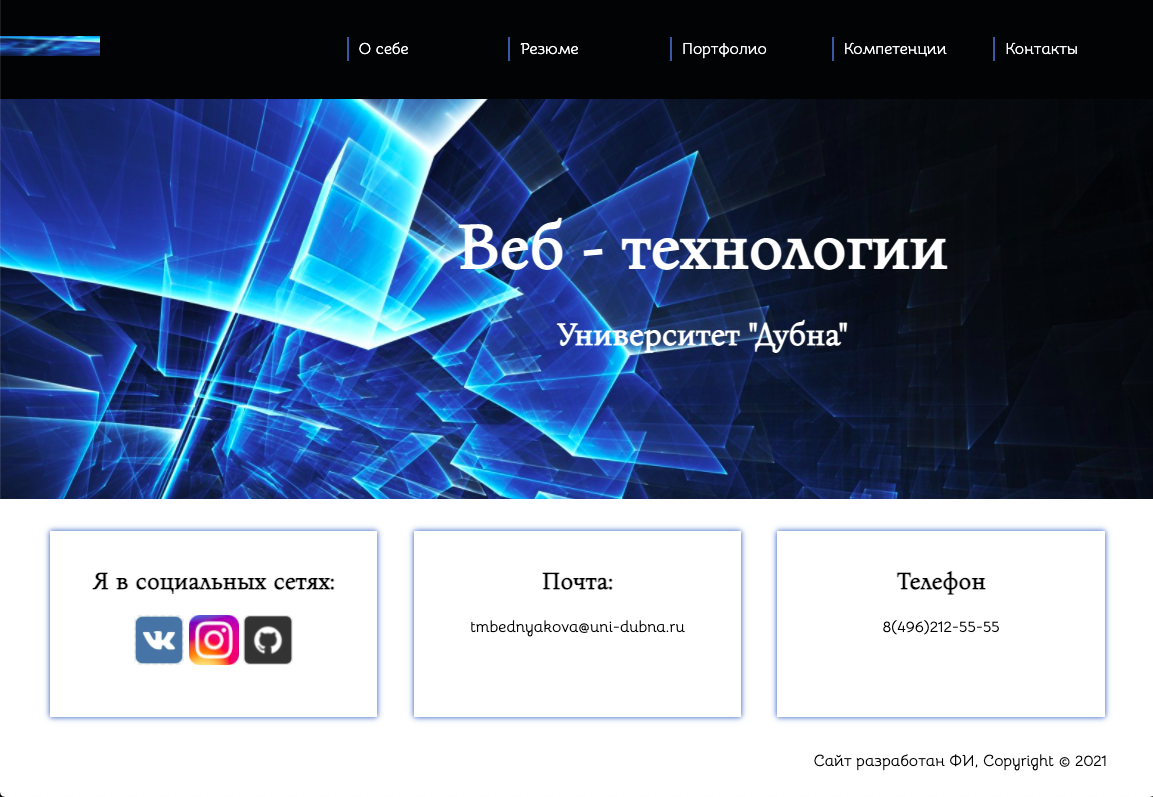
Для горизонтального выравнивания используется комбинация из двух свойств: блоку задается ширина (свойство width или max-width) и для полей справ и слева выставляется значение auto.

Данные свойства удобно задать для класса и обернуть в элемент с данным классом содержимое основных контейнеров.



## Задание 2.5.

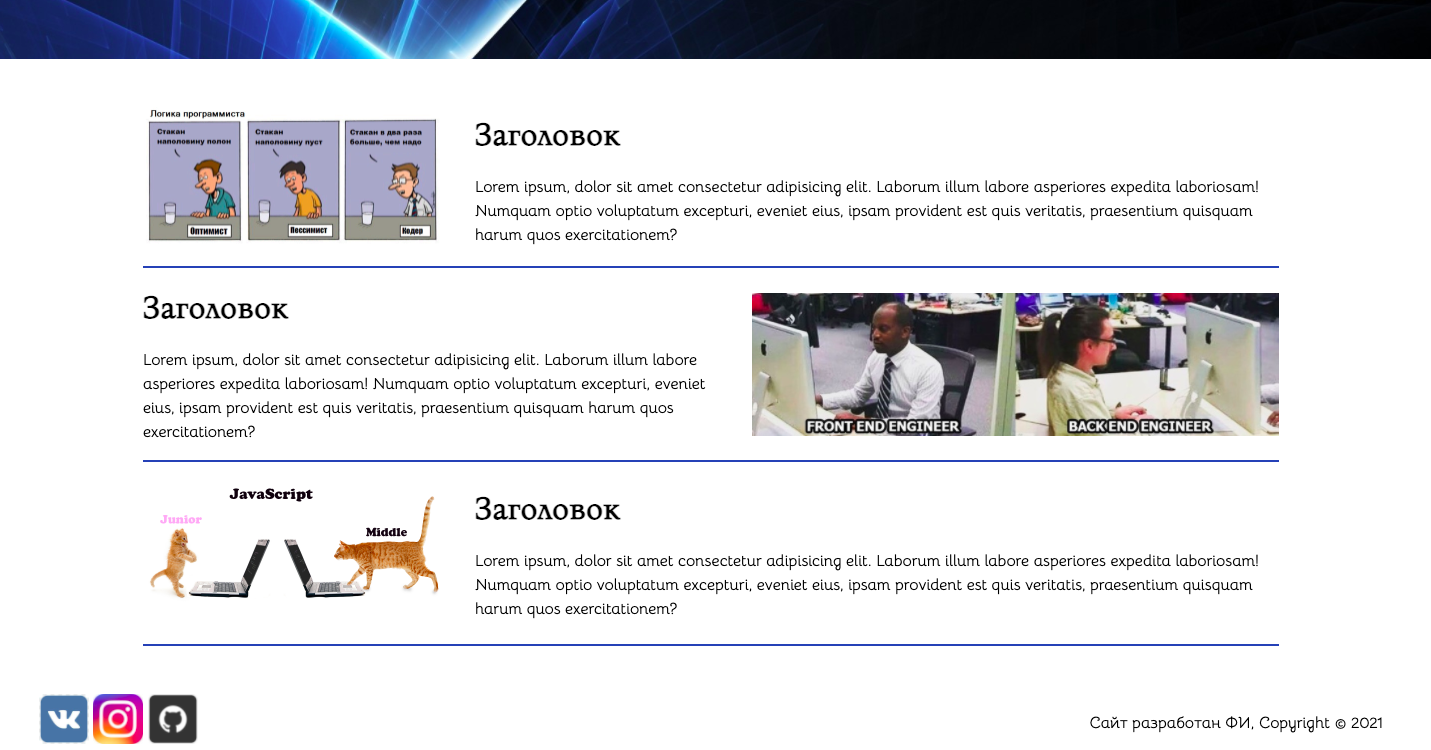
Оформите страницу Контакты, как на рисунке:



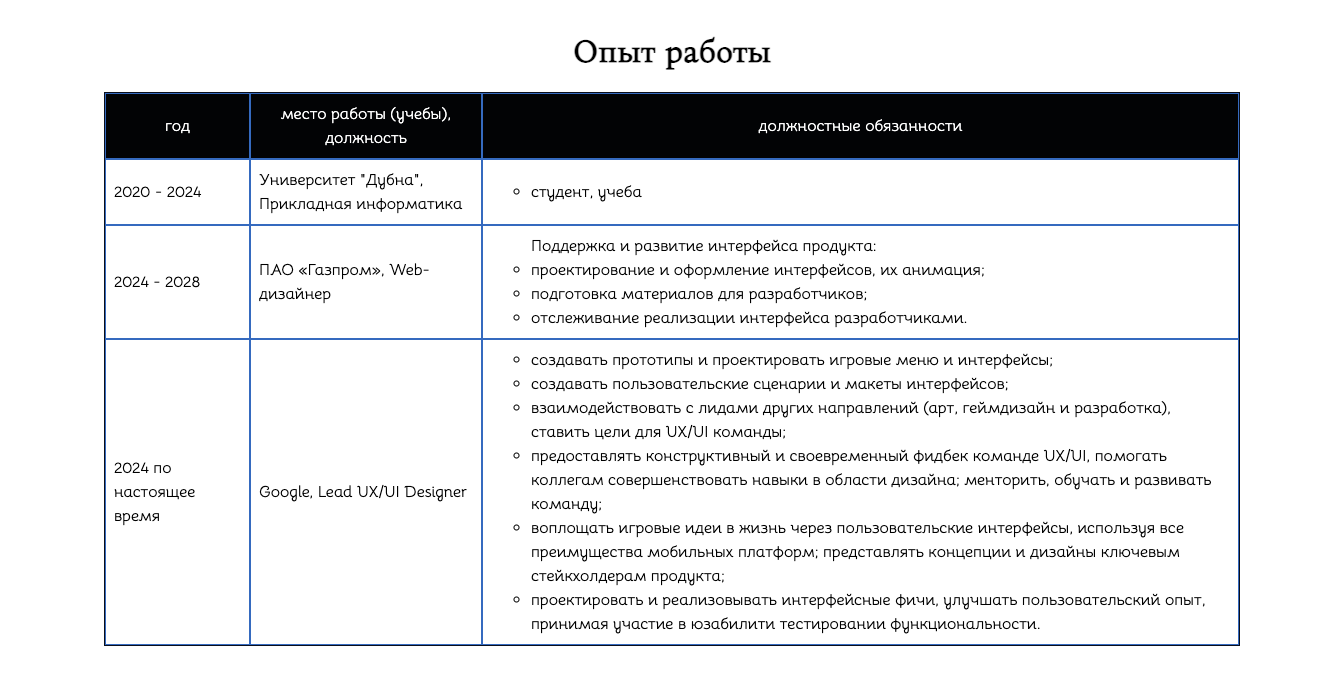
## 

## Задание 2.6.

Оформите страницу Портфолио, как на рисунке:



## Задание 2.7.

Оформите страницу Резюме, как на рисунке:

## Задание 3.

Адаптируйте сайт для мобильных устройств.