



АСПЕКТЫ ПРЕПОДАВАНИЯ ТЕМЫ «ОСНОВЫ ВЕБ-КОНСТРУИРОВАНИЯ» НА ПОВЫШЕННОМ УРОВНЕ

Е. Н. Войтехович

Лицей Белорусского государственного университета, Беларусь

Аннотация. В статье рассматриваются: аспекты изучения на повышенном уровне темы «Основы веб-конструирования» в рамках учебного предмета «Информатика», вопросы современного подхода к верстке веб-страниц. Предлагается авторская разработка практической работы по созданию веб-страницы на основе модели верстки Flexbox. Излагается теоретический материал, необходимый для выполнения этой работы.

Ключевые слова. Информатика, веб-конструирование, повышенный уровень, flexbox модель, практическая работа.

ASPECTS OF TEACHING TOPIC "BASIS OF WEB DESIGN" AT THE INCREASED LEVEL

E. N. Voytehovich

Lyceum of Belarusian State University, Belarus

Abstract. The article describes: aspects of studying at an advanced level the topic “Fundamentals of Web Designing” within the framework of the subject “Informatics”, as well as issues of a modern approach to layout web pages are considered. The author's development of practical work how to create a web page based on the Flexbox layout model is proposed. The theoretical material which is necessary to carry out this work is described.

Keywords. Informatics, web design, advanced level, flexbox model, practical work.

Введение

В последние годы в Республике Беларусь активно воплощается в жизнь идея профильного обучения в старших классах учреждений общего среднего образования. Вводится изучение различных учебных предметов на повышенном уровне [0]. В силу бурного развития информационно-коммуникационных технологий и их внедрения в

образовательный процесс, изучению учебного предмета «Информатика» отводится особая роль. С 2020 года планируется предоставление учащимся возможности изучения этого предмета на третьей ступени обучения на повышенном уровне.

В Лицее БГУ учебный предмет «Информатика» на повышенном уровне изучают с 2015 года. Учебная программа рассчитана на 210 часов (3 часа в неделю) для X-XI классов. Содержание учебной программы поддерживает все основные дидактические линии изучения информатики на базовом уровне. Вместе с тем, эта программа позволяет уделить изучению некоторых тем большее количество учебных часов. Одна из дидактических линий программы – овладение навыками в области информационных технологий и компьютерных коммуникаций. В учебной программе XI класса эта линия реализована, в том числе, и в рамках темы «Основы веб-конструирования». Для изучения этой темы на повышенном уровне в Лицее БГУ отводится 22 часа, тогда как на базовом уровне – 12 часов [2]. При этом разница заключается не только в соотношении часов, отводимых на изучение темы, но и в глубине изучения, ширине охвата теоретического и практического материала.

В современном мире веб-конструирование – востребованное направление деятельности, и можно с большой вероятностью предположить, что многие из сегодняшних учащихся в дальнейшем столкнутся с этим видом деятельности. Поэтому важно уже в школьном курсе показать основные идеи и современные подходы к разработке веб-страниц.

Нами были изучены разработки дидактических материалов (практических работ) по теме «Основы веб-конструирования» как в отечественных [3, 4], так и зарубежных источниках [5, 6]. Однако, не удалось найти дидактические материалы, ориентированные на учащихся и отвечающие современному уровню развития технических средств и программного обеспечения. Поэтому и возникла необходимость разработки таких материалов. В настоящей статье представлено описание одной из практических работ по теме «Основы веб-конструирования» и краткий теоретический материал, необходимый для ее выполнения. Работу можно выполнять с поддержкой любой операционной системы, понадобится только какой-либо текстовый редактор с подсветкой html-кода и браузер.

Блочная верстка. Flexbox модель

Современные подходы к созданию веб-страниц основаны на блочной верстке. Блоки – это контейнеры `div`, которые не имеют определенного смыслового содержания. В них можно помещать различные элементы html-кода. Для оформления блоков используются стандартные свойства: `border` (задает параметры границ блока), `margin`

(задает отступы от внешней границы блока до границ страницы или до соседних элементов), `padding` (задает отступы от внешней границы блока до его содержания).

Одна из современных моделей верстки веб-страниц – CSS Flexible Box Layout Module (CSS модуль для макетов с гибкими блоками), коротко – flexbox (гибкая коробка), создана, чтобы избежать недостатков при создании самых разных html-конструкций, в том числе адаптированных под различные ширину и высоту, и сделать верстку логичной и простой. Такая модель верстки становится все популярнее за счет удобного использования при создании каркасов как веб-страниц в целом, так и отдельных элементов [7].

Основные преимущества flexbox:

- все блоки легко делаются «резиновыми»;
- элементы могут сжиматься и растягиваться по заданным правилам, занимая нужное пространство;
- легко установить выравнивание элементов по вертикали, горизонтали и базовой линии текста;
- элементы могут автоматически выстраиваться в несколько строк/столбцов.

Flexbox состоит из контейнера (flex-контейнера) и дочерних элементов (flex-блоков) (рис. 1).

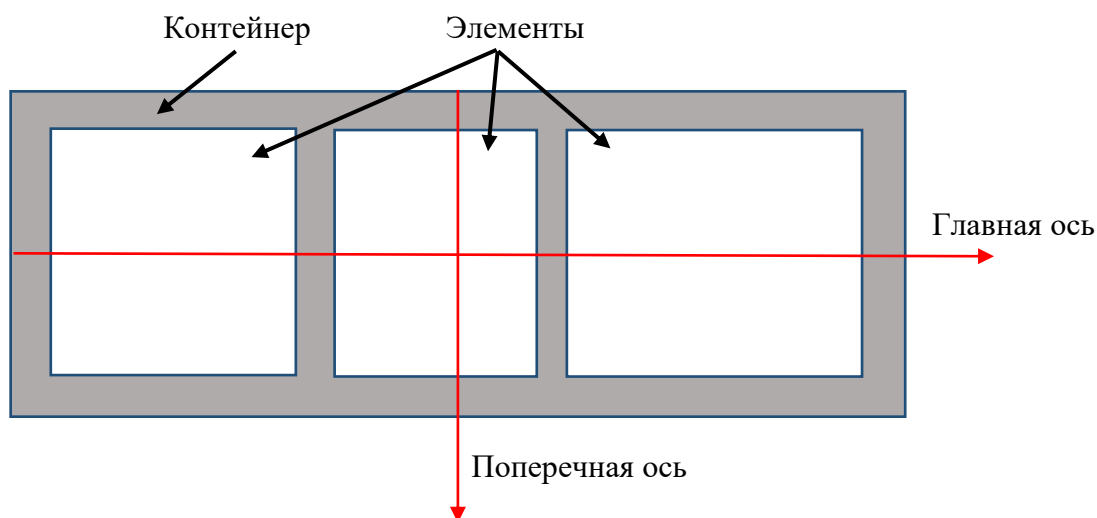


Рисунок 1 – Структура Flexbox

Для включения flexbox, любому html-элементу достаточно присвоить css-свойство `display:flex`. Пример, когда свойство `display:flex` присвоено блоку, в котором находятся три дочерних блока:

```

< style>
  .flex{ display: flex; }
</style>
<div class="flex">
  <div>блок 1</div>
  <div>блок 2</div>
  <div>блок 3</div>
</div>

```

После включения flex-свойства, внутри контейнера создаются две оси (рис. 1). Главной осью flex-контейнера является направление, в соответствии с которым располагаются все его дочерние элементы. Поперечной осью называется направление, перпендикулярное главной оси. Главная ось по умолчанию располагается слева направо. Поперечная – сверху вниз. Направление главной оси flex-контейнера можно задавать, используя базовое css-свойство `flex-direction`.

Flexbox содержит разные css-свойства для управления всей flex-конструкцией. Одни нужно применять к flex-контейнеру, а другие к элементам этого контейнера.

Свойство flex-контейнера `justify-content` задает выравнивание дочерних элементов по главной оси и может принимать различные значения (рис. 2):

- `flex-start` (значение по умолчанию), дочерние элементы прижаты к началу главной оси;
- `flex-end`, дочерние элементы прижаты к концу главной оси;
- `center`, дочерние элементы располагаются в центре главной оси;
- `space-between`, первый дочерний элемент располагается в начале главной оси, последний дочерний элемент – в конце, все остальные дочерние элементы равномерно распределены в оставшемся пространстве;
- `space-around`, все дочерние элементы равномерно распределены вдоль главной оси, разделяя все свободное пространство поровну.

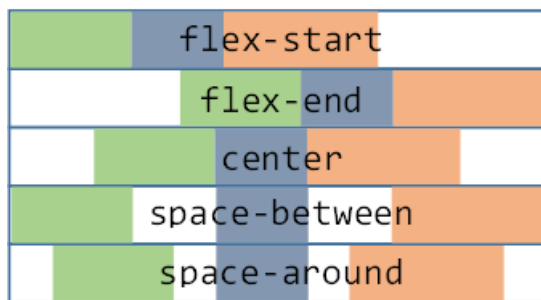


Рисунок 2 – Значения свойства `justify-content`

Свойство `align-items` flex-контейнера задает выравнивание дочерних элементов по поперечной оси внутри ряда и может принимать следующие значения (рис. 3):

- `flex-start`, дочерние элементы прижаты к началу поперечной оси;
- `flex-end`, дочерние элементы прижаты к концу поперечной оси;
- `center`, дочерние элементы располагаются в центре поперечной оси;
- `stretch` (значение по умолчанию), дочерние элементы растянуты, занимая все доступное место по поперечной оси.

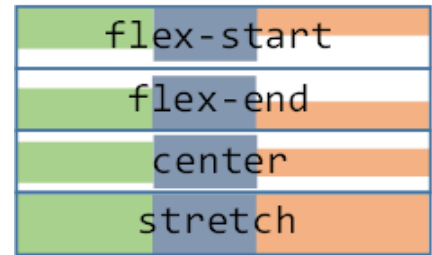


Рисунок 3 - Значения свойства `align-items`

По умолчанию дочерние элементы flex-контейнера располагаются в одну линию. За расположение содержимого flex-контейнера в несколько строк отвечает свойство `flex-wrap`, которое может принимать значения:


- `nowrap` (значение по умолчанию), блоки расположены в одну линию слева направо;
- `wrap`, блоки расположены в несколько горизонтальных рядов, следуя друг за другом слева направо (если не помещаются в один ряд);
- `wrap-reverse`: то-же что и `wrap`, но блоки располагаются в обратном порядке (справа налево).

Практическая работа

Задания выполняются с использованием заготовок html и css файлов и файлов с изображениями озер (см. Приложение 1).

Задание 1. Использование блоков

Внести указанные ниже изменения в документ `web_1.html`. Параметры форматирования описать в css-файле.

1. Добавить фоновое изображение.
2. Заголовок заключить в блок и применить к нему соответствующее форматирование:
 - а. Выбрать на <https://fonts.google.com> параметры шрифта по своему усмотрению.
 - б. Нажать кнопку  в блоке с выбранными параметрами.
 - в. Развернуть окно Family Selected и скопировать ссылку (например, `<link href="https://fonts.googleapis.com/css?family=PT+Sans+Caption:700&display=swap&subset=cyrillic" rel="stylesheet">`).

- г. Вставить ссылку в заголовок html-документа.
- д. Скопировать стиль css (например, `font-family: 'PT Sans Caption', sans-serif;`) и применить его для блока с заголовком.
3. Изображение и текст заключить в блоки.
4. Расставить в тексте спецсимволы (неразрывные пробелы). Применить к блоку с текстом стилевое описание по своему усмотрению (например: `background: aliceblue; width: 30%; padding: 8px; font: 13pt Arial;`).
5. Создать flex-контейнер и поместить в него блоки с изображением и текстом. Задать для flex-контейнера свойства (`display: flex; justify-content: center;`).

С примером выполнения задания можно ознакомиться в *Приложении 1*.

Задание 2. Выравнивание блоков

Продолжить работу с документом *web_1.html*:

1. Добавить заголовок веб-страницы. Форматирование шрифта выбрать самостоятельно (см. п. 2 задания 1).
2. Названия озер заключить в flex-контейнеры. Применить стилевое описание по своему усмотрению (например: `background: #22879b; margin: 3px; padding: 5px; font-size: 30pt; color: #E0F8F7; font-family: 'PT Sans Caption', sans-serif; width: 60%;`). Применить выравнивание: `justify-content: center;`
3. Создать блоки с изображением и описанием озер Беларуси, находящихся вблизи места вашего проживания (информацию найти в сети Интернет). Для этого повторить пункты 3–4 задания 1.
4. Разместить блоки на веб-странице так, чтобы блок с названием каждого озера находился над блоком с его изображением и описанием.
5. Заключить все созданные блоки в flex-контейнер. Применить выравнивание: `flex-direction: column; align-items: center;`

С примером выполнения задания можно ознакомиться в *Приложении 2*.

Разработанная практическая работа была предложена учащимся 11-х классов Лицея БГУ информатико-математического, математического и физического профилей в 2018/2019 и 2019/2020 учебных годах. Ребята с энтузиазмом восприняли использование современных подходов к веб-конструированию, предложенных в практической работе. Некоторые учащиеся еще до выполнения работы были знакомы с технологией Flexbox на практике, другие только слышали о таком методе верстки страниц. Но даже те

учащиеся, которые до изучения темы «Основы веб-конструирования» не создавали веб-страниц, смогли успешно выполнить предложенную работу. И в дальнейшем практически все учащиеся при выполнении проектов «Создание сайта по темам учебных предметов» использовали технологию Flexbox.

Приложение 1

Содержание файла-заготовки *web_1.html*:

```
<html>
  <head>
    <title>Использование блоков</title>
    <meta charset = "utf-8">
  </head>
  <body>
    НАРОЧЬ
    <img src = "img/ozero.jpg">
    Нарочь – самое большое озеро в Белоруссии. Площадь – 79,62 кв.км. Наибольшая глубина – 24,8 м.
    Длина наибольшего поперечника – 12,8 км. Длина береговой линии – 40 км (четвёртое место среди
    озёр Белоруссии). Объём воды – 0,71 кв.км (первое место среди озёр Белоруссии). Высота над
    уровнем моря – 165 м. Всего в озеро впадает шестнадцать ручьев, помимо них с восточной стороны
    в Нарочь впадает проток Схема из озера Мястро, а с юго-восточной – вытекает река Нарочь. Озеро
    Нарочь возникло несколько тысячелетий назад, при отступлении ледника.
  </body>
</html>
```

Содержание файла-заготовки *1.css*:

```
body{
  background-image: /*Фоновое изображение*/
}
.bz{
  padding: /*Отступ*/
  color: /*Цвет текста*/
  font-family: /*Шрифт*/
}
.bt{
  background: /*Цвет фона*/
  width: /*Ширина блока*/
  padding: /*Отступ*/
  font: /*Параметры шрифта*/
}
.flex-cont{
  display: flex;
  justify-content: center;
}
img {
  width: 400px;
  height: 100%;
}
```

Содержание файла *web_1.html* после выполнения задания:

```
<html>
  <head>
    <title>Использование блоков</title>
    <meta charset = "utf-8">
    <link rel = "stylesheet" type = "text/css" href = "css/1.css">
    <link href = "https://fonts.googleapis.com/css?family=PT+Sans+Caption:
    700&display=swap&subset=cyrillic" rel = "stylesheet">
  </head>
  <body>
    <div class = "bz">
      <h1 align = "center">НАРОЧЬ</h1>
    </div>
    <div class = "flex-cont">
      <div><img src = "img/ozero.jpg"></div>
      <div class = "bt">
        <!--Описание озера Нарочь-->
      </div>
    </div>
  </body>
</html>
```

Содержание файла *1.css*:

```
body{
  background-image: url("../img/fon1.jpg");
}
.bz{
  padding: 5px;
  color: #E0F8F7;
  font-family: 'PT Sans Caption', sans-serif;
}
.bt{
  background: aliceblue;
  width: 30%;
  padding: 8px;
  font: 13pt Arial;
}
.flex-cont{
  display: flex;
  justify-content: space-around ;
}
img {
  width: 400px;
  height: 100%;
}
```


Отображение в браузере:



Приложение 2

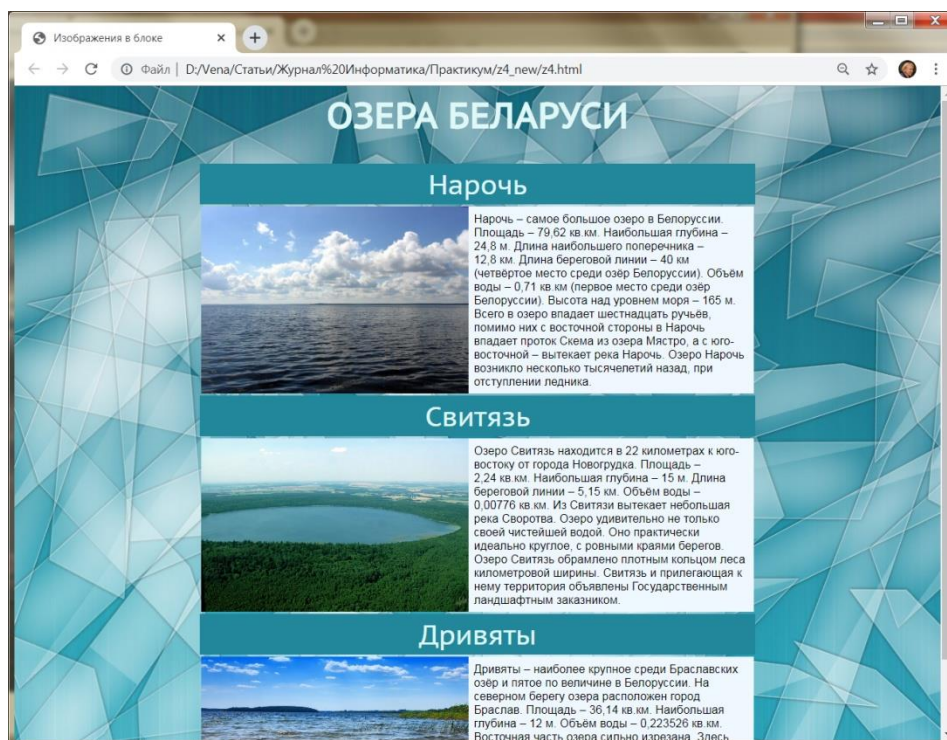
Содержание файла `web_1.html` после выполнения задания:

```
<html>
  <head>
    <title>Озера Беларуси</title>
    <meta charset = "utf-8">
    <link rel = "stylesheet" type = "text/css" href = "css/ style.css">
    <link href = "https://fonts.googleapis.com/css?family=PT+Sans+Caption:
    700&display=swap&subset=cyrillic" rel = "stylesheet">
  </head>
  <body>
    <h1 class = "z_web">Озера Беларуси</h1>
    <div class = "flex-all">
      <div class = "bz">Нарочь</div>
      <div class = "flex-cont">
        <div><img src = "img/ozero.jpg"></div>
        <div class = "bt">
          <!--Описание озера Нарочь-->
        </div>
      </div>
      <div class = "bz">Свитязь</div>
      <div class = "flex-cont">
        <div><img src = "img/svitiyz.jpg"></div>
        <div class = "bt">
          <!--Описание озера Свитязь-->
        </div>
      </div>
      <div class = "bz">Дривяты</div>
      <div class = "flex-cont">
        <div><img src = "img/driviyti.jpg"></div>
        <div class = "bt">
          <!--Описание озера Дривяты-->
        </div>
      </div>
    </div>
  </body>
</html>
```

Содержание файла 1.css:

```
body{
  background-image: url("../img/fon1.jpg");
}
.z_web{
  text-align: center;
  font-size: 40pt;
  color: #E0F8F7;
  font-family: 'PT Sans Caption', sans-serif;
}
.bz{
  background: #22879b;
  margin: 3px;
  padding: 5px;
  font-size: 30pt;
  color: #E0F8F7;
  font-family: 'PT Sans Caption', sans-serif;
  width: 60%;
  display: flex;
  justify-content: center;
}
.flex-cont{
  display: flex;
  justify-content: center;
}
.bt{
  background: aliceblue;
  width: 30%;
  padding: 8px;
  font: 13pt Arial;
}
.flex-all{
  display: flex;
  flex-direction: column;
  align-items: center;
}
img {
  width: 400px;
  height: 100%;
}
```

Отображение в браузере:



Список библиографических ссылок (на языке оригинала)

1. Материалы по вопросам организации допрофильной подготовки и профильного обучения. [Электронный ресурс]. *Национальный институт образования*. URL: <https://www.adu.by/ru/homepage/prof-oby-1.html> (дата обращения: 09.12.2019).
2. Учебная программа по учебному предмету «Информатика» для X – XI классов учреждений общего среднего образования с русским языком обучения и воспитания. утв. Пост. МО РБ от 27.07.2017;93.
3. Миняйлова Е.Л., Вербовников Д.А., Коледа Н.Р. Особенности реализации методики оформления веб-страниц средствами каскадных таблиц стилей (CSS). *Вестник МГИРО*. 2016;3(26).
4. Овчинникова Л.Г. Информатика. Рабочая тетрадь для 11 класса : пособие для учащихся учреждений общ. сред. образования с рус. (белорус.) яз. обучения. Минск: Аверсэв, 2016.
5. Паляков, К.Ю. Веб-дизайн: практикум. [Электронный ресурс]. Интернет-журнал «*Информатика*» 2013;(12). URL: <https://kpolyakov.spb.ru/download/inf-2013-12.pdf> (дата обращения: 09.12.2019).

6. Практические работы по HTML и CSS. [Электронный ресурс]. *Информационно-образовательный портал*. URL: <http://sgt-portal.ks.ua/ru/vneurochnaya-rabota/kruzhki/web-dizajn/205-html-laboratornaya-rabota-n1> (дата обращения: 09.12.2019).

7. How Flexbox works [Как работает CSS Flexbox]. Сборник статей «#CSS», 31 января 2017. [Электронный ресурс]. URL: <https://www.freecodecamp.org/news/an-animated-guide-to-flexbox-d280cf6afc35> (дата обращения: 23.01.2020). (In English)

References (на английском языке)

1. Materialy po voprosam organizacii doprofil'noj podgotovki i profil'nogo obucheniya [Materials on the organization of pre-profile training and profile training] [Electronic resource]. *Nacional'nyj institut obrazovaniya*. Available at: <https://www.adu.by/ru/homepage/prof-oby-1.html> (date of access: 09.12.2019). (In Russian)

2. Uchebnaya programma po uchebnomu predmetu «Informatika» dlya X – XI klassov uchrezhdenij obshchego srednego obrazovaniya s russkim yazykom obucheniya i vospitaniya. [The curriculum on the subject "Informatics" for the X - XI classes of institutions of general secondary education with the Russian language of instruction and education] – utv. Post. MO RB ot 27.07.2017;93. (In Russian)

3. Minyajlova E.L., Verbovnikov D.A., Koleda N.R. Osobennosti realizacii metodiki oformleniya veb-stranic sredstvami kaskadnyh tablic stilej (CSS) [Features of the implementation of web page design techniques using cascading style sheets (CSS)]. *Vestnik MGIRO*. 2016;3(26). (In Russian)

4. Ovchinnikova L.G. Informatika. Rabochaya tetrad' dlya 11 klassa [Informatics. Grade 11 workbook] : posobie dlya uchashchihsya uchrezhdenij obshch. sred. obrazovaniya s rus. (belorus.) yaz. obucheniya. Minsk: Aversev, 2016. (In Russian)

5. Palyakov K.YU. Veb-dizajn: praktikum [Web Design: Workshop]. [Electronic resource]. *Internet-zhurnal «Informatika»* 2013;(12). Available at: <https://kpolyakov.spb.ru/download/inf-2013-12.pdf> (date of access: 09.12.2019). (In Russian)

6. Practical work on HTML and CSS [Practical HTML and CSS]. [Electronic resource]. *Information and educational portal*. Available at: <http://sgt-portal.ks.ua/ru/vneurochnaya-rabota/kruzhki/web-dizajn/205-html-laboratornaya-rabota-n1> (date of access: 09.12.2019). (In Russian)

7. How Flexbox works. [Electronic resource]. Collection of articles "#CSS", 31 January 2017 Available at: <https://www.freecodecamp.org/news/an-animated-guide-to-flexbox-d280cf6afc35>. (date of access: 23.01.2020). (In English)