

Знакомство с HTML

На прошлом занятии

Мы узнали

- Что такое атрибуты элемента
- Подробно о тегах DIV и SPAN
- Что такое теги разметки страницы в HTML 5
- Почему не стоит повсеместно использовать DIV
- Какие бывают семантические теги и как их применять

Сегодня на занятии

Научимся

- Создавать страницы с использованием семантической разметки
- Создадим страницу с использованием различных видов форматирования

Основное содержание

Семантическая разметка - практика

В этом примере будут использованы CSS свойства, которые необходимы для большей наглядности. Не переживайте если вы сейчас не поймете работу этих свойств, так как нас сейчас больше интересует правильная разметка документа, а подробное изучение CSS будет рассмотрено позже.

Создание основы HTML-документа

Начнем с основного “каркаса” документа и потихонечку будем продвигаться к цели. Создайте новый html-файл, назовите его layout.html и добавьте базовую структуру страницы:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Семантическая Разметка</title>
  </head>
  <body>
  </body>
</html>
```

Раздел BODY

Добавим внутрь раздела BODY несколько элементов и подумаем зачем мы их сюда добавили:

```
<body>  
  <header>HEADER</header>  
  <nav>NAV</nav>  
  <div id="page"></div>  
  <footer>FOOTER</footer>  
</body>
```

DIV id="page"

Один элемент здесь кажется лишним, но это не так. DIV нам нужен, чтобы оформить нашу страничку более выразительно. Используя его, мы сможем применить к элементам внутри него некоторые стили.

Давайте посмотрим, что же будет внутри этого элемента с id="page":

```
<div id="page">  
  <main>MAIN</main>  
  <aside>ASIDE</aside>  
</div>
```

Да, да это теги MAIN и ASIDE. Мы ведь хотим увидеть в результате не надписи, безобразно собранные в кучу, а четкую структуру.

MAIN

Мы должны почувствовать связь того, что написали в html-файле и того, что видим в окошке браузера и именно ради этой связи мы с вами и продолжим. Вот так нам нужно подправить в MAIN:

```
<main>
```

```
    MAIN
```

```
    <article>ARTICLE</article>
```

```
</main>
```

Main - это у нас основной контент, то содержимое страницы, которое представляет наибольшую важность и в данном случае наиболее важным у нас будет Статья, содержащая завершенную, независимую композицию.

ARTICLE

Но не будет же у нас пустая статья, она же должна быть содержательной:

Наша статья будет состоять из нескольких частей: заголовков, два раздела группирующих связанное содержимое и подвал содержащий различные сноски и т.д.

```
<article>
```

```
  ARTICLE
```

```
  <header>HEADER</header>
```

```
  <section>SECTION</section>
```

```
  <section>SECTION</section>
```

```
  <footer>FOOTER</footer>
```

```
</article>
```

ASIDE

Перейдем к следующему тегу: ASIDE. Его мы тоже пустым не оставим. Давайте добавим туда пару элементов - HEADER и NAV. Должно получится вот так:

```
<aside>  
  ASIDE  
  <header>HEADER</header>  
  <nav>NAV</nav>  
</aside>
```

Здесь мы разместим некоторый контент отступающий от основного содержимого (MAIN)

Семантическая разметка - результат

```
<body>
  <header>HEADER</header>
  <nav>NAV</nav>
  <div id="page">
    <main>
      MAIN
      <article>
        ARTICLE
        <header>HEADER</header>
        <section>SECTION</section>
        <section>SECTION</section>
        <footer>FOOTER</footer>
      </article>
    </main>
    <aside>
      ASIDE
      <header>HEADER</header>
      <nav>NAV</nav>
    </aside>
  </div>
  <footer>FOOTER</footer>
</body>
```

HEADER
NAV
MAIN
ARTICLE
HEADER
SECTION
SECTION
FOOTER
ASIDE
HEADER
NAV
FOOTER

Семантическая разметка - CSS

Добавим к нашей странички немножечко магии... CSS. Добавлять будем в самый низ раздела HEAD, в тег STYLE.

Вот сюда:

```
<head>  
  <title>Семантическая Разметка</title>  
  <style>  
    Добавляем стили сюда  
  </style>  
</head>
```

CSS

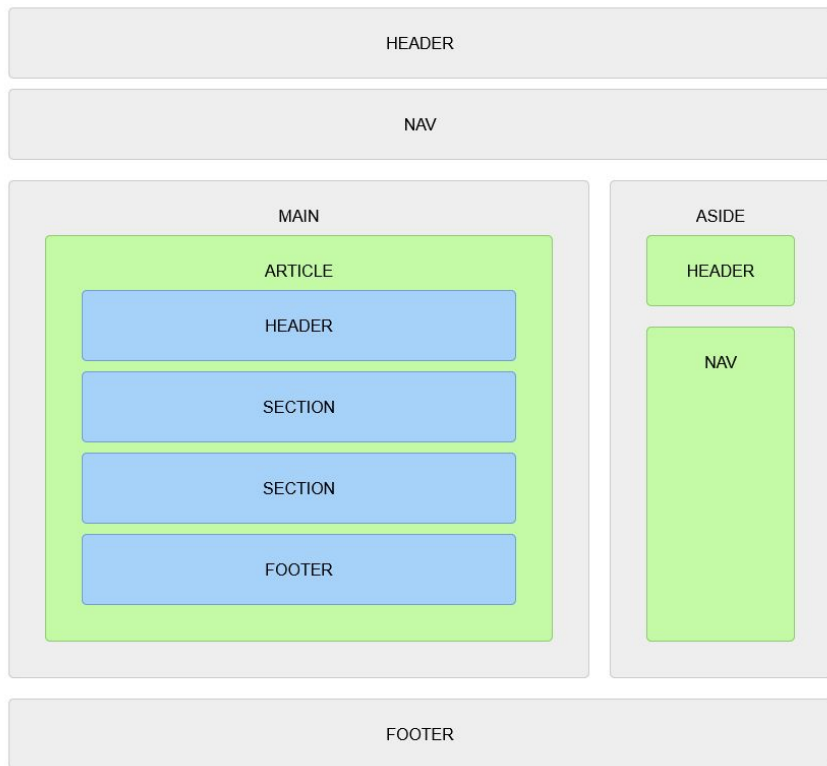
CSS:

```
body {  
  font-family: sans-serif;  
  padding: 10px;  
}  
body *:not(#page) {  
  padding: 25px;  
  margin: 10px;  
  background-color: #eee;  
  border: 1px solid #ccc;  
  border-radius: 5px;  
  text-align: center;  
}  
#page {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
}  
#page > main {  
  flex: 1;  
}
```

Еще CSS:

```
#page > aside {  
  display: flex;  
  flex-direction: column;  
  flex: 1;  
  max-width: 20%;  
}  
#page > aside > * {  
  background-color: #c4f9a6;  
  border-color: #91ca71;  
}  
#page > aside > nav {  
  flex: 1;  
}  
#page > main > article {  
  background-color: #c4f9a6;  
  border-color: #91ca71;  
}  
#page > main > article > * {  
  background-color: #a6d2f9;  
  border-color: #6ca1d0;  
}
```

Семантическая разметка + CSS - результат



Подведение итогов

Научились

- Создавать страницы с использованием семантической разметки
- Создадим страницу с использованием различных видов форматирования