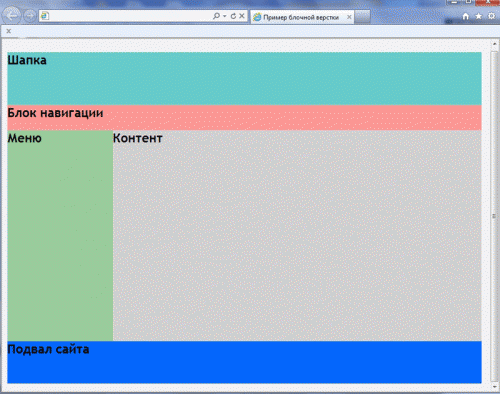
**Лабораторная работа 1. HTML + CSS. Разработка статической страницы (меню и несколько страниц)**

Сверстать структуру HTML-документа. В соответствии с шаблоном реализовать мини-сайт (не фиксированный: резиновый, адаптивный) с заданной тематикой, используя блочную верстку.

Пример тематики сайта

1. Библиотека (Меню – О библиотеке, Каталог книг, Контактная информация)
2. Туристическое агентство (Меню – О нас, Каталог стран, Контактная информация)
3. Медицинская клиника (Меню – О клинике, Каталог услуг, Контактная информация)
4. Сайт школы. (Меню – О школе, Список учителей, Контактная информация)
5. Автосервис. (Меню – Об автосервисе, Список услуг, Контактная информация)
6. Сайт кинотеатра. (Меню – О кинотеатре, Каталог фильмов, Контактная информация)
7. Мини-блог. (Меню – О сервисе, Перечень статей – записей блога, Контактная информация)
8. Гостевая книга. Регистрация пользователей и оставление отзывов.
9. Автосалон. (Меню – О нас, Перечень автомобилей, Контактная информация)
10. Мастерская по ремонту бытовой техники. (Меню – О мастерской, Список мастеров, Контактная информация)

Пример макета



Главная страница должна содержать: шапку, блок навигации, левое меню (на некоторых страницах оно не будет отображаться), контент, подвал сайта.

При выборе пункта меню (О нас, Контактная информация) в блоке навигации в области контента отображается соответствующая информация. При выборе меню (Каталог / Перечень / Список) в блоке контента отображается перечень соответствующих ссылок (а также появляется меню слева), при выборе любого пункта меню в блоке контента отображается информация о выбранном элементе.

**Технологические требования к разрабатываемому ресурсу**

1. Представить ссылку на внутренний ресурс, ссылку на внешний ресурс, ссылку на изображение
2. Использовать таблицу каскадных стилей CSS в отдельном файле. Сделать отдельную директорию для CSS
3. В контент страниц вынести информацию, содержащую изображения, параграфы, ссылки, списки, таблицы… (по усмотрению).
4. Заполнить меню, блок навигации, шапку, подвал в соответствии с тематикой сайта.
5. По возможности, использовать семантические теги (теги HTML5).
6. Стараться структурировать файлы при разработке (не валить в одну кучу). Статические страницы в одну папку, таблицы стилей – в другую