Создание простой CMS системы при помощи ООП PHP - Практика

<https://webformyself.com/minikurs/new/cms/cm1/lessons_page_6.html>

**Урок 1 - Создание базы данных. Единая точка входа**

cms - будет содержать пользовательскую и админскую часть.

Что необходимо для работы

1. шаблон страницы (html)
2. БД MySQL

1**. Создаем новую БД и наполняем её контентом**. (Пример стуктуры БД см. OpenServer через phpAdmin).

2. **Кодируем сайт**:

* формируем директории в корне сайта
* создем config.php - для хранения констант для соедиенения с БД и остальных основных настроек.
* создем index.php - **Единая точка входа на наш сайт** (см. 22:40 урок 1)

Основа сайта(вся логика) являет собой класс abstract class ACore , данный класс содержит все необходимые св-ва и методы для работы всего сайта, так же создаются его дочерние классы class Main, View, Category, Menu и т.д. Данные дочерние классы будут рендерить в браузер соответсвующие им данные (основной контент).

Роль index.php (Единая точка входа на наш сайт) , т.е. когда мы обращаемся к сайту через браузер, в URL мы указываем index.php это приведет нас на главную страницу сайта. Если же мы хотим загрузить какую либо другую страницу (дочерний класс) нашего сайта, то в URL необходимо передать помимо основного доменного имени еще и параметры(GET или POST) после символа ?  
Напрмер: http://minicms/index.php?option=main

Далее index.php на базе переданных параметров в URL, уже подгружает необходимый нам класс, который и отрендрить необходимую нам страницу в браузер. Т.е. вывод любых страниц сайта, в любом случае, происходит через скрипт index.php, который в свою очередь подгружает необходимый дочерний класс.

Создаем abstract class ACore

В классе index.php подключаем вспомог. скрипты через require\_once: **require\_once**(**"./config.php"**);  
**require\_once**(**"./classes/ACore.php"**)

Так как index.php это единая точка входа, то подразумевается, что в этот файл будут передаваться какие-то GET параметры. Поэтому необходимо описать **алгоритм принятия и обработки входных данных** от юзер в URL принятых через $\_GET['option'] в самом скрипте. см. список проверок в скрипте index.php

Определяем abstract class ACore

создаем его дочерниий классы class Main

Протестируем код: http://minicms/index.php?option=main

Кидаем в корень папку file (за ранее подготовленную) с фото

**Урок 2 - Создание логич. ядра пользовательской части**

1. При создании объекта, необходимо выполнить соединение с БД. Для этого в конструкторе класса ACore опишем данный метод.

2. Опишем методы по выводу статических частей страниц (Header, Menu, LeftBar, Footer) в классе ACore один раз

***~~Если не обновляются стили нажми F8 для остановки исполнеия скрипта, что бы он загрузился заново и подтянул обновленный style.css~~***

Чтобы заставить chrome загрузить css и js:

Вариант Windows 1: CTRL + SHIFT + R   
Вариант Windows 2: SHIFT + F5

Оригинальный ответ:

Изменено поведение Chrome. CTRL + R сделает это.

<http://qaru.site/questions/56886/how-to-force-chrome-browser-to-reload-css-file-while-debugging-in-visual-studio>

Далее необходимо вывести центральный блок и футтер

Центральный блок является динамическим блоком

**Урок 3. Создание и вывод основных страниц сайта**

Заключ. урок по созданию клиентской части.

**Урок 4. Вывод главной страницы админки**

ычычыч

Урок 5. Добавление, редактирование и удаление статей

Когда сайт готово для использования необходимо будет не забыдь убрать все выводы ошибок в браузер (mysqli\_error())