# Лекция 1

**Wordpress –** это свободно распространяемая, соответствующая стандартам, быстрая, лёгкая и бесплатная платформа для персонального блоггинга с практичными настройками и свойствами по умолчанию и с чрезвычайно гибким и настраиваемым ядром.

**Архитектура:** ядро+ тема + дополнения

**Преимущества:**

1. Наиболее популярная CMS

2. Бесплатная

3. Наиболее широкий набор плагинов, тем, виджетов и т.д.

4. WYSIWYG редактор

5. Технический опыт не обязателен

**Недостатки:**

1. Главная проблема - как использовать все её возможности правильно

2. Возможны проблемы при установке

**Управление и администрирование:**

* Управление пользователями
* Профили пользователей
* Динамическая генерация страниц
* Интернационализация и локализация

**Плагины**

* Обеспечение безопасности сайта
* Борьба со спамом
* Техническая оптимизация ресурса в соответствии с требованиями поисковых систем (Создание и обновление карты сайта, Канонизация URL, Оптимизация title страниц, Автоматическая генерация мета-данных страниц, Блокирование индексации дублированного контента, Создание микроразметки страниц)
* Повышение юзабилити и функциональность ресурса

**Меры безопасности:**

1. Двухфакторная аутентификация (ДФА)

2. Логины и пароли

3. Зашифровать все

4. Обновление CMS

**Основы кэширование в WordPress**

**Кэш** - это промежуточный буфер, который позволяет хранить наиболее часто используемые данные в памяти или на жёстком диске, что позволяют существенно ускорить процесс их выдачи.

**Три основных вида кэширования:**

1. Кэширование страниц (WP Super и W3 Total Cache)

2. Кэширование объектов (функции wp\_cache\_add, wp\_cache\_set, wp\_cache\_get): позволяет кэшировать данные произвольного типа. В ядре по умолчанию

3. Транзитное кэширование (функции get\_transient, set\_transient, delete\_transient)

**Плагины для адаптация под мобильные устройства:**

* WPTouch
* Jetpack
* WPSmart Mobile
* Responsible

# Лекция 2. Работа с контентом

**Таблицы базы данных:**

* По умолчанию префикс у всех таблиц: wp\_
* Основная таблица: wp\_posts
* Две таблицы не связаны ни с одной другой: wp\_options, wp\_links
* Таблицы wp\_users и wp\_comments не связаны между собой
* Установка плагинов создаёт доп. таблицы

**Виды контента в WordPress:**

* Сообщения (записи)
* Комментарии
* Пользователи
* Ссылки

**Первый вид контента**. Типы записей встроенных в WordPress:

1. **Запись, сообщение (post)**

* single-post-{ярлык\_записи}.php
* single-post.php
* ­single.php
* singular.php
* index.php
* Записи по умолчанию выводятся на главных страницах многих тем WordPress в порядке даты публикации (по убыванию)

1. **Страница (page)**

* http://ваш-сайт.ru/ярлык\_страницы
* ­{любое\_название}.php (шаблон страницы)
* page-{urldecode\_post\_name}.php
* page-{post\_name}.php
* page-{ID-записи}.php
* page.php
* singular.php
* index.php
* Страницы могут использовать произвольный файл из папки с темой в качестве своего HTML/PHP шаблона. Это значит, что в принципе каждая страница может кардинально отличаться от общего дизайна и верстки сайта
* Имеют иерархию
* Имеется возможность сортировать по порядку

1. **Вложение (attachment)**

* Предназначен для управления файлами, загруженными через стандартный загрузчик WP, а также для хранения данных о них

1. **Редакция (revision)**

* Черновики, которые автоматически сохраняются в процессе редактирования записи

1. **Элемент меню навигации (navigation menu item)**

* Хранит информацию об элементах навигации
* Это единственный тип, который не используются для работы с контентом
* Сохраняются как записи

**Второй вид контента**. Комментарии хранятся в отдельной таблице БД – wp\_comments. Комментарий может иметь прикреплённые через wp\_commentmeta метаданные. Комментарии могут быть связаны друг с другом через поля comment\_parent. Комментарии могут быть связаны с таблицей wp\_users

**Третий вид контента**. Пользователи имеют свои собственные таблицы wp\_users, а также метаданные wp\_usermeta

**Четвёртый вид контента**. Ссылки работают подобно записям. У них есть содержание, им можно присвоить элементы таксономии. Но им нельзя назначить автора.

**!** Три типа контента могут содержать метаданные: записи, комментарии и пользователи.

**Виджеты**

В WP есть области, в которых мы можем встраивать контент – **регионы**. Количество регионов в разных темах различное.

В WP существует определённое количество регионов, куда можно добавлять **виджеты**. Виджеты бывают типовые или созданные плагинами и темами или определены прямо в админке с помощью html-кода.

**Виджеты** – это независимые блоки содержимого, которые можно размещать в областях, предусмотренных темами.

# Лекция 3.

module in joomla - меню, его нельзя просто так создать

1. создаем новое меню
2. создаем пункты меню
3. создать модуль для меню в котором будет позиция для меню

**Как смотреть позицию:**

Ссылка: предварительный просмотр (расширения -> шаблоны)

Открывается страница разлинеенная и где названия позиций (позиция модулей)

Расширение - модули: ваши созданные модули, галочки активации, позиция и какой тип, доступ, на каких стр будет отображ

# Лекция 4. **Уровни доступа к системе**

**Группы пользователей интерфейса (Front-end Groups)**

|  |  |
| --- | --- |
| Гость | Просмотр сайта и общедоступных (public) статей |
| Зарегистрированный пользователь | - Привилегии Гостевой группы  - Просмотр статей доступных для пользователей |
| Автор | - Привилегии Группы зарегистрированных пользователей  - Создание новых статей  - Редактирование собственных стилей  - Просмотр специального контента |
| Редактор | - Привилегии Группы авторов  - Изменение всех статей, в том числе опубликованных |
| Издатель | - Привилегии Группы редакторов  - Публикация статей |

**Группы админки (Back-end Groups)**

|  |  |
| --- | --- |
| Менеджер | - Привилегии Группы издателей  - Доступ администратора |
| Администратор | - Привилегии Группы менеджеров  - Создание новых пользователей  - Установка расширений |
| Супер-администратор | - Права администратора  - Изменение шаблона сайта  - Изменение общих настроек |

**Управление пользователями**

Система общей настройки пользователей (Пользователи -> Пользователи -> Настройки):

- Уведомление администратора по электронной почте

- Администратор не может сделать себя или кого-либо ещё Супер-администратором.

- Можно изменить пользователей в настройках

**4 способа установки расширений в Joomla:**

1. Установить из Jet

2. Установить из пакета файлов

3. Установка из директории

4. Установка по URL

# Лекция 5.

**Drupal** - CMS, написанная на PHP и использует реляционные БД. Была разработана в 2000 г. Открытый исходный код.

**Основные понятия:**

1. Нода (node) - элемент контента, которой основывается на одной и той же базовой структуре.
2. Таксономия (taxonomy) - состоит из двух элементов: словари и термины.
3. Формат ввода (input format) - определенный набор прав, применяющийся при отображении страницы, после ввода содержимого.
4. Блоки
5. Права доступа
6. Меню

AdminToolBar - следует установить плагин.

Все модули и темы могут быть представлены в нескольких версиях:

* Стабильная версия
* Альфа/бета версия
* Разрабатываемая версия

**Страница темы или модуля имеет следующие элементы:**

* Название
* Описание
* Ссылки на скачивание
* Статистика
* Ссылки на страницу документацию, демо и трэкер багов

**Типы блоков:**

1. Предоставляемые модулями
2. Созданные вручную (пользовательские)

# Лекция 6. Конструктор CMS Drupal

**Типы блоков:**

1. Предоставляемые модулями
2. Созданные вручную (пользовательские)

Plugin API - позволяет создавать блоки.

Entity API - позволяет управлять настройками блока и определять настройки видимости блока.

*Суть лекции - описание интерфейса для создания блоков и пр. (одни скрины)*

Все модули можно найти на официальном сайте друпал.

Создание модуля:

* module/custom/[module:name]
* файл имя\_модуля.info.yml
* создаём в корневой папке модуля папку src
* создаём в папке src новую папку: Controller
* создаём файл с нашим контроллером имя\_контроллераController.php

# Лекция 7. Apex

Инструмент ускоренной разработки Web приложений для бд Oracle.

Oracle Apex позволяет создавать информационные системы с использованием веб-браузера и не требует от разработчиков большого опыта программирования.

Возможности Apex

* темы пользовательского интерфейса
* управление навигацией
* управление формами
* гибкие отчеты
* среда разработки имеет простой и эффективный web-интерфейс
* помощники миграции из настольных бд и электронных таблиц
* встроенные мастер генерации отчетов в формате pdf web-сервисами
* большое кол-во шаблонов пользовательского интерфейса
* интуитивно понятное управление рабочим пространством
* инструменты для интеграции web-сервисами
* управление объектами по принципу drag & drop
* графический помощник создания sql-запросов
* защищенность данных сессии после авторизации пользователя
* встроенный редактор pl\sql
* мастер создания диаграмм отчетов на сайте
* поддержка более 20 языков, включая русский

**Apex - кросс- платформенная программа**

**Языки:**

* php,
* Java,
* pl\sql - можно обойтись без промежуточного звена в виде веб-сервера Apache, html-код будет выдавать СУБД

**Во время своей работы Application Express**

* выполняет аутентификацию и авторизацию
* осущ контроль выполнения приложения на разных уровнях запуска
* выполняет проверку состояния качества и выполняет обработку запросов
* управляет состоянием сессии
* генерирует и обрабатывает страниц

Oracle Application Express размещается в бд. Представлеят собой набор таблиц с данными и pl/sql код.

Oracle применяет HTTP Server(Apache), в котором содержится плагин mod\_plsql. Этот плагин перенаправляет все запросы между элементами APEX В БД Oracle в реальном времени.

Oracle Application Express (APEX) генерирует из данных, находящихся в БД Oracle, динамические страницы, при этом позволяет нам обрабатывать их в режиме реального времени. При создании приложений, написанных на APEX, генерируются метаданные, которые хранятся в таблицах БД Oracle, модифицируя. Во время работы программы APEX считает эти метаданные и отображает их в браузере.

**Структура главной страницы:**

* Пользователи
* Статистика

**В состав APEX входят следующие четыре основных компонента:**

1. **App Builder**

Для создания HTML-интерфейса. Каждое приложение в APEX представляет собой коллекцию страниц (page), связанных друг с другом при помощи вкладок (buttons) или гипертекстовых ссылок (hypertext links).

Номер приложения — целое число, идентифицирующее приложение в системе в целом; номер страницы различает страницы внутри данного приложения и имеет структуру P. ? - целое положительное число

1. **SQL Workshop - среда управления объектами бд**
2. **Team Development**
3. **Gallery**

Object Browser позволяет использовать элемент управления деревьями для просмотра свойств объекта и создания новых объектов

**SQL-Commands позволяет вам вводить специальный sql-код**

* Запуск команд SQL или PL/SQL блоков
* Результаты запроса выводятся на экран в табличном виде
* Просмотр объясняет планы относительно операций DML запросов
* Описание объектов базы данных
* Сохранение SQL-запросов для более позднего использования
* Доступ к истории команд

**Utilities**

* Data Workshop — Загружает и разгружает данные в тексте, XML и форматах электронной таблицы
* Object Reports — Просматривают отчеты всех объектов, доступных для Вашего рабочего пространства
* Generate DDL — Создает DDL для всех или выбранных объектов в пределах ваших схем
* Recycle Bin — Корзина — Представление и восстановливает отброшенные объекты
* Schema Comparison — Сравнение схем — представление различий между двумя схемами, доступными для вашего рабочего пространства
* Database Monitor — Монитор базы данных

Схема – это логический контейнер для объектов бд

В списке Schema отображаются только схемы, назначенные вашему рабочему пространству.