

Бочаров Б.П., Воеводина М.Ю.

Базы данных в Moodle

Мы используем Moodle уже более пяти лет (в учебном процессе Харьковской национальной академии городского хозяйства). За это время определились две основные проблемы – изредка возникающие «глюки» системы и практически отсутствующая документация (например [1] – все, что создатели Moodle сочли нужным написать о базе данных).

В выпущенном нашей академией учебном пособии по системе Moodle [2] до детального описания сложных элементов курса просто не дошли руки. Мы попытались исправить это упущение для элемента «База данных». Другие сложные элементы (например, тесты) ожидают своей очереди.

Данная работа планировалась как небольшая статья, однако в процессе написания разрослась до размеров небольшой книги или большой брошюры. Оказалось, что по такому, казалось бы, частному вопросу можно сообщить достаточно много информации.

Мы постарались на конкретном примере очень подробно объяснить все наши действия по созданию и поддержке базы данных Moodle. Надеемся, что после изучения данной работы Вы узнаете, зачем нужна база данных Moodle и как она устроена. Мы осветили практически все аспекты создания и использования базы данных. Исключение составляет Шаблон RSS. На момент написания статьи этот шаблон (зачем он нужен и как работает) так и остался для нас абсолютно неизвестным.

Эта книга предназначена преподавателям, которые уже используют Moodle в своей профессиональной деятельности и хотят узнать больше о возможностях системы Moodle.

Для понимания материала, изложенного в работе, необходимы базовые знания по языку гипертекстовой разметки документов HTML, таблицам стилей CSS и языку программирования JavaScript. Все это в работе практически не описывается, основное внимание уделяется базе данных Moodle.

Отметим, что в рамках дисциплины «Вычислительная техника и алгоритмические языки», которые авторы преподают для студентов второго курса непрофильных (т.е. не связанных с IT-технологиями) специальностей изложено значительно больше информации, чем нужно для понимания этой книги.

Мы используем версию Moodle 1.9. Если у Вас такая версия, то база данных из резервной копии восстановится полностью, со всеми записями. В версии 2.x восстановится пустая база данных (только описание).

Переход на версию 2.x в Харьковской национальной академии городского хозяйства на момент написания работы практически невозможен, потому что средств автоматического переноса курсов нет, а у нас на сайте 529 курсов и 12966 пользователей.

Введение

Долгое время мы считали базу данных Moodle красивой (и не такой уж красивой) игрушкой. Мы и сейчас так считаем, однако окончательно выяснилось, что для некоторых задач других средств в moodle нет. Поэтому, «за неимением гербовой, будем писать на простой...» и, по возможности, писать красиво.

Начнем с того, что база данных Moodle – это не база данных в том смысле, в котором этот термин обычно употребляют. Термином «база данных» обычно называют *систему управления базами данных с реляционной моделью данных*, например Access или MySQL. *Реляционная база данных* представляет собой набор двумерных таблиц (например, лист Excel), связанных между собой.

База данных Moodle – это одна плоская (двумерная) таблица, поэтому она больше напоминает электронную таблицу, а не базу данных. Однако есть в этом элементе кое-что, частично оправдывающее такое название – зачаточная система разграничения полномочий и прав доступа. Преподаватель может сделать что угодно с любой записью. Студент может добавить запись, редактировать (удалять) свои записи, чужие – только просматривать. Это позволяет заполнять базу нескольким пользователям одновременно и избежать дублирования информации.

Кроме того, база данных Moodle снабжена и некоторыми средствами поиска, выбора и сортировки информации. Именно некоторыми, они значительно уступают электронной таблице Excel.

Зачем нужна база данных Moodle?

Это главный вопрос. Если Вы ответили на этот вопрос (еще точнее можно его сформулировать так: «зачем мне в моем курсе нужна база данных Moodle?»), то база данных в Вашем курсе обязательно появится и будет там жить. Если Вы не нашли ответ на этот вопрос, то, скорее всего, базы данных в Вашем курсе не будет.

К сожалению, на такие вопросы каждый отвечает самостоятельно, мы можем лишь поделиться своим практическим опытом и результатами долгих размышлений.

Если Вы собираетесь вносить информацию самостоятельно (или привлекая несколько студентов, работу которых Вы можете координировать), **то база данных Вам не нужна!** Таблица Excel подходит для хранения такой информации значительно больше.

База данных может оказаться полезной, если студенты сами выбирают объекты, информацию о которых вносят в базу. В этой работе описывается база данных трехмерных моделей, но в базу данных можно вносить темы рефератов, курсовых и т.д.

Мы надеемся, что в процессе очень подробного изучения нашей базы данных Вы приблизитесь к ответу на главный вопрос: «А зачем мне эта база

нужна?»).

Даже если Вы не смогли ответить на этот вопрос и решили, что база данных Вам в реальной жизни не нужна, то Вы все равно не зря потратили время. Вы точно узнали, что на элемент «База данных» надеяться не стоит.

Общие положения

Демонстрационный курс

Вы можете посмотреть базу данных в специальном курсе на сайте Moodle Харьковской национальной академии городского хозяйства.

Адрес курса <http://cdo.kname.edu.ua/course/view.php?id=889>

Для просмотра курса нужно зайти гостем.

ЦДО ХНАГХ

Вы не прошли идентификацию. (Вход)

ДО ХНАГХ > Зайти на сайт

Русский (ru)

Зайти на сайт

Введите свой логин и пароль
(Куки (Cookies) должны быть разрешены
в Вашем браузере) ?


Логин

Пароль

Некоторые курсы могут
позволять гостевой доступ

Забыли логин или пароль?

Вы в первый раз на нашем сайте?



Вниманию студентов ХНАГХ: При первом входе в систему в качестве логина и пароля использовать **индивидуальный идентификационный номер (ИИН) !**

Рекомендуется ИЗМЕНИТЬ ПАРОЛЬ после первого входа !

Затем нужно принять «Пользовательское соглашение» (естественно, сначала его нужно прочитать).

Пользовательское Соглашение

Уважаемый Пользователь!

Для продолжения работы на нашем портале Вы должны ознакомиться с [Правилами](#) и принять их условия.

Тем самым Вы подтверждаете свое согласие с требованиями Правил и обязуетесь их выполнять.

Я принимаю Пользовательское Соглашение и обязуюсь его выполнять.

Да Нет

Нужно ввести ключевое слово (**demo**) и нажать кнопку «Записаться на курс».

Этот курс требует "кодового слова". В качестве гостя Вам необходимо вводить его каждый раз при входе на курс. Вы можете получить этот пароль от :

[Бочаров Борис Петрович.](#)
[Воеводина Марія Юріївна.](#)
[Яковицький Ігор Леонідович.](#)


Кодовое слово:


После этого Вы попадаете в курс «Демонстрационные материалы».


Демонстрационные материалы

1

База данных трехмерных объектов города Харькова, размещенных в Google Earth.

 [Список трехмерных моделей города Харькова, размещенных в Google Earth](#)

 [Резервная копия этого курса](#)

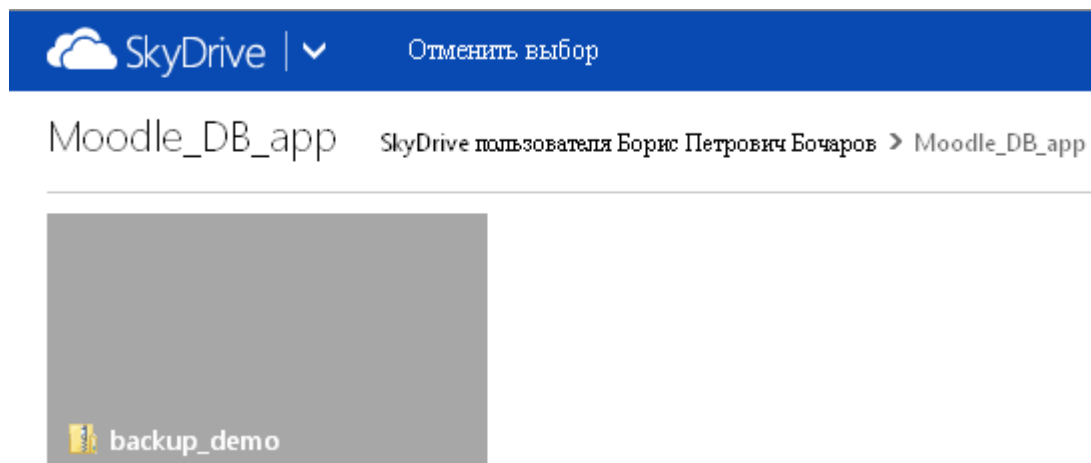
 [Прототипы шаблонов базы данных](#)

В курсе Вы увидите базу данных «Список трехмерных моделей города Харькова, размещенных в Google Earth». Это база, которая дальше подробно описывается в книге.

Резервная копия этого курса – ссылка на файл (физически находится на

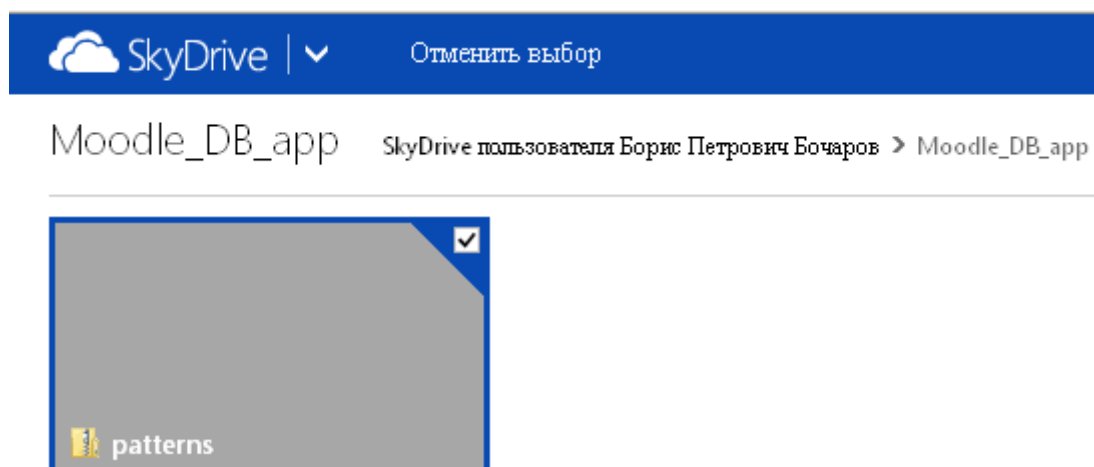
SkyDrive). Из этой копии Вы можете восстановить базу данных в своем курсе и внимательно ее изучить. Для скачивания файла достаточно щелкнуть по нему мышкой. Адрес файла:

<https://skydrive.live.com/redir?resid=74A1EAA18E3BAE10!131>



Прототипы шаблонов базы данных (zip-архив, физически находится на SkyDrive) – набор файлов, в которых записаны все шаблоны. Адрес файла:

<https://skydrive.live.com/redir?resid=74A1EAA18E3BAE10!132>



Все эти файлы будут рассмотрены очень подробно, пока ограничимся простым перечислением.

kha_1rec.html – прототип шаблона одиночной записи.

kha_edit_rec.html – прототип шаблона формы записи.

kha_list_rec.html – прототип шаблона списка.

kha_search.html – прототип шаблона расширенного поиска.

kha_pattern.css – прототип шаблона CSS.

kha_colors.html – вспомогательный файл для визуального подбора стилей.

kha_pattern.js – прототип шаблона Javascript.

Соглашения

В работе подробно рассматриваются тексты файлов, объясняется назначение практически каждой строчки. В тексте строка файла обозначается следующим образом (в начале строки указывается ее номер в файле):

```
21 var pri = 0;
```

Для ссылок на строки файла в тексте используется символ «№». Например, №21 – это ссылка на строку номер 21. Больше ни для каких целей знак «№» не используется. Ниже приведен пример описания фрагмента файла.

В цикле (№22-27) проверяется, записано ли значение поля «status» в массив **kha_StatArr** (№23).

Если да, то значение соответствующего элемента массива **kha_StatRecArr** увеличивается на 1 (№24). Переменной **pri** присваивается значение 1 (признак того, что значение поля «status» уже записано в массив **kha_StatArr**). Происходит выход из цикла (№26), а затем и из функции.

```
22 for(var i=0; i<kha_StatNum; i++)
23     if(kha_StatArr[i] == cur_s){
24         kha_StatRecArr[i]++;
25         pri=1;
26         break;
27     }
```

Имена всех файлов, переменных (за исключением локальных) и функций начинаются со строки (префикса) «**kha_**».

При формировании любой страницы Moodle используется обширная библиотека функций JavaScript. Мы не знаем точно, как организованы эти библиотеки и какие переменные и функции в них описаны. Поэтому, для гарантированного обеспечения уникальности имен предотвращения конфликтов, мы условились называть все наши внутренние переменные особым образом.

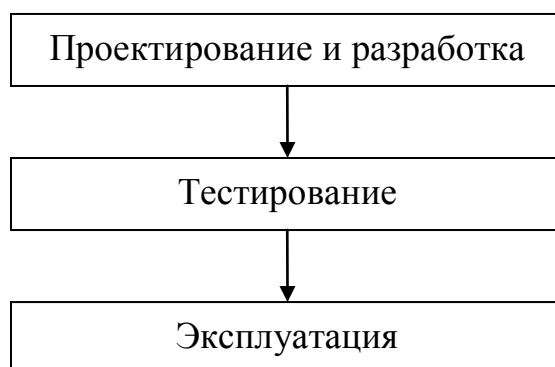
Файлы прототипов шаблонов называются так для единообразия. Однако такой стиль имен файлов может быть очень полезен, если Вы используете и поддерживаете несколько различных баз данных.

Проектирование базы данных

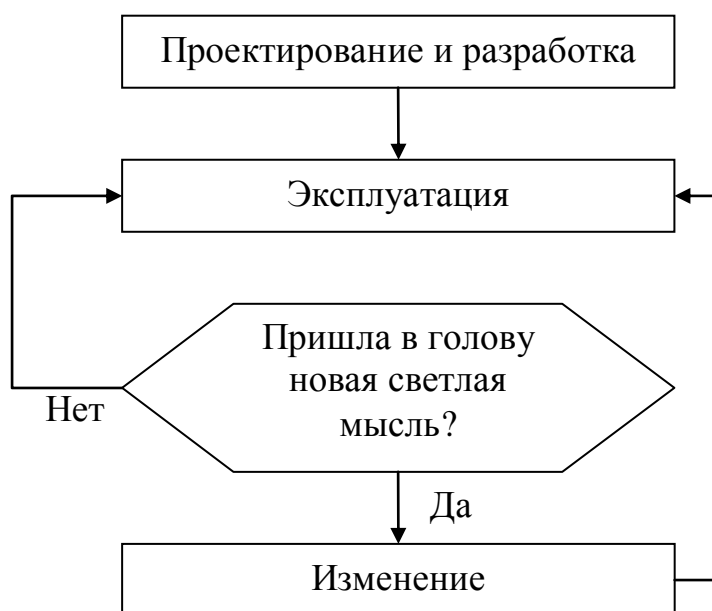
Проектирование базы данных (и курса в Moodle в целом) – это очень сложная и ответственная работа. Как правило, очень сложно заставить себя

сначала подумать, а потом уж «трясти». Это напоминает разработку некоммерческих программ.

В идеале процесс разработки и эксплуатации должен выглядеть так:



А на практике получается примерно так.



Мы описываем уже готовую базу, поэтому есть уникальная возможность излагать этапы разработки в правильном порядке.

1. Определение назначения базы данных.
2. Разработка информационной модели (т.е. содержание и структура информации, которую мы собираемся хранить в базе данных).
3. Моделирование действий пользователей:
 - алгоритм работы студента;
 - алгоритм работы преподавателя.
4. Техническая реализация базы данных
 - определение полей базы;

- разработка шаблонов.

База данных «Список трехмерных моделей города Харькова, размещенных в Google Earth» предназначена для: систематизации информации о трехмерных объектах, размещенных пользователями в программе Google планета Земля. Кроме того студенты получают навыки работы с программой Google планета Земля. Мы надеемся, что при заполнении этой базы студенты сделают некоторые типичные ошибки (или хотя бы прочитают о них в инструкции) и не будут делать эти ошибки при разработке и размещении в Google Earth своих моделей.

Мы собираемся брать информацию из программы Google Earth и из страницы модели в Google (Google продал программу SketchUp фирме «Trimble», это название видно на странице модели).

Студент вносит в базу данных следующую информацию.

1. Название объекта.
2. Широта и долгота объекта
3. Изображение объекта.
4. Адрес объекта (почтовый).
5. URL модели (интернетовский адрес).
6. Автор модели – URL (интернетовский адрес) аккаунта автора в Google.
7. Метка Google – специальный файл, который показывает объект в Google Earth и Google Maps.
8. Описание модели.
9. Сделана ли модель в ХНАГХ?

Преподавателю доступны еще два поля.

1. Одобрение записи – поле в котором мы будем записывать текущее состояние записи. (Да, Нет, Повтор, Исправить)
2. Комментарий к записи – здесь мы будем писать замечания к записи.

Описание схемы данных (информационной модели) базы данных мы совместим с подробной инструкцией по заполнению базы данных (моделирование действий студента).

Внесение информации в базу данных трехмерных объектов Google Earth

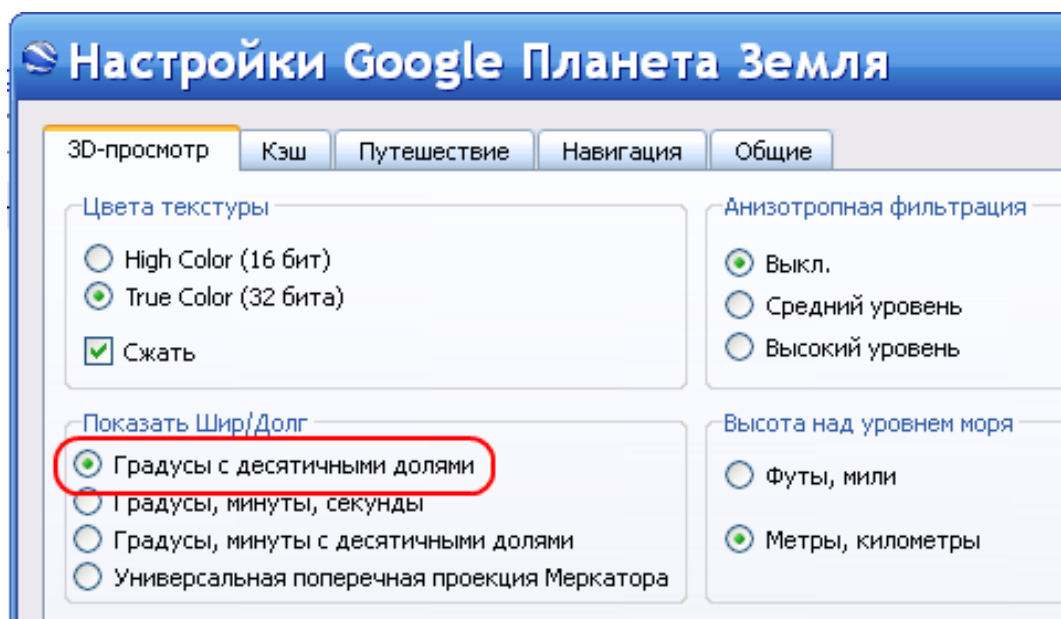
Перед началом работы с базой необходимо правильно настроить программу Google Планета Земля.

Настройка Google Earth

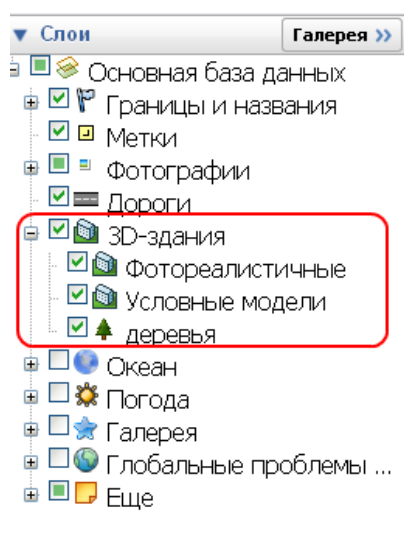
Первое, что нужно сделать – настроить правильное отображение координат. Широта и долгота должны отображаться в виде «Градусы с десятичными долями».

Внимание! Если вы оставите значение по умолчанию «Градусы, минуты,

секунды», то при вводе координат в поле базы данных Moodle округлит значения до целых чисел. Точность указания места при этом будет около 100 километров. **Не делайте эту очень распространенную ошибку!!!**



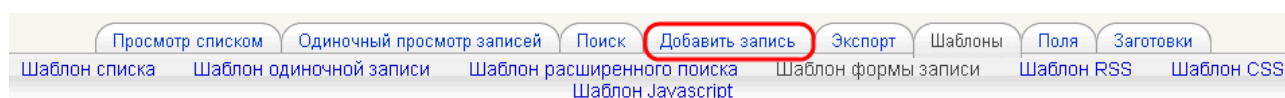
Следующий шаг – настройка отображения трехмерных объектов.



Теперь переходим к выбору объекта и занесения его в базу данных.

Начало добавления новой записи

В базе данных нажимаем ссылку «Добавить запись».



Мы увидим форму для ввода информации.

Новая запись

Изображение <input type="text"/> <input type="button" value="Обзор..."/>	Название объекта: <input type="text"/>
Альтернативный текст <input type="text"/>	Адрес объекта: <input type="text"/>
	Координаты объекта: Широта <input type="text"/> °N Долгота <input type="text"/> °E
	URL Ссылка: <input type="text"/> http://
	модели: Текст: <input type="text"/>
	Автор Ссылка: <input type="text"/> http://
	модели: Текст: <input type="text"/>

Файл, описывающий местоположение объекта:

Файл <input type="text"/> <input type="button" value="Обзор..."/>
Отображаемая ссылка на файл (необязательно) <input type="text"/>

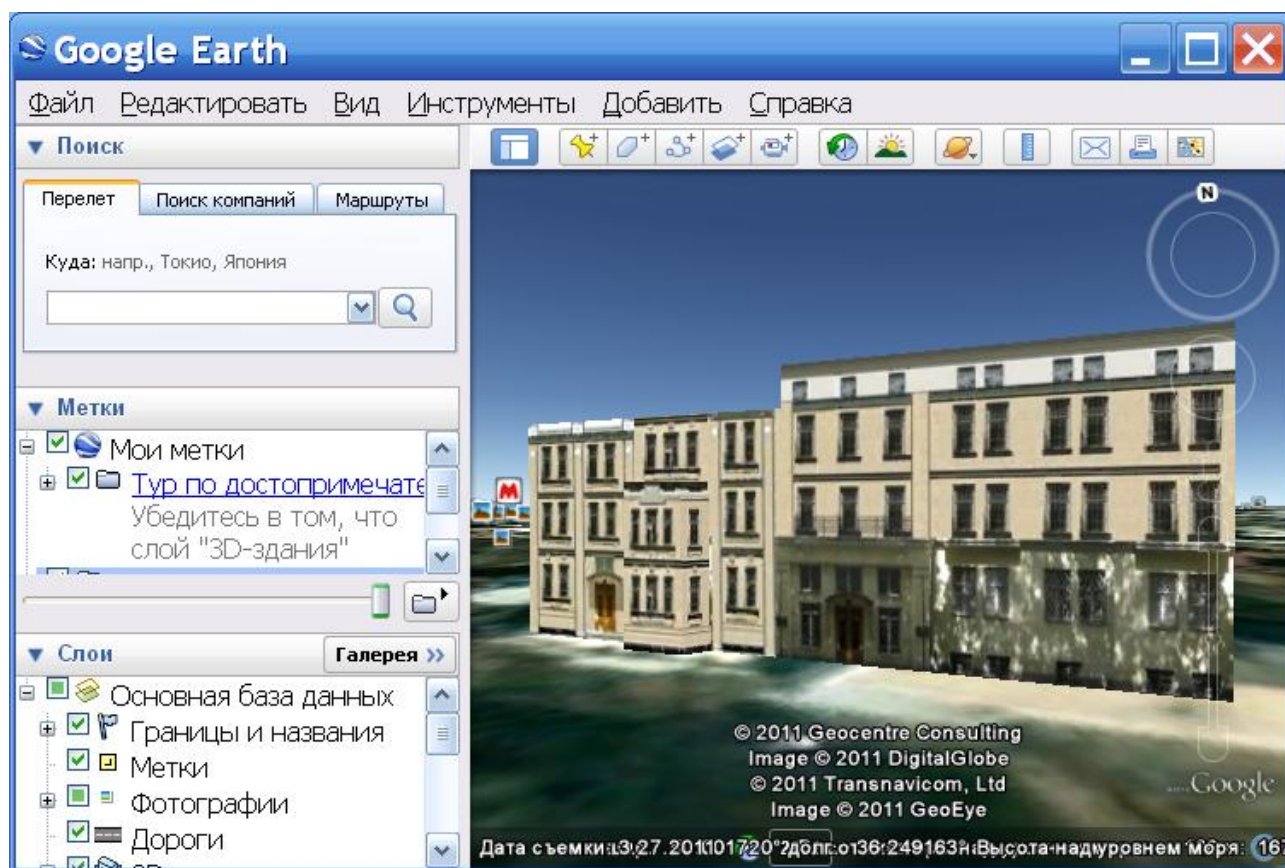
Описание объекта:

[HTML-редакторе](#) ?

Модель сделана:

Изображение

В программе Google Earth выбираем объект и подбираем правильную точку зрения



С помощью средства захвата экрана (кнопка «Print Screen») и графического редактора нужно подготовить изображение объекта.

Внимание! Изображение **обязательно** должно иметь размеры **300x150** пикселей.



Сохраняем файл (имя файла должно состоять **только** из латинских букв и цифр, например **ob1.jpg**).

Теперь вставляем в запись картинку.



В качестве альтернативного текста указываем название объекта

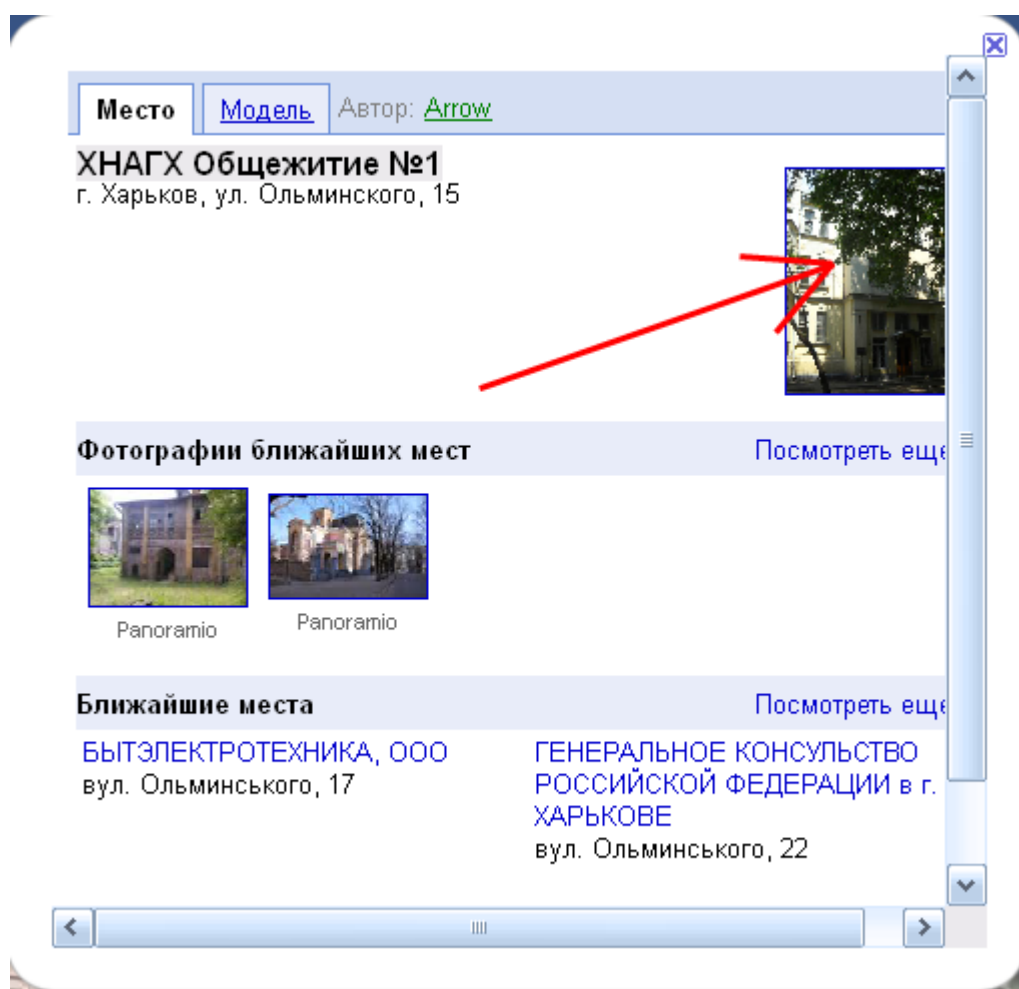
Новая запись

Изображение Обзор...

Альтернативный текст

Название и адрес объекта

Возвращаемся в программу Google Earth. Нажимаем на объект, чтобы получить его описание. Нажимаем на картинку, чтобы обратиться к странице модели (выводится в браузере).



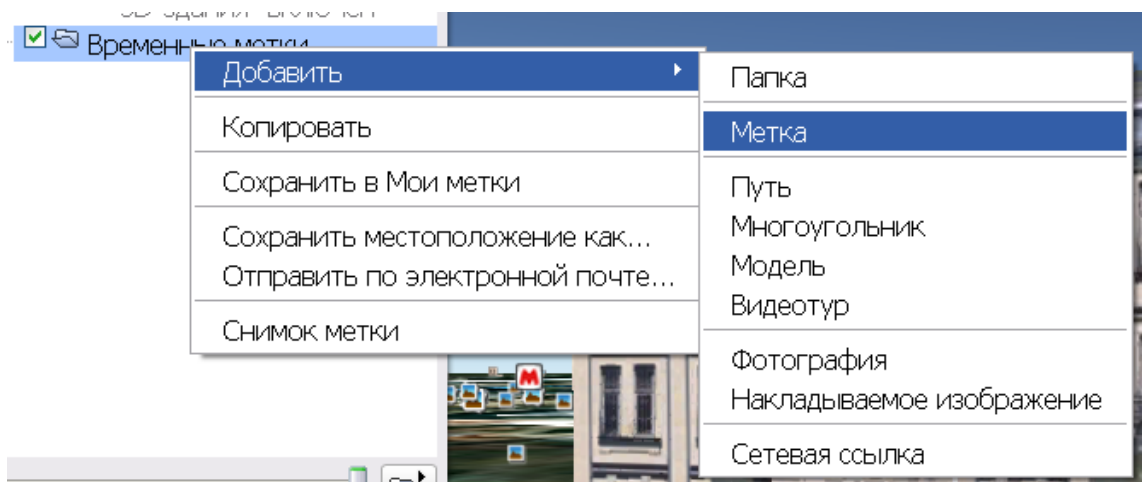
Переходим в Moodle и заполняем поля «**Название объекта**» и «**Адрес объекта**».

Название объекта:

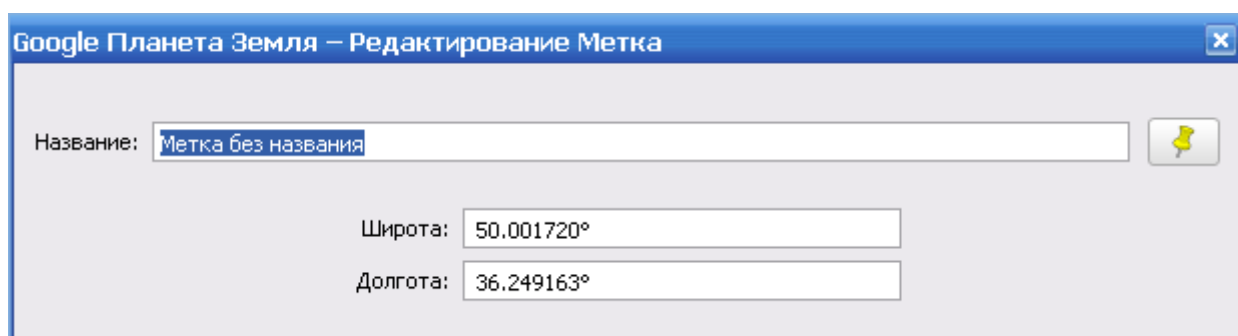
Адрес объекта:

Координаты объекта

Возвращаемся в программу Google Earth. Создаем временную метку.



В «свойствах метки» находим широту и долготу объекта.



Переходим в Moodle и заполняем поля «Широта» и «Долгота» в базе данных.

Координаты объекта:

Широта	50.001720	°N
Долгота	36.249163	°E

Переходим к странице модели.

URL (интернетовский адрес) страницы модели

ХНАГХ Общежитие №1 . А: x

sketchup.google.com/3dwarehouse/details?mid=ec3099ae25cbb9df33f3231560ace08&ct=3db1&hl=ru

3D-модели Архитектор SketchUp

Trimble 3D-модели powered by Google

ХНАГХ Общежитие №1 обновлено: Arrow дата: 09.07.11

Изображение Карта Просмотр улиц

Изображение 3D-обзор

Просмотры: 133 Загрузки: 15

Загрузить модель

Создать коллекцию Открыть общий доступ

Оценить эту модель

Просмотреть в Google Планета Земля

Просмотр на 3D-карте

ХНАГХ Общежитие №1 Санаторий-профилакторий «Градостроитель» -5-ти этажное кирпичное здание; -Общежитие располагает комнатами на три, четыре человека; - Кухни расположены отдельно на каждом этаже; -В общежитии расположен медицинский центр академии, оборудованы душевые. - основан в 1970 году, на базе Харьковской национальной академии городского хозяйства (ХНАГХ), для оздоровления и профилактического лечения студентов академии. -Санаторий открыт на 100 мест (80 стационарных, 20 амбулаторных с питанием).

Сообщить о нарушении правил

Модели Поиск Расширенный поиск

Portions ©2013 Trimble Navigation Limited - Portions ©2013 Google - Google's Privacy Policy - Trimble's Privacy Policy - Условия использования - Trimble Home -

Копируем URL (интернетовский адрес) модели.

ХНАГХ Общежитие №1 . А: x

sketchup.google.com/3dwarehouse/details?mid=ec3099ae25cbb9df33f3231560ace08&ct=3db1&hl=ru

3D-модели Архитектор SketchUp

Trimble 3D-модели powered by Google

ХНАГХ Общежитие №1

оби

Всю служебную информацию необходимо удалить.

[http://sketchup.google.com/3dwarehouse/details?
mid=ec3099ae25cbb9df33f3231560ace08&ct=3dbl&hl=ru](http://sketchup.google.com/3dwarehouse/details?mid=ec3099ae25cbb9df33f3231560ace08&ct=3dbl&hl=ru)

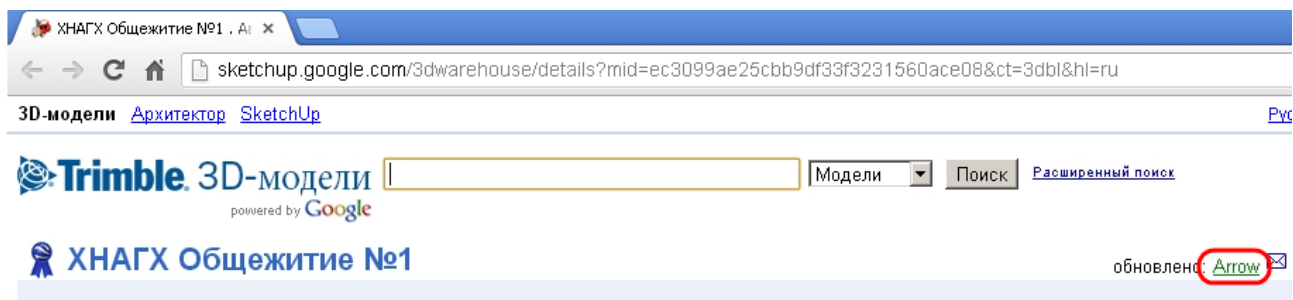
Итоговый URL:

[http://sketchup.google.com/3dwarehouse/details?
mid=ec3099ae25cbb9df33f3231560ace08](http://sketchup.google.com/3dwarehouse/details?mid=ec3099ae25cbb9df33f3231560ace08)

Переходим в Moodle и заполняем поле «**URL модели**». URL заносится в «**ссылку**» и «**текст**».

URL модели: Ссылка:
Текст:

Переходим в браузер и вызываем страницу автора модели.



URL (интернетовский адрес) страницы автора модели

ХНАГХ Общежитие №1 | Модели автора Arrow

sketchup.google.com/3dwarehouse/search?uq=1790197545758634092142008&scoring=m

3D-модели [Архитектор](#) [SketchUp](#)

Trimble 3D-модели powered by Google

Анастасия Танцур

[Просмотреть полный профиль](#)

1 модели, загруженные в 3D-модели

1 модели, принятые Google Планета Земля

[Подробнее о значках](#)

+1 0 Твитнуть 0 Мне нравится 0

Модели пользователя Arrow По date Резу

Определяем URL (интернетовский адрес) страницы автора.

ХНАГХ Общежитие №1 | Модели автора Arrow

sketchup.google.com/3dwarehouse/search?uq=1790197545758634092142008&scoring=m

3D-модели [Архитектор](#) [SketchUp](#)

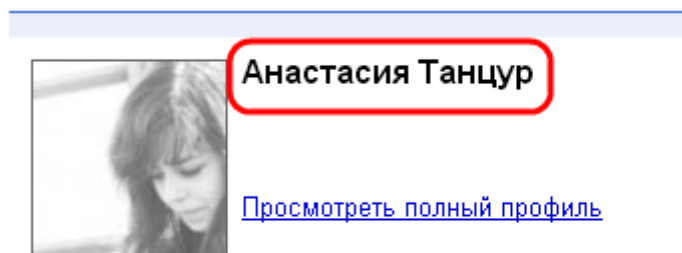
Всю служебную информацию из URL необходимо удалить.

http://sketchup.google.com/3dwarehouse/search?
uq=1790197545758634092142008&scoring=m

Итоговый URL:

http://sketchup.google.com/3dwarehouse/search?
uq=1790197545758634092142008

Определяем имя автора модели или его ник (псевдоним).



Переходим в Moodle и заполняем поле «Автор модели».

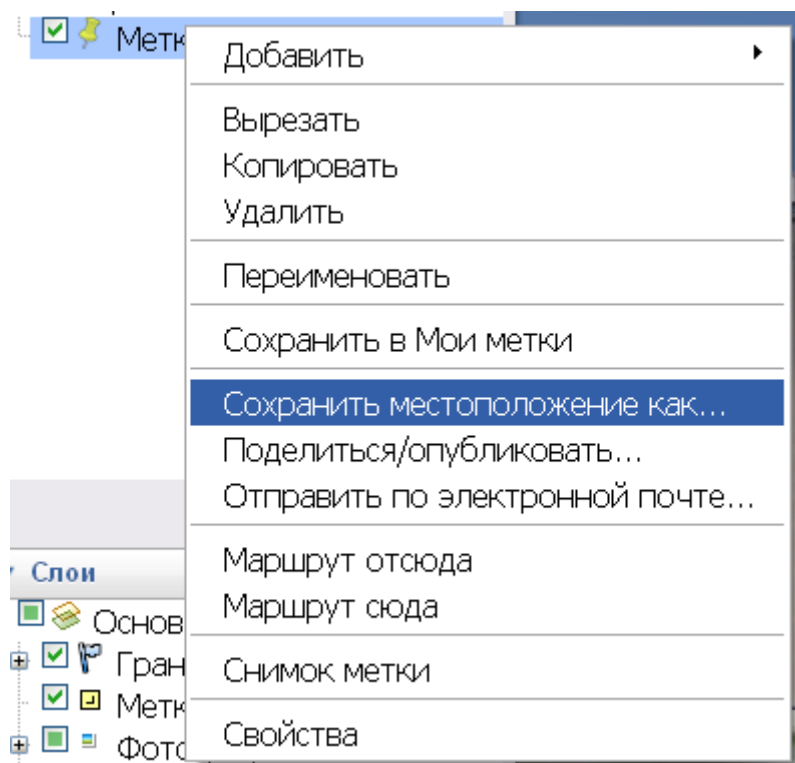
Автор модели:	Ссылка: <input type="text" value="http://sketchup.google.com/3dwarehouse/search?uq=179019754"/>
	Текст: <input type="text" value="Анастасия Танцур"/>

Метка Google.

Возвращаемся в программу Google Earth. Создаем файл описания местоположения объекта (файл **kml**).

Примечание. KML (от англ. Keyhole Markup Language — язык разметки Keyhole) — язык разметки на основе XML для представления трёхмерных геопространственных данных в программе «Google Планета Земля» («Keyhole» до её приобретения «Google»).

Сохраняем положение объекта (формат файла вывода – kml).



Сохраняем файл (имя файла должно состоять **только** из латинских букв и цифр, например **ob1.kml**).

Переходим в Moodle и заполняем поле «**Файл, описывающий местоположение объекта**».

Внимание!!! В поле «**Отображаемая ссылка на файл (необязательно)**» **необходимо** написать слово «**метка**».

Описание объекта

Переходим на страницу модели в браузере и копируем описание модели.



ХНАГХ Общежитие №1 Санаторий-профилакторий «Градостроитель» -5-ти этажное кирпичное здание; -Общежитие располагает комнатами на три, четыре человека; - Кухни расположены отдельно на каждом этаже; -В общежитии расположен медицинский центр академии, оборудованы душевые. - основан в 1970 году, на базе Харьковской национальной академии городского хозяйства (ХНАГХ), для оздоровления и профилактического лечения студентов академии. -Санаторий открыт на 100 мест (80 стационарных, 20 амбулаторных с питанием).

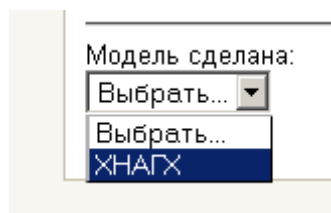
Переходим в Moodle и заполняем поле «**Описание объекта**».

Описание объекта:

ХНАГХ Общежитие №1 Санаторий-профилакторий «Градостроитель» -5-ти этажное кирпичное здание; -Общежитие располагает комнатами на три, четыре человека; -Кухни расположены отдельно на каждом этаже; -В общежитии расположен медицинский центр академии, оборудованы душевые. - основан в 1970 году, на базе Харьковской национальной академии городского хозяйства (ХНАГХ), для оздоровления и профилактического лечения студентов академии. -Санаторий открыт на 100 мест (80 стационарных, 20 амбулаторных с питанием).

Сделана ли модель в ХНАГХ?

Заполняем поле «**Модель сделана**». Если модель сделана студентами ХНАГХ, то нужно выбрать «ХНАГХ», в противном случае не нужно выбирать ничего (оставить поле без изменения).



Нажимаем кнопку «**Сохранить и просмотреть**».

Модель сделана:

Видим результат работы

Список трехмерных моделей города Харькова, размещенных в Google Earth

Страница: (Предыдущая) 1 ... 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 ...109 (Следующая)

Название объекта: ХНАГХ Общежитие №1 **ХНАГХ**
Адрес объекта: г. Харьков, ул. Ольминского, 15
Координаты объекта: 50.0017°N, 36.2492°E
URL модели: <http://sketchup.google.com/3dwarehouse/details?mid=ec3099ae25cbb9df33f3231560ace08>
Автор модели: Arrow
Файл, описывающий **метка**
местоположение объекта:
Автор записи: Тест Тест
Дата внесения: Суббота 12 Ноябрь 2011, 08:43
Последнее изменение: Воскресенье 21 Октябрь 2012, 06:38

Описание объекта: ХНАГХ Общежитие №1 Санаторий-профилакторий «Градостроитель» -5-ти этажное кирпичное здание; -Общежитие располагает комнатами на три, четыре человека; -Кухни расположены отдельно на каждом этаже; -В общежитии расположен медицинский центр академии, оборудованы душевые. - основан в 1970 году, на базе Харьковской национальной академии городского хозяйства (ХНАГХ), для оздоровления и профилактического лечения студентов академии. -Санаторий открыт на 100 мест (80 стационарных, 20 амбулаторных с питанием).

Запись одобрена:
 Комментарий:

Все описанные действия выполняют студенты, преподаватель может (и должен) заполнить еще два поля – «Запись одобрена» и «Комментарий».

Запись одобрена:

Да

Комментарий:

Trebuchet 1 (8 pt) Язык **B** *I* U ~~S~~ x₂ x² | [Иконки]

Это тестовая запись, введенная тестовым студентом.

Путь:

[О HTML-редакторе](#)

Создание полей

Вы видите уже готовую базу данных, поэтому процесс создания полей остался несколько в стороне.

В этом разделе мы покажем, как создавать поля с самого начала.

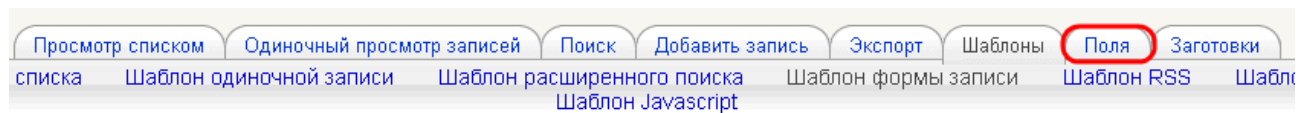
Очень советуем ответственно отнестись к выбору имен полей. Исправить их практически невозможно (при создании этой базы мы не продумали названия полей, поэтому они далеки от совершенства). Добавлять поля в процессе эксплуатации базы возможно, но не желательно.

Выберем имена и типы полей в базе данных.

Имя поля	Тип поля	Описание поля
Название	Текст	Название объекта
Координаты	Широта / долгота	Широта и долгота
Изображение	Изображение	Изображение объекта
Адрес	Текст	Адрес объекта (почтовый)
URL_модель	Ссылка	URL модели
Автормодели	Ссылка	Автор модели (URL автора)
kml	Файл	Метка Google
status	Выпадающий список	Одобрение записи.
comment	Текстовая область	Комментарий к записи
Описание	Текстовая область	Описание модели

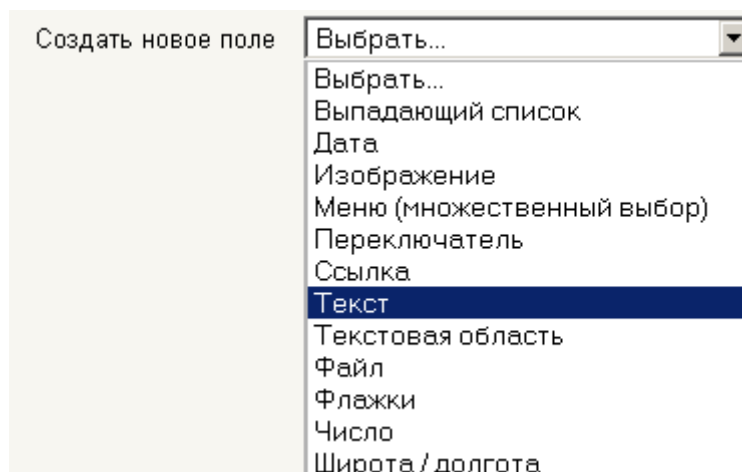
is_kname	Выпадающий список	Сделана ли модель в ХНАГХ?
----------	-------------------	----------------------------

Для того, чтобы вводить (редактировать) поля базы данных нужно нажать ссылку «Поля».



Поле «Название»

Добавляем поле «Название». Выбираем тип поля из списка



Вводим название и описание поля. Не разрешаем автосвязывание с глоссарием. Не забываем нажать кнопку «Сохранить».

Поле "Текст"

Имя поля

Описание поля

Разрешить автосвязывание

Поле «Координаты»

Добавляем поле «Координаты». Выбираем тип поля из списка. Вводим название и описание поля. Координаты будем использовать в программе Google Earth (параметр «Показать внешние связи»). Параметр «Как маркировать элементы в файлах KML (Google Earth)» не изменяем.

Поле "Долгота"/"Широта"

Имя поля	<input type="text" value="Координаты"/>
Описание поля	<input type="text" value="Широта и долгота объекта"/>
Показать внешние связи	<input type="checkbox"/> <ul style="list-style-type: none"> Google Maps <li style="background-color: #e0e0e0;">Google Earth Geabios OpenStreetMap Multimap
Как маркировать элементы в файлах KML (Google Earth)	<input type="text" value="Запись #"/>

Поле «Изображение»

Добавляем поле «Изображение». Выбираем тип поля из списка. Вводим название и описание поля. Мы будем вставлять изображения 300x150 пикселей. Записи, содержащие картинки других размеров, будем считать неправильными и одобрим их только после того, как студент вставит правильную картинку.

Поле "Изображение"

Имя поля	<input type="text" value="Изображение"/>
Описание поля	<input type="text" value="Изображение объекта"/>
Ширина при просмотре по одной записи, пикс.	<input type="text" value="300"/>
Высота при просмотре по одной записи, пикс.	<input type="text" value="150"/>
Ширина в отображении списком, пикс.	<input type="text" value="100"/>
Высота в отображении списком, пикс.	<input type="text" value="50"/>
Максимальный размер	<input type="text" value="10Мбайт"/>

Поле «Адрес»

Добавляем поле «Адрес». Выбираем тип поля из списка. Вводим название и описание поля.

Поле "Текст"

Имя поля	<input type="text" value="Адрес"/>
Описание поля	<input type="text" value="Адрес объекта"/>
Разрешить автосвязывание	<input type="checkbox"/>

Поле «URL_модель»

Добавляем поле «URL_модель». Выбираем тип поля из списка. Вводим название и описание поля.

Параметр «Принудительное название для ссылки» оставляем пустым. Обязательно устанавливаем флажок «Автоматически делать ссылкой».

Такое сочетание параметров установлено опытным путем. Оно обеспечивает возможность сортировать и фильтровать информацию по полю типа «Ссылка».

Поле "Ссылка"

Имя поля	<input type="text" value="URL_модель"/>
Описание поля	<input type="text" value="URL модели"/>
Принудительное название для ссылки	<input type="text"/>
Автоматически делать ссылкой	<input checked="" type="checkbox"/>

Поле «Автормодели»

Точно также добавляем поле «Автормодели».

Поле "Ссылка"

Имя поля	<input type="text" value="Автормодели"/>
Описание поля	<input type="text" value="Автор модели"/>
Принудительное название для ссылки	<input type="text"/>
Автоматически делать ссылкой	<input checked="" type="checkbox"/>

Поле «kml»

Добавляем поле «kml». Выбираем тип поля из списка. Вводим название и описание поля.

Файлы kml занимают немного места, поэтому выбираем минимальное значение параметра «Максимальный размер».

Поле "Файл"

Имя поля	<input type="text" value="kml"/>
Описание поля	<input type="text" value="Метка Google"/>
Максимальный размер	<input type="text" value="10Кбайт"/>

Поле «status»

Добавляем поле «status». Выбираем тип поля из списка. Вводим название и описание поля. Значения, которые можно записать в это поле задаем в окне «Варианты выбора (один на строку)».

Поле "Выпадающий список"

Имя поля	<input type="text" value="status"/>
Описание поля	<input type="text" value="Одобрение записи"/>
Варианты выбора (один на строку)	<div style="border: 1px solid red; padding: 2px;">Да Нет Повтор Исправить</div>

Поле «comment»

Добавляем поле «comment». Выбираем тип поля из списка. Вводим название и описание поля. Мы не предполагаем вводить длинные комментарии, поэтому вполне достаточно текстовой области шириной в 80 символов и высотой в 5 строчек.

Поле "Текстовая область"

Имя поля	<input type="text" value="comment"/>
Описание поля	<input type="text" value="Комментарий к записи"/>
Ширина	<input type="text" value="80"/> знаков
Высота	<input type="text" value="5"/> строк

Поле «Описание»

Добавляем поле «Описание». Выбираем тип поля из списка. Вводим название и описание поля. Описание объекта может быть достаточно большим, поэтому мы выбираем текстовой область шириной в 80 символов и высотой в 10 строчек.

Поле "Текстовая область"

Имя поля

Описание поля

Ширина знаков

Высота строк

Поле «is_kname»

Добавляем поле «is_kname». Предусмотрено только одно значение. Если модель сделана не студентами ХНАГХ, то это поле нужно оставить без изменений.

Поле "Выпадающий список"

Имя поля

Описание поля

Варианты выбора (один на строку)

Проектирование шаблонов

Если пользоваться общепринятой для баз данных терминологией, то шаблоны позволяют создавать «отчеты» (формы просмотра списка записей или одной записи) и «формы» ввода и редактирования. Шаблоны – единственное средство настройки поведения и внешнего вида базы данных.

В базе данных Moodle предусмотрены следующие шаблоны.

1. Шаблон списка
2. Шаблон одиночной записи
3. Шаблон расширенного поиска
4. Шаблон формы записи
5. Шаблон RSS
6. Шаблон CSS
7. Шаблон Javascript

Ниже приведено краткое описание каждого шаблона.

Шаблон списка.

Этот шаблон позволяет создавать фрагменты html-страницы, в которую выводится сразу несколько записей. В базе данных «Список трехмерных моделей города Харькова, размещенных в Google Earth» шаблон списка используется очень активно, причем задействованы практически все практически все возможности. Мы очень подробно все это опишем.

Шаблон одиночной записи.

Этот шаблон используется для описания фрагмента html-страницы, отображающей одну запись. Здесь должна отображаться вся информация, которая хранится в записи. Мы подробно опишем этот шаблон нашей базы данных.

Шаблон расширенного поиска.

Этот шаблон используется для определения критериев, по значениям которым можно искать конкретную информацию среди записей БД. Такие инструменты (расширенный поиск) очень часто можно встретить в сети.

Шаблон формы записи.

Данный шаблон создает форму, используемую для добавления и редактирования записей в базе данных. В существующей версии сохранение нового шаблона уничтожает старый. В дальнейшем планируется дать пользователю возможность создавать (а также именовать, импортировать и экспортировать) шаблоны для повторного использования. Мы обошли это ограничение таким образом – преподаватель добавляет специальные поля, в которые заносит свою оценку записи (поле «status») и комментарий (поле «comment»). Ниже приведена специальная инструкция для преподавателя по работе с базой.

Шаблон RSS.

Позволяет настраивать содержимое каналов RSS (новостной ленты) для записей базы данных. На момент написания статьи этот шаблон (зачем он нужен и как работает) так и остался для нас абсолютно неизвестным.

Шаблон CSS.

В этом шаблоне можно описать свои CSS (каскадные таблицы стилей). Moodle поместит эти таблицы в начале любой html-страницы базы данных. Используем этот шаблон и ниже его опишем.

Шаблон Javascript.

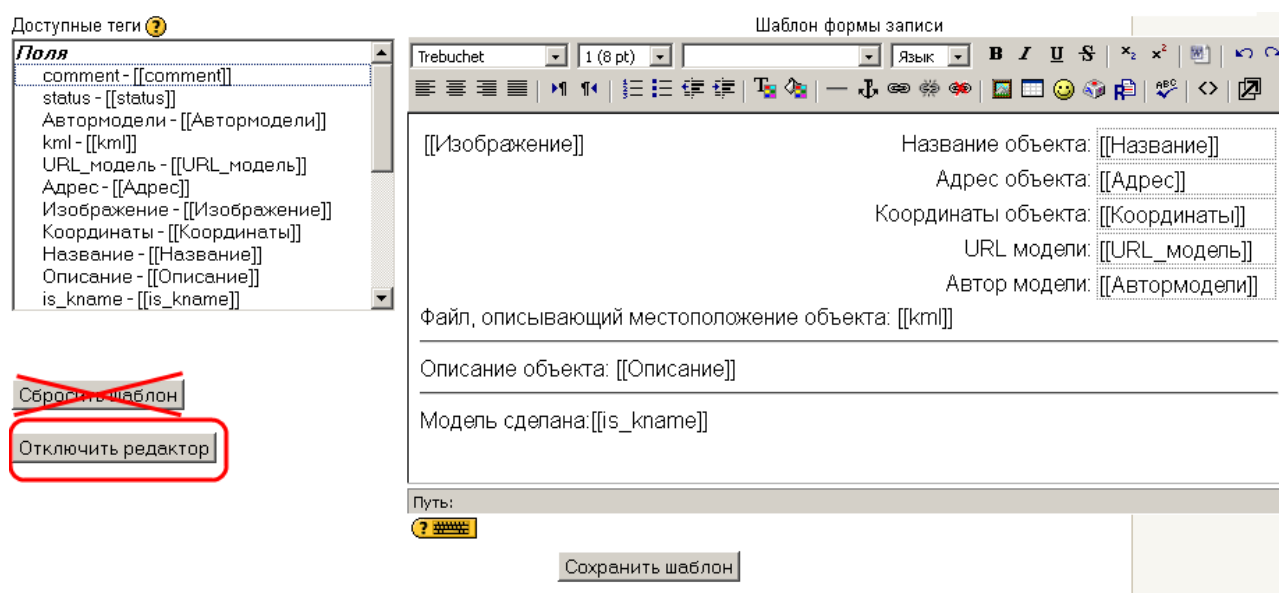
В этом шаблоне можно записывать описание общих функций и глобальных (для базы данных) переменных Javascript. В базе данных «Список трехмерных моделей города Харькова, размещенных в Google Earth» этот шаблон используется. Ниже мы его опишем.

В Moodle для редактирования шаблонов используется встроенный редактор. Мы его использовать не будем.

Мы будем создавать каждый шаблон в виде файла html, а затем переносим информацию (все строки файла между тегами **<body>** и **</body>**) в окно редактирования шаблона.

Кнопка «Сброс шаблона» используется для автоматического создания шаблона. Moodle. При этом старый шаблон уничтожается. Мы **НИКОГДА** не будем использовать кнопку «Сброс шаблона».

Работу с шаблонами мы начинаем с отключения стандартного редактора Moodle (по умолчанию он активирован).



Поля в шаблоне определяются так «[[имя_поля]]», например «[[Название]]». Поля называются «тегами», на наш взгляд название неудачное.

Кроме того определены специальные поля (теги).

##Edit## создает пиктограмму "Редактировать" (Рука с ручкой), щелчок по которой позволяет редактировать текущую запись (пиктограмма появляется только в том случае, если у пользователя есть права на редактирование)

##More## создает пиктограмму "Просмотр" (Лупа), щелчок по которой переводит пользователя в режим просмотра записей по одной, где может содержаться более подробная информация

##MoreURL## выполняет ту же функцию, что и предыдущий тег, позволяя вместо пиктограммы "Лупа" использовать любые гиперссылки (в частности, текстовые ссылки - например: "Подробнее").

Чтобы вставить данный тег, выделите текст или поле, которые будут служить ссылкой, щелкните по кнопке "Вставка ссылки" в визуальном редакторе вашего браузера и в поле "URL-адрес" введите **##MoreURL##**.

##Delete## создает пиктограмму "Удалить" (Косой крест), щелчок по которой позволяет удалить текущую запись (пиктограмма появляется только в том случае, если у пользователя есть права на редактирование)

##Approve## создает пиктограмму, которая позволяет одобрить запись, т.е. разрешить отображение текущей записи в базе данных (пиктограмма появляется только в том случае, если у пользователя есть право одобрять записи).

##Comments## создает ссылку для просмотра/редактирования комментариев (ссылка появляется только в том случае, если комментарии разрешены в установках БД).

##User## создает ссылку на личную карточку участника курса, который добавил запись в базу данных; текстом ссылки является имя участника

##timeadded## выводит время добавления записи.

##timemodified## выводит время изменения записи.

В приложении к статье находятся html-файлы всех шаблонов.

kha_list_rec.html – шаблон списка

kha_1rec.html – шаблон одиночной записи

kha_search.html – шаблон расширенного поиска

kha_edit_rec.html – шаблон формы записи

kha_pattern.css – шаблон CSS

kha_pattern.js – шаблон Javascript

Кроме того в приложении находится файл **kha_colors.html** – вспомогательный файл для подбора цветов отображения строк при просмотре списком.

Шаблон формы записи

Этот шаблон используется для ввода новой записи в базу данных и редактирование существующих.

Сначала рассмотрим файл этого шаблона полностью, а затем объясним назначение всех строк текста.

Полный текст шаблона формы записи

Полный текст шаблона приведен ниже (файл **kha_edit_rec.html**).

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
5 <title>Редактирование записи</title>
6 </head>
7 <body>
8 <table width="100%" border="0"><tbody>
9 <tr>
10 <td valign="top" style="border-style: none;
width: 305px; background-image: none; text-align: left;
vertical-align: top;" colspan="1" rowspan="5">
[[Изображение]]<br /><br />
11 </td>
```

```
12      <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;"> Название объекта:
13      </td>
14      <td valign="top">[[Название]]
15      </td>
16  </tr>
17  <tr>
18      <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">Адрес объекта: <br />
19      </td>
20      <td valign="top">[[Адрес]]
21      </td>
22  </tr>
23  <tr>
24      <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">Координаты объекта: <br />
25      </td>
26      <td valign="top">[[Координаты]]
27      </td>
28  </tr>
29  <tr>
30      <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">URL модели: <br />
31      </td>
32      <td valign="top">[[URL_модель]]
33      </td>
34  </tr>
35  <tr>
36      <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">Автор модели: <br />
37      </td>
38      <td valign="top">[[Автормодели]]
39      </td>
```

```

40     </tr> </tbody>
41 </table>
42 Файл , описывающий местоположение объекта: [[kml]] <hr
/>
43 Описание объекта: [[Описание]]<hr />
44 Модель сделана: [[is_kname]]
45 Запись одобрена: [[status]]
46 Комментарий: [[comment]]
47 </body>
48 </html>

```

В браузере это файл выглядит так:

[[Изображение]]	Название объекта: [[Название]]
	Адрес объекта: [[Адрес]]
	Координаты объекта: [[Координаты]]
	URL модели: [[URL_модель]]
	Автор модели: [[Автормодели]]

Файл, описывающий местоположение объекта: [[kml]]

Описание объекта: [[Описание]]

Модель сделана: [[is_kname]] Запись одобрена: [[status]] Комментарий: [[comment]]

Окно редактирования шаблона формы записи

В окне редактирования шаблона находятся строки №8-44 из этого файла.

Доступные теги ?

Поля

- comment - [[comment]]
- status - [[status]]
- Автормодели - [[Автормодели]]
- kml - [[kml]]
- URL_модель - [[URL_модель]]
- Адрес - [[Адрес]]
- Изображение - [[Изображение]]
- Координаты - [[Координаты]]
- Название - [[Название]]
- Описание - [[Описание]]
- is_kname - [[is_kname]]

~~Сбросить шаблон~~

~~Включить редактор~~

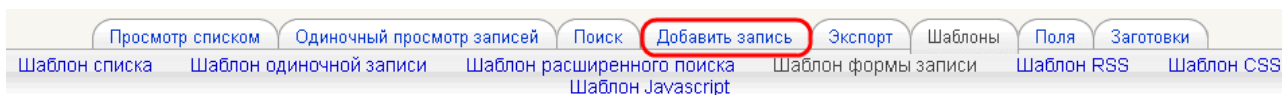
Шаблон формы записи

```

<table width="100%" border="0"><tbody>
  <tr>
    <td valign="top" style="border-style: none; width: 305px;
background-image: none; text-align: left; vertical-align: top;"
colspan="1" rowspan="5"> [[Изображение]]<br /><br />
    </td>
    <td valign="top" style="border-style: none; background-image: none;
text-align: right; vertical-align: top;"> Название объекта:
    </td>
    <td valign="top"> [[Название]]
    </td>
  </tr>
  <tr>
    <td valign="top" style="border-style: none; background-image: none;
text-align: right; vertical-align: top;">Адрес объекта: <br />
    </td>
    <td valign="top"> [[Адрес]]
    </td>
  </tr>
  <tr>
    <td valign="top" style="border-style: none; background-image: none;

```

Сохранить шаблон



Если нажать ссылку «Добавить запись», то появится окно ввода записи.

Новая запись

<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> Изображение <input style="width: 100%;" type="text"/> <input type="button" value="Обзор..."/> </div> <div style="border: 1px solid #ccc; padding: 5px;"> Альтернативный текст <input style="width: 100%;" type="text"/> </div>	<div style="margin-bottom: 5px;"> Название объекта: <input style="width: 100%;" type="text"/> </div> <div style="margin-bottom: 5px;"> Адрес объекта: <input style="width: 100%;" type="text"/> </div> <div style="margin-bottom: 5px;"> Координаты объекта: <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">Широта</td> <td style="width: 20%;"><input style="width: 80%;" type="text"/></td> <td style="width: 30%;">°N</td> </tr> <tr> <td>Долгота</td> <td><input style="width: 80%;" type="text"/></td> <td>°E</td> </tr> </table> </div> <div style="margin-bottom: 5px;"> URL Ссылка: <input style="width: 100%;" type="text"/> </div> <div style="margin-bottom: 5px;"> модели: <input style="width: 100%;" type="text"/> </div> <div style="margin-bottom: 5px;"> Автор Ссылка: <input style="width: 100%;" type="text"/> </div> <div style="margin-bottom: 5px;"> модели: <input style="width: 100%;" type="text"/> </div>	Широта	<input style="width: 80%;" type="text"/>	°N	Долгота	<input style="width: 80%;" type="text"/>	°E
Широта	<input style="width: 80%;" type="text"/>	°N					
Долгота	<input style="width: 80%;" type="text"/>	°E					

Файл, описывающий местоположение объекта:

<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> Файл <input style="width: 100%;" type="text"/> <input type="button" value="Обзор..."/> </div>	<div style="border: 1px solid #ccc; padding: 5px;"> Отображаемая ссылка на файл (необязательно) <input style="width: 100%;" type="text"/> </div>
---	--

Описание объекта:

HTML-редакторе ?

Модель сделана:

Выбрать...

Описание шаблона формы записи

№1-7 – стандартный заголовок файла html. Их не нужно переносить в окно редактирование шаблона.

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">

```

```

2 <html xmlns="http://www.w3.org/1999/xhtml">

```

```

3 <head>

```

```

4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />

```

5 <title>Редактирование записи</title>

6 </head>

7 <body>

№8 – начало таблицы, №9 – начало строки таблицы.

8 <table width="100%" border="0"><tbody>

9 <tr>

Поле «Изображение»

№10-11 описывают столбец таблицы, в котором информация вводится в поле «Изображение».

10 <td valign="top" style="border-style: none; width: 305px; background-image: none; text-align: left; vertical-align: top;" colspan="1" rowspan="5"> [[Изображение]]

11 </td>

Изображение <input type="text"/> Обзор... Альтернативный текст <input type="text"/>	Название объекта: <input type="text"/> Адрес объекта: <input type="text"/> Координаты объекта: <table border="0" style="margin-left: 20px;"> <tr> <td>Широта</td> <td><input type="text"/></td> <td>°N</td> </tr> <tr> <td>Долгота</td> <td><input type="text"/></td> <td>°E</td> </tr> </table>	Широта	<input type="text"/>	°N	Долгота	<input type="text"/>	°E
Широта	<input type="text"/>	°N					
Долгота	<input type="text"/>	°E					
	URL Ссылка: <input type="text" value="http://"/> модели: Текст: <input type="text"/>						
	Автор Ссылка: <input type="text" value="http://"/> модели: Текст: <input type="text"/>						

Обратите внимание на тег **rowspan="5"**. Это тег означает, что столбец объединяет 5 строк.

Поле «Название»

№12-15 описывают два столбца таблицы, которые предназначены для ввода информации в поле «Название». №16 завершает строку таблицы, начатую в строке №9.

12 <td valign="top" style="border-style: none; background-image: none; text-align: right; vertical-align: top;"> Название объекта:


```

13     </td>
14     <td valign="top">[[Название]]
15     </td>
16 </tr>

```

Название
объекта:

Поле «Адрес»

№17-22 описывают строку таблицы (и два столбца внутри нее), которая предназначены для ввода информации в поле «Адрес».

```

17 <tr>
18   <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">Адрес объекта: <br />
19   </td>
20   <td valign="top">[[Адрес]]
21   </td>
22 </tr>

```

Адрес
объекта:

Поле «координаты»

№23-28 описывают строку таблицы (и два столбца внутри нее), которая предназначены для ввода информации в поле «Координаты».

```

23 <tr>
24   <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">Координаты объекта: <br />
25   </td>
26   <td valign="top">[[Координаты]]
27   </td>
28 </tr>

```

Координаты объекта:	Широта <input type="text"/> °N
	Долгота <input type="text"/> °E

Поле «URL_модель»

№29-34 описывают строку таблицы (и два столбца внутри нее), которая предназначена для ввода информации в поле «URL_модель».

```

29 <tr>
30 <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">URL модели: <br />
31 </td>
32 <td valign="top">[[URL_модель]]
33 </td>
34 </tr>

```

URL Ссылка:	<input type="text" value="http://"/>
модели: Текст:	<input type="text"/>

Поле «Автормодели»

№35-40 описывают строку таблицы (и два столбца внутри нее), которая предназначена для ввода информации в поле «Автормодели». Кроме того, в №40-41 заканчивается описание таблицы, начатое в №8.

```

35 <tr>
36 <td valign="top" style="border-style: none;
background-image: none; text-align: right; vertical-
align: top;">Автор модели: <br />
37 </td>
38 <td valign="top">[[Автормодели]]
39 </td>
40 </tr> </tbody>
41 </table>

```

Автор Ссылка:	<input type="text" value="http://"/>
модели: Текст:	<input type="text"/>

Поле «kml»

В №42 вводится информация в поле «kml».

42 Файл, описывающий местоположение объекта: `[[kml]] <hr />`

Файл, описывающий местоположение объекта:

Файл <input type="text"/>	<input type="button" value="Обзор.."/>
Отображаемая ссылка на файл (необязательно) <input type="text"/>	

Поле «Описание»

В №43 вводится информация в поле «Описание».

43 Описание объекта: `[[Описание]]<hr />`

Описание объекта:

--

Поле «is_kname»

В №44 вводится информация в поле «is_kname».

44 Модель сделана: `[[is_kname]]`

Модель сделана:

<input type="button" value="Выбрать..."/>

№45-46 пока не заносим в шаблон. Ввод информации в поле «**status**» (№45) и «**comment**» (№46) мы рассмотрим позже. Теги `</body>` и `</html>` (№47-48) завершают файл **html**.

45 Запись одобрена: `[[status]]`

46 Комментарий: `[[comment]]`

47 `</body>`

48 `</html>`

Шаблон одиночной записи

Этот шаблон используется для просмотра одной записи базы данных. Сначала рассмотрим файл этого шаблона полностью, а затем объясним назначение всех строк текста.

Полный текст шаблона одиночной записи

Полный текст шаблона приведен ниже (файл `kha_1rec.html`).

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
5 <title>Просмотр 1 записи</title>
6 </head>
7 <body>
8 <table width="100%" border="0"><tbody>
9   <tr>
10     <td valign="top" style="width: 305px; background-
image: none; text-align: left; vertical-align: top;
border-style: none;" colspan="1" rowspan="9">
[[Изображение]]<br /><br />
11     </td>
12     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;"> Название объекта:
13     </td>
14     <td valign="top"><b>[[Название]]</b> <font
color="#CC9900"><b><i>[[is_kname]]</i></b></font>
15     </td>
16   </tr>
17   <tr>
18     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Адрес объекта:
19     </td>
```

```
20     <td valign="top"><b>[[Адрес]]</b>
21     </td>
22 </tr>
23 <tr>
24     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Координаты объекта:
25     </td>
26     <td valign="top"><b>[[Координаты]]</b>
27     </td>
28 </tr>
29 <tr>
30     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">URL модели:
31     </td>
32     <td valign="top"><b>[[URL_модель]]</b>
33     </td>
34 </tr>
35 <tr>
36     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Автор модели:
37     </td>
38     <td valign="top"><b>[[Автормодели]]</b>
39     </td>
40 </tr>
41 <tr>
42     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Файл, описывающий местоположение объекта:
43     </td>
44     <td valign="top"><b>[[kml]]</b>
45     </td>
46 </tr>
47 <tr>
48     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
```

```

none;">Автор записи:
49     </td>
50     <td valign="top"><b>##user##</b>
51     </td>
52 </tr>
53 <tr>
54     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Дата внесения:
55     </td>
56     <td valign="top"><b>##timeadded##</b>
57     </td>
58 </tr>
59 <tr>
60     <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Последнее изменение:
61     </td>
62     <td valign="top"><b>##timemodified##</b>
63     </td>
64 </tr> </tbody>
65 </table>
66 <hr /> Описание объекта: <b>[[Описание]]</b><hr />
67 Запись одобрена: <b>[[status]]</b><br />
68 <span style="color: rgb(255, 0,
0);">Комментарий:</span><b>
69 <span style="background-color: rgb(255, 255, 102);
color: rgb(0, 0, 255);">[[comment]]</span></b>
70 <br /> ##edit## ##delete##
71 </body>
72 </html>

```

В браузере этот файл будет выглядеть так:

[[Изображение]]

Название объекта: [[Название]]
[[is_kname]]

Адрес объекта: [[Адрес]]

Координаты объекта: [[Координаты]]

URL модели: [[URL_модель]]

Автор модели: [[Автормодели]]

Файл, описывающий [[kml]]

местоположение объекта:

Автор записи: ##user##

Дата внесения: ##timeadded##

Последнее изменение: ##timemodified##

Описание объекта: [[Описание]]

Запись одобрена: [[status]]

Комментарий: [[comment]]

##edit## ##delete##

Окно редактирования шаблона одиночной записи

В окне редактирования шаблона находятся строки №8-70 из этого файла.

Доступные теги ?

Поля

- comment - [[comment]]
- status - [[status]]
- Автормодели - [[Автормодели]]
- kml - [[kml]]
- URL_модель - [[URL_модель]]
- Адрес - [[Адрес]]
- Изображение - [[Изображение]]
- Координаты - [[Координаты]]
- Название - [[Название]]
- Описание - [[Описание]]
- is_kname - [[is_kname]]

Сбросить шаблон

Включить редактор


Сохранить шаблон

Шаблон одиночной записи

```
<table width="100%" border="0"><tbody>
  <tr>
    <td valign="top" style="width: 305px; background-image: none; text-align: left; vertical-align: top; border-style: none;" colspan="1" rowspan="9"> [[Изображение]] <br /><br />
    </td>
    <td valign="top" style="background-image: none; text-align: right; vertical-align: top; border-style: none;"> Название объекта:
    </td>
    <td valign="top"><b>[[Название]]</b> <font color="#CC9900"><b><i>[[is_kname]]</i></b></font>
    </td>
  </tr>
  <tr>
    <td valign="top" style="background-image: none; text-align: right; vertical-align: top; border-style: none;">Адрес объекта:
    </td>
    <td valign="top"><b>[[Адрес]]</b>
    </td>
  </tr>
</tbody>
</table>
```

Описание шаблона одиночной записи

В режиме «Просмотр списком» находим запись «Харьковская кенасса» и нажимаем иконку «подробнее»

	Харьковская кенасса ХНАГХ	49.0000°N, 36.0000°E метка	http://sketchup.google.com/3dwarehouse/details?mid=2ffd091484947a9720e9008a053603	Ева Мойковская	Воскресенье 9 Декабрь 2012, 00:03	Герасименко Анастасия	Исправить ✖
---	------------------------------	-------------------------------	---	----------------	---	--------------------------	----------------

Страница: (Предыдущая) 1 ... 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 ...109 (Следующая)



Название **Харьковская кенасса ХНАГХ**
 объекта:
 Адрес объекта: **Украина, г. Харьков, ул. Кузнечная, 24**
 Координаты **49.0000°N, 36.0000°E**
 объекта:
 URL модели: <http://sketchup.google.com/3dwarehouse/details?mid=2ffdc091484947a9720c9008a053603>
 Автор модели: **Eva Morkovskaya**
 Файл, **метка**
 описывающий
 местоположение
 объекта:
 Автор записи: **Герасименко Анастасия**
 Дата внесения: **Воскресенье 9 Декабрь 2012, 00:03**
 Последнее **Среда 2 Январь 2013, 07:34**
 изменение:

Описание объекта: **Харьковская кенасса — культовое сооружение караимов, памятник архитектуры конца XIX века. Кенасса была построена в 1891—1893 году по проекту архитектора Б. С. Покровского на углу Подольского переулка (ныне ул. Гамарника) и Кузнечной улицы. Караимская община была распущена советской властью в 1929 г., а в здании кенассы обосновался клуб-музей «Воинствующий безбожник». Позже в здании располагалась транспортная организация. В 2006 году кенасса была возвращена караимской религиозной общине.**

Запись одобрена: **Исправить**

Комментарий: **Координаты**



№1-7 – стандартный заголовок файла html. Их не нужно переносить в окно редактирование шаблона.

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
5 <title>Просмотр 1 записи</title>
6 </head>
7 <body>

```

№8 – начало таблицы, №9 – начало строки таблицы.

```

8 <table width="100%" border="0"><tbody>
9 <tr>

```


Поле «Изображение»

№10-11 описывают столбец таблицы, в котором информация вводится в поле «Изображение».

```
10 <td valign="top" style="width: 305px; background-
image: none; text-align: left; vertical-align: top;
border-style: none;" colspan="1" rowspan="9">
[[Изображение]]<br /><br />
```

```
11 </td>
```



Название Харьковская кенасса **ХНАГХ**
 объекта:
 Адрес объекта: Украина, г. Харьков, ул. Кузнечная, 24
 Координаты **49.0000°N, 36.0000°E**
 объекта:
 URL модели: <http://sketchup.google.com/3dwarehouse/details?mid=2ffdc091484947a9720c9008a053603>
 Автор модели: [Eva Morkovskaya](#)
 Файл, [?](#) метка
 описывающий
 местоположение
 объекта:
 Автор записи: [Герасименко Анастасия](#)
 Дата внесения: **Воскресенье 9 Декабрь 2012, 00:03**
 Последнее **Среда 2 Январь 2013, 07:34**
 изменение:

Обратите внимание на тег **rowspan="9"**. Это тег означает, что столбец объединяет 9 строк.

Поля «Название» и «is_kname»

В №12-15 выводятся на экран значения полей «Название» и «is_kname». В №16 закрывается 1-я строка таблицы (открывается в №9).

```
12 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;"> Название объекта:
```

```
13 </td>
```

```
14 <td valign="top"><b>[[Название]]</b> <font
color="#CC9900"><b><i>[[is_kname]]</i></b></font>
```

```
15 </td>
```

```
16 </tr>
```

Название Харьковская кенасса **ХНАГХ**
объекта:

Поле «Адрес»

В №17-22 на экран выводится поле «Адрес».

```
17 <tr>
18 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Адрес объекта:
19 </td>
20 <td valign="top"><b>[[Адрес]]</b>
21 </td>
22 </tr>
```

Адрес объекта: Украина, г. Харьков, ул. Кузнечная, 24

Поле «Координаты»

В №23-28 на экран выводится поле «Координаты».

```
23 <tr>
24 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Координаты объекта:
25 </td>
26 <td valign="top"><b>[[Координаты]]</b>
27 </td>
28 </tr>
```

Координаты **49.0000°N, 36.0000°E**
объекта:

В этой записи координаты введены неправильно (с точностью до целых градусов, ошибка около 100 км).

Поле «URL_модель»

В №29-34 на экран выводится поле «URL_модель».

```
29 <tr>
```

```

30      <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">URL модели:
31      </td>
32      <td valign="top"><b>[[URL_модель]]</b>
33      </td>
34 </tr>

```

URL модели: <http://sketchup.google.com/3dwarehouse/details?mid=2ffdc091484947a9720c9008a053603>

Поле «Автормодели»

В №35-40 на экран выводится поле «Автормодели».

```

35 <tr>
36 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Автор модели:
37 </td>
38 <td valign="top"><b>[[Автормодели]]</b>
39 </td>
40 </tr>

```

Автор модели: [Eva Morkovskaya](#)

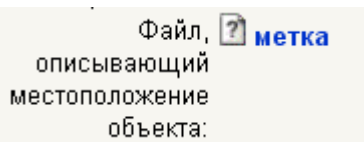
Поле «kml»

В №41-46 на экран выводится поле «kml» (метка Google).

```

41 <tr>
42 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Файл, описывающий местоположение объекта:
43 </td>
44 <td valign="top"><b>[[kml]]</b>
45 </td>
46 </tr>

```



Служебное поле «##user##»

В №47-52 на экран выводится служебное поле «##user##». Это – пользователь, который внес запись.

```

47 <tr>
48 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Автор записи:
49 </td>
50 <td valign="top"><b>##user##</b>
51 </td>
52 </tr>

```

Автор записи: Герасименко Анастасия

Служебное поле «##timeadded##»

В №53-58 на экран выводится служебное поле «##timeadded##». Это – время ввода записи.

```

53 <tr>
54 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Дата внесения:
55 </td>
56 <td valign="top"><b>##timeadded##</b>
57 </td>
58 </tr>

```

Дата внесения: Воскресенье 9 Декабрь 2012, 00:03

Служебное поле «##timemodified ##»

В №59-64 на экран выводится служебное поле «##timemodified##». Это – время последнего изменения записи. В №64-65 заканчивается описание таблицы (№8 – начало описания таблицы).

```

59 <tr>
60 <td valign="top" style="background-image: none;
text-align: right; vertical-align: top; border-style:
none;">Последнее изменение:
61 </td>
62 <td valign="top"><b>##timemodified##</b>
63 </td>
64 </tr> </tbody>
65 </table>

```

Последнее Среда 2 Январь 2013, 07:34
изменение:

Поле «Описание»

В №66 на экран выводится поле «Описание».

```
66 <hr /> Описание объекта: <b>[[Описание]]</b><hr />
```

Описание объекта: Харьковская кенасса — культовое сооружение караимов, памятник архитектуры конца XIX века. Кенасса была построена в 1891—1893 году по проекту архитектора Б. С. Покровского на углу Подольского переуллка (ныне ул. Гамарника) и Кузнечной улицы. Караимская община была распущена советской властью в 1929 г., а в здании кенассы обосновался клуб-музей «Воинствующий безбожник». Позже в здании располагалась транспортная организация. В 2006 году кенасса была возвращена караимской религиозной общине.

В №67 на экран выводится поле «**status**». Значение поля показывает, что запись нужно исправить.

```
67 Запись одобрена: <b>[[status]]</b><br />
```

Запись одобрена: **Исправить**

Поле «comment»

В №68-69 на экран выводится поле «**comment**». Значение поля указывает на ошибку (координаты введены неправильно).

```

68 <span style="color: rgb(255, 0,
0);">Комментарий:</span><b>
69 <span style="background-color: rgb(255, 255, 102);
color: rgb(0, 0, 255);">[[comment]]</span></b>

```


Комментарий: **Координаты**


Служебные поля «##edit##» и «##delete##»

В №70 на экран выводятся служебные поля «##edit##» (иконка редактирования записи) и «##delete##» (иконка удаления записи).

70 `
 ##edit## ##delete##`



Если нажать иконку редактировать () , то появится форма ввода и редактирования информации. Она описывается «шаблоном формы записи».

Изображение <input type="text"/> <input type="button" value="Обзор..."/> Альтернативный текст <input type="text" value="Харьковская кенасса"/> 	Название объекта: <input type="text" value="Харьковская кенасса"/> Адрес объекта: <input type="text" value="Украина, г. Харьков, ул. Кузнечная, 24"/> Координаты объекта: <table border="1"> <tr> <td>Широта</td> <td><input type="text" value="49"/></td> <td>°N</td> </tr> <tr> <td>Долгота</td> <td><input type="text" value="36"/></td> <td>°E</td> </tr> </table>	Широта	<input type="text" value="49"/>	°N	Долгота	<input type="text" value="36"/>	°E
Широта	<input type="text" value="49"/>	°N					
Долгота	<input type="text" value="36"/>	°E					
URL Ссылка: <input type="text" value="http://sketchup.google.com/3dwarehouse/"/> модели: <input type="text" value="http://sketchup.google.com/3dwarehouse/"/> Автор Ссылка: <input type="text" value="http://sketchup.google.com/3dwarehouse/"/> модели: <input type="text" value="Eva Morkovskaya"/>							
Файл, описывающий местоположение объекта: Файл <input type="text"/> <input type="button" value="Обзор..."/> Отображаемая ссылка на файл (необязательно) <input type="text" value="метка"/>							
<input type="button" value="метка"/>							
Описание объекта: Харьковская кенасса – культовое сооружение караимов, памятник архитектуры конца XIX века. Кенасса была построена в 1891–1893 году по проекту архитектора Б. С. Покровского на углу Подольского переулочка (ныне ул. Гамарника) и Кузнечной улицы. Караимская община была распущена советской властью в 1929 г., а в здании кенассы обосновался клуб-музей «Воинствующий безбожник». Позже в здании располагалась транспортная организация. В 2006 году кенасса была возвращена караимской религиозной общине.							
HTML-редакторе ?							
Модель сделана: <input type="text" value="ХНАГХ"/>							
<input type="button" value="Сохранить и просмотреть"/> <input type="button" value="Отмена"/>							

№71-72 завершают **html** – файл.

71 </body>

72 </html>

Работа преподавателя с базой

Вернемся теперь к работе с полями «status» (одобрение записи) и «comment» (комментарий к записи). Нам нужно сделать так, чтобы преподаватель мог изменять эти поля, а студент – только просматривать.

Это реализовано с помощью изменения «Шаблона формы записи». Алгоритм работы преподавателя с базой данных:

1. Закрыть базу для студентов.
2. Добавить в конец «Шаблона формы записи» строку:

Запись одобрена: [[status]] Комментарий: [[comment]]

3. Одобрять (или не одобрять записи), добавлять комментарии.
4. Удалить из «Шаблона формы записи» строку:

Запись одобрена: [[status]] Комментарий: [[comment]]

5. Открыть базу для студентов.

Дополнительная часть формы ввода и редактирования информации выглядит так.

Запись одобрена:



Исправить ▾

Комментарий:

Trebuchet 1 (8 pt) Язык **B** *I* U ~~S~~ x₂ x²

Координаты

Путь:

 [О HTML-редакторе](#) 

Сохранить и просмотреть Отмена

Шаблон расширенного поиска

Этот шаблон предназначен для описания критериев поиска информации.

Для вызова формы поиска информации нужно установить флажок «Расширенный поиск»

Сортировать по Расширенный поиск

URL модели:

Автор модели:

Имя автора записи:

Фамилия автора записи:

Модель сделана:

Запись одобрена:

Полный текст шаблона расширенного поиска

Сначала рассмотрим файл этого шаблона полностью, а затем объясним назначение всех строк текста.

Полный текст шаблона приведен ниже (файл `kha_search.html`).

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
5 <title>Поиск</title>
6 </head>
7 <div class="defaulttemplate">
8   <table cellpadding="5"><tbody>
9     <tr>
10      <td valign="top" align="right">URL модели:
11      </td>
12      <td>[[URL_модель]]
13      </td>
14    </tr>
15    <tr>
16      <td valign="top" align="right">Автор модели:

```



```

17         </td>
18         <td>[[Автормодели]]
19         </td>
20     </tr>
21     <tr>
22         <td valign="top" align="right">Имя автора
записи:
23         </td>
24         <td>##firstname##
25         </td>
26     </tr>
27     <tr>
28         <td valign="top" align="right">Фамилия автора
записи:
29         </td>
30         <td>##lastname##
31         </td>
32     </tr>
33     <tr>
34         <td valign="top" align="right">Модель сделана:
35         </td>
36         <td>[[is_kname]]
37         </td>
38     </tr>
39     <tr>
40         <td valign="top" align="right">Запись одобрена:
41         </td>
42         <td>[[status]]
43         </td>
44     </tr>
45 </tbody>
46 </table></div>
47 </body>
48 </html>

```

В браузере этот файл будет выглядеть так:

URL модели: [[URL_модель]]
 Автор модели: [[Автормодели]]
 Имя автора записи: ##firstname##
 Фамилия автора записи: ##lastname##
 Модель сделана: [[is_kname]]
 Запись одобрена: [[status]]

Окно редактирования шаблона расширенного поиска

В окне редактирования шаблона находятся строки №8-46 из этого файла.



Описание шаблона расширенного поиска

№1-7 – стандартный заголовок файла html. Их не нужно переносить в окно редактирование шаблона.

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

2 <html xmlns="http://www.w3.org/1999/xhtml">

3 <head>

4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

5 <title>Поиск</title>

```
6 </head>
7 <div class="defaulttemplate">
```

№8 – начало таблицы.

```
8 <table cellpadding="5"><tbody>
```

Поле «URL_модель»

№9-14 описывают строку таблицы, в которой информация вводится в поле «URL_модель».

```
9 <tr>
10 <td valign="top" align="right">URL модели:
11 </td>
12 <td>[[URL_модель]]
13 </td>
14 </tr>
```

URL модели:

Поле «Автормодели»

№15-20 описывают строку таблицы, в которой информация вводится в поле «Автормодели».

```
15 <tr>
16 <td valign="top" align="right">Автор модели:
17 </td>
18 <td>[[Автормодели]]
19 </td>
20 </tr>
```

Автор модели:

Специальное поле «##firstname##»

№21-26 описывают строку таблицы, в которой информация вводится в специальное поле «##firstname##».

```

21      <tr>
22          <td valign="top" align="right">Имя автора
записи:
23      </td>
24          <td>##firstname##
25      </td>
26      </tr>

```

Имя автора записи:

Специальное поле «##lastname##»

№27-32 описывают строку таблицы, в которой информация вводится в специальное поле «##lastname##».

```

27      <tr>
28          <td valign="top" align="right">Фамилия автора
записи:
29      </td>
30          <td>##lastname##
31      </td>
32      </tr>

```

Фамилия автора записи:

Поле «is_kname»

№27-32 описывают строку таблицы, в которой информация в поле «is_kname».

```

33      <tr>
34          <td valign="top" align="right">Модель сделана:
35      </td>
36          <td>[[is_kname]]
37      </td>
38      </tr>

```

Модель сделана:

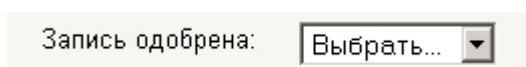
Поле «status»

№27-32 описывают строку таблицы, в которой информация в поле «status».

```

39 <tr>
40 <td valign="top" align="right">Запись одобрена:
41 </td>
42 <td>[[status]]
43 </td>
44 </tr>

```



№45-46 завершают таблицу (начало в №8). №47-48 завершают html – файл.

```

45 </tbody>
46 </table></div>
47 </body>
48 </html>

```

Шаблон CSS

Мы будем использовать стили CSS (каскадные таблицы стилей) для того, чтобы «раскрашивать» записи при просмотре списком. Записи имеющие разный статус (одобрено, не одобрено, исправить, повтор) будут иметь различный цвет фона и текста.

В принципе для этой задачи создавать отдельную таблицу стилей совсем не обязательно. Можно указать соответствующие опции каждого тега. Но, если Вам захочется изменить какой-нибудь параметр (цвет), то исправление в CSS – самый быстрый и эффективный способ.

Moodle добавляет файл шаблона CSS к любой своей странице, которая работает с базой (просмотр списком, просмотр одиночной записи, ввод и редактирование записи).

В исходном коде страницы шаблон CSS подгружается так.

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="ru" xml:lang="ru">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <link rel="stylesheet" type="text/css" href="http://cdo.kname.edu.ua/theme/standard/styles.php" />
6 <link rel="stylesheet" type="text/css" href="http://cdo.kname.edu.ua/theme/formal_white/styles.php" />
7 <link rel="stylesheet" type="text/css" href="http://cdo.kname.edu.ua/mod/data/css.php?d=738" /> <script
  type="text/javascript" src="http://cdo.kname.edu.ua/mod/data/js.php?d=738"></script>

```

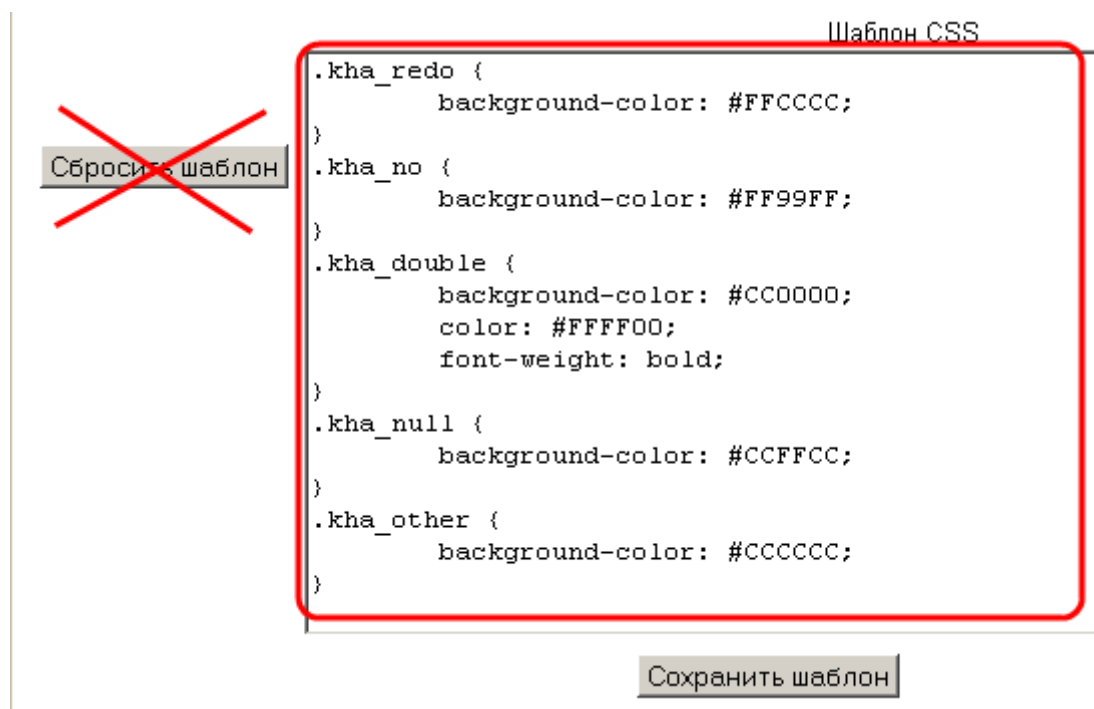
Полный текст шаблона CSS

Ниже приведен полный текст шаблона CSS (файл **kha_pattern.css**).

```
1 .kha_redo {
2   _background-color: #FFCCCC;
3 }
4 .kha_no {
5   _background-color: #FF99FF;
6 }
7 .kha_double {
8   _background-color: #CC0000;
9   _color: #FFFF00;
10  _font-weight: bold;
11 }
12 .kha_null {
13   _background-color: #CCFFCC;
14 }
15 .kha_other {
16   _background-color: #CCCCCC;
17 }
```

Окно редактирования шаблона CSS

Этот файл полностью записывается в окно редактирования шаблона.



Файл проверки стилей

Напомним, что для предотвращения конфликтов и обеспечения уникальности имен (в данном случае имен стилей) все имена начинаются с строки «**kha_**».

Предназначение каждого стиля рассмотрим на примере. Для подбора цветов мы создали специальный файл html (**kha_colors.html**). Текст этого файла приведен ниже.

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
```

```
2 <html xmlns="http://www.w3.org/1999/xhtml">
```

```
3 <head>
```

```
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
```

```
5 <title>Проверка стилей</title>
```

```
6 <link href="kha_pattern.css" rel="stylesheet"
type="text/css" />
```

```
7 </head>
```

```
8 <body>
```

```
9 <table width="100%" border="1" cellspacing="2"
cellpadding="5">
```

```
10 <tr>
11   <td>Одобрено</td>
12   <td>&nbsp;</td>
13 </tr>
14 <tr class="kha_redo">
15   <td>Исправить</td>
16   <td>kha_redo</td>
17 </tr>
18 <tr class="kha_no">
19   <td>Не одобрено</td>
20   <td>kha_no</td>
21 </tr>
22 <tr class="kha_double">
23   <td>Повтор</td>
24   <td>kha_double</td>
25 </tr>
26 <tr class="kha_null">
27   <td>NULL</td>
28   <td>kha_null</td>
29 </tr>
30 <tr class="kha_other">
31   <td>Other</td>
32   <td>kha_other</td>
33 </tr>_
34 </table>
35 </body>
36 </html>
```

В браузере этот файл отображается так.

Одобрено	
Исправить	kha_redo
Не одобрено	kha_no
Повтор	kha_double
NULL	kha_null
Other	kha_other

В каждой строке таблицы цветовая гамма отображения информации в зависимости от значения поля «**status**». При проектировании базы мы определили (неудачно) такие значения поля:

Поле "Выпадающий список"

Имя поля

Описание поля

Варианты выбора (один на строку)

Да
Нет
Повтор
Исправить

Значения «Да» и «Нет» требуют специального разъяснения. Поэтому мы их заменим на «Одобрено» и «Не одобрено». Практическая реализация этой замены будет описана ниже.

Кроме того мы предусмотрели ситуацию, когда полю «**status**» еще не присвоено значение (студент создал запись и она ожидает оценки преподавателя). В этом случае мы присваиваем полю значение «NULL». Если по какой-то причине поле «**status**» будет иметь другое значение, то мы будем использовать для отображения таких записей значение «Other».

Описание файла проверки стилей

Рассмотрим подробно работу файла **kha_colors.html**.

№1-8 – стандартное начало файла html. В №6 в исходный файл подгружается файл «**kha_pattern.css**».

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
```


Исправить	kha_redo
-----------	----------

Стиль для статуса «Не одобрено»

В №14-17 определены параметры выхода записей, имеющих статус «Не одобрено». Для отображения таких записей используется стиль (**class**) «**kha_no**». Стиль применяется к строке таблицы (№18).

```

18 <tr class="kha_no">
19   <td>Не одобрено</td>
20   <td>kha_no</td>
21 </tr>

```

Не одобрено	kha_no
-------------	--------

Стиль для статуса «Повтор»

В №22-25 определены параметры выхода записей, имеющих статус «Повтор». Для отображения таких записей используется стиль (**class**) «**kha_double**». Стиль применяется к строке таблицы (№22).

```

22 <tr class="kha_double">
23   <td>Повтор</td>
24   <td>kha_double</td>
25 </tr>

```

Повтор	kha_double
--------	------------

Стиль для неопределенного статуса «Повтор»

В №26-29 записаны параметры выхода записей, у которых поле «status» не определено. Для отображения таких записей используется стиль (**class**) «**kha_null**». Стиль применяется к строке таблицы (№26).

```

26 <tr class="kha_null">
27   <td>NULL</td>
28   <td>kha_null</td>
29 </tr>

```

NULL	kha_null
------	----------

Стиль для любого другого статуса

В №30-33 определены параметры выхода записей, у которых поле «status» имеет какое-то другое значение (отличается от значения, описанных выше). Для отображения таких записей используется стиль (**class**) «**kha_other**». Стиль применяется к строке таблицы (№30).

```

30 <tr class="kha_other">
31   <td>Other</td>
32   <td>kha_other</td>
33 </tr>

```

Other	kha_other
-------	-----------

В №34 закрывается таблица (начало в №9). №35-36 завершают файл html.

```

34 </table>
35 </body>
36 </html>

```

Шаблон Javascript

В этом шаблоне мы храним переменные и функции Javascript, которые используются в других шаблонах.

Moodle добавляет файл шаблона Javascript к любой своей странице, которая работает с базой (просмотр списком, просмотр одиночной записи, ввод и редактирование записи).

В исходном коде страницы шаблон Javascript подгружается так.

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="ru" xml:lang="ru">
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5   <link rel="stylesheet" type="text/css" href="http://cdo.kname.edu.ua/theme/standard/styles.php" />
6   <link rel="stylesheet" type="text/css" href="http://cdo.kname.edu.ua/theme/formal_white/styles.php" />
7   <link rel="stylesheet" type="text/css" href="http://cdo.kname.edu.ua/mod/data/css.php?d=738" /> <script
   type="text/javascript" src="http://cdo.kname.edu.ua/mod/data/js.php?d=738"></script>

```

Полный текст шаблона Javascript

Ниже приведен полный текст шаблона Javascript (файл **kha_pattern.js**).

```
1 var
2 i,n,s,s1;
3 var
4 kha_RecTotal= 0,
5 kha_RecKnameTotal = 0;
6 var
7 kha_StatNum = 0,
8 kha_StatArr = [], kha_StatRecArr = [];
9 var
10 kha_StatOtherClass = ' class="kha_other"',
11 kha_StatClass = [], kha_StatTypes = [];
12
13 kha_StatTypes[0]= 'Одобрено';
14   kha_StatClass[0]= '';
15 kha_StatTypes[1]= 'Исправить';
16   kha_StatClass[1]= ' class="kha_redo"';
17 kha_StatTypes[2]= 'Не одобрено';
18   kha_StatClass[2]= ' class="kha_no"';
19 kha_StatTypes[3]= 'Повтор';
20   kha_StatClass[3]= ' class="kha_double"';
21 kha_StatTypes[4]= 'NULL';
22   kha_StatClass[4]= ' class="kha_null"';
23 kha_StatTypes[5]= 'Other';
24   kha_StatClass[5]= ' class="kha_other"';
25
26
27 function kha_WriteStatItem(cur_s){
28 var pri = 0;
29   for(var i=0; i<kha_StatNum; i++)
30     if(kha_StatArr[i] == cur_s){
31       kha_StatRecArr[i]++;
32       pri=1;
33       break;
34     }
35   if(pri == 0){
36     kha_StatArr[kha_StatNum] = cur_s;
37     kha_StatRecArr[kha_StatNum]=1;
38     kha_StatNum++;
39   }
```

```

32     }
33 }
34
35 function kha_GetStatClass(cur_s) {
36 var_sclass = kha_StatOtherClass;
37     for(var i=0; i<kha_StatTypes.length; i++){
38         if(cur_s == kha_StatTypes[i]){
39             sclass = kha_StatClass[i];
40             break;
41         }
42     }
43     return sclass;
44 }

```

Окно редактирования шаблона Javascript

Этот файл полностью записывается в окно редактирования шаблона.

Шаблон Javascript

```

var
i,n,s,s1;
var
kha_RecTotal= 0,
kha_RecKnameTotal = 0;
var
kha_StatNum = 0,
kha_StatArr = [], kha_StatRecArr = [];
var
kha_StatOtherClass = ' class="kha_other"',
kha_StatClass = [], kha_StatTypes = [];

kha_StatTypes[0]= 'Одобрено';    kha_StatClass[0]= '';
kha_StatTypes[1]= 'Исправить';  kha_StatClass[1]= ' class="kha_redo"';
kha_StatTypes[2]= 'Не одобрено'; kha_StatClass[2]= ' class="kha_no"';
kha_StatTypes[3]= 'Повтор';    kha_StatClass[3]= '
class="kha_double"';
kha_StatTypes[4]= 'NULL';      kha_StatClass[4]= ' class="kha_null"';
kha_StatTypes[5]= 'Other';     kha_StatClass[5]= ' class="kha_other"';

function kha_WriteStatItem(cur_s) {

```

Переменные

Рассмотрим теперь шаблон Javascript подробно.

В №1-2 описываются служебные переменные. В JavaScript можно эти

переменные не описывать, но мы привыкли описывать все переменные, которые используем.

```
1 var  
2 i, n, s, s1;
```

Далее в шаблоне приведены глобальные переменные и функции. В шаблоне Javascript они только описываются, а используются в других шаблонах (в данном случае в шаблоне «просмотра списком»).

В №4 описана переменная **kha_RecTotal**, которая используется для подсчета количество записей в базе данных.

В №5 описана переменная **kha_RecKnameTotal**, она используется для подсчета количество записей в базе данных, в которых записана информация о 3-мерных моделях, сделанных в ХНАГХ.

Напомним, что для предотвращения конфликтов и обеспечения уникальности имен (в данном случае переменных и функций) все имена начинаются с строки «**kha_**».

```
3 var  
4 kha_RecTotal= 0,  
5 kha_RecKnameTotal = 0;
```

В № 7 описана переменная **kha_StatNum**, которая используется для определения числа различных статусов записей в базе данных.

В №8 описаны массивы **kha_StatArr** и **kha_StatRecArr**.

В массив **kha_StatArr** записываются значения поля «status», а в массив **kha_StatRecArr** количества записей с конкретным значением поля «status».

Подробно работу с этими (и другими) глобальными переменными мы рассмотрим в разделе «Шаблон просмотра списком».

```
6 var  
7 kha_StatNum = 0,  
8 kha_StatArr = [], kha_StatRecArr = [];
```

В №10 описана переменная, в которую записано название стиля (**class="kha_other"**) для записей с «нестандартным» полем «status».

```
9 var  
10 kha_StatOtherClass = ' class="kha_other" ',
```

В №11 описаны массивы **kha_StatClass** и **kha_StatTypes**.

```
11 kha_StatClass = [], kha_StatTypes = [];
```

```
12
```

В №13-18 элементам массива присваиваются конкретные значения. Записи, у которых значение поля «status» равно значению элемента массива **kha_StatTypes [i]**, выводятся с использованием стиля, записанного в элементе массива **kha_StatClass[i]** (переменная **i** может принимать значения от 1 до 6).

```
13 kha_StatTypes[0]= 'Одобрено';
```

```
    kha_StatClass[0]= '';
```

```
14 kha_StatTypes[1]= 'Исправить';
```

```
    kha_StatClass[1]= ' class="kha_redo"';
```

```
15 kha_StatTypes[2]= 'Не одобрено';
```

```
    kha_StatClass[2]= ' class="kha_no"';
```

```
16 kha_StatTypes[3]= 'Повтор';
```

```
    kha_StatClass[3]= ' class="kha_double"';
```

```
17 kha_StatTypes[4]= 'NULL';
```

```
    kha_StatClass[4]= ' class="kha_null"';
```

```
18 kha_StatTypes[5]= 'Other';
```

```
    kha_StatClass[5]= ' class="kha_other"';
```

```
19
```

Функции

В №20-33 описана функция **kha_WriteStatItem**.

Эта функция принимает в качестве параметра значение поля «status» и проверяет, записано ли это значение в массиве **kha_StatArr**.

Если да, то значение соответствующего элемента массива **kha_StatRecArr** (количество записей с этим значением поля «status») увеличивается на 1. В противном случае значение поля «status» записывается в новый элемент массива **kha_StatArr**, а соответствующему элементу массива **kha_StatRecArr** присваивается значение 1.

```
20 function kha_WriteStatItem(cur_s) {
```

В №21 описывается локальная переменная **pri**, которая используется для хранения результатов проверки «записано ли уже значение параметра в массив **kha_StatArr**». По умолчанию (№21) переменной **pri** присваивается значение 0 (значение параметра еще не записано в массив), если значение поля «status» будет найдено в массиве **kha_StatArr**, то переменной **pri** будет

присвоено значение **1**.

```
21 var pri = 0;
```

В цикле (№22-27) проверяется, записано ли значение поля «status» в массив **kha_StatArr** (№23).

Если да, то значение соответствующего элемента массива **kha_StatRecArr** увеличивается на **1** (№24). Переменной **pri** присваивается значение **1** (признак того, что значение поля «status» уже записано в массив **kha_StatArr**). Происходит выход из цикла (№26), а затем и из функции.

```
22 for(var i=0; i<kha_StatNum; i++)
23     if(kha_StatArr[i] == cur_s){
24         kha_StatRecArr[i]++;
25         pri=1;
26         break;
27     }
```

Если значение поля «status» еще не записано в массив **kha_StatArr**, то значение параметра записывается в новый элемент массива (№29), соответствующему элементу массива **kha_StatRecArr** (количество записей с этим значением поля «status») присваивается значение **1** (№30).

После этого происходит выход из функции.

```
28     if(pri == 0){
29         kha_StatArr[kha_StatNum] = cur_s;
30         kha_StatRecArr[kha_StatNum]=1;
31         kha_StatNum++;
32     }
33 }
34
```

В №35-44 описана функция **kha_GetStatClass**.

Эта функция принимает в качестве параметра значение поля «status» и возвращает соответствующий ему стиль вывода информации.

```
35 function kha_GetStatClass(cur_s) {
```

В №36 описана локальная переменная **sclass**, в которую записывается стиль (**class**) вывода информации. По умолчанию переменной присваивается значение **kha_StatOtherClass** (' class="kha_other"').

```
36 var_sclass = kha_StatOtherClass;
```

В цикле (№37-41) проверяется, есть ли значение поля «status» в массиве **kha_StatTypes** . Если да (№38), то переменной **sclass** присваивается значение стиль из соответствующего элемента массива **kha_StatClass** (№39) и происходит выход из цикла (№40). Если значение поля «status» в массиве **kha_StatTypes** не найдено, то переменная **sclass** не изменяется.

```
37 for(var i=0; i<kha_StatTypes.length; i++){
38     if(cur_s == kha_StatTypes[i]){
39         sclass = kha_StatClass[i];
40         break;
41     }
42 }
```

В №43-44 функция возвращает значение (**sclass**) и завершает свою работу.

```
43 return sclass;
44 }
```

Шаблон списка

Шаблон списка используется для просмотра нескольких записей одновременно. В нашей базе данных с помощью этого шаблона формируется таблица, каждая строка этой таблицы – запись базы данных.

Мы будем использовать стили из шаблона CSS (каскадные таблицы стилей) и функции из шаблона Javascript для того, чтобы «раскрашивать» строки таблицы. Записи, имеющие разный статус (одобрено, не одобрено, исправить, повтор) будут иметь различный цвет фона и текста.

Полный текст html-файла

Ниже приведен полный текст шаблона списка (файл **kha_list_rec.html**).

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
```

```
3 <head>
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
5 <title>Просмотр списка</title>
6 <link href="kha_pattern.css" rel="stylesheet"
type="text/css" />
7 <script language="JavaScript" type="text/javascript"
src="kha_pattern.js"></script>
8 </head>
9 <body>
10 <!------- Заголовок ----->
11 <table cellspacing="5" cellpadding="2" border="1">
12   <tr>
13     <th width="110" valign="top">Изображение
14   </th>
15     <th width="100" valign="top">Название
16   </th>
17     <th width="80" valign="top">Коорд.
18   </th>
19     <th width="300" valign="top">URL модели
20   </th>
21     <th width="100" valign="top">Автор модели
22   </th>
23     <th width="100" valign="top">Дата внесения записи
24   </th>
25     <th width="100" valign="top">Автор записи
26   </th>
27     <th width="50" valign="top">Зап.<br />одобр.
28   </th>
29 </tr>
30 <!------- Повторяющиеся записи ----->
31 <script type="text/javascript">//<![CDATA[
32   kha_RecTotal++;
33   s1 = '[[is_kname]]';
34   if(s1 == 'ХНАГХ') kha_RecKnameTotal++;
35   s = '[[status]]';
36   if(s == 'Да') s = 'Одобрено';
37   if(s == 'Нет') s = 'Не одобрено';
```

```

38     if(!s) s = 'NULL';
39     kha_WriteStatItem(s);
40     document.write('<tr'+kha_GetStatClass(s)+'>');
41 //]]></script>
42     <td width="110" valign="top">[[Изображение]]
43     </td>
44     <td width="100" valign="top">[[Название]]<br>
45     <font
color="#CC9900"><b>[[is_kname]]</b></font></td>
46     <td width="80" valign="top">[[Координаты]]<br
/>[[kml]]
47     </td>
48     <td width="300" valign="top">[[URL_модель]]
49     </td>
50     <td width="100" valign="top">[[Автормодели]]
51     </td>
52     <td width="100" valign="top">##timeadded##
53     </td>
54     <td width="100" valign="top">##user##
55     </td>
56     <td width="50" valign="top">[[status]]<br
/>##more####edit####delete##
57     </td>
58 </tr>
59 <!------- Нижний колонтитул ----->
60 </table>
61 <h1>Статистика</h1>
62 <script type="text/javascript">//<![CDATA[
63     document.write('Всего записей -
<b>'+kha_RecTotal+'</b>');
64     document.write('<font color="#CC9900">Объекты,
сделанные в ХНАГХ - <b>'+ kha_RecKnameTotal +
'</b></font></b>');
65 //]]></script>
66 <b><table cellspacing="5" cellpadding="2" border="1">
67 <tr><td align="center">Всего записей</td>
68 <script type="text/javascript">//<![CDATA[
69     n=0;
```

```

70   for(i=0; i<kha_StatArr.length; i++){
71       n += kha_StatRecArr[i];
72       s = kha_StatArr[i];
73       s1 = '<td align="center"' + kha_GetStatClass(s) +
'>';
74       document.write(s1+s+'</td>');
75   }
76   document.write('</tr><tr>');
77   document.write('<td align="center">'+n+'</td>');
78   for(i=0; i<kha_StatArr.length; i++){
79       n = kha_StatRecArr[i];
80       s = kha_StatArr[i];
81       s1 = '<td align="center"' + kha_GetStatClass(s) +
'>';
82       document.write(s1+n+'</td>');
83   }
84   //]]></script>
85 </tr></table></b>
86 </body>
87 </html>

```

В браузере этот шаблон будет выглядеть так.

Изображение	Название	Коорд.	URL модели	Автор модели	Дата внесения записи	Автор записи	Зап. одобр.
[[Изображение]]	[[Название]] [[is_kname]]	[[Координаты]] [[kml]]	[[URL_модель]]	[[Автормодели]]	##timeadded##	##user##	[[status]] ##more####edit####delete##

Статистика

Всего записей - 1. Объекты, сделанные в ХНАГХ - 0.

Всего записей	[[status]]
1	1

Окна редактирования шаблона списка

У шаблона списка есть три окна редактирования.

Доступные теги (?)

Поля

- comment - [[comment]]
- status - [[status]]
- Автормодели - [[Автормодели]]
- kml - [[kml]]
- URL_модель - [[URL_модель]]
- Адрес - [[Адрес]]
- Изображение - [[Изображение]]
- Координаты - [[Координаты]]
- Название - [[Название]]
- Описание - [[Описание]]
- is_kname - [[is_kname]]

Заголовок

```
<table cellpadding="5" cellspacing="2" border="1">
<tr>
<th width="110" valign="top">Изображение
</th>
<th width="100" valign="top">Название
</th>
<th width="80" valign="top">Коорд.
```

Повторяющиеся записи

```
<script type="text/javascript">/*!CDATA[
kha_RecTotal++;
s1 = '[[is_kname]]';
if(s1 == 'ХНАГХ') kha_RecKnameTotal++;
s = '[[status]]';
if(s == 'Да') s = 'Одобрено';
if(s == 'Нет') s = 'Не одобрено';
if(!s) s = 'NULL';
kha_WriteStatItem(s);
document.write('<tr'+kha_GetStatClass(s)+'>');
//]></script>
<td width="110" valign="top">[[Изображение]]
</td>
<td width="100" valign="top">[[Название]]<br>
<font color="#CC9900"><b>[[is_kname]]</b></font></td>
<td width="80" valign="top">[[Координаты]]<br />[[kml]]
</td>
<td width="300" valign="top">[[URL_модель]]
```

Нижний колонтитул

```
</table>
<h1>Статистика</h1>
<script type="text/javascript">/*!CDATA[
document.write('Всего записей - <b>'+kha_RecTotal+'</b>');
document.write('<font color="#CC9900">Объекты, сделанные в ХНАГХ -
<b>'+ kha_RecKnameTotal +'</b></font></b> ');
```

Сбросить шаблон

Включить редактор

Сохранить шаблон

1. **Заголовок** – в это окно записываются строки №11-29 из файла **kha_list_rec.html**.

2. **Повторяющиеся записи** – в это окно записываются строки №31-58 из файла **kha_list_rec.html**.

3. **Нижний колонтитул** – в это окно записываются строки №60-85 из файла **kha_list_rec.html**.

Стандартный редактор Moodle






В нашей базе шаблон списка – самый сложный элемент. При его разработке были решены проблемы, связанные с убогим описанием базы в документации.

Первая проблема – заключается в том, что стандартный редактор Moodle все время пытается выводить каждую запись из списка в виде отдельной таблицы, а не в виде строки одной таблицы. Это приводит к тому, что ячейки в каждом столбце имеют разный размер.




В нашем шаблоне в заголовок мы вынесли тег **<table>**, и строку названий полей. В окне «повторяющиеся записи» мы описали строку таблицы для записей. В нижнем колонтитуле у нас находится тег **</table>**.

Заметим, что скрипты на JavaScript мы внесли в шаблон позднее.

Результат нас полностью удовлетворил (на тот момент).

Изображение	Название	Коорд.	URL модели	Автор модели	Дата внесения записи	Автор записи	Зап. одобр.
	nine - storey apartment block house	50.0061°N, 36.2141°E ? метка	http://sketchup.google.com/3dwarehouse/details?mid=160528ca1bd056b38f756b1b56816013	Руслан Че	Пятница 2 Декабрь 2011, 14:55	Бичикашвили Элиза	Да ? ✕
	Звезда	49.9627°N, 36.3337°E ? метка	http://sketchup.google.com/3dwarehouse/details?mid=1854a327bb669fad68a636ed1513c019	Александр Юрчак	Пятница 8 Декабрь 2011, 23:23	Махова Анастасия	Да ? ✕
	Харьковская общеобразовательная школа №5 ХНАГХ	50.0036°N, 36.2401°E ? Метка	http://sketchup.google.com/3dwarehouse/details?mid=1a74cea66110505abd1d209f884da75a	Eva Morkovskaya	Понедельник 10 Декабрь 2012, 18:35	Лычман Владислав	Да ? ✕
	Харьковская ТЭЦ-5	49.9693°N, 36.1078°E ? метка	http://sketchup.google.com/3dwarehouse/details?mid=1aacd33d75d632c863995fb119e59971	Александр Чигиринский	Понедельник 21 Ноябрь 2011, 18:01	Балабанова Юлия	Да ? ✕
	Супермаркет АТБ	49.9531°N, 36.2590°E ? Метка	http://sketchup.google.com/3dwarehouse/details?mid=1bfba7b99e4c091c35f766a35d918a0	Fandango	Понедельник 28 Ноябрь 2011, 20:33	Балабанова Юлия	Да ? ✕

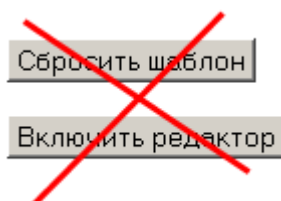
При создании шаблона мы не пользовались стандартным редактором Moodle. Однако при попытке безобидного исправления «Коорд.» на «Координаты» (при включенном редакторе) получилось следующее.

Изображение	Название	Координаты	URL модели	Автор модели	Дата внесения записи	Автор записи	Зап. одобр.
	nine - storey apartment block house	50.0061°N, 36.2141°E ? метка	http://sketchup.google.com/3dwarehouse/details?mid=160528ca1bd056b38f756b1b56816013	Руслан Че	Пятница 2 Декабрь 2011, 14:55	Бичикашвили Элиза	Да
	Звезда	49.9627°N, 36.3337°E ? метка	http://sketchup.google.com/3dwarehouse/details?mid=1854a327bb669fad68a636ed1513c019	Александр Юрчак	Пятница 8 Декабрь 2011, 23:23	Махова Анастасия	Да
	Харьковская общеобразовательная школа №5 ХНАГХ	50.0036°N, 36.2401°E ? Метка	http://sketchup.google.com/3dwarehouse/details?mid=1a74cea66110505abd1d209f884da75a	Eva Morkovskaya	Понедельник 10 Декабрь 2012, 18:35	Лычман Владислав	Да

Редактор «исправил» все наши «ошибки» со всей решительностью и неуклюжестью, которые вообще свойственны программной реализации системы Moodle.

Это – вторая принципиальная проблема с базой данных Moodle.

У этой проблемы одно решение – **не использовать стандартный редактор!**



Рассмотрим теперь шаблон списка подробно.

Начало файла html

№1-9 – стандартной начало файла html. В №6 загружается шаблон CSS, а в №7 – шаблон Javascript.

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
5 <title>Просмотр списка</title>
6 <link href="kha_pattern.css" rel="stylesheet"
type="text/css" />
7 <script language="JavaScript" type="text/javascript"
src="kha_pattern.js"></script>
8 </head>
9 <body>

```

Заголовок шаблона

В №10-29 выводится заголовок таблицы. В html – файле шаблона это выглядит так:

Изображение	Название	Коорд.	URL модели	Автор модели	Дата внесения записи	Автор записи	Зап. одобр.
-------------	----------	--------	------------	--------------	----------------------	--------------	-------------

а в Moodle так:

Изображение	Название	Коорд.	URL модели	Автор модели	Дата внесения записи	Автор записи	Зап. одобр.
-------------	----------	--------	------------	--------------	----------------------	--------------	-------------

В дальнейшем фрагменты отображения html – файла в браузере мы будем обозначать «html», а фрагменты окна Moodle – «Moodle».

```

10 <!------- Заголовок ----->

```

№11 начинает описание таблицы в которую будет выводиться информация. В №8 открывается строка заголовка таблицы.

11 `<table cellspacing="5" cellpadding="2" border="1">`

12 `<tr>`

В №13-14 выводится ячейка «Изображение».

13 `<th width="110" valign="top">Изображение`

14 `</th>`

Html	Moodle
Изображение	Изображение

В №15-16 выводится ячейка «Название».

15 `<th width="100" valign="top">Название`

16 `</th>`

Html	Moodle
Название	Название

В №17-18 выводится ячейка «Коорд.».

17 `<th width="80" valign="top">Коорд.`

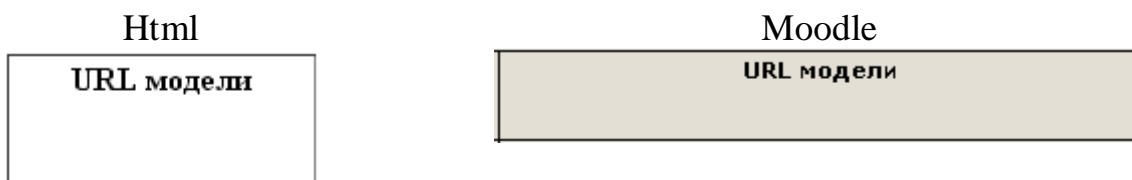
18 `</th>`

Html	Moodle
Коорд.	Коорд.

В №19-20 выводится ячейка «URL модели».

19 `<th width="300" valign="top">URL модели`

20 `</th>`



В №21-22 выводится ячейка «**Автор модели**».

21 `<th width="100" valign="top">Автор модели`
 22 `</th>`



В №23-24 выводится ячейка «**Дата внесения записи**».

23 `<th width="100" valign="top">Дата внесения записи`
 24 `</th>`



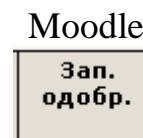
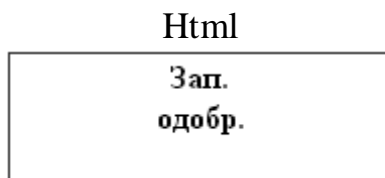
В №25-26 выводится ячейка «**Автор записи**».

25 `<th width="100" valign="top">Автор записи`
 26 `</th>`



В №27-28 выводится ячейка «**Зап. одобр.**».

27 `<th width="50" valign="top">Зап.
одобр.`
 28 `</th>`



В №29 строка заголовка закрывается.

29 `</tr>`



Повторяющиеся записи

В №31-58 выводятся повторяющиеся записи.

В файле html строка повторяющихся записей выглядит так:

[[Изображение]]	[[Название]] [[is_kname]]	[[Координаты]] [[kml]]	[[URL_модель]]	[[Автормодели]]	##timeadded##	##user##	[[status]] ##more####edit####delete##
-----------------	------------------------------	---------------------------	----------------	-----------------	---------------	----------	--

а в странице вывода Moodle – так (для записи «Харьковская кенасса»):

	Харьковская кенасса ХНАГХ	49.0000°N, 36.0000°E метка	http://sketchup.google.com/3dwarehouse/details?mid=2fddc091484947a9720c9008a053603	Eva Morkovskaya	Воскресенье 9 Декабрь 2012, 00:03	Герасименко Анастасия	Исправить 
---	------------------------------	-------------------------------	---	-----------------	---	--------------------------	--

30 `<!----- Повторяющиеся записи ----->`

Обработка статуса записи

В №31-41 представлен фрагмент программы (скрипта) на языке JavaScript.

№31 объявляет браузеру, что в текст html – страницы внедрен скрипт (№41 показывает конец скрипта).

Конструкция

```
//<![CDATA[ ---
Фрагмент программы
//]]>
```

защищает скрипт от чересчур ретивых валидаторов (систем проверки и интерпретации исходного кода). Судя по первой исходного кода страницы просмотра списка

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Moodle использует XHTML и поэтому может неправильно интерпретировать код скрипта.

```
31 <script type="text/javascript">//<![CDATA[
```

В №32 значение переменной **kha_RecTotal** увеличивается на **1**. В этой переменной хранится общее число записей.

```
32 kha_RecTotal++;
```

В №33 локальную переменную **s1** записывается значение поля **is_kname**.

```
33 s1 = '[[is_kname]]';
```

Если в значение поля **is_kname** равно **'ХНАГХ'**, то значение переменной **kha_RecKnameTotal** увеличивается на **1**. В этой переменной хранится число моделей, сделанных в ХНАГХ.

```
34 if(s1 == 'ХНАГХ') kha_RecKnameTotal++;
```

В №35 локально переменной **s** присваивается значение поля **status**.

```
35 s = '[[status]]';
```

В №36-37 исправляется неточность в проектировании базы. Если значение переменной **s** равно **'Да'**, то этой переменной присваивается значение **'Одобрено'**. Если значение переменной **s** равно **'Нет'**, то этой переменной присваивается значение **'Не одобрено'**. Естественно, при этом значение поля **status** не изменяется.

```
36 if(s == 'Да') s = 'Одобрено';
```

```
37 if(s == 'Нет') s = 'Не одобрено';
```

В № 38 обрабатывается ситуация, когда поле **status** не определено. В этом случае переменной **s** присваивается значение **'NULL'**.

```
38 if(!s) s = 'NULL';
```

В №39 вызывается функция **kha_WriteStatItem**.

Эта функция принимает в качестве параметра значение поля «status» и проверяет, записано ли это значение в массиве **kha_StatArr**.

Если да, то значение соответствующего элемента массива **kha_StatRecArr** (количество записей с этим значением поля «status»)

увеличивается на 1. В противном случае значение поля «status» записывается в новый элемент массива **kha_StatArr**, а соответствующему элементу массива **kha_StatRecArr** присваивается значение 1. Эти массивы мы будем использовать при формировании нижнего колонтитула.

```
39 kha_WriteStatItem(s) ;
```

В №40 в страницу выводится тег html **<tr...>**. Опция тега определяется функцией **kha_GetStatClass**. Эта функция принимает в качестве параметра значение поля «status» и возвращает соответствующий ему стиль вывода информации.

```
40 document.write('<tr'+kha_GetStatClass(s)+'>');
```

Очевидно, что в html файле шаблона не происходит присваивание значений переменным базы данных, поэтому переменной **s** присваивается непосредственно значение **'[[status]]'** (№37). Этого значения нет среди элементов массива **kha_StatTypes**, поэтому функция **kha_GetStatClass** возвращает значение **kha_StatOtherClass(' class="kha_other"')**.

В этом случае №40 выводит следующую строку:

```
<tr. class="kha_other">
```

Поэтому в браузере мы видим серую строчку, т.к. в стиле **kha_other** определен серый цвет фона:

```
.kha_other {
    background-color: #CCCCCC;
}
```

[[Изображение]]	[[Название]]	[[Координаты]]	[[URL_модель]]	[[Автормодели]]	##timeadded##	##user##	[[status]]
	[[is_kname]]	[[kml]]					##more####edit####delete##



Рассмотрим теперь вывод информации системой Moodle (на примере записи «Харьковская кенасса»).

Переменной **s** присваивается значение **'Исправить'** (№37). Это значение есть среди элементов массива **kha_StatTypes**, определенных в шаблоне Javascript (**kha_StatTypes[1] = 'Исправить'**), поэтому функция **kha_GetStatClass** возвращает значение элемента 1 массива **kha_StatClass**:

```
kha_StatClass[1]= ' class="kha_redo"'
```

В браузере мы видим розовую строчку, т.к. в стиле `kha_redo` определен розовый цвет фона:

```
.kha_redo {
    background-color: #FFCCCC;
}
```

	Харьковская кенасса ХНАГХ	49.0000°N, 36.0000°E метка	http://sketchup.google.com/3dwarehouse/details?mid=2ffdc091484947a9720c9008a053603	Eva Morkovskaya	Воскресенье 9 Декабрь 2012, 00:03	Герасименко Анастасия	Исправить 
---	------------------------------	-------------------------------	---	-----------------	---	--------------------------	--

№41 завершает фрагмент программы.

```
41 //]]></script>
```

Вывод полей списка

В №42-43 выводится значение поле «Изображение».

```
42 <td width="110" valign="top">[[Изображение]]
43 </td>
```

Html

```
[[Изображение]]
```

Moodle

(для записи «Харьковская кенасса»)



В №44-45 выводятся значения полей «Название» и «is_kname».

```
44 <td width="100" valign="top">[[Название]]<br>
45 <font color="#CC9900"> <b>[[is_kname]]</b>
</font> </td>
```

Html

```
[[Название]]
[[is_kname]]
```

Moodle

(для записи «Харьковская кенасса»)

```
Харьковская кенасса
ХНАГХ
```

В №46-47 выводятся значения полей «Координаты» и «kml».

```
46 <td width="80" valign="top">[[Координаты]]<br />
```

```
[[kml]]
```

```
47 </td>
```

Html

```
[[Координаты]]
[[kml]]
```

Moodle

(для записи «Харьковская кенасса»)

```
49.0000°N, 36.0000°E
? метка
```

В №48-49 выводится значение поля «URL_модель».

```
48 <td width="300" valign="top">[[URL_модель]]
```

```
49 </td>
```

Html

```
[[URL_модель]]
```

Moodle

(для записи «Харьковская кенасса»)

```
http://sketchup.google.com/3dwarehouse
/details?mid=2ffdc091484947a9720c9008a053603
```

В №50-51 выводится значение поля «Автормодели».

```
50 <td width="100" valign="top">[[Автормодели]]
```

```
51 </td>
```

Html

```
[[Автормодели]]
```

Moodle

(для записи «Харьковская кенасса»)

```
Eva Morkovskaya
```

В №52-53 выводится значение специального поля «##timeadded##» (время занесения записи).

```
52 <td width="100" valign="top">##timeadded##
```

```
53 </td>
```

Html

```
##timeadded##
```

Moodle

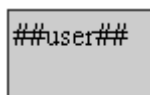
(для записи «Харьковская кенасса»)

```
Воскресенье
9 Декабрь
2012, 00:03
```

В №54-55 выводится значение специального поля «**##user##**» (автор записи – пользователь, который ввел запись).

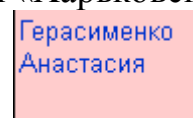
```
54 <td width="100" valign="top">##user##
55 </td>
```

Html



Moodle

(для записи «Харьковская кенасса»)



В №53-54 выводится значение поля «**status**» и специальные поля (теги):

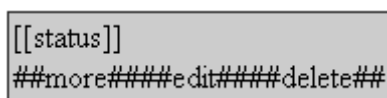
«**##more##**» – иконка просмотра одной записи,

«**##edit##**» – иконка редактирования записи,

«**##delete##**» – иконка удаления записи,

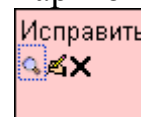
```
56 <td width="50" valign="top">[[status]]<br
/>##more####edit####delete##
57 </td>
```

Html



Moodle

(для записи «Харьковская кенасса»)



№58 завершает строку таблицы, начальный тег записан в html – документ с помощью скрипта JavaScript (№40).

```
58 </tr>
```

Нижний колонтитул

В №60-85 выводится нижний колонтитул.

В файле html строка нижний колонтитул выглядит так:

Статистика

Всего записей - 1. Объекты, сделанные в ХНАГХ - 0.

Всего записей	[[status]]
1	1

а в странице вывода Moodle – так:

Статистика				
Всего записей - 109. Объекты, сделанные в ХНАГХ - 49.				
Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

59 <!----- Нижний колонтитул ----->

№ 60 закрывает таблицу, в которую выводятся заголовок шаблона и повторяющиеся записи (№11 – начало).

60 </table>

Вывод статистической информации.

В №61 заголовок статистики

61 <h1>Статистика</h1>

Html
Статистика

Moodle

Статистика

В №62-65 представлен фрагмент программы (скрипта) на языке JavaScript, который выводит общее количество записей в базе данных и количество моделей, сделанных в ХНАГХ.

62 <script type="text/javascript">/*! [CDATA[

В №63 выводится количество общее количество записей в базе данных.

63 document.write('Всего записей -
'+kha_RecTotal+'.');

Html
 Всего записей - 1.

Moodle
 Всего записей - 109.

В №64 выводится количество моделей, сделанных в ХНАГХ.

```
64 document.write('<font color="#CC9900">Объекты,
сделанные в ХНАГХ - <b>'+ kha_ResKnameTotal +
'</b></font></b>');
```

Html
 Объекты, сделанные в ХНАГХ - 0

Moodle
 Объекты, сделанные в ХНАГХ - 49

В №65 завершается фрагмент программы (начало в №52).

```
65 //]]></script>
```

В №66 начинается таблица статистической информации.

```
66 <b><table cellpadding="5" cellspacing="2" border="1">
```

В №67 начинается первая ячейка таблицы статистической информации.

```
67 <tr><td align="center">Всего записей</td>
```

Html
 Всего записей

Moodle
 Всего записей

В №68-84 представлен фрагмент программы (скрипта) на языке JavaScript, который выводит остальную часть таблицы статистической информации.

```
68 <script type="text/javascript">/*! [CDATA[
```

Заголовок таблицы статистической информации

Заголовок таблицы статистической информации выгладит на экране так.

Html
 Всего записей [[status]]

Moodle
 Всего записей | Одобрено | NULL | Исправить | Повтор

В №69-75 выводится оставшаяся часть заголовка таблицы статистической

информации. Кроме того осуществляется альтернативный способ подсчета количества записей (как суммы количеств записей с одинаковым статусом).

Значение этой величины (количество записей) будет накапливаться в переменной **n** (№69).

```
69    n=0;
```

В цикле (№70-75) перебираются все элементы массивов **kha_StatArr** (все значения поля «статус», которые встречаются в базе данных) и массива **kha_StatRecArr** (количества записей для каждого значения поля «статус»).

kha_StatArr.length – количество элементов этих массивов (одно и тоже для каждого массива).

```
70    for(i=0; i<kha_StatArr.length; i++){
71        n += kha_StatRecArr[i];
```

В №72 переменной **s** присваивается значение текущего элемента массива **kha_StatArr**.

```
72        s = kha_StatArr[i];
```

В №73 переменной **s1** присваивается строка, в которой записано начало очередной ячейки заголовка (тег `<td...>`).

```
73        s1 = '<td align="center"' + kha_GetStatClass(s) +
'>';
```

В №74 выводится ячейка заголовка.

```
74        document.write(s1+s+'</td>');
75    }
```

Рассмотрим подробно работу этого алгоритма отдельно для `html` – файла и страницы Moodle.

Для `html` – файла у нас массив **kha_StatArr** имеет только 1 элемент

```
kha_StatArr[0] = '[[status]]'
```

Массив **kha_StatRecArr** тоже имеет только 1 элемент.

```
kha_StatRecArr[0] = '1'
```

Длина ЭТИХ массивов `kha_StatArr.length = 1`

На 1-м шаге цикла

```
70    i=0
71    n += kha_StatRecArr[0] = 0+1 = 1
72    s = kha_StatArr[0] = '[[status]]'
```

Функция `kha_GetStatClass('[[status]]')` возвращает значение `' class="kha_other"'`.

```
73    s1 = '<td align="center" class="kha_other">';
74    document.write(
'<td align="center" class="kha_other">[[status]]</td>');
```

Всего записей	[[status]]
1	1

На следующем шаге цикла `i=1`, условие `i < kha_StatArr.length` не выполняется, цикл завершает свою работу.

При реальном выводе информации в страницу Moodle массив `kha_StatArr` имеет 4 элемента

```
kha_StatArr[0] = 'Одобрено'
kha_StatArr[1] = 'NULL'
kha_StatArr[2] = 'Исправить'
kha_StatArr[3] = 'Повтор'
```

Массив `kha_StatRecArr` тоже имеет 4 элемента.

```
kha_StatRecArr[0] = '91'
kha_StatRecArr[1] = '11'
kha_StatRecArr[2] = '5'
kha_StatRecArr[3] = '2'
```

Длина ЭТИХ массивов `kha_StatArr.length = 4`

На 1-м шаге цикла

```

70  i=0
71  n += kha_StatRecArr[0] = 0+91 = 91
72  s = kha_StatArr[0] = 'Одобрено'

```

Функция `kha_GetStatClass('Одобрено')` возвращает значение '' (пустая строка).

```

73  s1 = '<td align="center">';
74  document.write(
'<td align="center">Одобрено</td>');

```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На 2-м шаге цикла

```

70  i=1
71  n += kha_StatRecArr[1] = 91+11 = 102
72  s = kha_StatArr[1] = 'NULL'

```

Функция `kha_GetStatClass('NULL')` возвращает значение ' class="kha_null" '.

```

73  s1 = '<td align="center" class="kha_null">';
74  document.write(
'<td align="center" class="kha_null">NULL</td>');

```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На 3-м шаге цикла

```

70  i=2
71  n += kha_StatRecArr[2] = 102+5 = 107
72  s = kha_StatArr[2] = 'Исправить'

```

Функция `kha_GetStatClass('Исправить')` возвращает значение ' class="kha_redo" '.

```

73  s1 = '<td align="center" class="kha_redo">';

```

```
74 document.write(
'<td align="center" class="kha_redo">Исправить</td>');
```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На 4-м шаге цикла

```
70 i=3
71 n += kha_StatRecArr[3] = 107+2 = 109
72 s = kha_StatArr[3] = 'Повтор'
```

Функция `kha_GetStatClass('Повтор')` возвращает значение `' class="kha_double"'`.

```
73 s1 = '<td align="center" class=" kha_double ">';
74 document.write(
'<td align="center" class=" kha_double "> Повтор </td>');
```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На следующем шаге цикла `i=4`, условие `i<kha_StatArr.length` не выполняется, цикл завершает свою работу.

Числовая строка таблицы статистической информации

Числовая строка таблицы статистической информации выгладит на экране так.

Html		Moodle				
1	1	109	91	11	5	2

В №76-77 выводится первая ячейка числовой строки (общее количество записей).

```
76 document.write('</tr><tr>');
77 document.write('<td align="center">'+n+'</td>');
```

Html		Moodle				
Всего записей	[[status]]	Всего записей	Одобрено	NULL	Исправить	Повтор
1	1	109	91	11	5	2

В цикле (№78-83) перебираются все элементы массивов **kha_StatArr** (все значения поля «статус», которые встречаются в базе данных) и массива **kha_StatRecArr** (количества записей для каждого значения поля «статус»).

kha_StatArr.length – количество элементов этих массивов (одно и тоже для каждого массива).

```
78 for(i=0; i<kha_StatArr.length; i++){
```

В №79 переменной **n** присваивается значение текущего элемента массива **kha_StatRecArr**.

```
79 n = kha_StatRecArr[i];
```

В №80 переменной **s** присваивается значение текущего элемента массива **kha_StatArr**.

```
80 s = kha_StatArr[i];
```

В №81 переменной **s1** присваивается строка, в которой записано начало очередной ячейки числовой строки (тег **<td...>**).

```
81 s1 = '<td align="center"' + kha_GetStatClass(s) + '>';
```

В №82 выводится ячейка числовой строки.

```
82 document.write(s1+n+'</td>');
```

```
83 }
```

Рассмотрим подробно работу этого алгоритма отдельно для **html** – файла и страницы **Moodle**.

Для **html** – файла у нас массив **kha_StatArr** имеет только 1 элемент

```
kha_StatArr[0] = '[[status]]'
```

Массив **kha_StatRecArr** тоже имеет только 1 элемент.

```
kha_StatRecArr[0] = '1'
```

Длина ЭТИХ массивов `kha_StatArr.length = 1`

На 1-м шаге цикла

```
78  i=0
79  n = kha_StatRecArr[0] = 1;
80  s = kha_StatArr[0] = '[[status]]';
```

Функция `kha_GetStatClass('[[status]]')` возвращает значение `' class="kha_other"'`.

```
81  s1 = '<td align="center" class="kha_other">';
82  document.write(
'<td align="center" class="kha_other">1</td>');
```

Всего записей	[[status]]
1	1

На следующем шаге цикла `i=1`, условие `i < kha_StatArr.length` не выполняется, цикл завершает свою работу.

При реальном выводе информации в страницу Moodle массив `kha_StatArr` имеет 4 элемента

```
kha_StatArr[0] = 'Одобрено'
kha_StatArr[1] = 'NULL'
kha_StatArr[2] = 'Исправить'
kha_StatArr[3] = 'Повтор'
```

Массив `kha_StatRecArr` тоже имеет 4 элемента.

```
kha_StatRecArr[0] = '91'
kha_StatRecArr[1] = '11'
kha_StatRecArr[2] = '5'
kha_StatRecArr[3] = '2'
```

Длина ЭТИХ массивов `kha_StatArr.length = 4`

На 1-м шаге цикла

```
78 i=0
79 n = kha_StatRecArr[0] = 91;
80 s = kha_StatArr[0] = 'Одобрено';
```

Функция `kha_GetStatClass('Одобрено')` возвращает значение '' (пустая строка).

```
81 s1 = '<td align="center">';
82 document.write(
'<td align="center">91</td>');
```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На 2-м шаге цикла

```
78 i=1
79 n = kha_StatRecArr[1] = 11;
80 s = kha_StatArr[1] = 'NULL';
```

Функция `kha_GetStatClass('NULL')` возвращает значение ' class="kha_null" '.

```
81 s1 = '<td align="center" class="kha_null">';
82 document.write(
'<td align="center" class="kha_null">11</td>');
```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На 3-м шаге цикла

```
78 i=2
79 n = kha_StatRecArr[2] = 5;
80 s = kha_StatArr[2] = 'Исправить';
```

Функция `kha_GetStatClass('Исправить')` возвращает значение ' class="kha_redo" '.

```

81     s1 = '<td align="center" class="kha_redo">';
82     document.write(
'<td align="center" class="kha_redo">5</td>');

```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На 4-м шаге цикла

```

78     i=3
79     n = kha_StatRecArr[3] = 2;
80     s = kha_StatArr[3] = 'Повтор';

```

Функция `kha_GetStatClass('Повтор')` возвращает значение
' class="kha_double" '.

```

81     s1 = '<td align="center" class=" kha_double ">';
82     document.write(
'<td align="center" class=" kha_double "> Повтор </td>');

```

Всего записей	Одобрено	NULL	Исправить	Повтор
109	91	11	5	2

На следующем шаге цикла `i=4`, условие `i<kha_StatArr.length` не выполняется, цикл завершает свою работу.

№84 завершает фрагмент программы (начало в №68)

```
84 //]]></script>
```

№85 завершает описание таблицы статистической информации

```
85 </tr></table></b>
```

№86-87 завершают файл html.

```
86 </body>
```

```
87 </html>
```

База данных и глоссарий

Глоссарий в Moodle действует интересным образом – просто добавляет в страницу большой кусок кода специального вида. При этом все наши скрипты просто рушатся, хорошо, что браузер игнорирует такие ошибки и продолжает свою работу.

Рассмотрим конкретный пример. Создадим глоссарий, и внесем в него термин «Повтор» – результат невнимательности, который влечет за собой существенное снижение оценки. Зададим автосвязывание для этого слова.

До включения глоссария фрагмент базы данных с повторяющимися записями выглядел так.

	Харьковская областная государственная администрация ХНАГХ	50.0040°N, 36.2362°E метка	http://sketchup.google.com/3dwarehouse/details?mid=ded46e8440857cbbf7977f36b6d5323b	Дарья Лыкова	Воскресенье 11 Ноябрь 2012, 09:30	Шульга Анастасия	Да ✖
	Porsche Centre Kharkov	50.0028°N, 36.2176°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e06dbacf1c630dc94519ac65c4e84241	Руслан Че	Четверг 13 Декабрь 2012, 21:36	Конюхова Анастасия	Повтор ✖
	Центр "Porsche"	50.0027°N, 36.2177°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e06dbacf1c630dc94519ac65c4e84241	Руслан Че	Воскресенье 13 Ноябрь 2011, 17:13	Маслова Ксения	Да ✖
	Спорткомплекс ХАИ ХНАГХ	50.0452°N, 36.2841°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e87a4b37e49a113bf01f15a4da57e7a	Анастасия Зорина	Среда 21 Ноябрь 2012, 15:29	Кравченко Сергей	Повтор ✖
	Спорткомплекс ХАИ ХНАГХ	50.0450°N, 36.2841°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e87a4b37e49a113bf01f15a4da57e7a	Anastasiia Zorina	Среда 14 Ноябрь 2012, 00:18	Вересова Милослава	Да ✖

После включения глоссария тот же фрагмент выглядел так.

	Харьковская областная государственная администрация ХНАГХ	50.0040°N, 36.2362°E метка	http://sketchup.google.com/3dwarehouse/details?mid=ded46e8440857cbbf7977f36b6d5323b	Дарья Лыкова	Воскресенье 11 Ноябрь 2012, 09:30	Шульга Анастасия	Да ✖
	Porsche Centre Kharkov	50.0028°N, 36.2176°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e06dbacf1c630dc94519ac65c4e84241	Руслан Че	Четверг 13 Декабрь 2012, 21:36	Конюхова Анастасия	Повтор ✖
	Центр "Porsche"	50.0027°N, 36.2177°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e06dbacf1c630dc94519ac65c4e84241	Руслан Че	Воскресенье 13 Ноябрь 2011, 17:13	Маслова Ксения	Да ✖
	Спорткомплекс ХАИ ХНАГХ	50.0452°N, 36.2841°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e87a4b37e49a113bf01f15a4da57e7a	Anastasiia Zorina	Среда 21 Ноябрь 2012, 15:29	Кравченко Сергей	Повтор ✖
	Спорткомплекс ХАИ ХНАГХ	50.0450°N, 36.2841°E метка	http://sketchup.google.com/3dwarehouse/details?mid=e87a4b37e49a113bf01f15a4da57e7a	Anastasiia Zorina	Среда 14 Ноябрь 2012, 00:18	Вересова Милослава	Да ✖

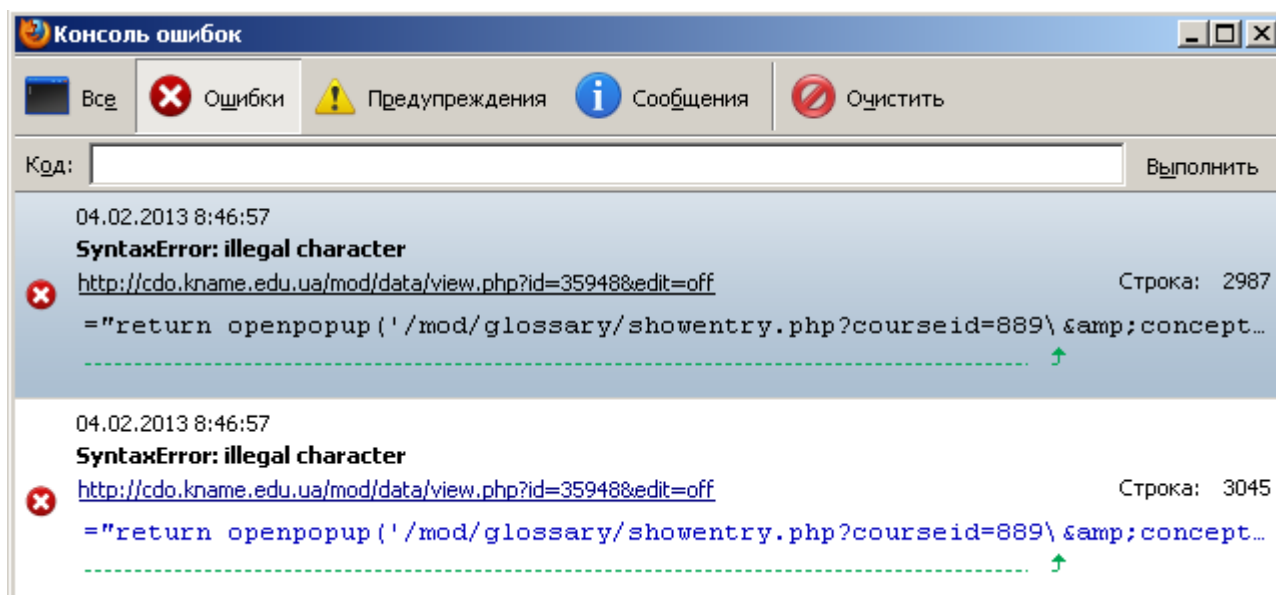
Форматирование строки потеряно, зато при нажатии на слово «Повтор» появилось такое откровение.

Повтор

:

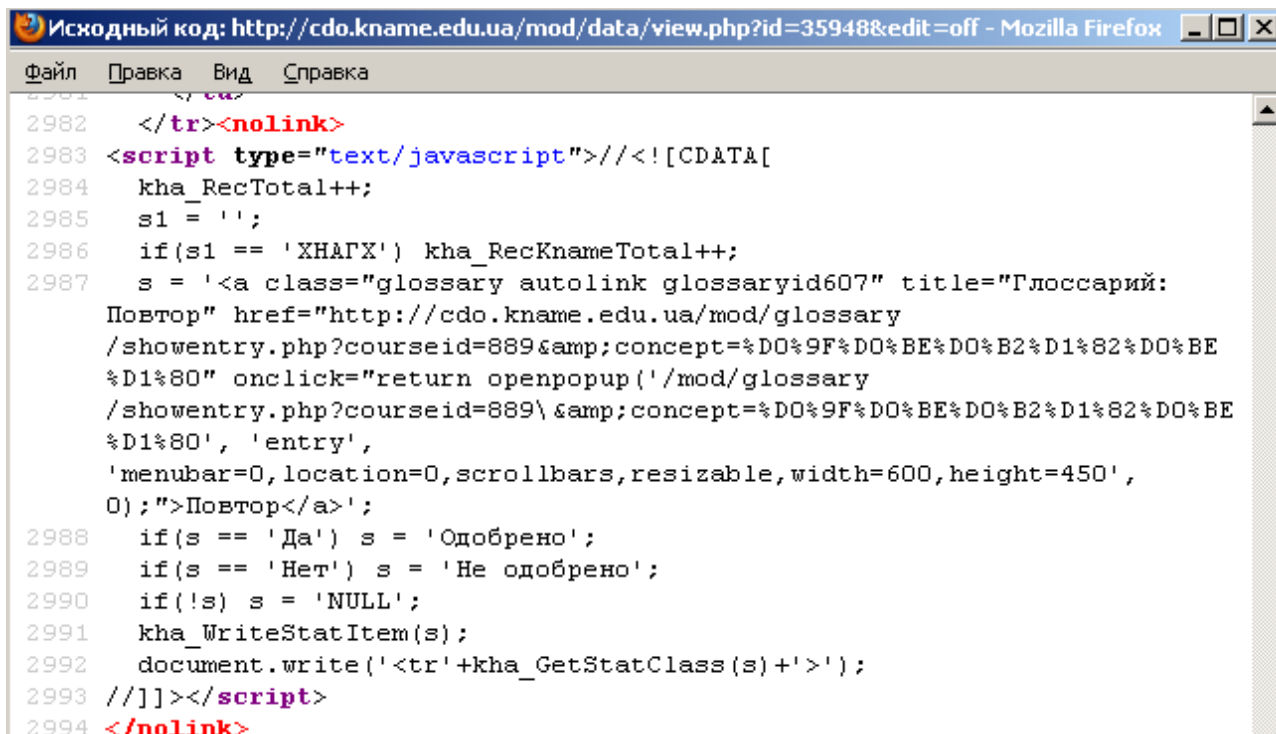
Результат невнимательности, который влечет за собой существенное снижение оценки.

В консоли ошибок (мы используем Mozilla Firefox 18.0.1) браузера мы видим, в каком месте обрушился скрипт.



В исходном коде видим, что переменной **s** вместо значения 'Повтор' присваивается несколько строк текста. Естественно эта строка нарушает правила синтаксиса, и весь фрагмент кода отбрасывается.

Отметим, что рекомендованные теги `<nolink>...</nolink>` в данном случае не срабатывают.



Как видно из примера глоссарий засоряет страницы Moodle мусором и его применение, с нашей точки зрения, должно иметь серьезное обоснование.

Заключение

Мы очень надеемся, что после изучения этой работы все читатели разобрались с нашей базой данных Moodle, а у некоторых даже появились идеи как сделать что-нибудь свое (используя нашу базу как основу).

Мы будем очень рады, если Вы напишете, что у Вас получилось. Пишите нам по адресу:

itedu@lib-journal.ru

Если у Вас есть замечания и предложения – пишите!

Литература

1. Database activity module
[http://docs.moodle.org/22/en/Database_activity_module].
2. Анисимов А.М. Работа в системе дистанционного обучения Moodle. – Харьков: ХНАГХ, 2009. – 292 с.