**БД «ВУЗ»**

Задача заключается в построении схемы базы данных для произвольного ВУЗа. Наша модель данных будет достаточно ограничена, то есть не будет включать в себя все возможности, но будет в основных сущностях отражать предметную область.

**Модель «ВУЗа»**

ВУЗ имеет полное название, краткое название, почтовый адрес, юридический адрес. Учтите, что свойства ВУЗа могут быть расширяемы неограниченно (например, адрес сайта, телефон, факс и тд). Можно считать, что в нашей БД планируется хранение только одного ВУЗа. Но подумайте над тем, что изменится, если в этой же БД придется хранить более одного ВУЗа. При желании можете самостоятельно расширить схему БД на такую ситуацию. В данном случае это прекрасный пример того, что проектировщик БД должен сразу задумываться над ответом на вопрос: планируется ли расширение функциональности БД, если да, то в какую сторону, и как минимизировать затраты в случае подобного расширения БД.

В ВУЗе есть различные подразделения: Ректорат, Деканаты, Приемная Комиссия, Кафедры, Отдел по работе с выпускниками, Бухгалтерия и тд, можете расширить список на примере Вашего ВУЗа. У каждого подразделения есть адрес, телефон подразделения, e-mail адрес, факс, приемные часы.

В БД должен содержаться список всех сотрудников, которые работают в ВУЗе. У сотрудника могут быть заведены следующие свойства: персональный идентификатор, ФИО, год рождения, ученая степень, ученое звание, должность, подразделение, к которому относится сотрудник. Пусть в нашей модели можно являться сотрудником только одного подразделения. Подумайте над тем, что список свойств сотрудника может расширяться до бесконечности, возможно, в какой-то момент времени возникнет необходимость добавить СНИЛС, номер паспорта, информацию из трудовой книжки и тд. Должность – это какую должность сотрудник занимает в текущий момент в подразделении. Например, ректор, декан, секретарь кафедры, заведующий кафедрой, доцент кафедры, старший преподаватель, главный специалист (например, отдела по работе с выпускниками).

Во всех подразделениях есть ответственные должности (например, секретарь или заведующий кафедрой), чьи контактные данные (телефон, приемные часы, возможно факс, возможно e-mail, и тд) тоже должны быть заведены в базе.

Обратите внимание, что секретарь кафедры может измениться, но при этом телефон секретаря кафедры всё равно останется тем же самым. При этом возможна и другая ситуация, секретарь кафедры остается тем же, кто и раньше, но изменяется e-mail адрес должности. То есть контактные данные – это свойство именно должности на кафедре, а не человека, который сейчас занимает эту должность. Кто именно в текущий момент времени занимает эту должность, будет понятно из текущего состояния таблицы с сотрудниками.

Каждый сотрудник может иметь право преподавать 0 и более предметов (0 – если это непреподавательская должность, в целом колво предметов для преподавания одним преподавателем сверху не ограничено), и каждый предмет имеют право вести от 0 до произвольного количества преподавателей. Соответственно, необходимо завести соответствие сотрудника и преподаваемых им предметов. В нашей БД не требуется добавление признака, какая должность (или какой конкретно сотрудник) имеет право на преподавание. Вы можете сделать это расширение предметной области по собственному желанию.

Предметы – у ВУЗа есть список предметов, которые могут преподаваться в этом ВУЗе. Каждый предмет имеет персональный идентификатор, название, количество часов лекций, количество часов семинаров. Будем считать, что каждый предмет читается не более одного семестра. Вы можете заметить, что на Физтехе, например, теорвер читается практически всем факультетам, но в разном объеме, с разной программой. В нашей модели будет значительное упрощение: будем считать, что если предмет имеет определенное название, то он читается одинаковом объеме (указанном в свойствах) для всех факультетов, которые выбрали этот предмет в учебный план. В нашей простой модели можно предложить и следующее разделение таких предметов: «Теорвер: средняя сложность», «Теорвер: максимальная сложность», «Теорвер для ФАЛТа» и тд. Это не лучший способ, но для знакомства с проектированием БД подходит.

Каждый предмет преподается не более чем одной кафедрой (но могут быть предметы, которые в этом учебном году по каким-то причинам не проводит ни одна кафедра).

Обратите внимание, что преподаватель, который относится к какому-то подразделению, может иметь право вести предметы не только этого подразделения.

На каждом факультете учатся студенты. У студентов есть персональный идентификатор, ФИО, номер группы, год поступления, статус. Статус – первый год обучения, второй год обучения, …., отчислен после N семестра, закончил, академ и тд.

В нашей модели перевод между факультетами запрещен. Номер группы у студента может изменяться (внутри факультета). Необходимо сделать так же таблицу с историей, в каком семестре в какой группе студент обучался. Перевод между группами в нашей модели возможен только после завершения прошедшего семестра и началом нового.

В нашей модели даже если студент является (станет после окончания) сотрудником ВУЗа, то связь между его идентификатором студента и идентификатором преподавателя поддерживать не обязательно. Но Вы можете предложить идею, как создать связь идентификатор студента – идентификатор преподавателя.

У каждого факультета есть свой учебный план для каждого года обучения. В нашей модели считаем, что этот план вечный. Учебный план включает в себя список предметов с указанием формы оценки (без зачета, зачет без оценки, зачет с оценкой, экзамен), номера семестра, когда эти предметы должны изучаться студентами этого факультета. У разных групп на одном курсе на одном факультете учебный план одинаков. Считаем, что деление на группы нужно, например, для базовых кафедр, которые у разных групп могут быть разными, что мы в нашей БД ВУЗов не отображаем.

Соответственно на 2017-2018 учебный год в БД должно храниться расписание, у какой группы, в каком семестре, какой предмет какой преподаватель вел. При желании можете добавить информацию: в какой день недели, какой парой и выделить день под базовую кафедру.

Кроме того, в конце каждого семестра проходят зачетные и экзаменационные недели, на которых заполняются ведомости. Одна ведомость на каждого преподавателя и каждую попытку сдачи каждого предмета. Например, экзамен у 1го курса одного факультета принимается несколькими преподавателями, и у каждого из преподавателей будет отдельная ведомость:

Номер ведомости

Дата сдачи

Номер семестра, за который сдается предмет

Принимающий преподаватель (даже в случае зачетов может не совпадать с ведущим семинарские занятия преподавателем)

Предмет

Список студентов с итоговыми оценками (неуд, уд, хор, отл, зачет или не зачет в случае явки и неявка, если студент не пришел).

На основе этих ведомостей создаются таблицы с данными ведомостей.

По заполненным ведомостям по каждому студенту к началу следующего семестра создается успеваемость: студент, номер прошедшего семестра, предмет, форма отчетности, оценка (последняя), колво попыток сдачи, дата последней сдачи, номер ведомости по сдаче с последней оценкой.

Запросы для заполнения таблиц нужными данными (как в случае «по ведомостям создать успеваемость») не нужны. Достаточно только схемы данных с небольшим количеством данных.

**Необходимо создать схему данных, то есть:**

- Названия таблиц и поля в них;

- Для полей должны быть указаны ключи и прописаны типы данных;

- Для каждой таблицы необходимо привести не менее 5-ти строк данных, которые планируется хранить в этой таблице.

- Дополнительно (будет оцениваться отдельно) просьба сформулировать список ограничений, накладываемых на базу, то есть какие проверки/связи необходимо создать, чтобы в базе данных в процессе работы сохранялась целостность и непротиворечивость информации.

База данных должна быть не менее чем в 3НФ, в обосновании должно быть доказательство этого факта. Если Вы планируете сделать базу не в 3НФ, то просьба обосновать, чем Ваша структура лучше, почему её не стоит приводить к 3НФ. Кроме того необходимо привести обоснование, почему приведенная схема данных отвечает всем требованиям ТЗ.

Все дополнительные предположения (если в ТЗ что-то не указано) должны быть в приложенном описании/обосновании.

Прислать схему данных достаточно в текстовом файле или xls (любом spreadsheet формате) в виде:

Название таблицы, названия и типы полей, ключевые поля, 5 строчек данных и тд для каждой таблицы.

Если владеете xml или json, то данные для таблиц можете прислать в xml или json формате.

**Доп вопрос:**

Подумайте, а как студентов и преподавателей хранить в одной таблице? (в этом случае не понадобилась бы связь между студентами и преподавателями, если бы студент стал преподавателем).

Комментарий: есть подходы к работе, когда ТЗ дописывается в процессе работы с заказчиком. То есть в процессе выяснения ответов на вопросы по неохваченным темам ТЗ может постепенно дополняться.

В Вашем случае заказчиком выступаю я, ТЗ точно не отвечает на все вопросы, которые возникнут у Вас, поэтому предлагаю Вам следующий вариант работы с ТЗ: все неясные моменты Вы решаете по Вашему желанию. Но еще раз просьба все Ваши предположения явно прописать в письме или в файле со схемой данных.