

## 1) Установка MySQL

Рассмотрим установку пакета программ для Windows, содержащего **MySQL server**, среду для разработки и администрирования **MySQL Workbench** и много других полезных компонентов. (Источник - <http://www.mysql.com/downloads/installer/> )

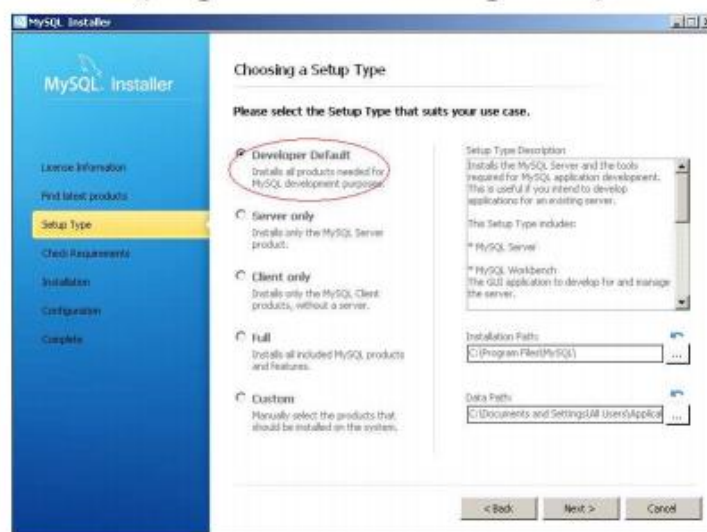
1. Для начала нужно установить Microsoft .NET 4.0 Framework, если его нет.

2. Установить Visual C++ Redistributable Packages for Visual Studio 2013.

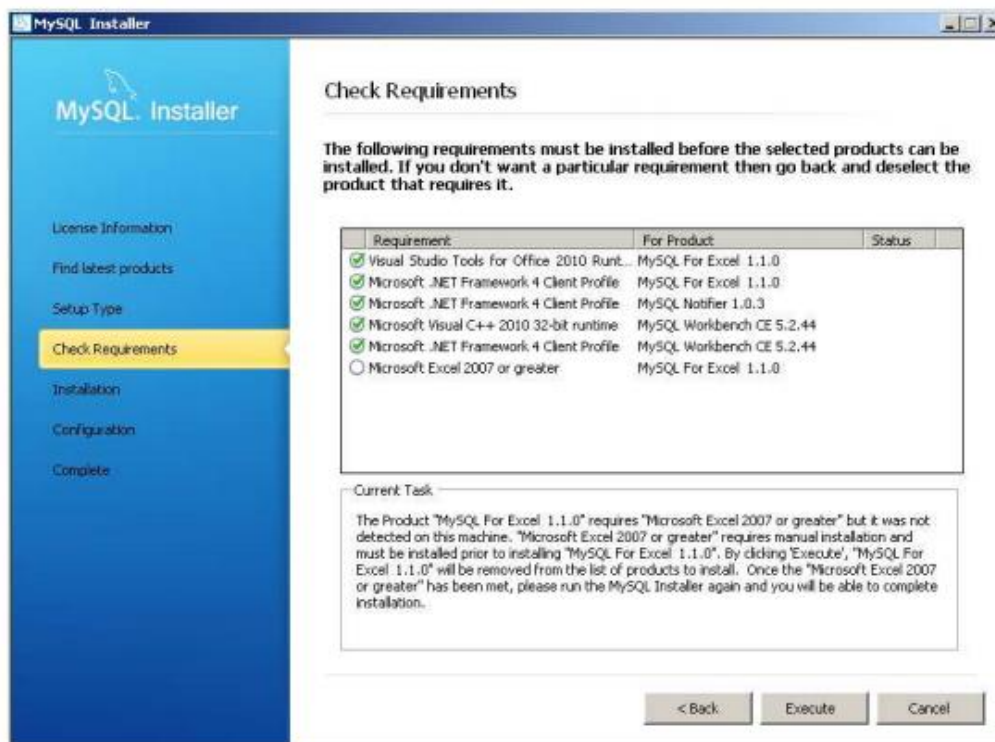
Теперь можно устанавливать **MySQL Workbench** - mysql-installer-community-5.5.28.3.msi или более новую версию:



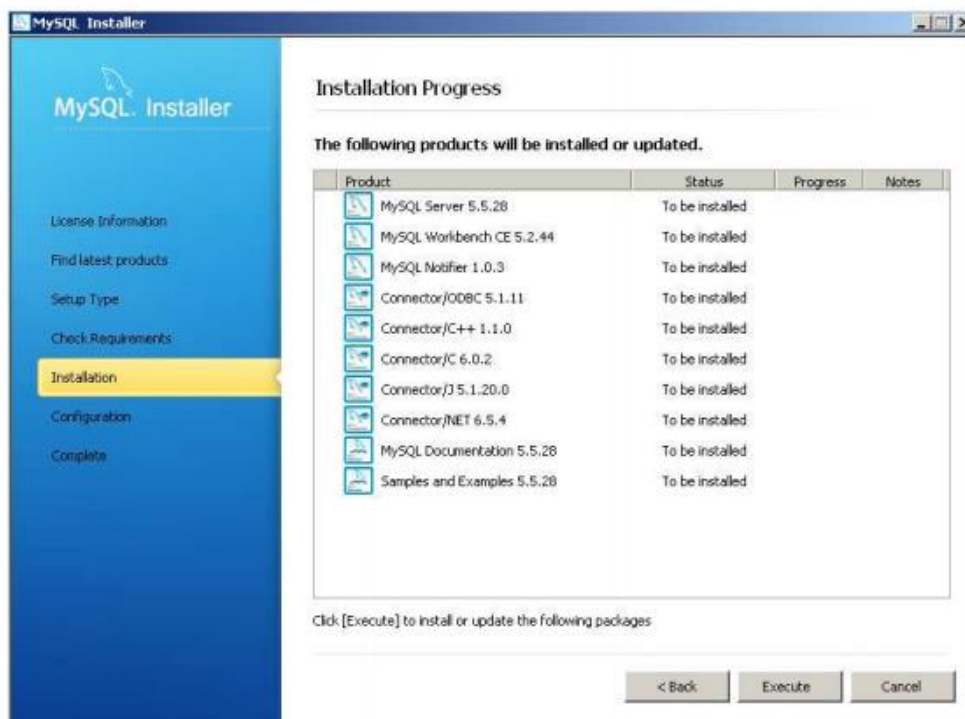
На 2 странице принимаем лицензию. На 3 странице **не проверяем обновления** (skip the check for updates).



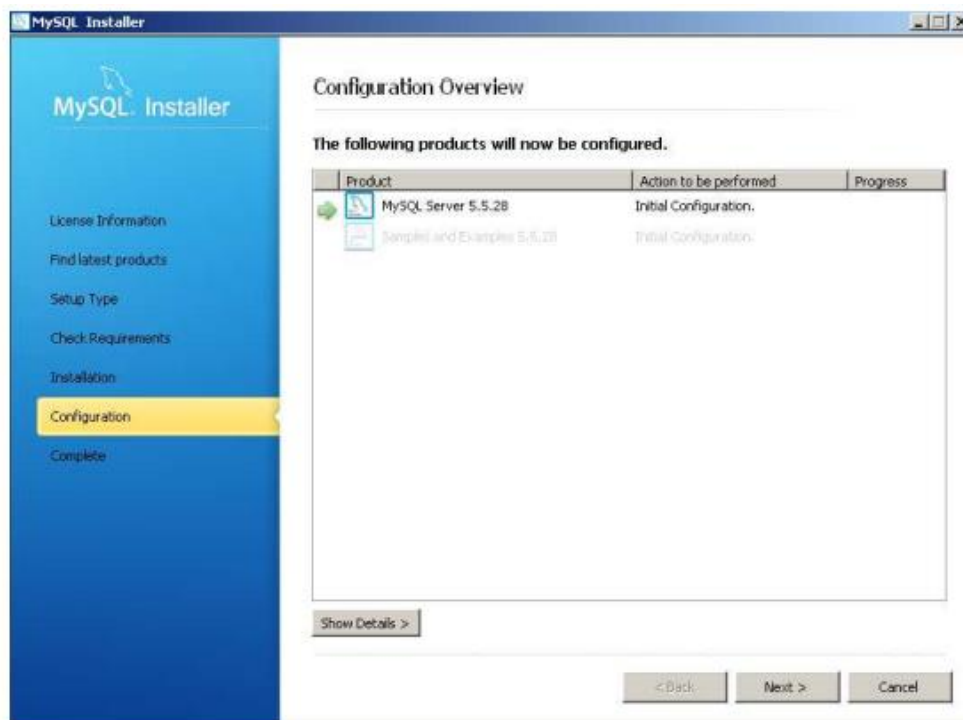
Проверяются необходимые условия для установки:



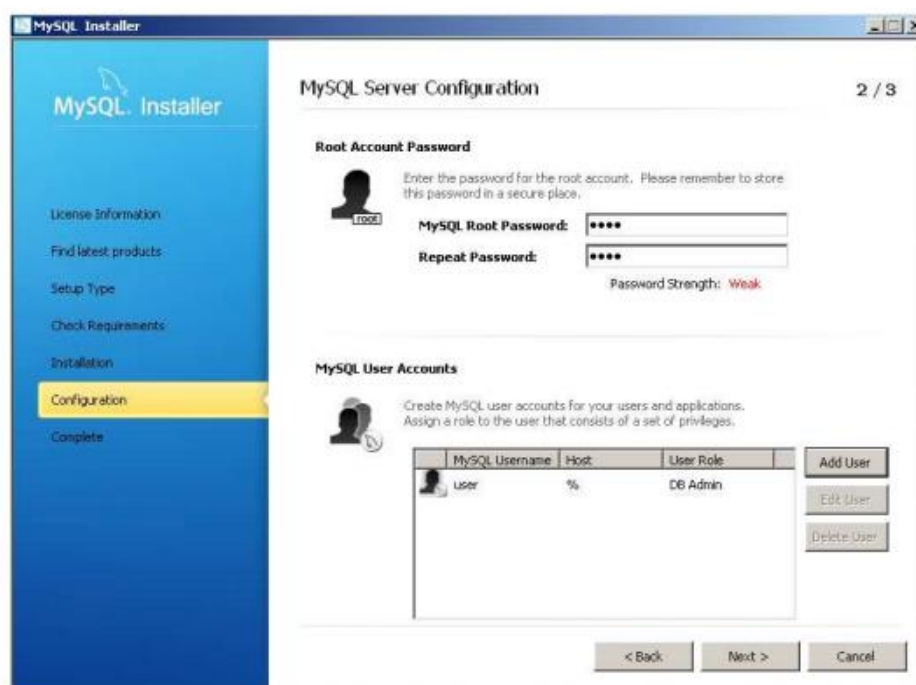
Устанавливаем следующие продукты: (первый пункт – это сам сервер, второй пункт – среда разработчика)



Конфигурирование сервера:



Для администратора по имени **root** зададим пароль. **Запомните его!** А также создадим пользователя по имени **user** с паролем. **Тоже запомните его!** Можно сделать пароли, совпадающие с именем пользователя: **root** и **user**, хотя в смысле безопасности это плохой вариант, но для учебных целей годится.



На следующих страницах ничего не изменяем. Имя сервера по умолчанию MySQL55. Установка успешно завершена.



## 2)Создание EER-диаграммы

Среда **mySQL Workbench** предназначена для визуального проектирования баз данных и управления сервером **mySQL**.

Для построения моделей предназначена секция **Data Modeling**:



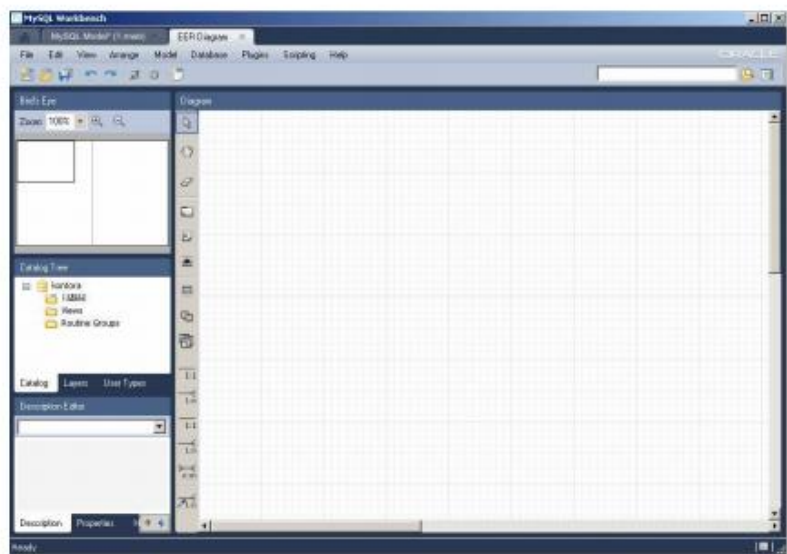
Выберем пункт **Create new EER Model**.


**EER model** расшифровывается как **Extended Entity-Relationship Model** и переводится как **Расширенная модель сущностей-связей**.

По умолчанию имя созданной модели **myDB**. Щелкните правой кнопкой мыши по имени модели и выберите в появившемся меню пункт **Edit schema**. В появившемся окне можно изменить имя модели. Назовем ее, например, **kontora**. В именах таблиц и столбцов нельзя использовать русские буквы.

В этом окне также **нужно** настроить так называемую «кодировку страницы» для корректного отображения русских букв **внутри** таблиц. Для этого выберите из списка пункт **«cp1251-cp1251\_general\_ci»**. Окно свойств можно закрыть.

Диаграмму будем строить с помощью визуальных средств. Щелкнем по пункту **Add diagram**, загрузится пустое окно диаграммы:



Создать новую таблицу можно с помощью пиктограммы . Нужно щелкнуть по этой пиктограмме, а

потом щелкнуть в рабочей области диаграммы. На этом месте появится таблица с названием по умолчанию **table1**. Двойной щелчок по этой таблице открывает окно редактирования, в котором можно изменить имя таблицы и настроить её структуру.

Будем создавать таблицу **Отделы** со следующими столбцами: **номер\_отдела**, **полное\_название\_отдела**, **короткое\_название\_отдела**.

Переименуем **table1** в **k\_dept** и начнем создавать столбцы.

Каждый столбец имеет:

- имя (не используйте русские буквы в имени!),
- тип данных. Самые распространенные типы данных:
  - INT – целое число;
  - VARCHAR(размер) – символьные данные переменной длины, в скобках указывается максимальный размер;
  - DECIMAL(размер, десятичные\_знаки) – десятичное число;
  - DATE – дата;
  - DATETIME – дата и время.

Далее располагаются столбцы, в которых можно настроить дополнительные свойства поля, включив соответствующий флажок:

- PK (primary key) – первичный ключ;
- NN (not null) – ячейка не допускает пустые значения;
- UQ (unique) – значение должно быть уникальным в пределах столбца;
- AI (auto incremental) – это свойство полезно для простого первичного ключа, оно означает, что первичный ключ будет автоматически заполняться натуральными числами: 1, 2, 3, и т.п.;
- DEFAULT – значение по умолчанию, т.е., значение, которое при добавлении новой строки в таблицу автоматически вставляется в ячейку сервером, если пользователь оставил ячейку пустой.

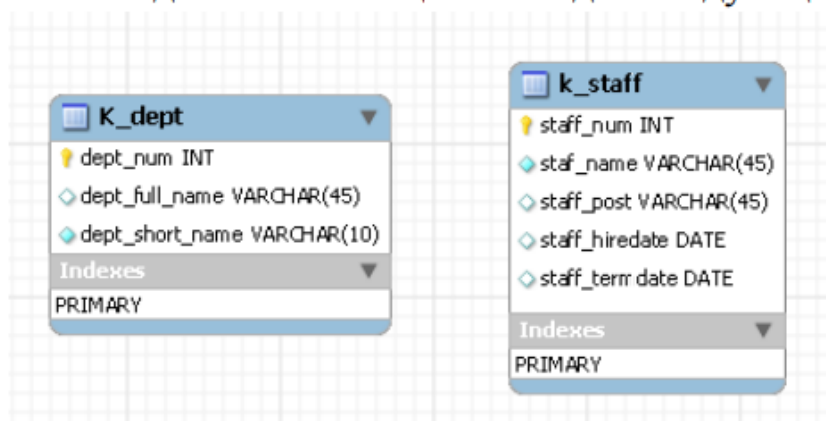
Таблица **Отделы** имеет следующий вид:

K_dept									
Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
dept_num	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
dept_full_name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
dept_short_name	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	


Далее создадим таблицу **Сотрудники** со следующими столбцами: номер\_сотрудника, имя\_сотрудника, должность, дата\_начала\_контракта, дата\_окончания\_контракта

k_staff										
Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default	
staff_num	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
staf_name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
staff_post	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
staff_hiredate	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
staff_termdate	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

На диаграмме созданные таблицы выглядят следующим образом:

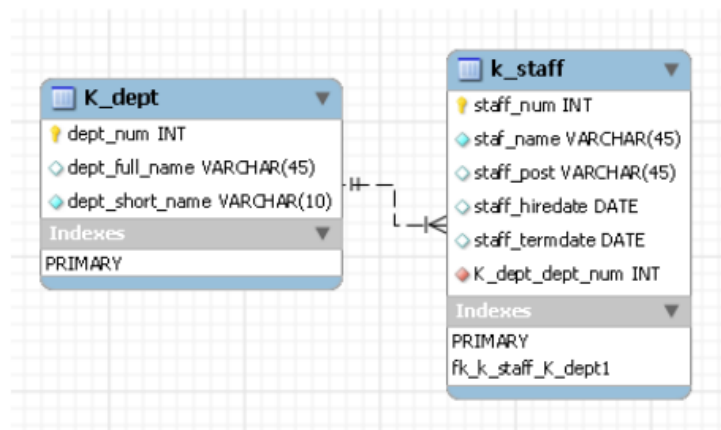


Обратите внимание, что при создании первичного ключа автоматически создается **индекс** по этому первичному ключу. **Индекс** представляет собой вспомогательную структуру, которая служит, прежде всего, для **ускорения поиска** и **быстрого доступа** к данным.


Теперь свяжем эти таблицы. Сначала создадим связь «Работает» между **Сотрудником** (дочерняя таблица) и **Отделом** (родительская таблица), степень связи M:1. Для создания связей M:1 служит пиктограмма на панели инструментов  (с пунктирной линией). С ее помощью создается так называемая «неидентифицирующая связь», т.е. обыкновенный внешний ключ, при этом первичный ключ родительской таблицы добавляется в список столбцов дочерней таблицы.

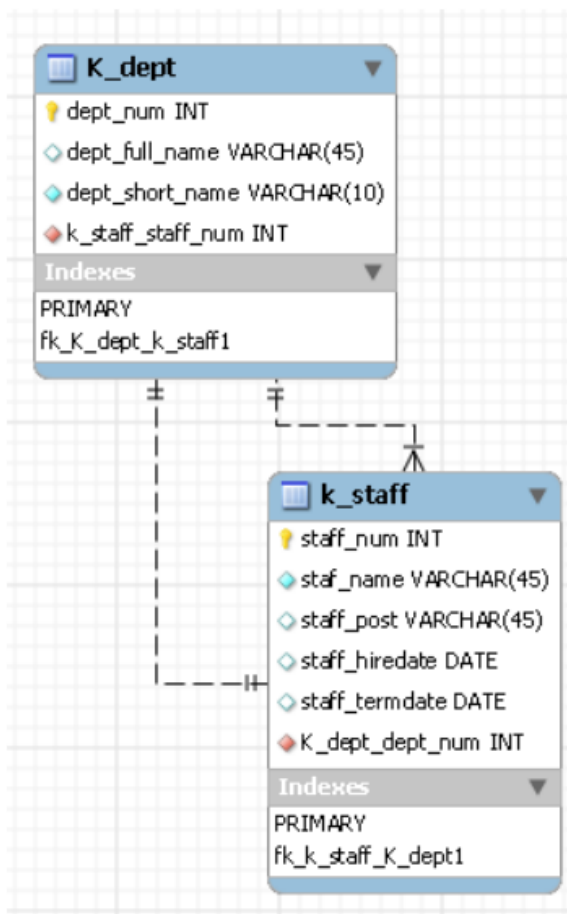
Итак, щелкнем на пиктограмме, затем щелкнем на дочерней таблице **Сотрудники**, затем на родительской таблице **Отделы**:





Обратите внимание, что при этом произошло. Между таблицами образовалась пунктирная линия; в сторону «к одному» она отмечена двумя черточками, в сторону «ко многим» - «куриной лапкой». Кроме того, в таблице **Сотрудники** образовался дополнительный столбец, которому автоматически присвоено имя *k\_dept\_dept\_num* (т.е., имя родительской таблицы плюс имя первичного ключа родительской таблицы). А в группе **Индексы** создан индекс по внешнему ключу.

Теперь добавим связь между этими же таблицами «Руководит» 1:1. Выберем пиктограмму , затем щелкнем по **Отделам**, затем по **Сотрудникам**.



Чтобы 2 связи на картинке не «завязывались узлом», мы их разместили друг под другом.

Обратите внимание, что в таблицу Отделы был автоматически добавлен столбец *k\_staff\_staff\_num*, а также индекс по внешнему ключу.

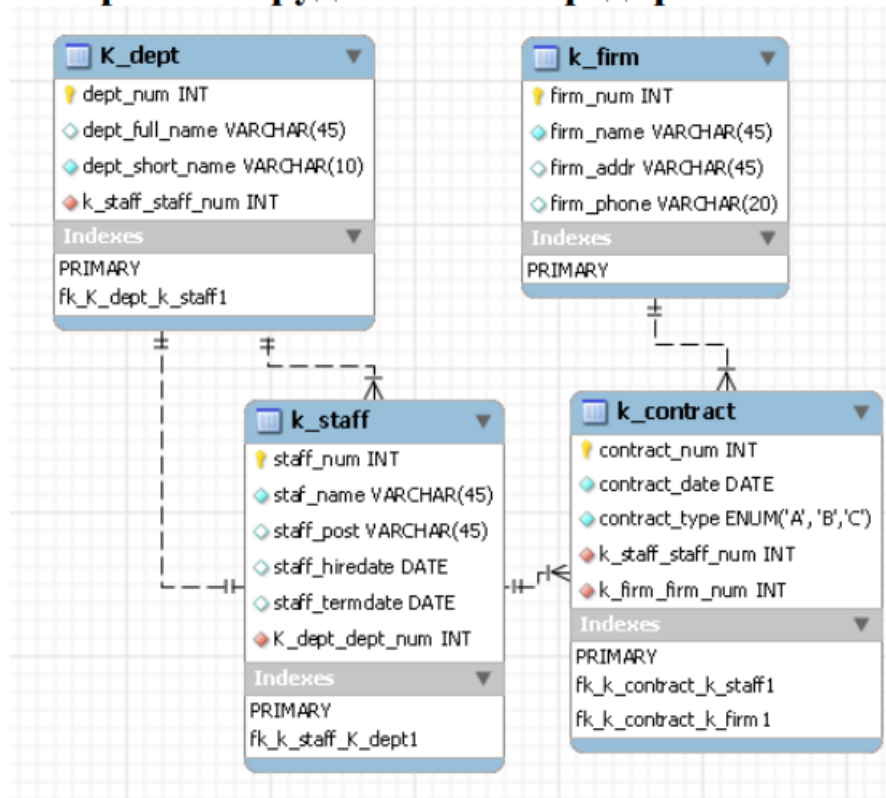
Создадим таблицу **Предприятия**:

k_firm										
Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default	
firm_num	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
firm_name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
firm_addr	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
firm_phone	VARCHAR(20)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Создадим таблицу **Договоры**. У столбца Тип\_договора зададим следующий формат: это буква из списка 'А', 'В', 'С'.

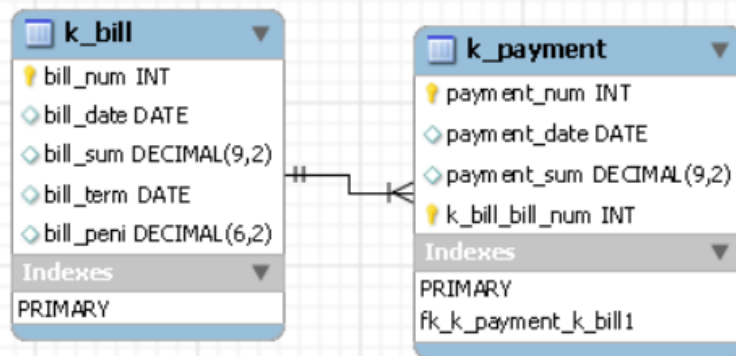
k_contract										
Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default	
contract_num	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
contract_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
contract_type	ENUM('A', 'B', 'C')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Свяжем **Договоры** с **Сотрудниками** и **Предприятиями** связями М:1.






Затем создадим **Счета и Платежи**:

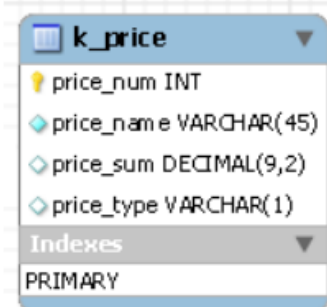



Поскольку сущность **Платеж** была «слабой», у нее нет полноценного первичного ключа, и каждый платеж однозначно идентифицируется группой атрибутов (номер\_счета, номер платежа). Отметим в качестве ключевого поля

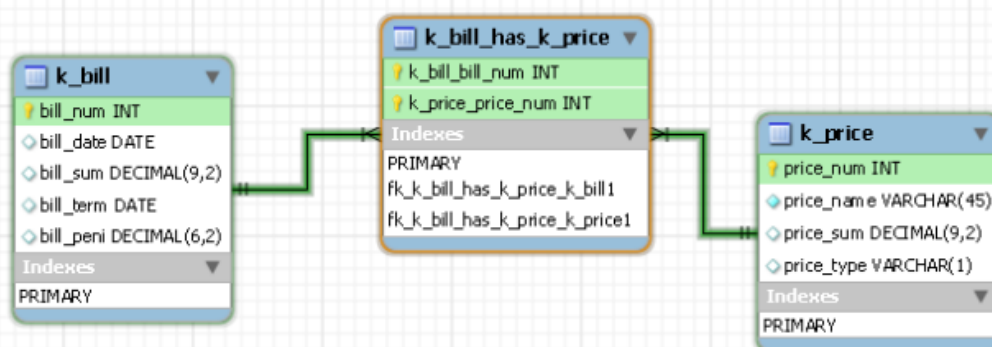
*payment\_num*, а затем создадим **идентифицирующую** связь между **Счетом и Платежом**.

Идентифицирующая связь создается с помощью пиктограммы  (со сплошной линией). При этом новый столбец *k\_bill\_bill\_num* становится не только внешним ключом в таблице **Платеж**, но и частью первичного ключа.

Далее создадим таблицу **Прайс-лист** со столбцами (номер\_товара, название\_товара, цена\_товара и тип\_товара).



Между объектами **Счет** и **Прайс-лист** имеется связь «многие - ко многим». Для создания этой связи нужно использовать пиктограмму . Следует щелкнуть мышью по этой пиктограмме, а затем последовательно щелкнуть по связываемым таблицам. Между ними появится новая таблица, обратите внимание на ее столбцы, первичный ключ и внешние ключи:



Для удобства переименуем эту таблицу в *k\_protokol* (ПротоколСчета), добавим столбцы *kolvo* и *price\_sum*.

k_protokol										
Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default	
k_bill_bill_num	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
k_price_price_num	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
kolvo	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
price_sum	DECIMAL(9,2)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Теперь EER-диаграмма имеет такой вид:

