**Лабораторная работа № 3. Проектирование реляционной базы данных**

Процесс **проектирования** базы данных включает в себя следующие шаги:

1. Инфологическое проектирование: определение задач, стоящих перед базой данных, описание особенностей предметной области, создание модели предметной области, определение групп пользователей.

2. Выбор аппаратной и программной платформы.

3. Выбор системы управления базой данных (СУБД) и других инструментальных программных средств.

4. Логическое проектирование базы данных: определение сущностей, их атрибутов и связей между сущностями, создание схем отношений, определение типов данных и ограничений целостности, нормализация отношений, определение прав доступа пользователей к объектам БД.

|  |  |
| --- | --- |
| **Задание** | **Краткие теоретические сведения** |
| 1. Создать *диаграмму вариантов использования* базы данных в соответствии со своим вариантом из таблицы п. 8. | Модель предметной области можно представить в виде диаграммы вариантов использования базы. Например, для базы данных ПРОДАЖИ фрагмент такой диаграммы может выглядеть следующим образом: |
| 2. Выбрать *аппаратную* и *программную* платформу, а также *систему управления базой* данных. | Необходимо знать объём работы организации, характер и интенсивность запросов. Например, для базы данных ПРОДАЖИ можно принять ориентировочно, что осуществляется около пятидесяти заказов в день (по 1К на каждый заказ); выполнение каждого заказа состоит в среднем из четырёх этапов (по 0,5К на этап); в компании работают 10 сотрудников (по 0,5К на сотрудника); устаревшие данные переводятся в архив. Тогда объём памяти для хранения данных в год примерно составит:  Mд = 2(50\*1+50\*4\*0,5+10\*0,5) = 310 \* 365 ≈ 114Мб  Если заказы будут осуществляться через сайт, то для реализации требуемых функций лучше всего подходит MySQL. |
| 3. Определить сущности, их атрибуты (поля), создать *схе-мы отношений*, определить типы данных и ограничения целостности. | Например, схема отношения сущности **Заказы:**   |  |  |  | | --- | --- | --- | | **Имя поля** | **Тип** | **Примечания** | | Номер заказа | Числовой | PK, nvarchar(10) | | Наименование товара | Текстовый | FK, nvarchar(20), not null | | Цена продажи | Денежный | real | | Количество | Числовой | int | | Дата заказа | Дата / время | date | | Дата поставки | Дата / время | date, больше даты заказа | | Заказчик | Текстовый | FK, nvarchar(20) | | Телефон | Текстовый | nvarchar(50) | | Оплачено | Текстовый | nvarchar(3) |   Схема отношения сущности **Товары:**   |  |  |  | | --- | --- | --- | | **Имя поля** | **Тип** | **Примечания** | | Наименование | Текстовый | PK, nvarchar(20), not null | | Цена | Денежный | real | | Количество | Числовой | int | | Изображение | Поле объекта OLE | varbinary | | Описание товара | Текстовый | nvarchar(120) |   Схема отношения сущности **Заказчики:**   |  |  |  | | --- | --- | --- | | **Имя поля** | **Тип** | **Примечания** | | Наименование фирмы | Текстовый | PK, nvarchar(20), not null | | Адрес | Текстовый | nvarchar(20) | | Телефон | Текстовый | nvarchar(50) | | Расчетный счет | Текстовый | nvarchar(15) | |
| 4. Привести таблицы базы данных к первой нормальной форме. | Пусть исходная информация о продажах имеет следующие поля:  **Наименование\_товара, Цена, Количество\_на\_складе, Наименование\_заказанно-го\_товара, Цена\_заказанного\_товара, Количество\_заказанного\_товара, Общая\_стоимость, Дата\_поставки, Заказчик, Адрес\_заказчика, Расчетный\_счет\_заказчика, Телефон\_заказчика**.  Для уменьшения избыточности данных, надо разбить таблицу на две:  **Наименование\_товара, Цена, Количество\_на\_складе**  и  **Наименование\_заказанного\_товара, Цена\_заказанного\_товара, Количество\_зака-занного\_товара, Общая\_стоимость, Дата\_поставки, Заказчик, Адрес\_заказчика, Расчетный\_счет\_заказчика, Телефон\_заказчика**.  Первая таблица соответствует первой нормальной форме, а во второй таблице имеется избыточность. Если один заказчик купит много товаров, то в таблице будут повторяться одни и те же исходные реквизиты заказчика. Поле **Общая\_стоимость** может быть вычислено по полям **Цена\_заказанного\_товара** и **Количество\_заказанного\_товара**.  Поэтому вторую таблицу надо разбить на две:  **Заказчик, Адрес\_заказчика, Расчетный\_счет\_заказчика**  и  **Наименование\_заказанного\_товара, Цена\_заказанного\_товара, Количество\_за-казанного\_товара, Дата\_поставки, Заказчик, Телефон\_заказчика**.  Поле **Общая\_стоимость**, как вычисляемое, в структуру таблицы не включается. |
| 5. Привести таблицы базы данных ко второй нормальной форме. | Из полученных трех таблиц первые две соответствуют второй нормальной форме, а в третьей таблице нет ключевого поля. Если в качестве такового взять поле **Наименование\_заказанного\_товара**, то оно может принимать одно и то же значение для различных заказчиков. Поэтому требуется ввести новое поле, которое являлось бы первичным ключом для всех остальных.  Таким ключом может стать поле **Номер\_заказа**, которое надо добавить, т. е. третья таблица теперь должна содержать поля:  **Номер\_заказа, Наименование\_заказанного\_товара, Цена\_заказанного\_товара, Количество\_заказанного\_товара, Дата\_поставки, Заказчик, Телефон\_заказчика**.  Определим имена таблиц (ключевые поля подчеркнуты):  ТОВАРЫ (**Наименование товара, Цена, Количество\_на\_складе**);  ЗАКАЗЧИКИ (**Заказчик, Адрес\_заказчика, Расчетный\_счет\_заказчика**);  ЗАКАЗЫ (**Номер заказа, Наименование\_заказанного\_товара, Цена\_заказанного\_ товара, Количество\_заказанного\_товара, Дата\_поставки, Заказчик, Телефон\_заказчика*)***. |
| 6. Привести таблицы базы данных к третьей нормальной форме. | В таблице ЗАКАЗЫ поля **Заказчик** и **Телефон\_заказчика** взаимно зависимы. Чтобы привести таблицу к 3 нормальной форме надо поле **Телефон\_заказчика** переместить в таблицу ЗАКАЗЧИКИ.  Теперь полученные таблицы соответствуют всем требованиям.  ТОВАРЫ (**Наименование товара, Цена, Количество\_на\_складе**);  ЗАКАЗЧИКИ (**Заказчик, Адрес\_заказчика, Расчетный\_счет\_заказчика, Телефон\_заказчика**);  ЗАКАЗЫ (**Номер заказа, Наименование\_заказанного\_товара, Цена\_заказанного\_ товара, Количество\_заказанного\_товара, Дата\_поставки, Заказчик*)***. |
| 7. Определить группы пользователей и права доступа к каждой таблице. | Права доступа к таблицам для групп пользователей:   |  |  |  | | --- | --- | --- | | **Таблицы** | **Группы пользователей (роли)** | | | **Продавцы** | **Заказчики** | | Товары | siud | s | | Заказчики | s | suid | | Заказы | s | s |   Используются сокращения: s – чтение данных (select); i – добавление данных (insert); u – модификация данных (update); d – удаление данных(delete). |

8. В соответствии со своим вариантом, номер которого определяет преподаватель, выполнить проектирование базы данных, используя данные из таблицы. При необходимости добавить информационные поля или удалить, сформировать ориентировочно в базе данных пять таблиц.

| **№ варианта** | **Исходная информация** |
| --- | --- |
| 15 | **Информационная система военного округа**. Информационные поля: название вида войск, страна дислокации, город, адрес, занимаемая площадь, кол-во сооружений, номер части в списке частей, место дислокации, вид войск, кол-во рот, кол-во техники, кол-во вооружений в списке вооружений, название техники в списке техники, часть, характеристики, название вооружения, часть, характеристики. |