МИНИСТЕРСТВО ОБРАЗОВАНИЯ РФ

НОВГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИМЕНИ ЯРОСЛАВА МУДРОГО

ИНСТИТУТ ЭКОНОМИКИ И УПРАВЛЕНИЯ

КАФЕДРА СЭММ

# ЛАБОРАТОРНАЯ РАБОТА № 1

**Проектирование базы данных.**

Выполнила:

Студентка гр. 2873

Иванова К.В.

Проверила:

Челпанова М.Б.

Великий Новгород

2008

**Цель работы:** научиться проектировать базу данных.

**Выполнение работы.**

**База данных** – это совокупность сведений (о реальных объектах, процессах, событиях), относящихся к определенной теме, организованная таким образом, чтобы обеспечить удобное представление этой совокупности, как в целом, так и в любой ее части.

**Система управления базами данных** – это комплекс программных и языковых средств, необходимых для создания БД, поддержания их в актуальном состоянии и организации поиска в них необходимой информации.

Система **СУБД Access** является универсальным средством для создания и обслуживания базы данных, обеспечения доступа к данным и их обработки.

Каждая база данных хранится в виде файла с расширением\***.mdb**

База данных создается пользователем для решения определенных задач. Работа над созданием БД должна начинаться с постановки задач и целей, основных функций, выполняемых БД и информации, содержащейся в ней. Эта работа выполняется внекомпьютерной сфере. При проектировке таблиц нужно помнить, что информация в таблице не должна дублироваться, не должно быть повторений и между таблицами. Если определенная информация хранится только в одной таблице, то и изменять ее придется только в одном месте. Это делает работу более эффективной, а также исключает возможность несовпадения информации в разных таблицах.

Основные цели, преследуемые при создании реляционной БД:

1. обеспечить быстрый доступ к данным в таблицах;

2. исключить ненужное повторение данных, которое может являться причиной ошибок при вводе и нерационального использования дискового пространства компьютера;

3. обеспечение целостности данных таким образом, чтобы при изменении одних объектов автоматически происходило соответствующее изменение связанных с ними объектов.

Имеется салон **«Ирина»**, который оказывает парикмахерские, косметические, стилистические и визажистские услуги.

Экономическая цель: наша база данных создается для учета всех оказанных салоном видов услуг каждому заказчику (посетителю), расчета стоимости этих услуг и учета стоимости материалов.

Для реализации этой базы данных мне нужны следующие виды информации:

* учет клиентов;
* учет оказанных услуг по клиентам;
* учет оказанных услуг по услугам исполнителями;
* учет оказанных услуг;
* учет оказанных услуг по видам;
* учет потребности в косметических средствах.

Это и есть экономическая сущность нашей базы данных. Все это образует одну большую таблицу «Салон **«Ирина»** ».

В ней содержится большое количество повторяющейся информации:

* ИНН посетителя;
* Наименование предприятия посетителя;
* Фамилия посетителя;
* Имя посетителя;
* Отчество посетителя;
* Должность посетителя;
* Телефон посетителя;
* Адрес посетителя;
* Используемые косметические средства;
* Цена, руб.;
* Стоимость косметических средств;
	+ Заработная плата;
* Фамилия исполнителя;
* Имя исполнителя;
* Отчество исполнителя;
* Вид услуг.

На основании этой таблицы мы разработали многотабличную базу данных, которая исключит повторение одной и той же информации и обеспечит автоматическое обновление связанной информации при внесении изменения. Для преодоления избыточности информации мы разбили эту таблицу на несколько маленьких таблиц (см. Приложение).

**«Посетители»***.* Каждая запись содержит данные об одном из посетителей.

**«Исполнитель»***.* Каждая запись содержит запись об одном из исполнителей.

**«Виды услуг»**- содержит записи о видах услуг.

**«Используемые косметические средства»** *-* содержит записи об используемых косметических средствах.

**«Расценка на заработную плату»** *-* содержит записи о расценках на заработную плату.

**«Спрос на услуги салона*»*** *-* содержит записи о спросе на услуги салона **«Ирина».**

Перед сохранением таблицы мы задали ключевое поле. Если записи не повторяются, задается *первичный ключ*. Он содержит информацию, которая однозначно определяет каждую запись. Мы выбрали тип данных *текстовый* и размер поля – 5. Выполнили команду: *Правка – Ключевое поле* или соответствующий значок на панели инструментов.

*Вторичный ключ* – это поле, значения которого могут повторятся в разных записях.

**Вывод:** мы научились проектировать свою собственную базу данных.