

## **Лабораторная работа №6**

### **Организация работы с базами данных**

#### **Задание (базовый уровень)**

##### **0. Проектирование базы данных:**

- спроектировать физическую модель базы согласно заданию (структуру базы согласовать с преподавателем).

##### **1. Разработка 1-й версии клиентского приложения MDI Win32:**

- в клиентской части должен быть создан объект типа DataSet содержащий физическую модель базы данных;
- в клиенте предусмотреть возможность отображения, добавления, редактирования и удаления данных из таблиц на форме;
- в качестве первичных ключей необходимо использовать тип int с возможностью автоматического увеличения при добавлении данных;
- информация о схеме данных и данных в таблицах должна сохраняться в XML файл.

##### **2. Разработка 2-й версии клиентского приложения MDI Win32:**

- в качестве СУБД необходимо использовать базу данных Access (можно использовать любую другую локальную версию СУБД);
- в качестве первичных ключей использовать тип Счетчик (для других СУБД предусмотреть аналогичный тип данных);
- доступ к СУБД осуществлять с использованием технологии ADO.NET;
- предусмотреть возможность добавления, редактирования, удаления и отображения данных из таблиц на форме с возможностью быстрого поиска данных в таблицах по определенному полю.

##### **3. Разработка 3-й версии клиентского приложения MDI Win32:**

- в качестве СУБД использовать базу данных MS SQL Server 2008 и выше или СУБД PostgreSQL;
- в данной версии должны быть реализованы все функции 2-й версии;
- в качестве первичных ключей для справочных таблиц необходимо использовать тип int с возможностью автоматического увеличения при добавлении данных, для основных таблиц использовать типа GUID;
- все запросы, связанные с изменением данных в базе данных, поиска информации и выбора информации для отчетов должны осуществляться на основе хранимых процедур;
- для отображения данных таблиц на форме использовать просмотры;
- доступ к СУБД осуществлять с использованием технологии ADO.NET;
- выборка данных для формирования отчетов должно осуществляться как минимум с использованием 2 таблиц;
- один отчет должен сохраняться в формате MS Word, второй в формате MS Excel;
- запрос данных и формирование отчета производить в отдельном потоке.

#### **Задание (повышенный уровень)**

##### **0. Проектирование базы данных:**

- Спроектировать физическую модель базы данных согласно заданию (структуру базы согласовать с преподавателем).

##### **1. Разработка 1-й версии клиентского приложения MDI WPF или с использованием языка Python:**

- в клиентской части возможно использование объекта типа DataSet (C#) содержащего физическую модель базы данных или использование

соответствующих библиотек Python для работы с XML или файлами формата .JSON;

- в клиенте предусмотреть возможность отображения, добавления, редактирования и удаления данных из таблиц на форме;
  - в качестве первичных ключей необходимо использовать тип `int` с возможностью автоматического увеличения при добавлении данных;
  - информация о схеме данных и данных в таблицах должна сохраняться в файле типа XML или Json.
2. Разработка 2-й версии клиентского приложения MDI WPF или с использованием языка Python:
- в качестве СУБД необходимо использовать любую локальную СУБД;
  - в качестве первичных ключей необходимо использовать тип `int` с возможностью автоматического увеличения при добавлении данных.
  - доступ к СУБД осуществлять либо с использованием технологии ADO.NET или на основе другой технологии подключения к базе данных;
  - предусмотреть возможность добавления, редактирования, удаления и отображения данных из таблиц на форме с возможностью быстрого поиска данных в таблицах по определенному полю.
3. Разработка 3-й версии клиентского приложения MDI WPF или с использованием языка Python:
- в качестве СУБД использовать любую сетевую СУБД;
  - в данной версии должны быть реализованы все функции 2-й версии;
  - в качестве первичных ключей для справочных таблиц необходимо использовать тип `int` с возможностью автоматического увеличения при добавлении данных, для основных таблиц использовать типа GUID;
  - доступ к СУБД осуществлять с использованием технологий Entity Framework Core (C#) или же ей подобной, при использовании языка Python необходимо использовать технологию ORM;
  - выборка данных для формирования отчетов должно осуществляться как минимум с использованием 3 таблиц;
  - один отчет должен сохраняться в формате MS Word, второй в формате MS Excel;
  - запрос данных и формирование отчета производить в отдельном потоке.

**Вариант задания определяется по номеру студента в списке группы.**

*Вариант 1*

«Учёт работ строительной компании»

Информация хранится в базе данных следующей структуры:

- подразделение (код подразделения, название подразделения);
- работник (код работника, ФИО работника, дата рождения, ИНН, № пенсионного страхового свидетельства, код подразделения, паспортные данные);
- справочник работ (код работы, название работы);
- заказчик (код заказчика, наименование, телефон, адрес, ИНН);
- заказ (код заказа, код заказчика, название объекта, содержание работ, дата начала работы, дата окончания работы);
- работа (код заказа, код работы, код работника, дата начала работы, дата окончания работы, описание работы).

### *Вариант 2*

#### «Учёт материалов на складе»

Информация хранится в базе данных следующей структуры:

- материал (код материала, название материала, код категории);
- категория (код категории, название категории, единицы измерения);
- поступление материала (код поступления, код материала, количество, код накладной);
- накладная (код накладной, дата оформления);
- расход материала (код расхода, код материала, количество, код накладной).

### *Вариант 3*

#### «Учёт студентов, проживающих в общежитиях»

Информация хранится в базе данных следующей структуры:

- группа (код группы, название группы, название факультета);
- общежитие (код общежития, название, адрес);
- комната (код комнаты, № комнаты, код общежития);
- студент (код студента, ФИО студента, дата рождения, пол, код группы, серия паспорта, номер паспорта, кем и когда выдан);
- информация о заселении (код комнаты, код студента, дата заселения, дата выселения).

### *Вариант 4*

#### «Учёт работы поликлиники»

Информация хранится в базе данных следующей структуры:

- отделение (код отделения, название отделения);
- должность (код должности, название должности);
- врач (код врача, ФИО врача, дата рождения, код отделения, код должности, пол);
- пациент (код пациента, ФИО пациента, дата рождения, пол, № медицинского полиса, паспортные данные, пол);
- приём у врача (код приёма, код пациента, код врача, дата приёма, время приёма, отчёт о приёме).

### *Вариант 5*

#### «Учёт отпусков сотрудников»

Информация хранится в базе данных следующей структуры:

- отдел (код отдела, название отдела);
- должность (код должности, название должности);
- сотрудник (табельный номер, ФИО сотрудника, дата рождения, ИНН, № пенсионного страхового свидетельства, паспортные данные);
- вид отпуска (код вида отпуска, вид отпуска);
- рабочее место (код работы, код сотрудника, код должности, код отдела, дата начала работы, дата завершения работы);
- отпуск (код отпуска, код вида отпуска, код работы, дата начала отпуска, дата окончания отпуска).

### *Вариант 6*

#### «Учёт арендуемых помещений»

Информация хранится в базе данных следующей структуры:

- здание (код здания, название, адрес);
- помещение (код помещения, название помещения, площадь, код здания);
- арендатор (код арендатора, название фирмы, юридический адрес, ФИО руководителя, контактный телефон);
- аренда (код аренды, код помещения, код арендатора, № договора, дата оформления договора, дата начала аренды, дата окончания аренды).

### *Вариант 7*

#### «Учёт работы автомастерской»

Информация хранится в базе данных следующей структуры:

- вид работ (код вида работ, вид работ);
- модели машин (код модели, название модели);
- автовладелец (код автовладельца, ФИО автовладельца, серия паспорта, номер паспорта, кем и когда выдан);
- автомобиль (код автомобиля, код модели, код автовладельца, номер автомобиля);
- мастер (код мастера, ФИО мастера, паспортные данные, дата рождения);
- работа (код автомобиля, код вида работ, дата начала работ, код мастера, дата окончания работ, описание проделанной работы).

### *Вариант 8*

#### «Учёт работы туристического агентства»

Информация хранится в базе данных следующей структуры:

- туроператор (код туроператора, название туроператора, ФИО контактного лица, контактный телефон, факс, адрес в WWW, e-mail);
- место назначения (код места назначения, название);
- курорт (код курорта, код места назначения, название курорта);
- клиент (код клиента, ФИО клиента, паспорт, контактный телефон);
- тур (код тура, код курорта, начало тура, окончание тура, код клиента, № договора, дата оплаты, стоимость).

### *Вариант 9*

#### «Учёт библиотечного фонда»

Информация хранится в базе данных следующей структуры:

- книга (код книги, наименование, год издания, цена, ISBN, код типа книги);
- тип книги (код типа книги, тип книги);
- поступления (№ экземпляра, код книги, код типа поступления, дата поступления);
- тип поступления (код типа поступления, тип поступления);
- списание (код списания, № экземпляра, дата списания, № акта, причина списания);
- абонемент (код записи в абонементе, № экземпляра, ФИО читателя, дата получения, дата возвращения).

### *Вариант 10*

#### «Учёт материальных ценностей»

Информация хранится в базе данных следующей структуры:

- материально-ответственное лицо (код сотрудника, ФИО сотрудника, паспортные данные, подразделение);
- тип ценности (код типа ценности, тип ценности);
- акт приёма (код акта приёма, № акта, дата оформления);
- акт списания (код акта списания, № акта, дата оформления);
- материальная ценность (код ценности, инвентарный номер, название, стоимость, код акта приёма, код акта списания, код типа ценности, код сотрудника).

### *Вариант 11*

#### «Учёт выступлений студентов на научных конференциях»

Информация хранится в базе данных следующей структуры:

- группа (код группы, название группы);
- студент (код студента, ФИО студента, № зачётки, дата рождения, пол студента, код группы);
- научный руководитель (код сотрудника, ФИО сотрудника, кафедра, должность);
- конференция (код конференции, название, место проведения, дата начала, дата окончания);
- доклад (код доклада, название доклада, дата выступления, код конференции, код студента, код сотрудника).

### *Вариант 12*

#### «Учёт работы научных конференций»

Информация хранится в базе данных следующей структуры:

- конференция (код конференции, название конференции, дата начала, дата окончания);
- руководители секций (код сотрудника, ФИО сотрудника, кафедра, должность);
- секция (код секции, название секции, дата начала работы, дата завершения работы, код конференции, код сотрудника);
- участник (код участника, ФИО участника, место и должность работы, e-mail, контактный телефон, адрес);
- доклад (код доклада, название доклада, дата выступления, код секции, код участника).

### *Вариант 13*

#### «Учёт успеваемости студентов»

Информация хранится в базе данных следующей структуры:

- специальность (код специальности, название специальности);
- учебная группа (код группы, название группы, код специальности);
- студент (код студента, ФИО студента, № зачётки, дата рождения, код группы);
- преподаватель (код преподавателя, табельный №, ФИО преподавателя, кафедра, должность);
- дисциплина (код дисциплины, название дисциплины, количество лекционных часов, количество часов практических занятий, количество часов лабораторных занятий, семестр);
- успеваемость (код записи, код дисциплины, код преподавателя, код студента, форма контроля, оценка, дата сдачи).

### *Вариант 14*

#### «Учёт посещаемости студентов»

Информация хранится в базе данных следующей структуры:

- студент (код студента, ФИО студента, № зачётки, дата рождения, группа);
- вид занятия (код вида занятия, название вида занятия);
- преподаватель (код преподавателя, табельный №, ФИО преподавателя, кафедра, должность);
- дисциплина (код дисциплины, название дисциплины, семестр);
- занятия (код занятия, код дисциплины, код преподавателя, код вида занятия, дата, № пары, тема занятия);
- посещаемость (код занятия, код студента).

### *Вариант 15*

#### «Расписания движения поездов»

Информация хранится в базе данных следующей структуры:

- поезд (код поезда, № поезда, название, код станции отправления, код станции прибытия);
- станция (код станции, название станции);
- расписание поезда (код, код поезда, время в пути, время прибытия, время стоянки, время отправления, код станции);
- состав (код состава, код поезда, номер состава, дата отправления, ФИО начальника поезда).

### *Вариант 16*

#### «Учет домашних животных»

Информация хранится в базе данных следующей структуры:

- домашнее животное (код животного, код породы, кличка, дата рождения, код владельца);
- владельцы (код владельца, ФИО, адрес проживания);
- порода (код породы, название породы);
- прививки (код прививки, название прививки, комментарий);
- прививки выставленные (код, код животного, код прививки, дата).

### *Вариант 17*

#### «Учет банковских операций»

Информация хранится в базе данных следующей структуры:

- счет (код счета, счет, код владельца счета, дата открытия, код банка);
- владельцы (код владельца, ФИО, адрес проживания);
- валюта (код валюты, название валюты, кратность);
- банки (код банка, название банка);
- валютные операции (код, код счета с которого производится перечисление, код счета на который производится перечисление, сумма операции в валюте, код валюты, сумма операции в рублях).

### *Вариант 18*

#### «Учет матчей»

Информация хранится в базе данных следующей структуры:

- матчи (код, код первой команды, код второй команды, дата проведения матча, код стадиона, код турнира);
- команды (код команды, название команды, адрес дислокации);
- стадион (код стадиона, название стадиона, адрес стадиона);
- турниры (код турнира, название турнира, код спонсора, дата начало проведения, дата конца проведения);
- спонсоры (код спонсора, код турнира, название спонсора).

### *Вариант 19*

#### «Поликлиника»

Информация хранится в базе данных следующей структуры:

- прием (код приема, код больного, код назначения, дата приема, диагноз, дата окончания болезни);
- врачи (код врача, ФИО врача, код специализации);
- больные (код больного, ФИО больного, адрес проживания);
- специализации (код специализации, название специализации);
- назначения (код назначения, дата назначения, код лекарства);
- лекарства (код лекарства, код назначения, название лекарства).

### *Вариант 20*

#### «Регистрация автомобилей»

Информация хранится в базе данных следующей структуры:

- регистрация (код регистрации, код владельца, код автомобиля, дата регистрации, номер автомобиля);
- владельцы (код владельца, ФИО, адрес проживания);
- автомобиль (код автомобиля, код марки автомобиля, номер кузова, номер двигателя);
- марка (код марки, название марки, код производителя);
- производитель (код производителя, название производителя, адрес).

Литература:

1. Описание СУБД <https://metanit.com/sql/>
2. Руководство по ADO.NET и работе с базами данных  
<https://metanit.com/sharp/adonet/>
3. Руководство по Entity Framework Core  
<https://metanit.com/sharp/entityframeworkcore/>
4. Руководство по WPF <https://metanit.com/sharp/wpf/>