**Лабораторная работа № 3.** **T-SQL − язык реляционной базы данных**

**Вариант 12**

Цель: изучить работу с операторами базы данных Transact-SQL; составить запрос для создания полноценной базы данных.

Язык реляционной базы данных в **SQL Server** называется **Transact-SQL** (**T-SQL**). Операторы языка делятся на несколько групп, основными из которых является язык определения данных (**Data Definition Language**, **DDL**) и язык манипулирования данными (**Data Manipulation Language**, **DML**).

Язык **DDL** содержит три обобщенных оператора: **create** *объект* (создание объекта базы данных), **alter** *объект* (изменение характеристик объекта) и **drop** *объект* (удаление существующего объекта). Эти опера­торы создают, изменяют и удаляют объекты базы данных, такие как сама база данных, таблицы, столбцы и индексы.

Язык **DML** содержит операторы, которые манипулируют данными, осуществляя выборку информации (**SELECT**), добавление (**INSERT**), удаление (**DELETE**) и изменение (**UPDATE**). При записи операторов можно использовать на клавиатуре любой регистр.

Первое задание заключалось в удалении базы данных из лабораторной работы №2 и создание её с помощью языка Transact-SQL – рисунок 1.1.



Рисунок 1.1 ­–Создание базы данных

Далее требовалось создать необходимые таблицы – рисунок 1.2.

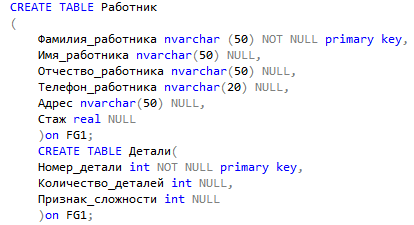


Рисунок 1.2 ­– Пример создания таблицы

При создании таблиц используются различные ограничения. Ограничения, накладываемые на столбцы таблиц баз данных, предотвращают появление данных, не соответствующих предварительно заданным свойствам таблиц. Эти ограничения называются ограничениями целостности. В таблице приведено описание ограничений целостности.

|  |  |
| --- | --- |
| Ограничение | Действие ограничения целостности |
| **data type**  тип данных | Предотвращает появление в столбце значений, не соответствующих типу данных |
| **not null**  запрет значений **null** | Предотвращает появление в столбце значений null |
| **default**  знач. по умолчанию | Устанавливает значение в столбце по умолчанию при выполнении операции INSERT |
| **primary key**  первичный ключ | Предотвращает появление в столбце повторяющихся значений и пустого значения |
| **foreign key**  внешний ключ | Устанавливает связь между таблицей со столбцом, имеющим свойство **foreign key** и таблицей, имеющей столбец со свойством **primary key**; |
| **unique**  уникальное значение | Не допускает пустые и повторяющиеся значения, не может быть использовано для связи с полем другой таблицы |
| **check**  проверка значений | Предотвращает появление в столбце значения, не удовлетворяющего логическому условию |

Для модификации существующих объектов базы данных или сервера применяется оператор ALTER. Добавление столбца в таблицу – рисунок 1.3.



Рисунок 1.3 ­– Работа с ALTER

Оператор INSERT используется для ввода информации в таблицу – рисунок 1.4.

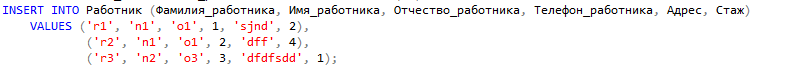


Рисунок 1.3 ­– Пример работы с INSERT

SELECT − основной оператор для поиска информации в базе данных, также существует функция count(\*), которая используется для подсчета количества строк в таблице – рисунок 1.4.



Рисунок 1.4 ­– Пример работы с SELECT

Для изменения строк таблицы используется оператор UPDATE – рисунок 1.6.



Рисунок 1.6 – Работа с UPDATE

БД представляет собой набор файлов операционной системы трех типов: первичный файл (расширение mdf), вторичные файлы (ndf) и файлы журнала транзакций (log). Все файлы БД, кроме файлов журнала транзакций, распределены по файловым группам. Файловые группы – это поименованный набор файлов БД – рисунок 1.7.

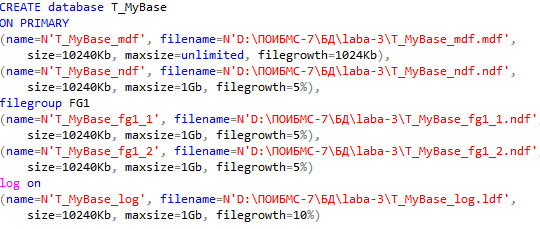


Рисунок 1.7 – Работа с набором файлов базы данных

**Вывод**

В ходе лабораторной работы были изучены операторы баз данных Transact-SQL. Получены знания по составлению запросов на языке Transact-SQL. Составлен запрос для создания базы данных. Отработаны навыки взаимодействия с наборами файловой системы баз данных.