## Учреждение образования

## Белорусский государственный технологический университет

Кафедра полиграфического оборудования и

системы обработки информации

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 9**

по дисциплине «Стандартизация и сертификация

информационных систем и технологий»

Тема

«Ознакомление со стандартом SQL»

Выполнил студент:

Миневич Кристина

ф-та Информационных технологий, специальности «Программное обеспечение информационных технологий» 4 к. 4 гр.

Проверил:

кандидат технических наук

Сулим П.Е.

Отчет по лабораторной работе

защищен с отметкой баллов

Минск 2024

**Лабораторная работа № 9**

**ОЗНАКОМЛЕНИЕ СО СТАНДАРТОМ SQL**

1. **Стандарт *SQL***

Источники:

[https://ru.wikipedia.org/wiki/SQL](https://ru.wikipedia.org/wiki/SQL#Стандартизация)

<https://learnsql.com/blog/history-of-sql-standards/>

<https://docs.oracle.com/en/database/oracle/oracle-database/21/sqlrf/SQL-Standards.html#GUID-BCCCFF75-D2A4-43AD-8CAF-C3C97D92AC63>

**SQL** («язык структурированных запросов») — [декларативный](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BA%D0%BB%D0%B0%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F), применяемый для создания, модификации и управления данными в [реляционной базе данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляемой соответствующей [системой управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).

Является, прежде всего, [информационно-логическим языком](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA), предназначенным для описания, изменения и извлечения данных, хранимых в [реляционных базах данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). В общем случае SQL (без ряда современных расширений) считается [языком программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) не [полным по Тьюрингу](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D0%BD%D0%BE%D1%82%D0%B0_%D0%BF%D0%BE_%D0%A2%D1%8C%D1%8E%D1%80%D0%B8%D0%BD%D0%B3%D1%83), но вместе с тем стандарт языка спецификацией [SQL/PSM](https://ru.wikipedia.org/wiki/SQL/PSM) предусматривает возможность его [процедурных расширений](https://ru.wikipedia.org/wiki/SQL#%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D0%B4%D1%83%D1%80%D0%BD%D1%8B%D0%B5_%D1%80%D0%B0%D1%81%D1%88%D0%B8%D1%80%D0%B5%D0%BD%D0%B8%D1%8F).

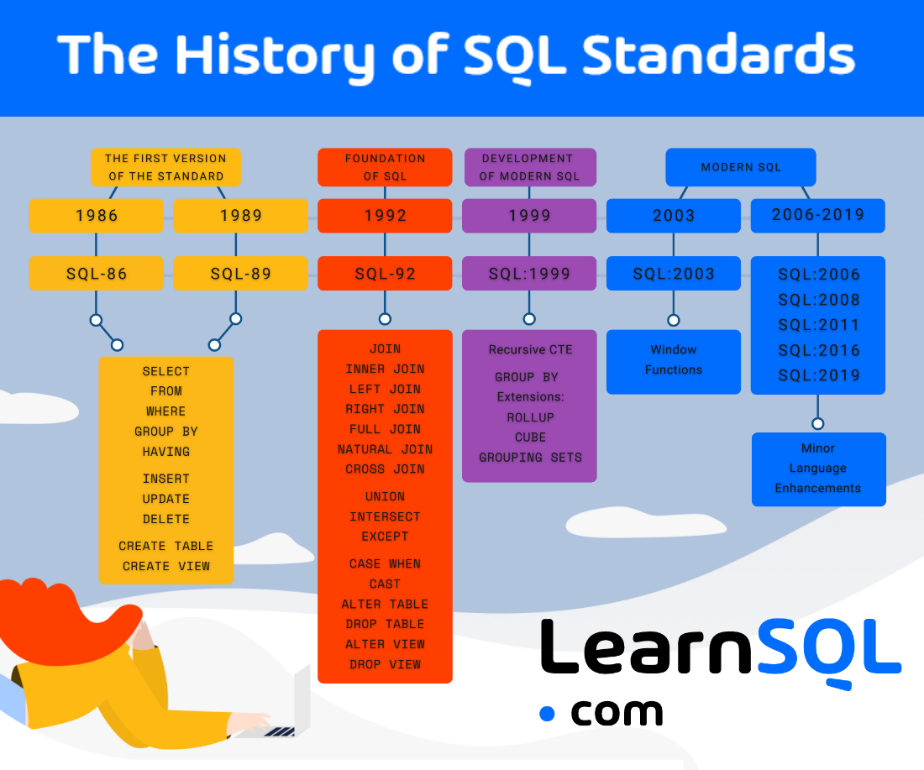
Изначально SQL был основным способом работы пользователя с [базой данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) и позволял выполнять следующий набор операций:

* создание в базе данных новой таблицы;
* добавление в таблицу новых записей;
* изменение записей;
* удаление записей;
* выборка записей из одной или нескольких таблиц (в соответствии с заданным условием);
* изменение структур таблиц.

Со временем SQL усложнился — обогатился новыми конструкциями, обеспечил возможность описания и управления новыми хранимыми объектами (например, индексы, представления, триггеры и хранимые процедуры) — и стал приобретать черты, свойственные языкам программирования.

SQL был [создан в начале 1970](https://learnsql.com/blog/history-of-sql/)-х годов компанией IBM. В 1974 году Дональд Чемберлин и Рэймонд Бойс опубликовали статью [SEQUEL: A Structured English Query Language](https://dl.acm.org/doi/10.1145/800296.811515), которая представила миру SQL. Статья содержала [грамматику BNF](https://en.wikipedia.org/wiki/Backus%E2%80%93Naur_form) (подробное техническое описание синтаксиса языка) для языка, который IBM позже сочла ошибкой. Публикация грамматики позволила различным поставщикам реализовать SQL в своих собственных движках баз данных. Тем не менее, было довольно много различий между реализациями SQL поставщиков, поскольку каждый поставщик представил свои собственные расширения для SQL. К середине 1980-х годов между различными версиями SQL возникла существенная несовместимость.

Также в 1980-х годах Американский национальный институт стандартов (ANSI) начал работать над стандартом SQL. Цель введения стандарта состояла в том, чтобы обеспечить переносимость SQL-приложений между соответствующими продуктами.



Первым стандартом SQL был **SQL-86**. Он был опубликован в 1986 году как стандарт ANSI и в 1987 году как стандарт Международной организации по стандартизации (ISO). Отправной точкой для стандарта ISO была реализация стандарта SQL IBM. Эта версия стандарта SQL также известна как **SQL 1**.

Стандарт SQL-86 составляет 120 страниц. Он содержит синтаксис операторов, SELECT, INSERT, и UPDATE DELETE. Синтаксис позволяет фильтровать с SELECT помощью, использовать WHERE с несколькими столбцами (но не по выражениям) и фильтровать группы с GROUP BY помощью. Там нет явного HAVING синтаксиса; вы можете перечислить несколько таблиц в предложении, чтобы получить по существу перекрестное соединение, которое вы можете позже фильтровать JOIN. Нет никакого способа сделать FROM с этим синтаксисом. Однако WHERE синтаксис допускает подзапросы. OUTER JOIN SELECT

Стандарт также имеет синтаксис определения схемы: вы можете создавать таблицы (UNIQUE только с NOT NULL ограничениями и), создавать представления и предоставлять привилегии. Типы данных, введенные в стандарте, - это символьные строки и числа (NUMERIC, DECIMAL, INTEGER, SMALLINT, FLOAT и REAL). Нет никаких DOUBLE PRECISION утверждений или ALTER. Существуют также встраивания SQL в различные языки программирования (Cobol, Fortran, Pascal и PL/1). DROP REVOKE

Как правило, вы можете купить стандартные документы от ISO или ANSI, даже старые и отозванные версии. Однако, как делится Эван Кэрролл в своем посте Stack Overflow, в настоящее время нелегко заполучить стандарт SQL-86.

Следующим стандартом SQL был **SQL-89**, опубликованный в 1989 году. Это был незначительный пересмотр более раннего стандарта, надмножество SQL-86, которое заменило SQL-86. Размер стандарта не изменился.

Наиболее важными новыми функциями были ограничения целостности: в дополнение к UNIQUE NOT NULL DEFAULT уже доступным параметрам, и check в представлениях были добавлены первичные ключи, внешние ключи CHECK и ограничения and. Стандарт также добавил новые языковые привязки для C и Ada.

Следующей редакцией стандарта был **SQL-92** – и это была серьезная редакция. Язык, введенный SQL-92, иногда называют SQL 2**. Стандартный документ вырос со 120 до 579 страниц. Тем не менее, большая часть роста была обусловлена более точными спецификациями существующих функций.**

Наиболее важными новыми функциями были:

* Явный JOIN синтаксис и введение внешних соединений: LEFT JOIN, RIGHT JOIN, FULL JOIN.
* Введение NATURAL JOIN и CROSS JOIN
* Введение операций множеств (объединение множеств, пересечение множеств и разность множеств).
* Введение условного выражения CASE WHEN.
* Новые скалярные операции: конкатенация строк, извлечение подстрок, математика даты и времени.
* CAST Оператор, который позволяет явное приведение значений в типы.

**SQL:1999** (также называемый **SQL 3**) был четвертой редакцией стандарта SQL. Начиная с этой версии в названии стандарта вместо дефиса использовалось двоеточие, чтобы соответствовать названиям других стандартов ISO. Этот стандарт был опубликован несколькими частями в период с 1999 по 2002 год.

Первая часть SQL:1999 состояла из пяти частей:

* Часть 1: SQL/Framework (100 страниц) определила основные понятия SQL.
* Часть 2: SQL/Foundation (1050 страниц) определяет основной синтаксис и операции SQL: типы, схемы, таблицы, представления, операторы запроса и обновления, выражения и так далее. Эта часть является наиболее важной для обычных пользователей SQL.
* Часть 3: SQL/CLI (интерфейс уровня вызова) (514 страниц) определил интерфейс прикладного программирования для SQL.
* Часть 4: SQL/PSM (Persistent Stored Modules) (193 страницы) определенные расширения, которые делают SQL процедурным.
* Часть 5: SQL/Привязки (270 страниц) определены методы встраивания операторов SQL в прикладные программы, написанные на стандартном языке программирования.

1. **Диалекты SQL. DSL**

Источники:

<https://dou.ua/lenta/articles/to-make-things-easy-dsl/>

<https://www.codeproject.com/Articles/112006/DSL-and-SQL-Server-Modeling-CTP>

<https://info-comp.ru/programmirovanie/749-what-is-sql.html>

<https://ru.wikipedia.org/wiki/SQL>

Язык **SQL** – это стандарт, он реализован во всех реляционных базах данных, но у каждой СУБД есть расширение этого стандарта, есть собственный язык работы с данными, его обычно называют диалектом SQL, который, конечно же, основан на SQL, но предоставляет больше возможностей для полноценного программирования, кроме того, такой внутренний язык дает возможность получать системную информацию и упрощать SQL запросы.

Вот некоторые диалекты языка SQL:

* **Transact-SQL** (сокращенно [T-SQL](https://self-learning.ru/what-is-t-sql)) – используется в Microsoft SQL Server;
* **PL/SQL** (Procedural Language / Structured Query Language) – используется в Oracle Database;
* **PL/pgSQL** (Procedural Language/PostGres Structured Query Language) – используется в PostgreSQL.

**Domain Specific Language**, или язык предметной области, — это язык, созданный для конкретной области применения. Построение его, или структуры данных, отражают специфику решаемых им задач.

В чем преимущество предметно-ориентированных языков? Зачем они нужны? Использование DSL несет в себе целый ряд преимуществ. Самое очевидное из них заключается в том, что использование языка и движка трансформации позволит вам значительно повысить эффективность определенного этапа процесса разработки, поскольку вам больше не придется вручную выполнять утомительную работу.

В отличие от интерпретации, генерация исходного кода из программ, написанных на DSL, позволяет использовать специализированные абстракции без каких-либо накладных расходов при выполнении кода, поскольку генератор, подобно компилятору, способен удалять абстракции и генерировать высокопроизводительный код. Благодаря тому, что вы сможете выражать понятия из нужной вам области на языке, который тесно с ней связан, продумывать логику станет легче, поскольку код будет свободен от подробностей реализации. Иными словами, использование DSL позволит вам отделить главное от второстепенных деталей.

Благодаря тому, что нотации, набор абстракций и область применения предметно-ориентированных языков максимально соответствуют соглашениям, принятым среди экспертов в соответствующей области

(т.е. непрограммистов), такие языки способствуют тесной интеграции между разработчиками и специалистами других направленностей.

Использование предметно-ориентированных языков в совокупности с механизмом выполнения позволяет сделать логику приложения, написанного на DSL-коде, независимой от целевой платформы. Использование DSL также позволяет повысить качество создаваемого продукта: снизить число ошибок, обеспечить более точное соответствие архитектуры техническому заданию и повысить легкость сопровождения. Это достигается благодаря упразднению «лишних» степеней свободы и дублирующегося кода, а также автоматизации однообразной работы.

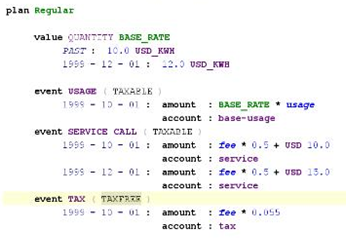
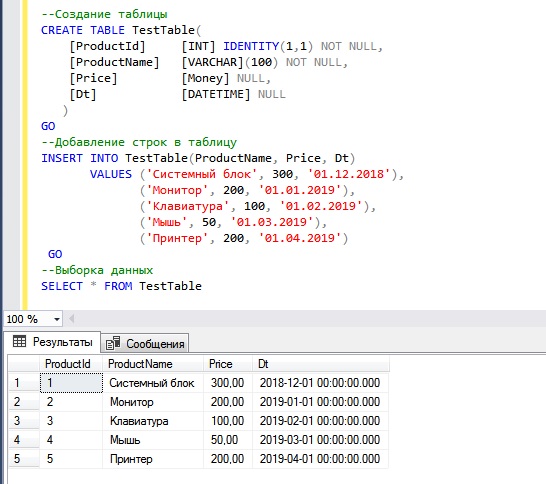
 

Рисунок 4 – Пример DSL Рисунок 5 – Пример SQL

**Вывод:**

В ходе лабораторной работы, был изучен SQL (structured query language – «язык структурированных запросов») – декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных, и его диалекты.