Учреждение образования

Белорусский государственный технологический университет

Кафедра полиграфического оборудования и

системы обработки информации

**Отчет по лабораторной работе**

«Ознакомление со стандартом SQL»

по дисциплине «Стандартизация и сертификация

информационных систем и технологий»

Выполнил студент

ФИТ ПОИТ 5-2 Валдайцев А. Д.

Проверил

кандидат технических наук

Сулим П.Е.

Отчет по лабораторной работе

защищен с отметкой баллов

Минск 2024

**Стандарт *SQL*.**

**Диалекты *SQL. DSL*.**

## Стандарт SQL

SQL (Structured Query Language) является непременным инструментом в мире реляционных баз данных, обеспечивая декларативный подход к работе с данными. Этот язык предоставляет элегантные средства для создания, изменения и управления информацией в базах данных. В своей основе SQL предназначен для описания, изменения и извлечения данных, лежащих в основе реляционной модели.

Изначально, SQL предоставлял базовый, но мощный набор операций, включающий создание таблиц, добавление и изменение записей, выборку данных из одной или нескольких таблиц и изменение структуры таблиц. В течение времени, SQL стал более сложным и разнообразным, обогащаясь новыми конструкциями и возможностями. Такие дополнения как индексы, представления, триггеры и хранимые процедуры позволили SQL приобрести черты, свойственные языкам программирования.

## SQL1

Когда на рынке существовало множество различных вариантов СУБД с собственными реализациями языка запросов, стало очевидным необходимость в стандартизации. Так родился первый стандарт SQL, изначально известный как **SQL1**, который был принят в **1989** году ANSI. Самая первая версия называлась SQL/86, но она была крайне сырой, а SQL1 считается версия SQL/89.

Здесь были определены основные операции синтаксиса SQL. У этого языка имеется масса недостатков: многие важные понятия не определены, много отдано на откуп реализациям. В этом стандарте полностью отсутствуют такие важные разделы, как манипулирование схемой БД и динамический SQL.

## SQL2

Этот стандарт, хоть и был значительным шагом вперед, был дополнен и расширен в следующие годы, в частности в **1992** году, когда был выпущен SQL92, также известный как **SQL2**.

SQL2 охватывает практически все аспекты, необходимые для реализации приложений: манипулирование схемой БД, управление транзакциями и сессиями (сессия – это последовательность транзакций, в пределах которой сохраняются временные отношения), подключения к БД, динамический SQL, отношения-каталоги БД.

## SQL/CLI

После принятия стандарта SQL92 были добавлены несколько дополнительных стандартов, которые расширили функциональность языка. В 1995 году был принят стандарт **SQL/CLI** (Call Level Interface), позже переименованный в CLI95.

SQL/CLI представляет собой набор спецификаций интерфейсов процедур, вызовы которых позволяют выполнять динамически задаваемые операторы SQL. По сути, представляет собой замену динамическому SQL.

Стандарт SQL/CLI послужил основой для создания повсеместно распространенных сегодня интерфейсов ODBC (Open Database Connectivity) и JDBC (Java Database Connectivity).

## SQL/PSM

Также на следующий год 1996 был принят стандарт **SQL/PSM** (Persistent Stored Modules), который стал известен как PSM-96. Основная цель этой спецификации – стандартизировать способы определения и использования хранимых процедур, то есть специальным образом оформленных программ, включающих операторы SQL, которые сохраняются в базе данных, могут вызываться приложениями и выполняются внутри СУБД.

## SQL3

Следующим этапом стал стандарт SQL:1999 (**SQL3**).

Хотя существует международный стандарт ANSI SQL, многие компании, занимающиеся разработкой СУБД, вносят изменения в язык SQL.

Вводится система типов языка, формулируются правила определения функциональных зависимостей и возможных ключей, определяются синтаксис и семантика основных операторов SQL. Добалвяет поддержка регулярных выражений, рекурсивных запросов и триггеров.

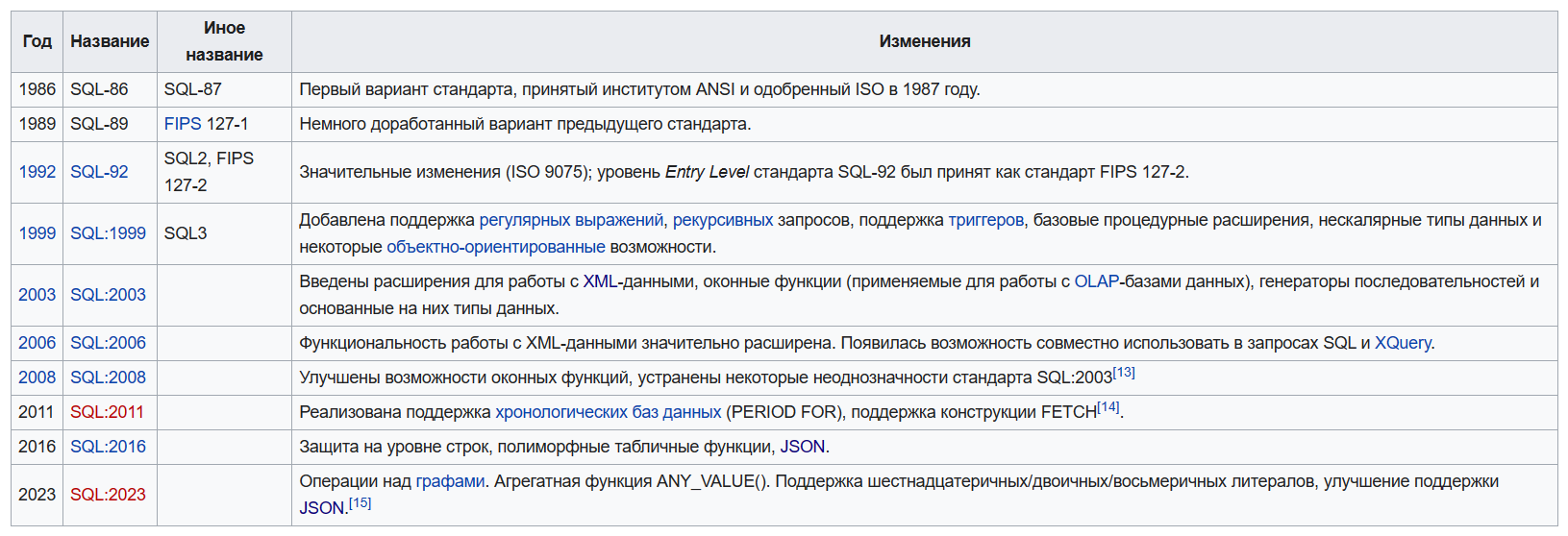


Рисунок 1.1 – Нововведения стандартов SQL

## Диалекты SQL

Хотя существует международный стандарт ANSI SQL, многие компании, занимающиеся разработкой СУБД, вносят изменения в язык SQL. Каждая реализация языка SQL в конкретной СУБД называется **диалектом** – подмножеством стандарта SQL. Функции, добавленные разработчиками коммерческих реализаций, называются расширениями.

Производители СУБД, такие как Oracle, Microsoft, Borland, Informix, Sybase, применяют реализации SQL, соответствующие как минимум начальному уровню и содержащие некоторые расширения, специфические для данной СУБД.

*Не существует двух совершенно идентичных диалектов.* Разработчики постоянно расширяют свои диалекты, что делает их более различными.

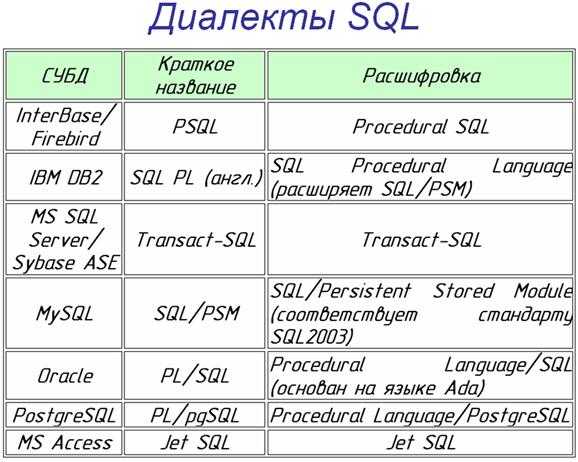


Рисунок 1.2 – Диалекты SQL

## DSL

Предметно-ориентированные языки (DSL) представляют собой языки программирования, которые обладают более высоким уровнем абстракции и специализируются на конкретных предметных областях, отражая их специфику и решая соответствующие задачи. Отличительной чертой DSL является то, что они разрабатываются совместно с экспертами в соответствующих областях, и их использование не требует высокой квалификации разработчика.

Преимущества предметно-ориентированных языков являются очевидными и значительными. Они позволяют существенно повысить эффективность процесса разработки, поскольку устраняют необходимость ручного выполнения трудоемкой работы. В отличие от интерпретации, генерация исходного кода на DSL позволяет использовать специализированные абстракции без дополнительных накладных расходов при выполнении кода. Это обеспечивает генерацию высокопроизводительного кода, так как генераторы, подобно компиляторам, могут удалять абстракции.

Использование предметно-ориентированных языков также облегчает процесс разработки, поскольку позволяет выражать понятия из конкретной предметной области на тесно связанном с ней языке, свободном от деталей реализации. Это позволяет разработчикам сосредоточиться на главных аспектах задачи, отделив их от второстепенных деталей.

DSL способствуют интеграции между разработчиками и специалистами других направлений, поскольку нотации, абстракции и область применения DSL соответствуют соглашениям, принятым экспертами в соответствующей области. Таким образом, DSL способствуют более тесному сотрудничеству и обмену знаниями между участниками проекта.

Использование предметно-ориентированных языков в сочетании с механизмом выполнения позволяет создавать приложения, независимые от конкретной платформы. Также использование DSL способствует повышению качества разрабатываемого продукта, снижает вероятность ошибок, обеспечивает более точное соответствие архитектуры техническому заданию и упрощает сопровождение кода. Это достигается путем устранения излишних степеней свободы и дублирующегося кода, а также автоматизации рутинных задач.

**Примеры DSL-языков:**

* [TeX/LaTeX](https://ru.wikipedia.org/wiki/TeX) для подготовки ([компьютерной вёрстки](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%B2%D1%91%D1%80%D1%81%D1%82%D0%BA%D0%B0)) текстовых документов;
* [Perl](https://ru.wikipedia.org/wiki/Perl) для манипулирования текстами;
* [SQL](https://ru.wikipedia.org/wiki/SQL) для [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%A3%D0%91%D0%94);
* [Tcl/Tk](https://ru.wikipedia.org/wiki/Tcl/Tk) для [графического интерфейса пользователя](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F);
* [HTML](https://ru.wikipedia.org/wiki/HTML) и [SGML](https://ru.wikipedia.org/wiki/SGML) для разметки документов;
* [Verilog](https://ru.wikipedia.org/wiki/Verilog) и [VHDL](https://ru.wikipedia.org/wiki/VHDL) для описания аппаратного обеспечения;
* [Mathematica](https://ru.wikipedia.org/wiki/Mathematica) и [Maple](https://ru.wikipedia.org/wiki/Maple) для символьных вычислений;
* [AutoLisp](https://ru.wikipedia.org/wiki/AutoLisp) для компьютерного моделирования ([САПР](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%B0%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F));
* [Prolog](https://ru.wikipedia.org/wiki/Prolog) для задач, сформулированных в терминах [исчисления предикатов](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D1%80%D0%B5%D0%B4%D0%B8%D0%BA%D0%B0%D1%82%D0%BE%D0%B2);
* [ML](https://ru.wikipedia.org/wiki/ML) и [Haskell](https://ru.wikipedia.org/wiki/Haskell" \o "Haskell) для задач, сформулированных в терминах функций.

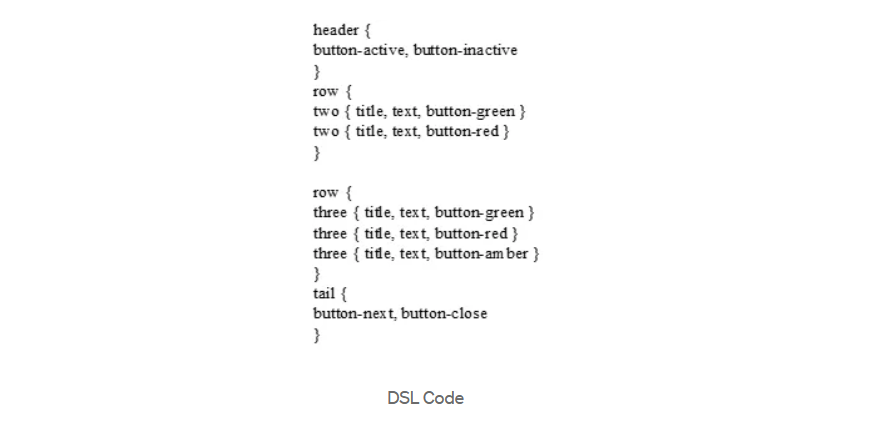


Рисунок 1.3 – Пример DSL-кода

## Источники

[SQL — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/SQL)

[The History of SQL Standards (learnsql.com)](https://learnsql.com/blog/history-of-sql-standards/)

[История языков программирования: SQL- стандартизация длиною в жизнь / Хабр (habr.com)](https://habr.com/ru/articles/311288/)

[ANSI SQL 101 — Understanding the Syntax and Concepts | by Alex Merced - Tech Evangelist | Feb, 2024 | Medium](https://alexmercedcoder.medium.com/ansi-sql-101-understanding-the-syntax-and-concepts-9eeaf2cbbea5)

[s3.amazonaws.com/artifacts.opencypher.org/website/materials/SQLStandards\_2017-10-20](https://s3.amazonaws.com/artifacts.opencypher.org/website/materials/SQLStandards_2017-10-20.pdf)

[Предметно-ориентированный язык — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D0%B4%D0%BC%D0%B5%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA#:~:text=%D0%9F%D1%80%D0%B5%D0%B4%D0%BC%D0%B5%D1%82%D0%BD%D0%BE%2D%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B9%20%D1%8F%D0%B7%D1%8B%D0%BA%20(%D0%B0%D0%BD%D0%B3%D0%BB.%20domain%2Dspecific%20language%2C,%D1%83%D1%87%D0%B8%D1%82%D1%8B%D0%B2%D0%B0%D1%8E%D1%89%D0%B5%D0%BC%D1%83%20%D0%BE%D1%81%D0%BE%D0%B1%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D0%B8%20%D0%BA%D0%BE%D0%BD%D0%BA%D1%80%D0%B5%D1%82%D0%BD%D1%8B%D1%85%20%D1%81%D1%84%D0%B5%D1%80%20%D0%B7%D0%BD%D0%B0%D0%BD%D0%B8%D0%B9))

## Выводы

В ходе выполнения лабораторной работы были рассмотрены различные стандарты языка SQL. Каждый из этих стандартов имеет свои особенности и преимущества, которые могут быть полезны при работе с базами данных.

Каждый следующий стандарт SQL представляет собой расширение и улучшение предыдущего. Новые версии стандартов вносят дополнительные функциональные возможности и расширяют поддержку языка SQL для более эффективной и удобной работы с данными. Также стоит отметить, что при переходе на последние версии стандартов SQL, разработчики имеют доступ к более современным и мощным инструментам для работы с базами данных.

Стандарты позволяют эффективно использовать возможности языка SQL и оптимизировать работу с данными. Кроме того, они обеспечивают совместимость и переносимость кода между различными базами данных.