

# Базы данных

## А3. Язык SQL



Московский государственный технический университет  
имени Н.Э. Баумана

**Факультет ИБМ**

Фев 2025 года

Москва

Артемьев Валерий Иванович © 2025

# Возможные интерфейсы к СУБД

## ■ Интерфейсы разных пользователей к СУБД

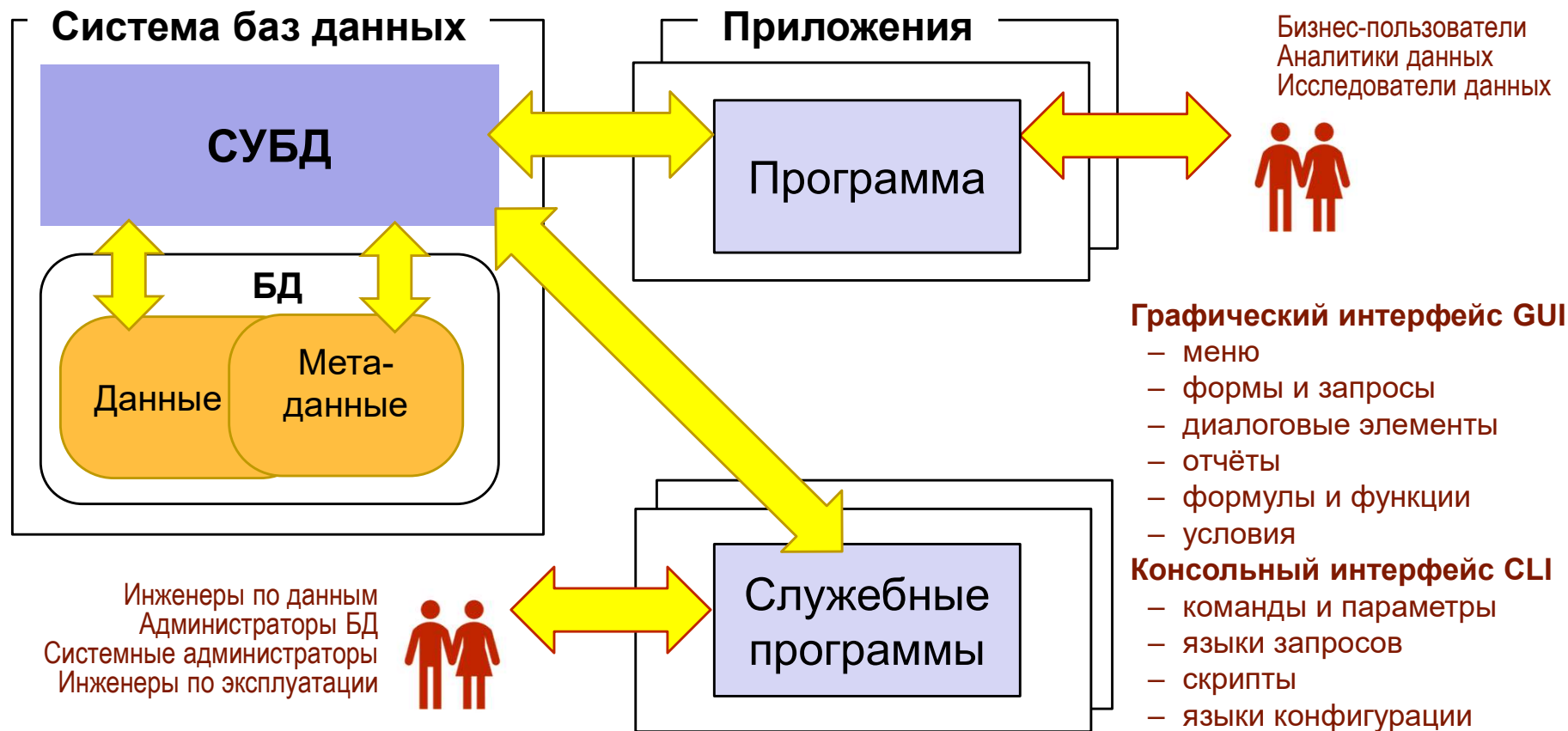
Опосредованное взаимодействие с СУБД бизнес-пользователя через приложение, а инженера данных, аналитика и исследователя данных через инструменты анализа данных.

Опосредованное взаимодействие с СУБД архитекторов и модельеров данных, разработчиков БД и приложений, администраторов БД и системных администраторов через инструментальные среды, системные утилиты и языки программирования.

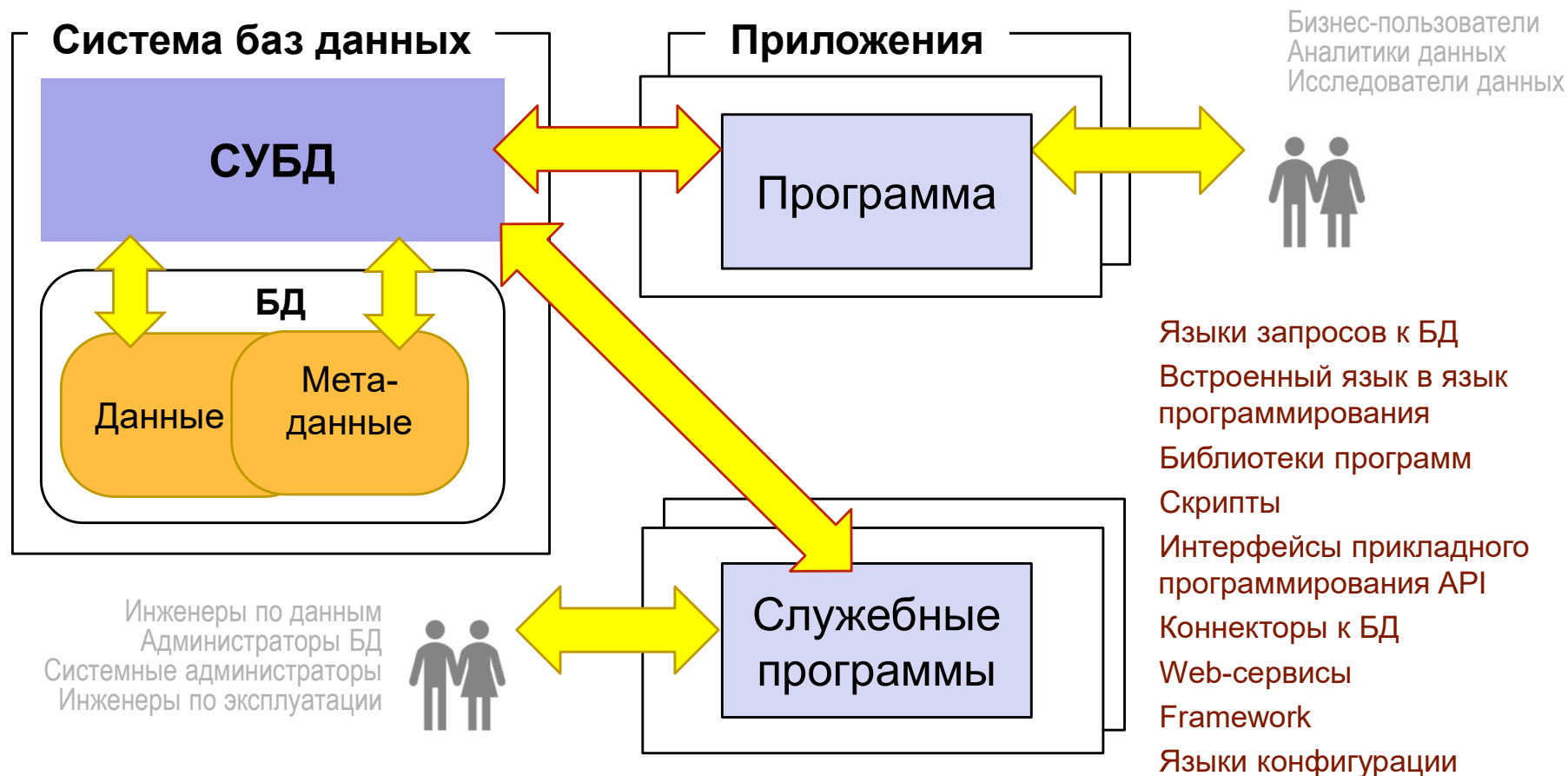
## ■ Программные интерфейсы к СУБД

Большое разнообразие интерфейсов, где часто применяются языки запросов, например, SQL для реляционных СУБД, в окружении языков и библиотек программирования.

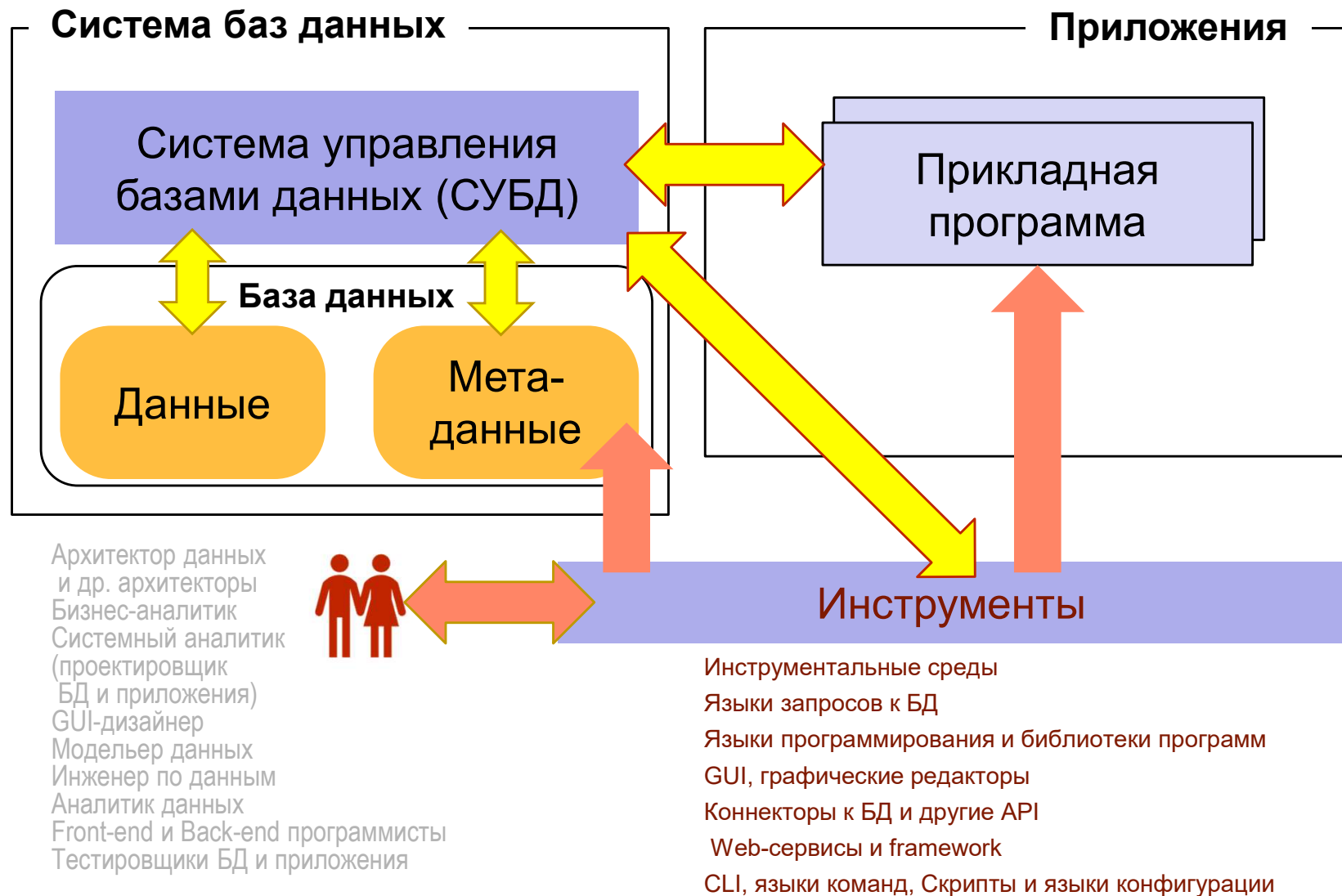
# Интерфейсы разных категорий пользователей к СУБД



# Программные интерфейсы к СУБД



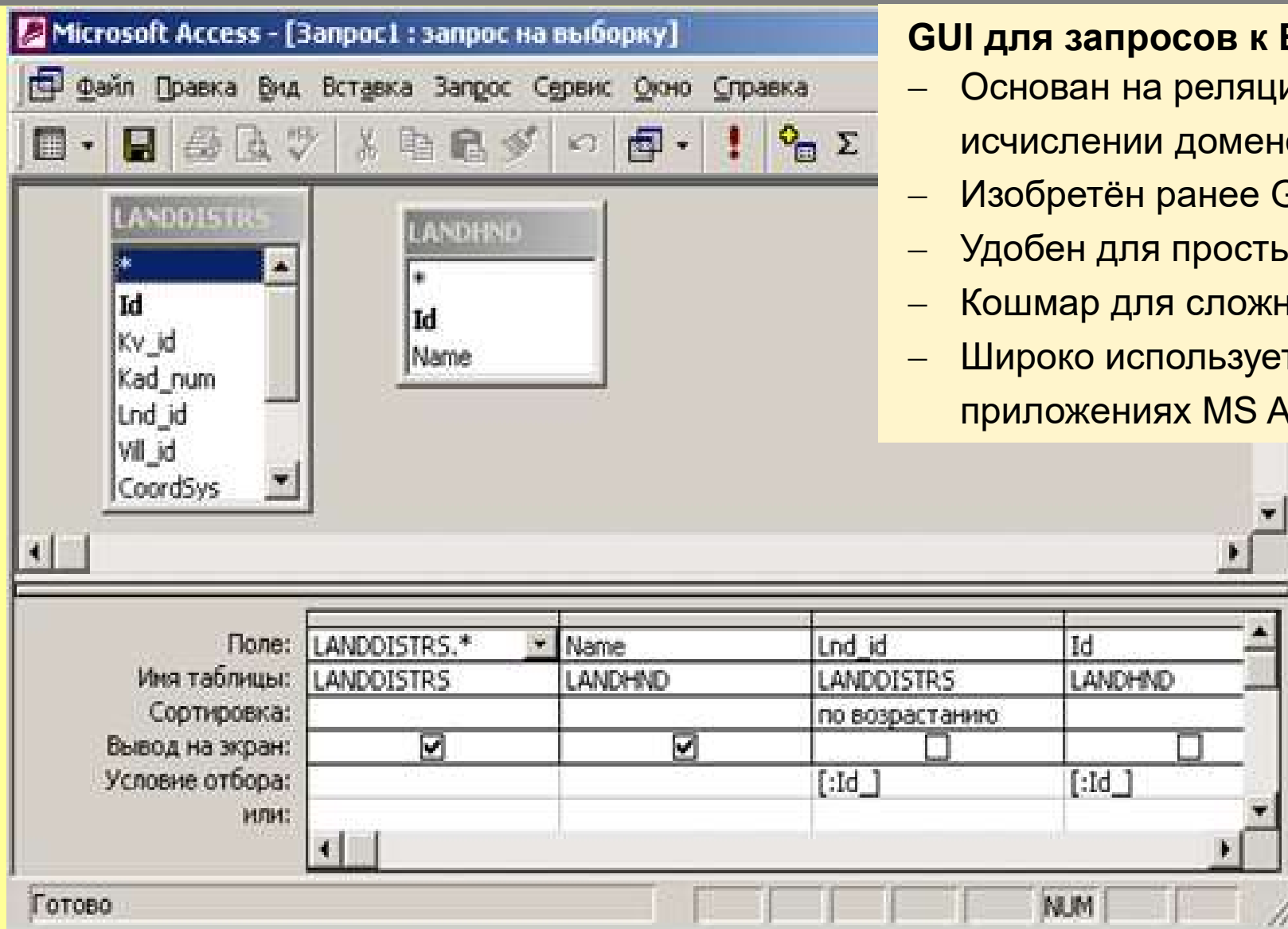
# Интерфейсы разработчиков БД и приложения



# Языки для работы с БД

- **Графический язык QBE (Query By Example)**  
Графический интерфейс для запросов
- **Язык для работы с реляционными БД SQL**  
Реляционная модель данных
- **Язык анализа данных DAX (Data Analysis eXpressions)**  
Многомерная, реляционная и колоночная модели данных
- **Клоны языка SQL для БД NoSQL**
- **Языки программирования триггеров и хранимых процедур**  
Oracle PL/SQL, MS T-SQL, pgPL/SQL, Java
- **Язык запросов к многомерным БД MDX (Multi Dimensional eXpressions)**  
Многомерная модель данных
- **Специальные языки запросов для БД NoSQL**

# Язык QBE запросов по образцу



## GUI для запросов к БД

- Основан на реляционном исчислении доменов DRC
- Изобретён ранее GUI
- Удобен для простых запросов
- Кошмар для сложных запросов
- Широко используется в офисных приложениях MS Access, MS Excel

# Примеры условий запроса в MS Access

## **>25 and <50**

Применяется к числовому полю, например «Цена», для выбора записей, в которых поле содержит значение больше 25 и меньше 50.

## **DateDiff ("гггг", [ДатаРождения], Date()) > 30**

Условие для поля «ДатаРождения» для выбора записей, в которых разность лет между датой рождения человека и текущей датой (возраст) больше 30.

## **IS NULL**

Условие выбора записей, в которых значение указанного поля равно NULL.

## **"Китай"**

Возвращает записи, в которых поле «Страна» имеет значение «Китай».

## **NOT "Мексика"**

Возвращает записи, в которых значение поля «Страна» не является «Мексика».

## **LIKE "С\*"**

Возвращает записи, в которых значение поля начинаются с буквы «С».

## **LIKE "Корея"**

Возвращает записи, в которых поле содержит строку «Корея».



# Язык DAX для многомерных запросов

Функциональный предметно-ориентированный язык для многомерных запросов (*формульный язык a la Excel*), создан Microsoft взамен языка MDX.

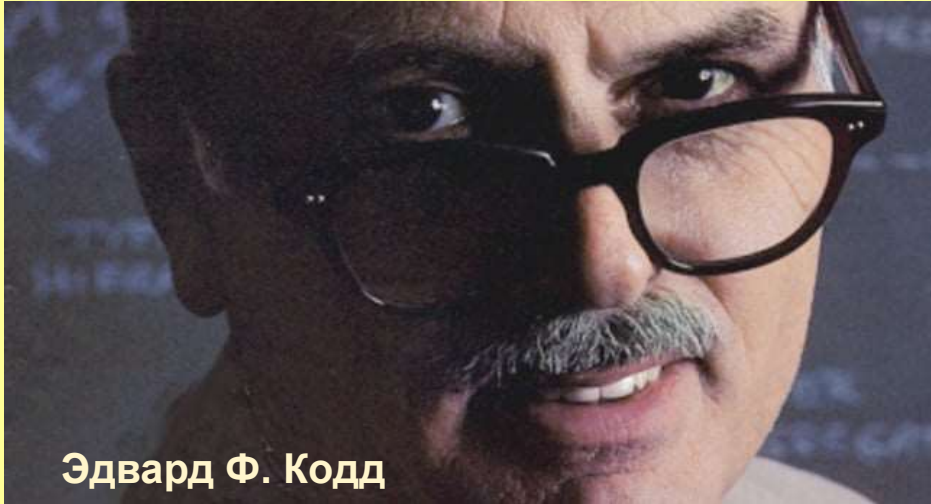
*Реляционная, многомерная и колоночная модели данных.* Относится к средствам *бизнес-аналитики*, ориентирован на обработку данных в памяти (*in memory*).

Широко применяется в продуктах MS Excel/PowerPivot, MS Power BI, MS Analysis Services, MS SQL Server, MS Azure.

Использование DAX свободно от лицензионных отчислений.

Познакомиться с языком можно в MS Excel (с версии 13) и в свободной версии MS Power BI Desktop.

# История появления и развития SQL



**Эдвард Ф. Кодд**

Дональд Д. Чемберлин и Раймонд Ф. Бойс



**1970:** Публикация работы Эдгара Ф. Кодда "A Relational Model of Data for Large Shared Data Banks", которая заложила основы реляционной модели данных.

Реляционная алгебра и язык DSL/Alpha

**1974:** Дональд Д. Чемберлин и Рэймонд Ф. Бойс из IBM разработали первый прототип языка SEQUEL.

**1979:** Oracle выпустила первую коммерческую версию SQL, что стало важным шагом в распространении языка.

**1986:** ANSI и ISO приняли первый стандарт SQL, что способствовало унификации и стандартизации языка.

# Общая характеристика языка SQL

Язык SQL – *высокоуровневый языковой интерфейс* обращения программ к реляционным СУБД для выполнения любых действий с базами данных. Запрашиваемые действия относятся к БД, её объектам, полям и параметрам.

Считается *непроцедурным языком*, а по виду *процедурно-декларативный*

*Глагол в повелительном наклонении – процедурная составляющая во всех операторах SQL. декларативный характер имеют операторы созданию и изменению таблиц и представлений на основе ФМД.*

**Большинство не считает SQL языком программирования**

*Нет явных средств управления порядком выполнения (циклов и условных операторов).  
Обработка множества строк, а не отдельных строк и полей данных.  
SQL является подязыком для языков программирования.*

**Обработка данных основана на теории отношений и реляционной алгебре**

*Исходными и результирующими объектами обработки являются отношения (таблицы и представления), т.е. множество кортежей (строк) данных. Для отбора данных выполняется ассоциативная фильтрация без явного использования навигационных указателей.*

# Синтаксис языка SQL

Близок к естественному английскому языку  
Оператор SQL обязателен в любом обращении к СУБД  
Немного видов операторов, что несколько усложняет синтаксис  
и восприятие перегруженного оператора SELECT

---

В начале ключевое слово, определяющее вид оператора  
Позиционно-ключевая форма параметров  
Параметры: имена объектов и полей БД, выражения с встроенными функциями, константы и позиции подстановки значений переменных  
Допустима вложенность операторов  
Точка с запятой в конце оператора

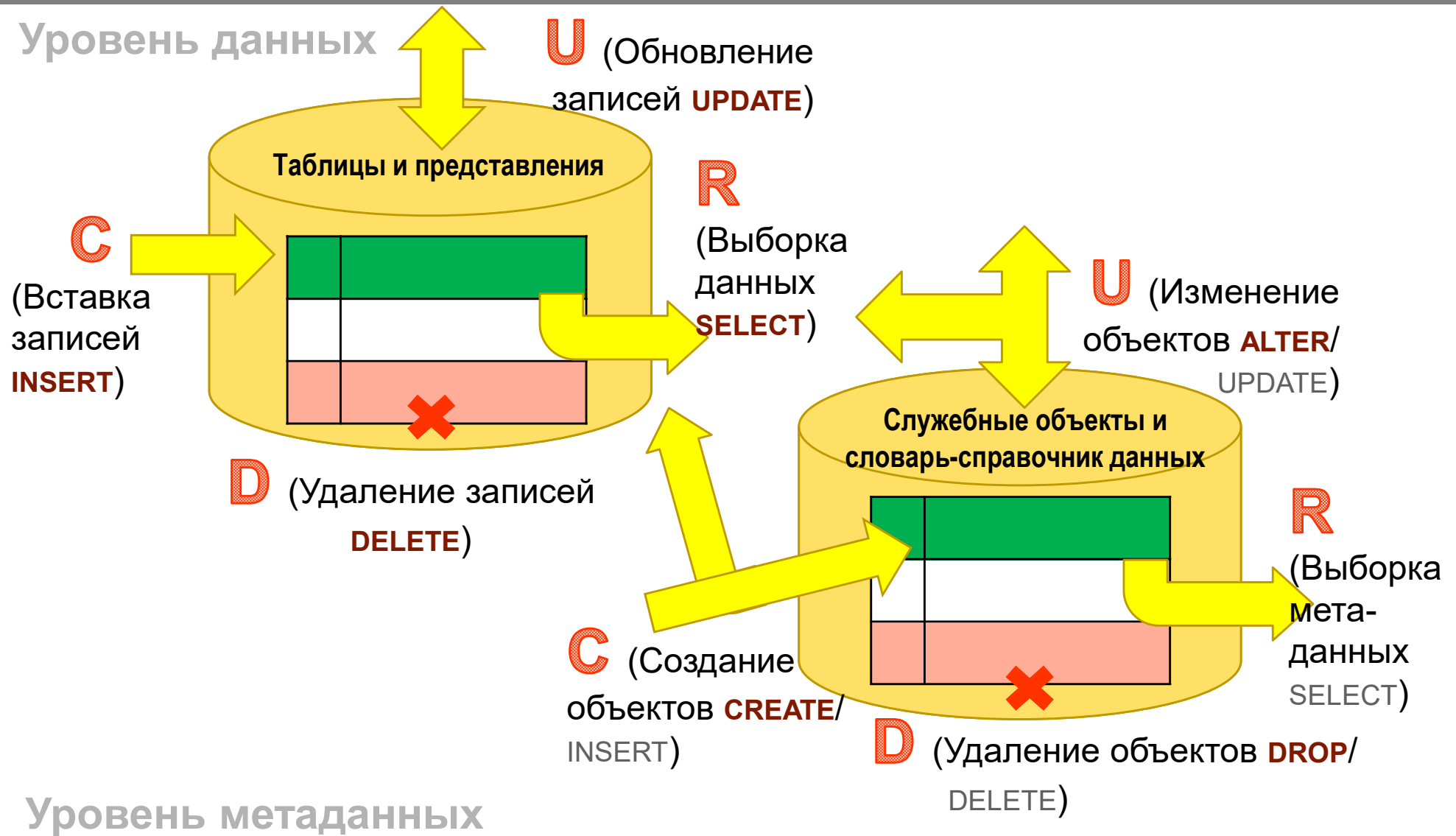
---

Прописные и строчные не различаются, кроме символьных констант и идентификаторов в двойных кавычках или в квадратных скобках  
Принято записывать ключевые слова прописными буквами  
Не использовать для имён ключевые слова SQL

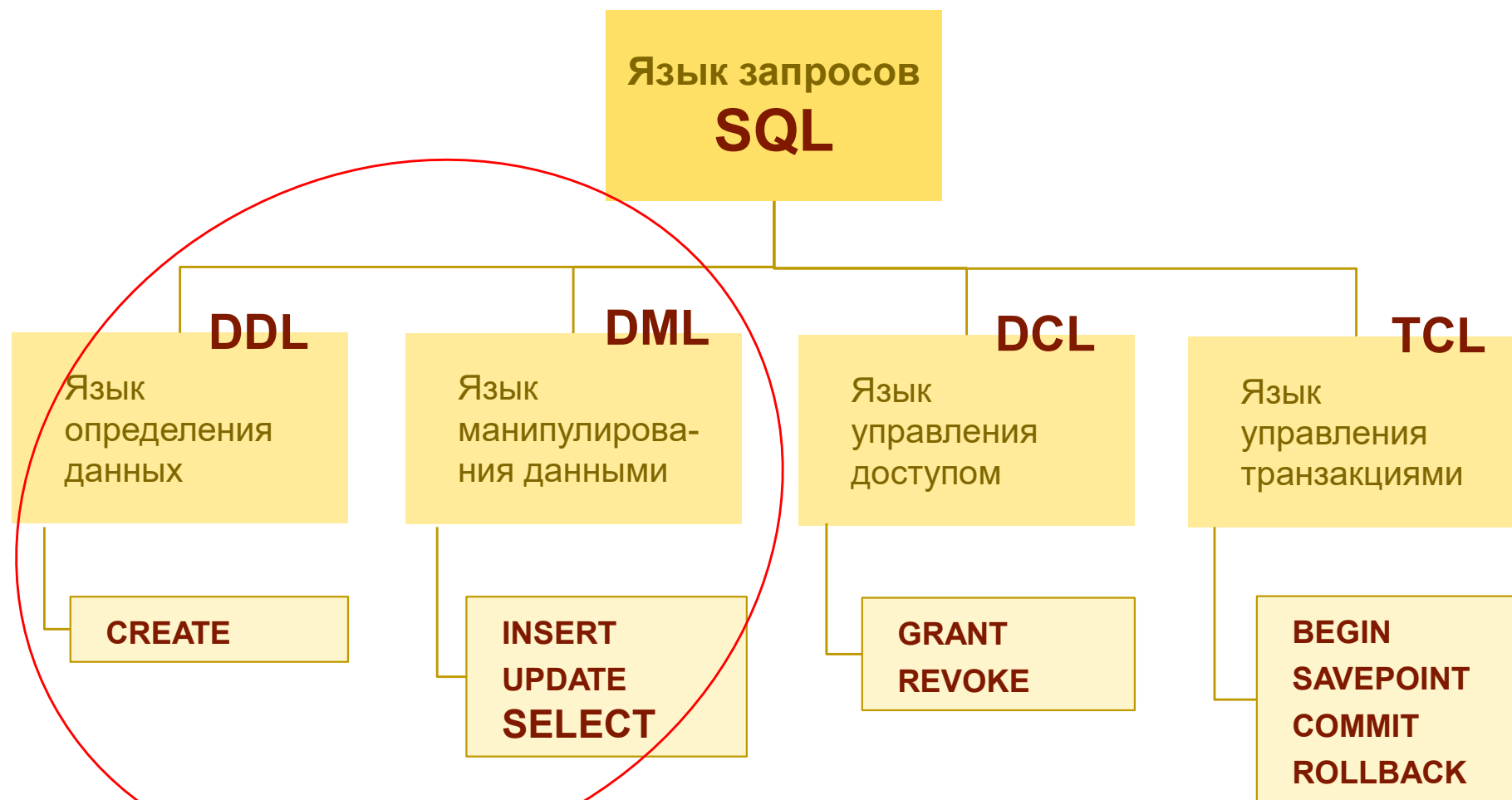
# Выполнение требований CRUD в языке SQL

CRUD	Уровень записей (строк)	Уровень объектов БД
<b>Create</b>	<b>INSERT</b> – добавление новой записи в таблицу	<b>CREATE / INSERT</b> – создание таблиц, представлений, индексов и их описаний
<b>Read Retrieval</b>	<b>SELECT</b> – выборочное чтение и обработка записей таблиц и представлений	<b>SELECT / SHOW</b> – чтение метаданных из словаря- справочника данных
<b>Update</b>	<b>UPDATE</b> – выборочное обновление данных в таблице	<b>ALTER / UPDATE</b> – изменение свойств и параметров объектов
<b>Delete</b>	<b>DELETE</b> – выборочное удаление записей в таблице	<b>DROP / DELETE</b> – удаление объектов и их описаний

# Операции SQL с данными/ метаданными



# Классификация языковых средств SQL



# Дорожная карта оператора SELECT в SQL





# Простые запросы SELECT

Инструкция SELECT выбирает указанные поля из таблицы (FROM) и вычисляет выражения над полями, используя условия (WHERE) с полями и выражениями, сортирует выборку (ORDER BY) по указанным полям по возрастанию или убыванию (DESC).

**SELECT** *список полей и выражений*

**FROM** *таблица*

**WHERE** *условия с полями и выражениями*

**ORDER BY** *список полей или их номеров* **DESC**

**LIMIT** *количество записей*

У Microsoft вместо LIMIT – TOP после SELECT

# Выражения в запросах SELECT

Кроме имён полей таблицы, в запросах после SELECT используются разные выражения:

- литералы (константы): числа, строки, даты и время
- псевдонимы полей
- строковые выражения
- арифметические выражения
- выражения с датами
- выражения с временем
- логические выражения
- встроенные функции
- пользовательские функции

# Различия в написании

**Имена таблиц и полей на естественном языке:**

[название поля] – MS Access, MS SQL Sever

“название поля” – в других СУБД

**Строки текста:**

“строка текста” – MS Access, MS SQL Sever

‘строка текста’ – в других СУБД

**Константы даты и времени**

#дата\_время# - MS Access, MS SQL Sever

“дата\_время” – в других СУБД

**Числовые константы и списки параметров**

3.14,0.5 – без включения русского языка

3,14;0,5 – с включением русского языка

# Простые запросы SELECT

```
SELECT Date()
```

Запрос *сегодняшней даты*

```
SELECT * FROM [СОТРУДНИКИ]
```

Запрос *всех строк со всеми полями* из таблицы СОТРУДНИКИ

```
SELECT [Фамилия], [Имя], [Отчество] FROM [СОТРУДНИКИ]
```

Запрос *списка сотрудников* с фамилиями, именами и отчествами

```
SELECT [Фамилия]+' '+[Имя]+' '+[Отчество] AS [ФИО]  
FROM [СОТРУДНИКИ]
```

Запрос *списка ФИО сотрудников* (склеены фамилии, имена, отчества)

```
SELECT [Фамилия], [Имя], [Отчество], Floor((Date()-[Дата  
рождения])/365) AS [Возраст] FROM [СОТРУДНИКИ]
```

Запрос *списка сотрудников с указанием возраста*

# Сортировка данных ORDER BY

```
SELECT [Фамилия], [Имя], [Отчество] FROM [СОТРУДНИКИ]  
ORDER BY 1,2,3
```

*Сортировка по фамилии, имени и отчеству сотрудников*

```
SELECT [Фамилия]+” ”+[Имя]+” “+[Отчество] [ФИО] FROM [СОТРУДНИКИ]  
ORDER BY 1
```

*Сортировка по ФИО сотрудников (склеены поля фамилии, имена и отчества)*

```
SELECT TOP 1 * FROM [СОТРУДНИКИ] ORDER BY [Дата рождения]
```

*Выбор старейшего сотрудника*

```
SELECT TOP 1 * FROM [СОТРУДНИКИ] ORDER BY [Дата рождения] DESC
```

*Выбор самого молодого сотрудника*

```
SELECT [Пол], [Фамилия]+” ”+[Имя]+” “+[Отчество] [ФИО]  
FROM [СОТРУДНИКИ] ORDER BY 1,2
```

*Сортировка по полу и ФИО сотрудников*

# Выборки по фильтрам WHERE

```
SELECT [Фамилия], [Имя], [Отчество] FROM [СОТРУДНИКИ]  
WHERE [Пол] = "муж"
```

Выбор сотрудников *мужчин*

```
SELECT * FROM [СОТРУДНИКИ] WHERE [Пенсионер]
```

Выбор сотрудников *пенсионеров*

```
SELECT TOP 1 * FROM [СОТРУДНИКИ] WHERE [Пол] = "муж"  
ORDER BY [Дата рождения]
```

Выбор *самого старейшего мужчины* из сотрудников

```
SELECT TOP 1 * FROM [СОТРУДНИКИ] WHERE [Пол] = "жен"  
ORDER BY [Дата рождения] DESC
```

Выбор *самой молодой женщины* из сотрудников

```
SELECT [Фамилия], [Имя], [Отчество] FROM [СОТРУДНИКИ]  
WHERE (Now-[Дата рождения])/365 <= 40 AND [Пол] = "муж"
```

Выбор *мужчин не старше 40 лет*

# Специальные фильтры в SELECT

## Вхождение или невхождение в список значений

Числовое поле/ выражение **IN** (список значений)

**NOT IN** (список значений)

## Вхождение или невхождение в диапазон значений

Числовое поле, дата и время/ выражение **BETWEEN** начало **AND** конец

## Соответствие или несоответствие образцу/ регулярному выражению

Текстовое поле/ выражение **LIKE** образец/ регулярное выражение

**NOT LIKE** образец/ регулярное выражение

## Наличие неопределённого и определённого значения

Поле **IS NULL**

**IS NOT NULL**

# Строковые функции в запросах

## **Len**

длина строки

## **Left, Right, Mid**

выделение подстроки слева, справа или посередине

## **Replace**

замена подстроки в строке

## **LTrim, RTrim, Trim**

усечение пробелов слева, справа или с обеих сторон

## **LCase, UCase**

перевод символов в нижний или в верхний регистр

## **Str**

перевод числа в строку

## **Split**

разделение строки по словам или по разделителю

## **Format**

форматирование значения

## **InStr**

определение позиции подстроки в строке





# Функции даты и времени в запросах

## Date

сегодняшняя дата

## Year, Month, Day, Weekday

определение номера года, месяца, дня недели по дате

## DateValue

возвращает дату по строке

## DateAdd

добавление интервала к дате

## DateDiff

разность дат или времени в заданных интервалах

## MonthName, WeekdayName

название месяца и дня недели

## Time

текущее время

## Hour, Minute, Second

определение часов, минут и секунд по времени

## TimeValue

возвращает время по строке

## Format

форматирование значения даты и времени



**Терпения и удачи всем, кто  
связан с базами данных**

**Спасибо за внимание!**

**Валерий Иванович Артемьев**

**МГТУ имени Н.Э. Баумана, кафедра ИУ-5**

**Банк России**

**Департамент данных, проектов и процессов**

**Тел.: +7(495) 753-96-25**

**e-mail: [viart@bmstu.ru](mailto:viart@bmstu.ru)**