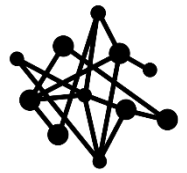
Давайте знакомиться ☺



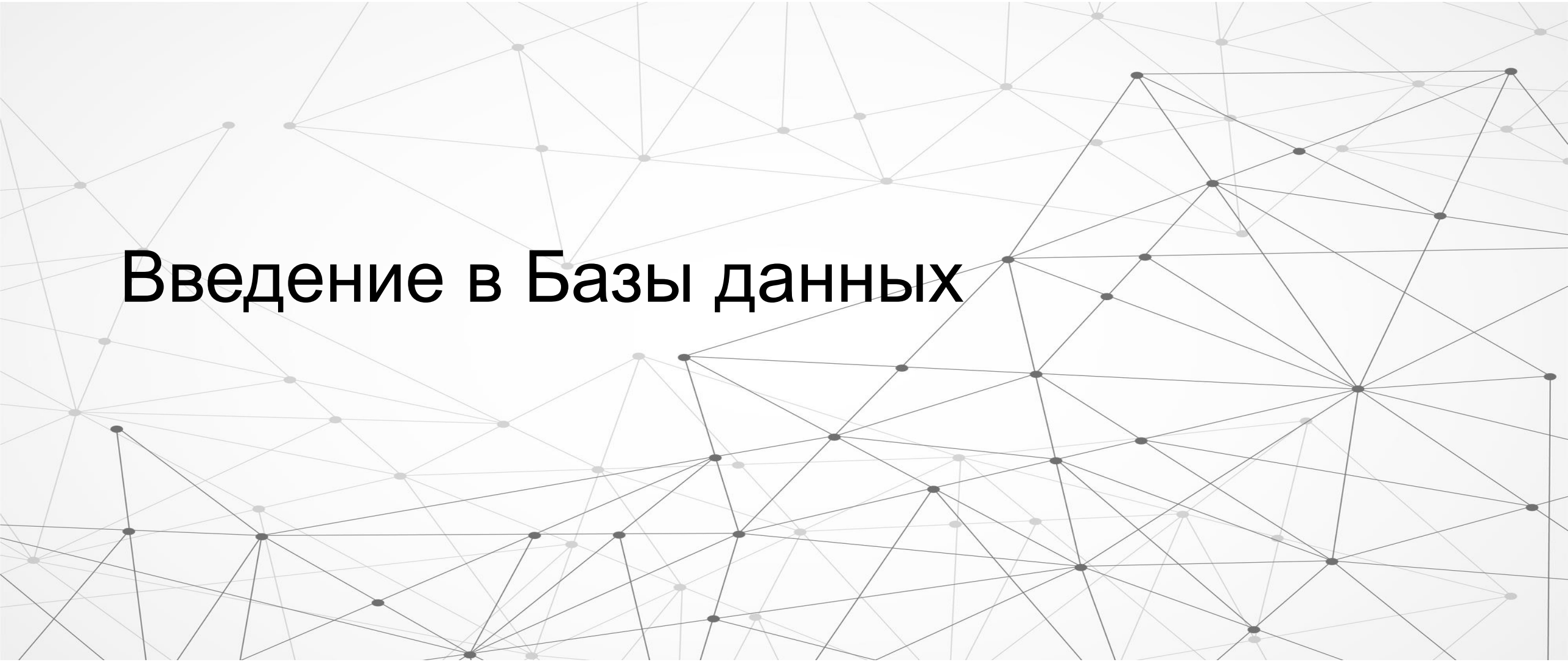
Data
Science
Academy

В КОНЦЕ ЗАНЯТИЯ ВЫ:

- будете знать, зачем нужны БД
- познакомитесь с инструментарием курса
- потренируетесь запускать контейнеры и просматривать атрибуты БД
- напишете свой первый запрос в SQL
- будете знать список возможностей инструкций SQL SELECT
- будете знать тему выполнения инструкции SELECT, FROM, WHERE, ORDER BY

Data Science Academy

Введение в Базы данных



Data Science Academy

Зачем нужны БД

• Основная функция базы данных – предоставление единого хранилища для всей информации, относящейся к определенной теме.

• Вместо того чтобы выискивать нужные сведения в документах Word, таблицах Excel, текстовых файлах, сообщениях электронной почты и самоклеющихся заметках, их можно взять из единой базы.

• База данных может содержать все что угодно, будь-то список приглашенных на свадьбу гостей или информация о каждом клиенте, посетившем Web-сайт электронного магазина и разместившего там свои заказы.

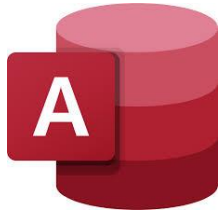
Основные понятия

• Базы данных (*БД) - это структурная совокупность взаимосвязанных* данных определенной предметной области (реальных объектов, процессов, явлений и т.д.).

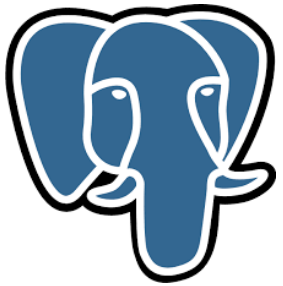
• Пример: БД о наличии медикаментов, БД документов учеников школы, картотека отдела кадров.

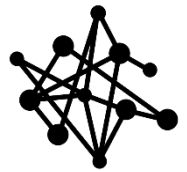
Data Science Academy

Типы БД



- Файл-серверные: Microsoft Access
- Клиент-серверные: MySQL, PostgreSQL, Oracle DB
- Встраиваемые: SqLite





Data Science Academy

Что такое СУБД?

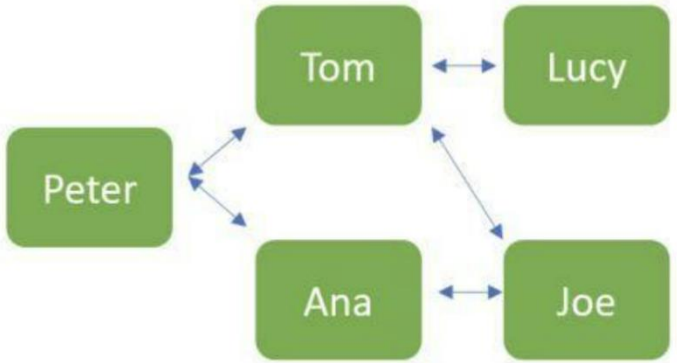
• СУБД - это программа, с помощью которой осуществляется хранение, обработка и поиск информации в базе данных.

• СУБД бывают как реляционные так и не реляционные.

• Реляционные: Oracle DB, PostgresSQL, MS Sql.

• Не реляционные, ещё по другому NoSQL БД, можно отнести MongoDB, Cassandra, HBase

Отличие NoSQL и SQL



NODE	FRIENDS
Peter	{Tom, Ana}
Tom	{Joe, Lucy, Peter}
Ana	{Joe, Lucy, Peter}
Lucy	{Tom}
Joe	{Tom, Ana}

```
var cars = [
{ Model: "BMW", Color: "Red", Manufactured: 2016 },
{ Model: "Mercedes", Type: "Coupe", Color: "Black", Manufactured: "1-1-2017" }
];
```

```
SELECT Orders.OrderID, Customers.Name, Orders.Date
FROM Orders
INNER JOIN Customers
ON Orders.CustID = Customers.CustID
```

Data Science Academy

Функции СУБД

- ввод
- хранение
- манипулирование
- обработку запросов к БД
- поиск
- выборку
- сортировку
- обновление
- защиту данных от несанкционированного доступа или потери

Реляционная СУБД

• Почему именно реляционную СУБД?

• Реляционная алгебра, на базе этой теории строятся все современные СУБД. Реляционная алгебра определяет систему операции над отношениями (таблицами): объединение, деление, вычитание, соединение.

• Это можно сказать на языке математики:

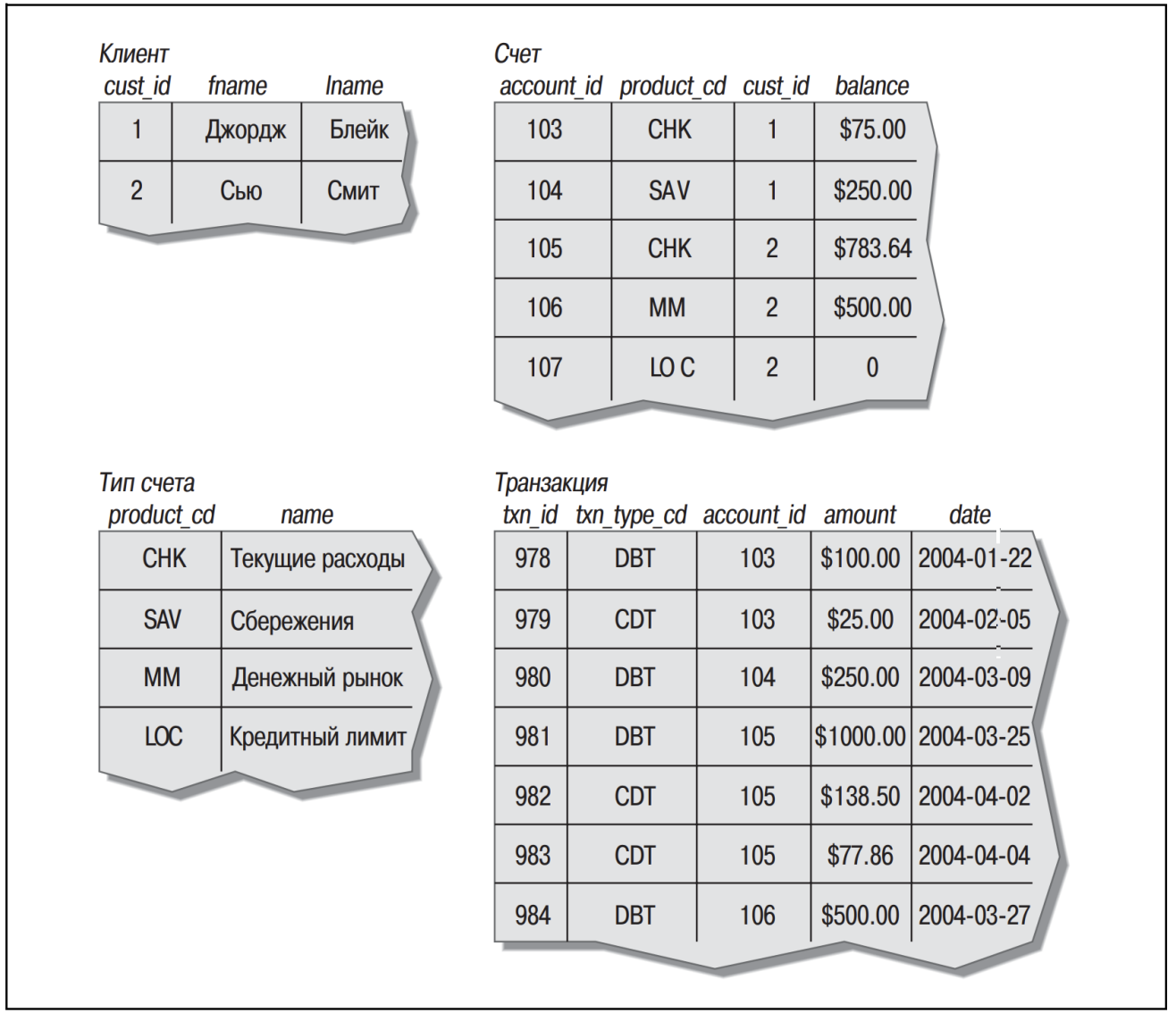
+ (плюс) - (минус) * (умножение) / (деление)

• Все эти операции выражаются на языке запросов SQL

Реляционная модель

Модель представляет собой фиксированную структуру математических понятий, которая описывает то, как будут представлены данные. Базовой единицей данных в пределах реляционной модели является таблица.

- Сущность (entity)
- Столбец (column)
- Строка (row)
- Таблица (table)
- Первичный ключ (primary key)
- Внешний ключ (foreign key)



Клиент

cust_id	fname	lname
1	Джордж	Блейк
2	Сью	Смит

Счет

account_id	product_cd	cust_id	balance
103	CHK	1	$75.00
104	SAV	1	$250.00
105	CHK	2	$783.64
106	MM	2	$500.00
107	LO C	2	0

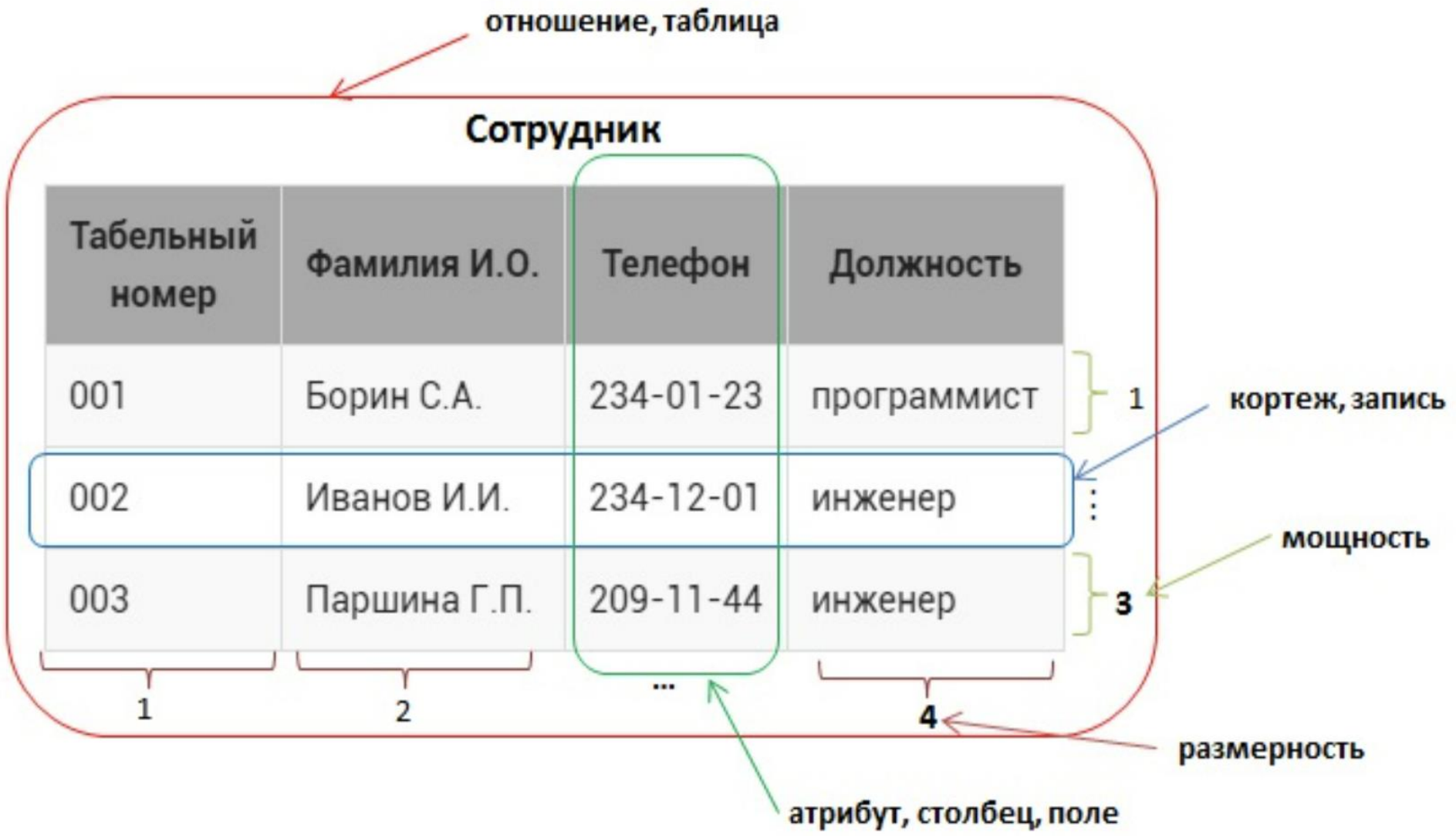
Тип счета

product_cd	name
CHK	Текущие расходы
SAV	Сбережения
MM	Денежный рынок
LOC	Кредитный лимит

Транзакция

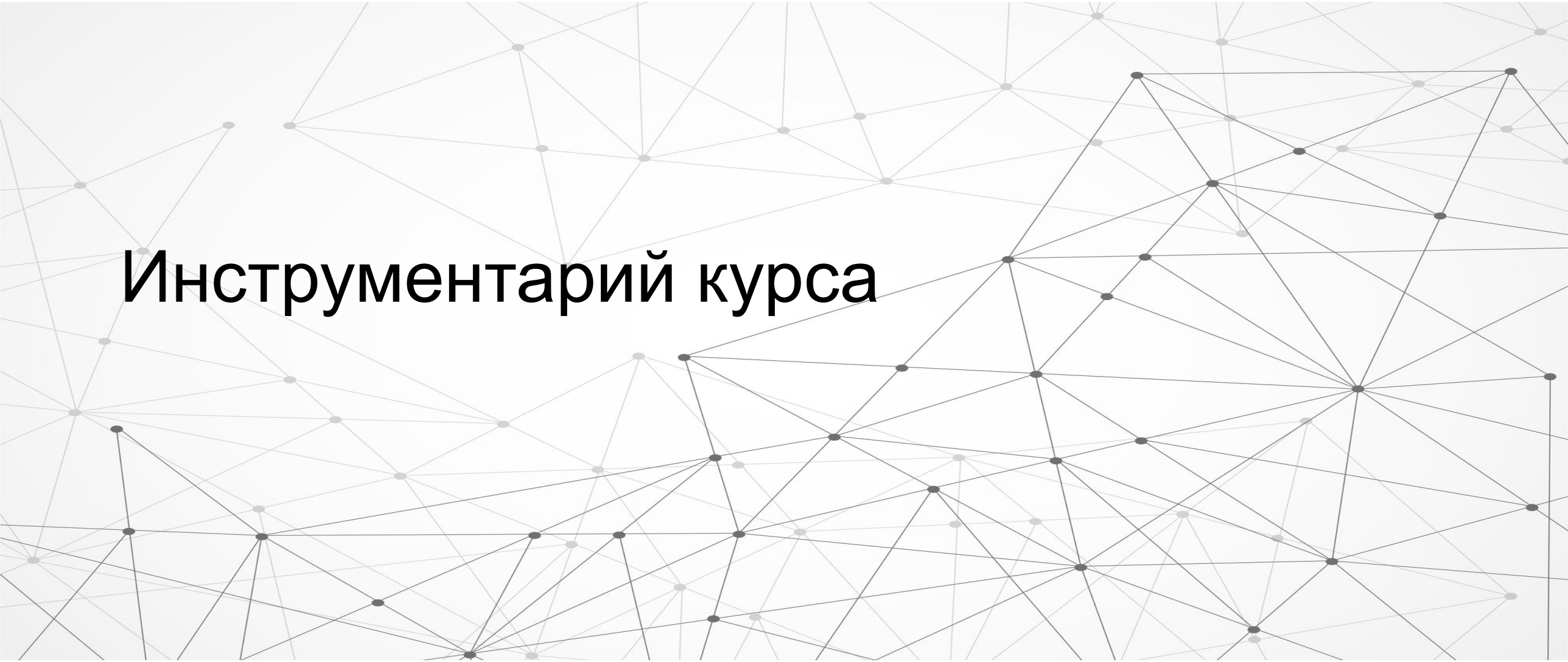
txn_id	txn_type_cd	account_id	amount	date
978	DBT	103	$100.00	2004-01-22
979	CDT	103	$25.00	2004-02-05
980	DBT	104	$250.00	2004-03-09
981	DBT	105	$1000.00	2004-03-25
982	CDT	105	$138.50	2004-04-02
983	CDT	105	$77.86	2004-04-04
984	DBT	106	$500.00	2004-03-27

Data Science Academy



Data Science Academy

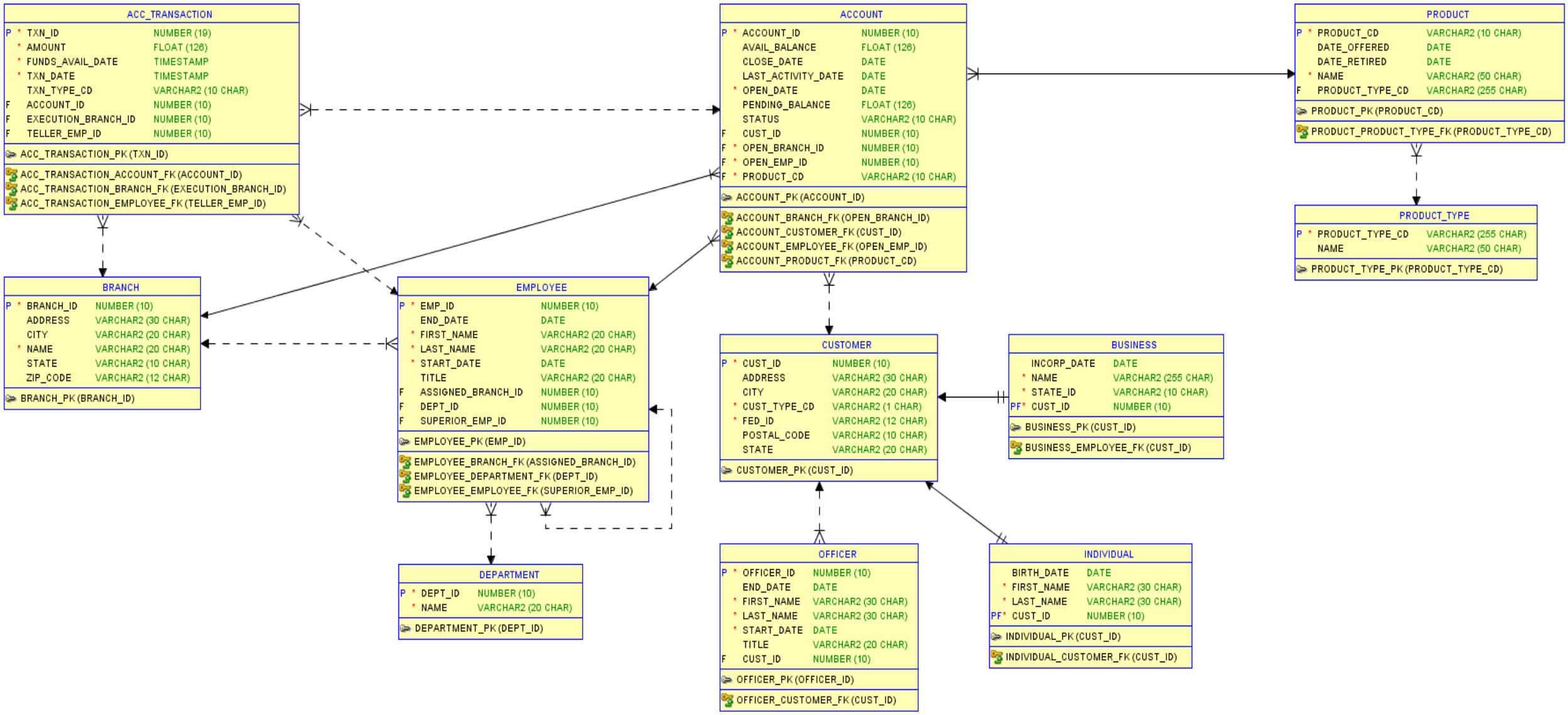
Инструментарий курса



Data Science Academy

С чем мы будем работать?

1. Oracle DB – СУБД.
2. Зарегистрируемся на сайте livesql.oracle для работы с БД.
3. Напишем “Hello World!”
4. Загрузим БД Банк.
5. Docker – легкие «виртуальные» машины.
6. SQL Developer – основная среда разработки.
7. Немного про другие среды разработки.



Data Science Academy

Знакомство с БД Банк детально.

Account - Конкретный счет, открытый для конкретного клиента

Branch – Филиал банка

Business – Клиент-юридическое лицо (подтип таблицы Customer)

Customer - Физическое или юридическое лицо, известные банку

Department - Группа сотрудников банка, реализующая определенную банковскую функцию

Employee - Человек, работающий в банке

Individual – Клиент-физическое лицо (подтип таблицы Customer)

Officer - Человек, которому разрешено вести дела от лица клиента юридического лица

Product - Услуга банка, предлагаемая клиентам

Product_type - Группа функционально схожих услуг

Transaction - Изменение баланса счета

Data Science Academy

Поговорим про SQL

Data Science Academy

Введение в SQL

- SQL это непроцедурный язык
- Процедурные языки нужны для реализации какой-либо логики
- SQL это декларативный язык (туда же входит HTML)

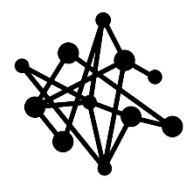
```sql
SELECT * FROM Users WHERE Country='Mexico';
```

```html
<article>
  <header>
    <h1>Declarative Programming</h1>
    <p>Sprinkle Declarative in your verbiage to sound smart</p>
  </header>
</article>
```

Data Science Academy

Инструкции SQL

SELECT INSERT UPDATE DELETE MERGE	Язык манипулирования данными (DML)
CREATE ALTER DROP RENAME TRUNCATE COMMENT	Язык определения данных (DDL)
GRANT REVOKE	Язык управления данными (DCL)
COMMIT ROLLBACK SAVEPOINT	Управление транзакциями (TCL)



Data
Science
Academy

- Результатом SQL запроса является результирующий набор.
- Язык SQL делится на команды DDL, DML, TCL, DCL
- DDL (Data Definition Language) – CREATE, ALTER, DROP, TRUNCATE
- DML (Data Manipulation Language) – SELECT, INSERT, UPDATE, DELETE
- TCL (Transaction Control Language) – COMMIT, ROLLBACK, SAVEPOINT
- DCL (Data Control Language) – GRANT, REVOKE
- ANSI (1989, 1992, 1999 и 2003 гг.)
- Процедурный SQL. PL/SQL, PostgreSQL, TransactSQL.

Data Science Academy

Блок	Назначение
Select	Определяет столбцы, которые должны быть включены в результирующий набор запроса
From	Указывает таблицы, из которых должны быть извлечены данные, и то, как эти таблицы должны быть соединены
Where	Ограничивает число строк в окончательном результирующем наборе
Group by	Используется для группировки строк по одинаковым значениям столбцов
Having	Ограничивает число строк в окончательном результирующем наборе с помощью группировки данных
Order by	Сортирует строки окончательного результирующего набора по одному или более столбцам

Типы данных Oracle DB

- CHAR, VARCHAR2
- NUMBER
- BOOLEAN
- DATE
- LOB, CLOB, XMLTYPE

Выбор всех столбцов

SQL Worksheet

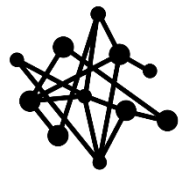
```
1  select * from account;
```

ACCOUNT_ID	AVAIL_BALANCE	CLOSE_DATE	LAST_ACTIVITY_DATE	OPEN_DATE	PENDING_BALANCE	STATUS	CUST_ID	OPEN_BRANCH_ID	OPEN_EMP_ID	PRODUCT_CD
1	1057.75	-	04-JAN-05	15-JAN-00	1057.75	ACTIVE	1	2	10	CHK
2	500	-	19-DEC-04	15-JAN-00	500	ACTIVE	1	2	10	SAV
3	3000	-	30-JUN-04	30-JUN-04	3000	ACTIVE	1	2	10	CD
4	2258.02	-	27-DEC-04	12-MAR-01	2258.02	ACTIVE	2	2	10	CHK

Выбор конкретных столбцов

```
select account_id, status from account;
```

ACCOUNT_ID	STATUS
1	ACTIVE
2	ACTIVE
3	ACTIVE
4	ACTIVE
5	ACTIVE
6	ACTIVE
7	ACTIVE



Data
Science
Academy

Арифметические выражения

Получение выражений с данными в числовом формате или формате дат с помощью арифметических операторов

Оператор	Описание
+	Сложение
-	Вычитание
*	Умножение
/	Деление

Использование арифметических операторов

```
SELECT
account_id, avail_balance, avail_balance + 1000 as balance_2
FROM account
```

ACCOUNT_ID	AVAIL_BALANCE	BALANCE_2
1	1057.75	2057.75
2	500	1500

Data Science Academy

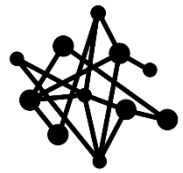
Приоритет операторов

```
SELECT
account_id, avail_balance, 2 * avail_balance + 1000
FROM account
```

ACCOUNT_ID	AVAIL_BALANCE	2*AVAIL_BALANCE+1000
1	1057.75	3115.5

```
SELECT
account_id, avail_balance, 2 * (avail_balance + 1000)
FROM account
```

ACCOUNT_ID	AVAIL_BALANCE	2*(AVAIL_BALANCE+1000)
1	1057.75	4115.5



Data Science Academy

Использование псевдонимов столбцов

```
SELECT
account_id, avail_balance, avail_balance + 1000 as balance_2
FROM account
```

ACCOUNT_ID	AVAIL_BALANCE	BALANCE_2
1	1057.75	2057.75
2	500	1500

Data Science Academy

Оператор конкатенации

- «Склеивает» строчки
- Обозначается двумя вертикальными палками ||

```
SELECT
first_name || last_name as emp_name
FROM employee
```

EMP_NAME
MichaelSmith
SusanBarker

Data Science Academy

Использование символьных строк литералов

```
SELECT
first_name ||' is a '||title as emp_position
FROM employee
```

EMP_POSITION
Michael is a President
Susan is a Vice President
Robert is a Treasurer

Data Science Academy

Дубли строк

```
select distinct product_cd from account;
```

PRODUCT_CD
MM
SAV
BUS
SBL
CHK
CD

Data Science Academy

- Средства ограничения числа строк:
  - Предложение WHERE
  - Условия сравнения с операторами

    =, <=, BETWEEN, IN, LIKE и NULL
  - Логические условия с операторами AND, OR и NOT
- Правила приоритета операторов, используемых в выражении
- Сортировка строк с использованием предложения ORDER BY

Data Science Academy

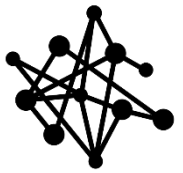
Ограничение числа выбранных строк

• Ограничение набора возвращаемых строк с помощью предложения WHERE:

SELECT *|{[DISTINCT] столбец|выражение [псевдоним],...}

FROM таблица [WHERE условия];

• Предложение WHERE следует за предложением FROM.



Data Science Academy

Использование выражения WHERE

```
select emp_id, first_name, last_name, start_date, title
FROM employee
where emp_id = 1
```

EMP_ID	FIRST_NAME	LAST_NAME	START_DATE	TITLE
1	Michael	Smith	22-JUN-01	President

Data Science Academy

Символьные строки и даты

• Символьные строки и значения дат заключаются в одиночные кавычки.

• Символьные значения проверяются с учетом регистра, а значения дат – с учетом формата.

```
select emp_id, first_name, last_name, start_date, title
FROM employee
where first_name = 'Michael';

select emp_id, first_name, last_name, start_date, title
FROM employee
where start_date = '22-JUN-01';
```

Data Science Academy

Операторы сравнения

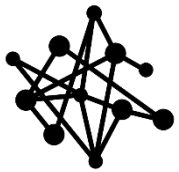
=	Равно
>	Больше
>=	Больше или равно
<	Меньше
<=	Меньше или равно
<>	Не равно
`BETWEEN ...AND...`	Между двумя значениями
`IN(set)`	Совпадает с любым значением из списка
`LIKE`	Соответствует шаблону символов
`IS NULL`	Является значением null

Data Science Academy

Использование операторов сравнения

```
select account_id, avail_balance from account
where avail_balance < 1000;
```

ACCOUNT_ID	AVAIL_BALANCE
2	500
5	200
8	534.12



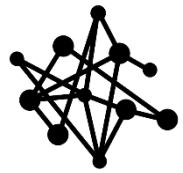
Data
Science
Academy

Условия диапазона, использующие оператор BETWEEN

Оператор BETWEEN используется для вывода строк на основе диапазона значений:

```
select account_id, avail_balance from account
where avail_balance between 1000 and 2000;
```

ACCOUNT_ID	AVAIL_BALANCE
1	1057.75
6	1057.75
19	1500



Data Science Academy

Сопоставление с шаблоном, использующее оператор LIKE

• Оператор LIKE применяется при поиске допустимых строковых значений с использованием подстановочных знаков.
• Условие поиска может включать символьный или числовой литерал:
  – % обозначает произвольное число символов (в том числе, возможно, нулевое).
  – _ обозначает строго один символ.

```
select emp_id, first_name, last_name, start_date, title
FROM employee
where first_name like 'M%';
```

```
SELECT last_name
FROM employee
WHERE last_name LIKE '_a%e%';
```

Data Science Academy

Поиск строк, содержащих % или _

Идентификатор ESCAPE позволяет искать сами символы % и _.

```
select
name
from
(select 'Smith_' as name from dual
)
where name like '_m%?_' ESCAPE '?'
```

NAME
Smith_

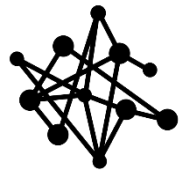
Data Science Academy

Использование условий NULL

Проверка на наличие пустых значений (null) производится с помощью оператора IS NULL.

```
SELECT
emp_id, first_name, last_name, start_date, title
FROM employee
WHERE end_date IS NULL;
```

EMP_ID	FIRST_NAME	LAST_NAME	START_DATE	TITLE
1	Michael	Smith	22-JUN-01	President
2	Susan	Barker	12-SEP-02	Vice President



Data Science Academy

Определение условий с использованием логических операторов

`AND`	Возвращает `TRUE`, если *оба* составляющих условия истинны
`OR`	Возвращает `TRUE`, если *любое* из составляющих условий истинно
`NOT`	Возвращает `TRUE`, если условие ложно

Data Science Academy

Использование оператора AND

В случае оператора AND необходимо, чтобы оба составляющих условия были истинными:

```
select first_name, last_name, start_date
from employee
where start_date > '17-MAR-04'
and first_name like 'C%';
```

FIRST_NAME	LAST_NAME	START_DATE
Chris	Tucker	15-SEP-04

Data Science Academy

Использование оператора OR

В случае оператора OR необходимо, чтобы любое из составляющих условий было истинным:

```
select first_name, last_name, start_date
from employee
where start_date > '17-MAR-01'
or first_name like 'C%';
```

FIRST_NAME	LAST_NAME	START_DATE
Michael	Smith	22-JUN-01
Susan	Barker	12-SEP-02
Susan	Hawthorne	24-APR-02
John	Gooding	14-NOV-03

Использование оператора NOT

```
SELECT first_name, last_name, start_date, title
FROM employee
WHERE title NOT IN ('President','Loan Manager','Teller');
```

FIRST_NAME	LAST_NAME	START_DATE	TITLE
Susan	Barker	12-SEP-02	Vice President
Robert	Tyler	09-FEB-00	Treasurer
Susan	Hawthorne	24-APR-02	Operations Manager
Helen	Fleming	17-MAR-04	Head Teller

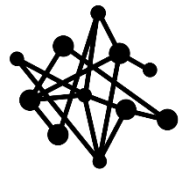
Правила приоритета

```
SELECT
first_name,
last_name,
title,
start_date
FROM employee
WHERE title = 'Teller'
AND first_name LIKE 'S%'
OR start_date > '22-JUN-01';
```

FIRST_NAME	LAST_NAME	TITLE	START_DATE
Susan	Barker	Vice President	12-SEP-02
Susan	Hawthorne	Operations Manager	24-APR-02
John	Gooding	Loan Manager	14-NOV-03

```
SELECT
first_name,
last_name,
title,
start_date
FROM employee
WHERE title = 'Teller'
AND (first_name LIKE 'S%'
OR start_date > '22-JUN-01'
);
```

FIRST_NAME	LAST_NAME	TITLE	START_DATE
Chris	Tucker	Teller	15-SEP-04
Sarah	Parker	Teller	02-DEC-02
Jane	Grossman	Teller	03-MAY-02



Data Science Academy

Использование предложения ORDER BY

• Сортировка извлеченных строк с помощью предложения ORDER BY:

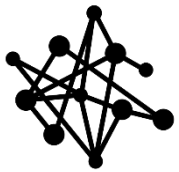
  – ASC: в порядке возрастания (по умолчанию)

  – DESC: в порядке убывания

• Предложение ORDER BY должно быть последним в инструкции SELECT:

```
select
account_id,
avail_balance,
open_date
from account
order by open_date
```

ACCOUNT_ID	AVAIL_BALANCE	OPEN_DATE
1	1057.75	15-JAN-00
2	500	15-JAN-00
9	767.77	15-JAN-00



Data Science Academy

Сортировка

- Сортировка в порядке убыванию:

```
select
account_id,
avail_balance,
open_date
from account
order by open_date DESC;
```

- Сортировка по псевдониму столбца:

```
select
account_id,
avail_balance,
open_date
from account
order by avail_balance;
```

Data Science Academy

Сортировка

- Сортировка по номеру позиции столбца:

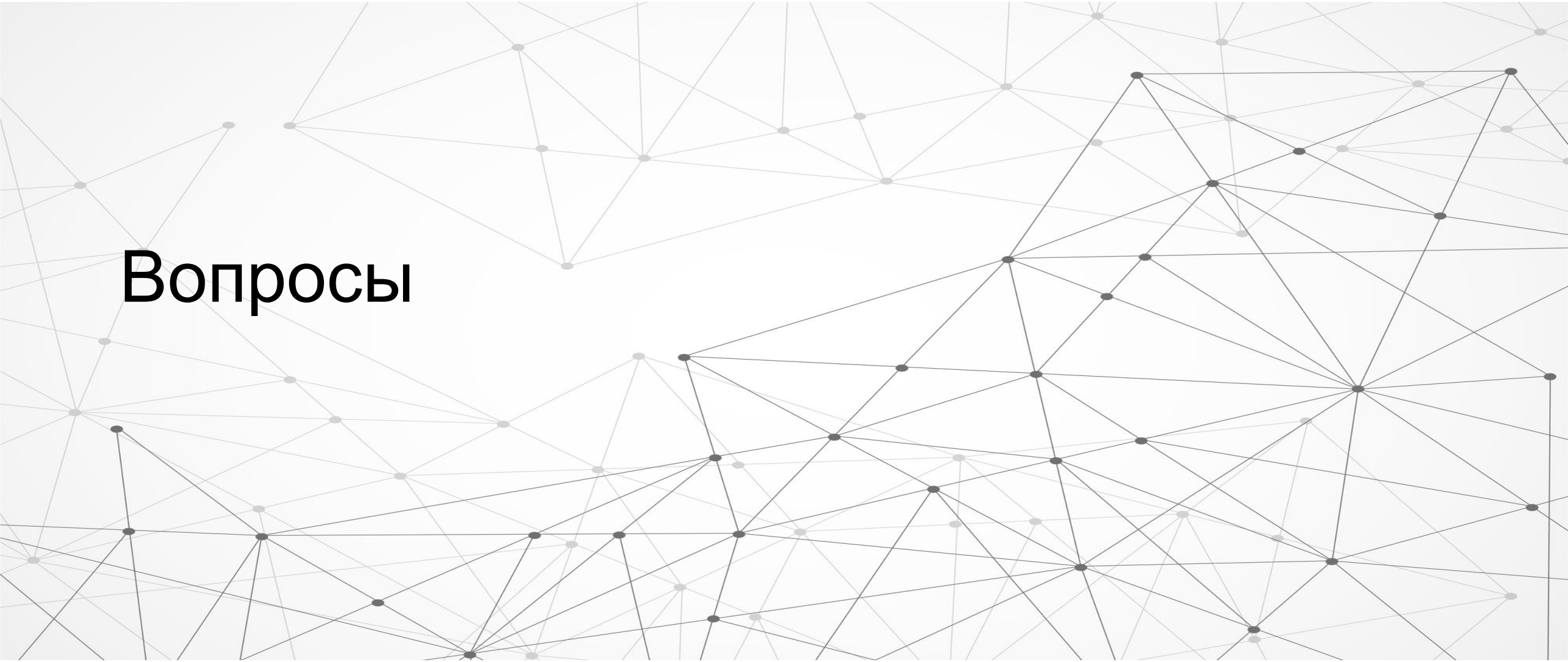
```
select
account_id,
avail_balance,
open_date
from account
order by 3 desc;
```

- Сортировка по нескольким столбцам:

```
select
account_id,
avail_balance,
open_date
from account
order by account_id, open_date;
```

Data Science Academy

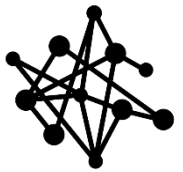
Вопросы



Data Science Academy

Домашнее задание.

1. Необходимо дома развернуть Docker контейнер.
2. Установить Docker образ БД Oracle 12c
3. Установить SQL Developer, и установить соединение с БД
4. Запустить скрипт для создания учебной БД Банк
5. Сделать select к каждой таблице и изучить поля и соединения.
6. Выбор всех данных из различных таблиц
7. Выполнить ДЗ по пройденной теме



Data
Science
Academy

Инструкция по установке Docker и DB Oracle 12c (вышлем отдельную инструкцию)

1. Установить Docker и зарегистрироваться на сайте https://www.docker.com/get-started
2. Установить Оракл ДБ
3. Зайти в настройки Docker – Resources – WSL Integration – и убрать галочку.
4. Установить SQL Developer https://www.oracle.com/tools/downloads/sqldev-downloads.html
5. Установить соединение через SQL Developer и загрузить учебную БД Банк.

Data Science Academy