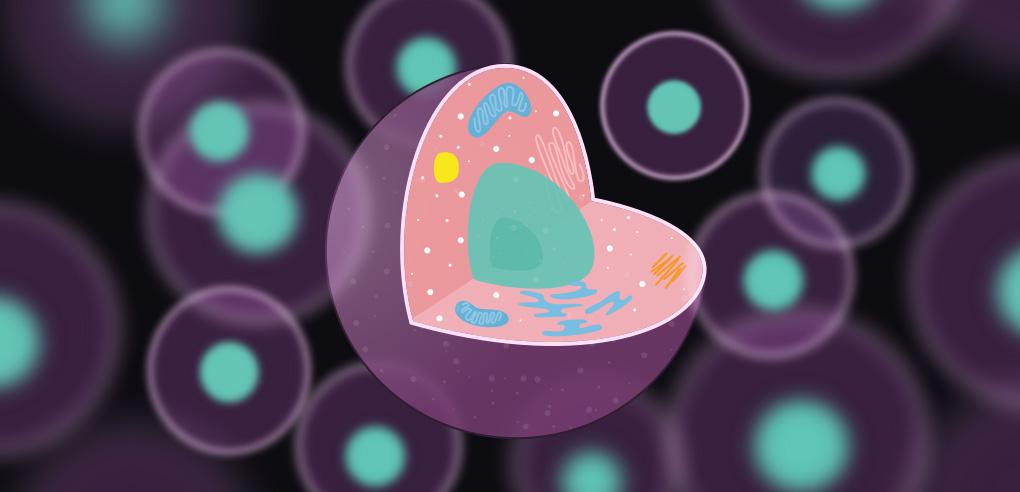
**SQL coursework**

ANISHCHANKA DANIIL WRITER

Отчет   
По анализу данных

**\_\_\_**

By [Daniil Anishchanka](mailto:den382505@gmail.com)



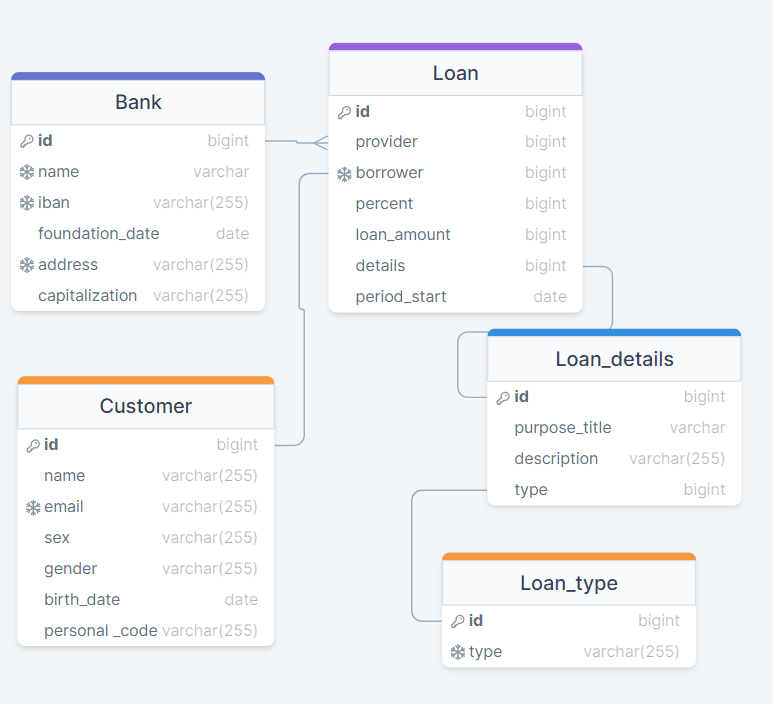
# 1. Design ER-diagram

For the chart design, I chose the online tool [drawsql.app](https://drawsql.app/), which allows me to select the type of database and follow the appropriate constraints.

For strategic time saving purposes, it was decided to reduce the number of tables from eight to five. This had a positive future impact on the volume and number of scripts required to be written at all stages of the assignment.

A diagram describing the tables is attached below. For example, since the topic of my project is a banking application, it was decided to put the debt type in a separate table, because in general all types can be described by several unique categories.

Since, for example, iban is unique, corresponding constraints have been imposed on it and other unique fields.



## 2. Develop OLTP solution

Once the diagram has been created, we can start making it a reality.

I chose Docker to host the PostgreSQL server, deploying the necessary container with the image of the latest PostgreSQL version. This solution introduced a significant number of limitations because Docker communicates with the computer system through a port, which appeared to be a bottleneck.

I created database, and added admin user there.

Full database creation script can be found by this [link](https://github.com/dadadaniil/sql_coursework/blob/main/scripts/db_creation_script.sql).

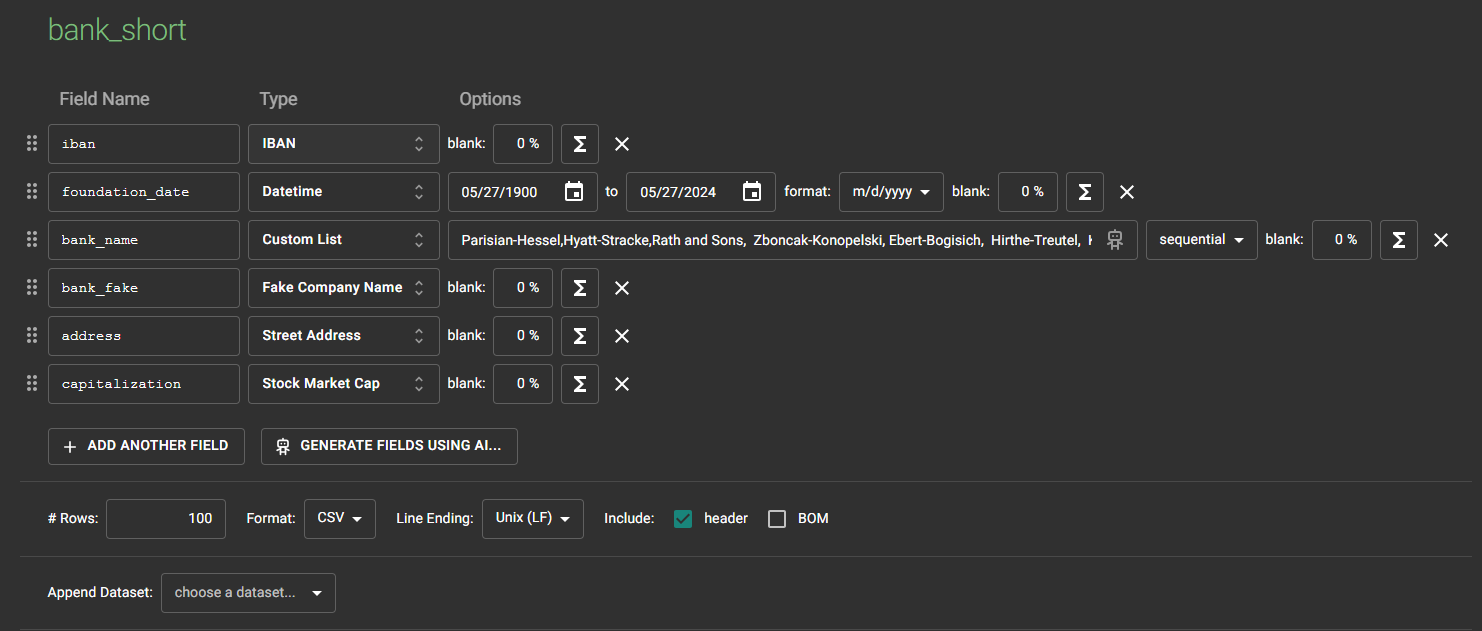
I managed to create all databases and after apply constraints to them. During first creation procedure, execution of script allows inserting prepared data with one action and after apply foreign keys for all the databases.

## 3. Prepare data to load to your OLTP database

By requirements, we had 200 rows of information in our datasets, so it’s make impossible to write datasets by your own. For purpose of data generation I used website [mockaroo](https://www.mockaroo.com), where located both of my dataset’s schemas: [1](https://www.mockaroo.com/schemas/625877), [2](https://www.mockaroo.com/schemas/625875).

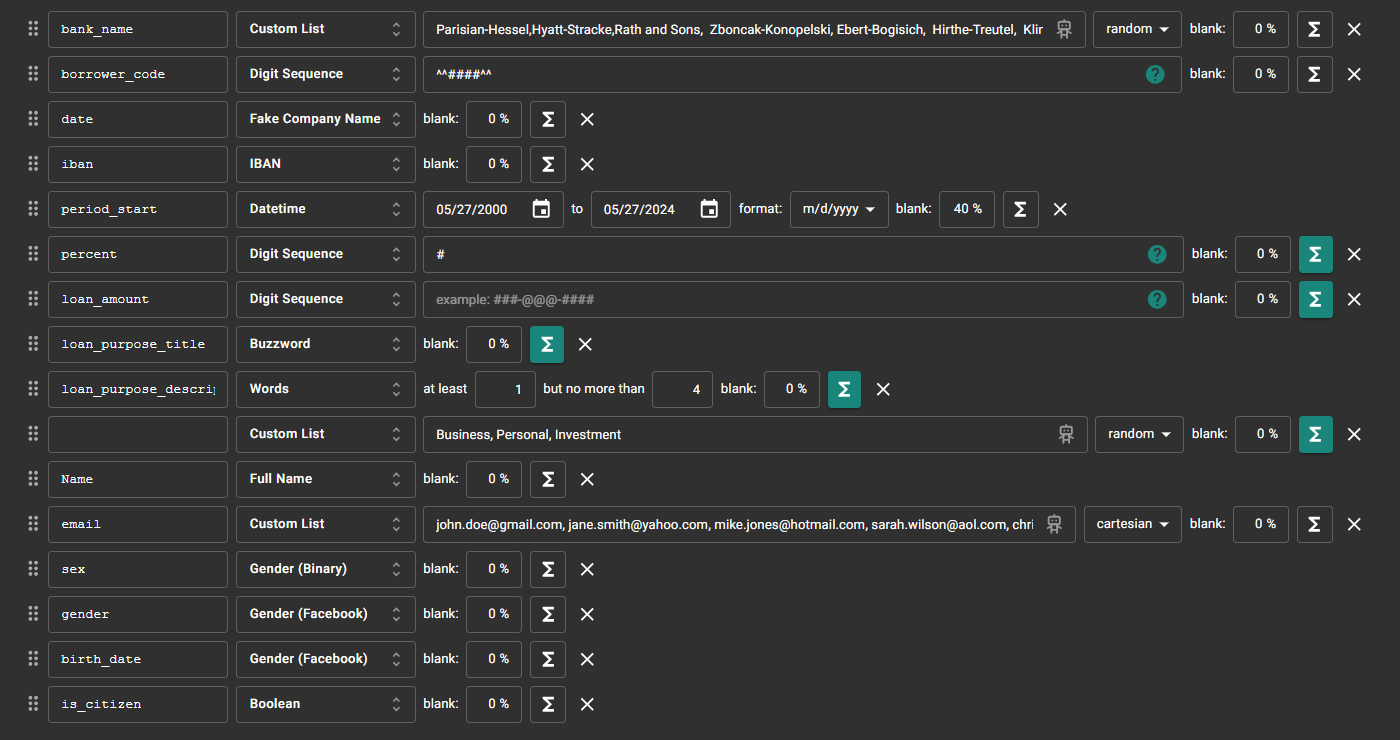
To generate relevant data, I was forced to learn the internal programming language for data generation.

**Bank dataset**

First database generated bank names from a list (these bank names are used as business keys in other database with customer loans) and after all names used, generations start with random names. Main loan providers are:

* Parisian-Hessel
* Hyatt-Stracke
* Rath and Sons
* Zboncak-Konopelski
* Ebert-Bogisich
* Hirthe-Treutel
* Kling LLC
* Lynch Group
* Keebler Group

**Customer dataset**

Here I added some logic in “percent”, “loan\_amount”, “loan\_purpose\_title”, “loan\_purpose\_description” and “loan\_type” fields to generate actual data instead of random digits and dates, which can not be applicable for an example in a context of birthdate.

Some users don’t have loans, so generation of other fields connected to loan also disabled. Their empty fields are not null, but stores values which represents lack of loans.

Basing on this data, scripts will sort users by borrower-or-not categories.

**Both datasets can be found by** [**link**](https://github.com/dadadaniil/sql_coursework/tree/main/datasets)**.**

**4. Prepare script to load data from CSV to your OLTP database.**

Besides the fact that data is already prepared and provided in right manner, not-null checks, digit-to-digit-cell, date-check etc. were implemented.

Before all, I decided to create temporary tables with all heap data. This move serves on the same purpose as sql’s CREATE TEMP TABLE, but I need to have data from a previous get-data-script run.

At this stage server, runned in Docker makes itself known by adding some constraints, so I was forced to use \COPY command because of accesses rights, which i didn’t succeed to change in Windows settings.

Unfortunately, sometimes, script can be runned without inserting any data: <https://stackoverflow.com/questions/54031813/i-am-trying-to-copy-a-file-but-getting-error-message>

Dates parsed from text to date format with *TO\_DATE*(period\_start, 'MM/DD/YYYY')

Full script can be found by [link](https://github.com/dadadaniil/sql_coursework/blob/main/scripts/get_data.sql).

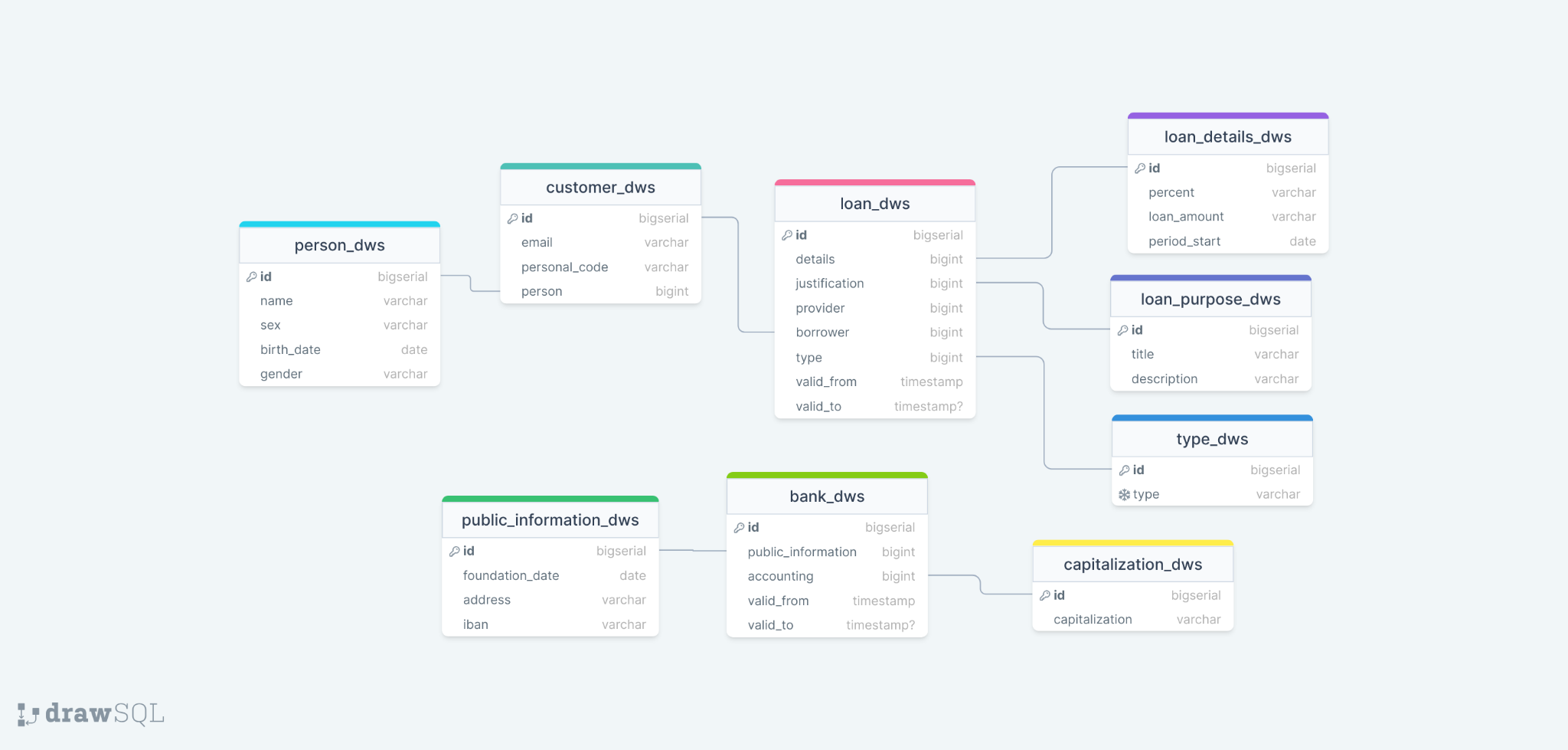
**5. Develop OLAP solution – design snowflake DWH**

For the banking application, I recognize the need for a robust and efficient system to manage business loans, borrowing trends, financial reporting. To address this, I have developed an OLAP solution designed to automate these tasks seamlessly.

By analogy, DWH was designed in [drawsql.app](https://drawsql.app/). I added fields valid\_from and valid\_to, where the second of the is nullable. The basic rule of designing snowflake is that in fact table no real values, but only reference keys to other databases.

Taking into account the fact that I have snowflake type, it becomes apparent table linear inheritance.

Diagram is provided by [link](https://drawsql.app/teams/dadadaniils-team/diagrams/bank-dws).



**6. Develop ETL process to move data from OLTP database to OLAP database**

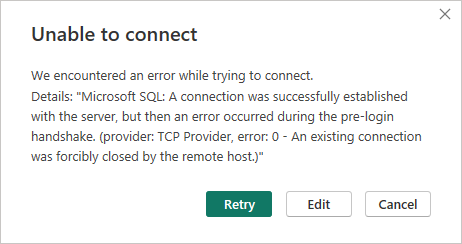
I created a function which is executed on each update of the original database. So in one place there located logic for searching corresponding element in DWS, replacing valid\_from/to field and adding fully new element.

So the whole process can be described with a long function which executes everything: finding such row, adding changes, in DHW, waiting for queries in source database etc.

Changes are detected with help of trigger, which reacts on any update of source main tables in primary database.

[Script link](https://github.com/dadadaniil/sql_coursework/blob/main/scripts/transfer_to_dws.sql)

**7. Create visual report based on your OLAP solution**This point took me has caused a lot of inconvenience again because of Docker:



Addition: <https://stackoverflow.com/questions/9890971/sql-server-a-connection-was-successfully-established-with-server-existing-c>

What hasn’t become a problem because I loaded data manually, exporting data with .xml to PowerBI.

Here I added measures of bank names. On the first picture I added a filter to exclude all years with less than two bank foundations. Also, we need to put attention to the fact that from 99 banks there are only 85 unique addressed, which means that some banks settled on the same address. We can assume that it means existence of subsidiary companies.

Link to report file is [here](https://github.com/dadadaniil/sql_coursework/blob/main/powerBI_report/report.pdf).

