Data modeling. Database Design

# Preparation

Find your username and password for the dedicated Postgres database

Check the connection through the client software. For example, DBeaver.

# Exercise 1

Bring the table with the list of resolved requests first to the first and then to the second normal form. The solved\_cases column contains a list of technical support requests made by a certain employee. Each employee has a unique employee\_num id.

|  |  |  |
| --- | --- | --- |
| **employee\_num** | **employee** | **solved\_cases** |
| 900 | Ivan A. | 1012, 1001 |
| 901 | Aleksandr R. | 1002, 1003, 1004 |
| 902 | Sofia T. | 1010 |

Employees and resolved cases.

**Task**

1. Fix the table. Migrate the data to the new table.
   1. Write a DDL script to create the corrected table.
   2. Write DML commands to populate the tables with data from the initial one.

# Exercise 2

Examine the table that records data about instructors and the courses they teach in different groups.

Divide the initial table considering the requirements of 3NF.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **emp\_id** | **emp\_name** | **pos\_id** | **position** | **salary** | **experience** | **dep\_id** | **department** | **subject** | **stud\_group\_id** | **class\_type** |
| 1 | Ivanov I. | 10 | tutor | 500 | 5 | BAC | Business analytics | Data management | 256 | 02 - Workshop |
| Statistics | 123 | 02 - Workshop |
| 12 | Petrov M. | 20 | Professor | 800 | 7 | BAC | Business analytics | Data management | 256 | 01 - Lecture |
| Accounting | 256 | 02 - Workshop |
| 13 | Sidorov N. | 55 | Associate professor | 750 | 10 | BAC | Business analytics | Statistics | 123 | 01 - Lecture |
| Accounting | 256 | 01 - Lecture |
| Accounting | 300 | 01 - Lecture |
| Accounting | 300 | 02 - Workshop |
| 45 | Yegorov V. | 55 | Associate professor | 750 | 5 | IBC | IT business | E-Commerce | 244 | 01 - Lecture |

This table also describes the responsibility of teachers for certain formats of classes. For example, in one group, a teacher in the "Accounting" course can conduct seminars and lectures, and in another — only lectures.

One teacher can hold only one position. Each position and department has a unique code.

Salary is determined by the position.

The names of the disciplines can sometimes change, and in the new database, users would like to be able to easily correct the text of the titles. The same is true about the types of classes.

A full description of the fields is given below:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Field** | **Description** | | emp\_id | Teacher's personnel number | | emp\_name | Instructor Name | | pos\_id | Job code | | position | Post | | salary | Salary | | experience | Experience, years | | dep\_id | Departmental code | | department | Name of Department | | subject | Discipline | | stud\_group\_id | Student group code | | class\_type | Format of classes | |

**Task**

1. Main task:
   1. Decompose the database schema to 3NF. Prepare a SQL DDL script for the new database tables. Create these tables in your database
2. Additional task:
   1. Add a table to save the history of changes to the position details (title and rate). To do this, copy the table of positions and add a timestamp to the PC. Try writing a few changes to a table using DML.
   2. Write a DDL script and DML

# Exercise 3

You have been asked to develop a database for a car-sharing application. The service itself is arranged as follows:

1. there are a number of cars with a known geolocation in the city;
2. the client finds and books a car;
3. before the start, checks it for damage and other flaws, after which he begins the trip;
4. The user finishes the trip and pays for it.

**There are many different cars in the company's fleet**. For each car, the model, color, VIN, state number and year of manufacture are known. Each new car is assigned a unique number.

The vehicles are associated with categories. Each category is known by its name and price per minute of rent. One car always belongs to only one category.

Only cars in the "Ready" status are available for booking through the app.

The machine has a geolocation set by latitude and longitude.

**To register** in the service, you need to specify your data and send it for verification. Each application for registration has a unique number and time of creation. The result of consideration of the application includes information on whether the person can drive.

The fields of the registration form are listed below.

|  |
| --- |
| Driver's license number |
| Driver's license issued date |
| Passport series |
| Passport number |
| Phone number |
| Surname |
| Name |
| Patronymic |

When **the user books a car**, the trip starts in the "Booking" state. If the driver decides to go by car, the status of the trip changes to "Active" and the start time of the trip is filled in. Before the start of the trip, you can take several photos of the car and/or write one comment.

The driver can cancel the trip, and then the status changes to "Canceled".

If the driver **has completed the trip**, the status of the trip is changed to "Completed", the time and coordinates of the end of the trip are recorded. The coordinates of the car at the end of the trip are updated.

After the rental is completed, the total amount to be paid is formed. Fines and additional payments (for example, for an unscheduled wash) are also added to the trip with an indication of the time of invoicing. Accruals are marked with a flag if they have been paid.

For accruals, there is a list with a number, name and description.

The amount to be paid with details is visible in the application in each completed trip.

## Task

1. Create a new ER diagram in ERDplus (<https://erdplus.com/>) based on the case description.
2. Make sure that all entities have attributes and that the cardinality of the relationships is set according to the description. You can add additional attributes to the model.
3. Create a relational schema based on ERD. Check its correctness. Specify the data types you need.
4. Generate a SQL DDL script.