# Tartu Narva College

Databases

NEXT

Author: Sergei Ivanov

Major: IT systems development

Subject: Data Bases

Teacher: Chen-Wan Lin

21.08.2023

Table of Contents

[Tartu Narva College 1](#_Toc147335530)

[Abstract 3](#_Toc147335531)

[Revision history 4](#_Toc147335532)

[Introduction 5](#_Toc147335533)

[Background Information 5](#_Toc147335534)

[Research Questions 5](#_Toc147335535)

[Objectives 5](#_Toc147335536)

[Methodology overview 6](#_Toc147335537)

[MySQL 6](#_Toc147335538)

[Introduction 6](#_Toc147335539)

[Keys 6](#_Toc147335540)

[Homework 7](#_Toc147335541)

[Part I 7](#_Toc147335542)

[References 10](#_Toc147335543)

# Abstract

This paper discusses how to create NEXT application and avoid entering data manually, but using databases instead, such as Prisma which is abstraction from MySQL, and provided some automation. For example, MySQL does not automatically escape queries, while Prisma, provided by MySQL, does so for a developer. Hence, Prisma helps the developer to protect its app against notorious SQL injections which are actual even till these days.

Apart from Prisma, we also implement various applications using MongoDB, which is known as NOSQL database. Keywords: Mongoose, mongod, MySQL, SQLite, PostgreSQL, NEXT, React, TypeScript, JavaScript, CSS, HTML, Layout, Response, Request, Props, Children, and many others.

# Revision history

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Organization/Point of Contact | Description of Changes |
| 1.0 | 29.09.2023 | Sergei Ivanov | Initialization of the paper |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tab. 1.: The log of changes committed to this document

# Introduction

## Background Information

The data base is not any server, but an organized collection of logically connected data gathered to carry out data analysis to support scientific or commercial research. In this paper, I aim to unravel the multifaceted nature of databases and explore their applications, functionalities, and integration with modern JavaScript frameworks.

One of the most fundamental aspects that distinguishes a database from CSV (comma separated values) is organization. It uses specific syntax to operate over data in terms of CRUD: Create, read, update, and delete. To define data structure,

Secondly, a database is not any data structure. The difference is that data structure is abstraction while a database implements it. For example, we can organize our data as a red-black tree, and then implement it using SQL.

Thirdly, there are server and serverless databases, which means that syntax might vary from database to another, as well as the interaction between a developer or a user and a database. To illustrate that, we can mention MySQL and SQLite: the server and serverless ones, respectively. The first one does require a server to host a data, commit changes, and maintain it, while the other one does not.

## Research Questions

My questions:

1. What is a database?
2. What is the purpose of a database?
3. What are applications of a database?
4. What databases exist?
5. What are examples of using databases?

## Objectives

My primary goal is to demonstrate how to use databases in combination with modern JS framework known as NEXT and built on top of React. In doing so, I explore various concepts related to databases and implement a few applications leveraging the functionality of Prisma, Mongoose, Oracle, mySQL, and SQLite.

Hence, my objectives are

1. Answer the research questions.
2. Introduce and demonstrate daemons (mongod).
3. Create servers and connect to databases.
4. Operate over data and introduce data structures that are implemented by database.

## Methodology overview

To achieve my goals, I have to conduct research on various data bases referring to academic papers and official documentations provided by Mongoose, NEXT, VERCEL, Prisma, and Oracle. Therefore, I can write my methods:

1. Research
2. Study
3. Analyze
4. Demonstrate
5. Build applications.

# MySQL

## Introduction

MySQL is lower-level RDBMS (Relational Database Management System) when compared to high-level ORM (Object Relational Modeling) such as Prisma, or ODM (Object Data Modeling) such as MongoDB. It is owned by Oracle company.

Remark. Even though MySQL is owned by Oracle company, Oracle DB is different database management system.

## Keys

In MySQL system, there the following commonly used keys:

1. Primary: for unique data; it is unique and does not contain NULL value.
2. Alternate: also, unique key used for complementing primary key.
3. Candidate: it is a unique identifier without unnecessary redundancy.
4. Foreign: It establish a link among tables; an attribute in a table that refers to the primary key.
5. Super: A composite candidate key,

# Homework

## Part I

My first homework assignment is that I must determine primary, alternate, super, candidate and foreign keys for the following tables:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| propertyNo | street | city | postcode | type | rooms | rent | ownerNo | staffNo | branchNo |
| PA14 | 16 Holhead | Aberdeen | AB7 5SU | House | 6 | 650 | CO46 | SA9 | B007 |
| PL94 | 6 Argyll St | London | NW2 | Flat | 4 | 400 | CO87 | SL41 | B005 |
| PG4 | 6 Lawrence St | Glasgow | G11 9QX | Flat | 3 | 350 | CO40 | - | B003 |
| PG36 | 2 Manor Rd | Glasgow | G32 4QX | Flat | 3 | 375 | CO93 | SG37 | B003 |
| PG21 | 18 Dale Rd | Glasgow | G12 | House | 5 | 600 | CO87 | SG37 | B003 |
| PG16 | 5 Novar Dr | Glasgow | G12 9AX | Flat | 4 | 450 | CO93 | SG14 | B003 |

Tab. 2: PropertyForRent

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| clientNo | fName | lName | telNo | prefType | maxRent | eMail |
| CR76 | John | Kay | 0207-774-5632 | Flat | 425 | [john.kay@gmail.com](mailto:john.kay@gmail.com) |
| CR56 | Aline | Stewart | 0141-848-1825 | Flat | 350 | [astewart@hotmail.com](mailto:astewart@hotmail.com) |
| CR74 | Mike | Ritchie | 01475-392178 | House | 750 | [mritchie@yahoo.co.uk](mailto:mritchie@yahoo.co.ukm) |
| CR62 | Mary | Tregear | 01224-196720 | Flat | 600 | [maryt@hotmail.co.uk](mailto:maryt@hotmail.co.uk) |

Tab. 3: Client

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ownerNo | fName | lName | address | telNo | eMail | password |
| CO46 | Joe | Keogh | 2 Fergus Dr, Aberdeen AB2 7SX | 01224-861212 | [jkeogh@lhh.com](mailto:jkeogh@lhh.com) | \*\*\*\*\*\*\*\* |
| CO87 | Carol | Farrel | 6 Achray St, Glasgow G32 9DX | 0141-357-7419 | [cfarrel@gmail.com](mailto:cfarrel@gmail.com) | \*\*\*\*\*\*\*\* |
| CO40 | Tina | Murphy | 63 Well St, Glasgow G42 | 0141-943-1728 | [tinam@hotmail.com](mailto:tinam@hotmail.com) | \*\*\*\*\*\*\*\* |
| CO93 | Tony | Shaw | 12 Park Pl, Glasgow G4 0QR | 0141-225-7025 | [tony.shaw@ark.com](mailto:tony.shaw@ark.com) | \*\*\*\*\*\*\*\* |

Tab. 3: PrivateOwner

As it can be seen, some keys are reused in other tables, and other ones are unique. Let us determine those keys:

1. Primary key: propertyNo, clientNo, ownerNo;
2. Candidate key: propertyNo, clientNo, ownerNo;
3. Alternate key: postcode, telNo, eMail;
4. Foreign key: ownerNo, staffNo, branchNo;
5. Super key: (propertyNo, street), (clientNo, fName, lName, rent, maxRent);
6. Regular keys: fName, lName, rent, maxRent, rooms, city, type, prefType, passwordHash.

Let us define the models for these tables, using SQL syntax:

CREATE TABLE PropertyForRent

(  
 propertyNo VARCHAR(255),

city VARCHAR(255),

postcode VARCHAR(255) UNIQUE,

type ENUM(‘house’, ‘flat’),

rooms INT,

rent INT,

ownerNo VARCHAR(255) UNIQUE,

staffNo VARCHAR(255),

branchNo VARCHAR(255),

PRIMARY KEY (propertyNo)

);

CREATE TABLE Client (

clientNo VARCHAR(255) UNIQUE,

fName VARCHAR(255),

lName VARCHAR(255),

telNo VARCHAR(255) UNIQUE,

prefType ENUM(‘house’, ‘flat’),

maxRent INT,

eMail VARCHAR(255) UNIQUE,

PRIMARY KEY (clientNo)

);

CREATE TABLE PrivateOwner (

ownerNo VARCHAR(255) UNIQUE,

fName VARCHAR(255),

lName VARCHAR(255),

address VARCHAR(255),

telNo VARCHAR(255) UNIQUE,

eMail VARCHAR(255) UNIQUE,

passwordHash VARCHAR(255),

PRIMARY KEY (ownerNo)

);

# References

1. Sergei Ivanov, Computer Hardware, 2023, Narva Tartu College
2. Vercel Inc., NEXT.js, 2023
3. MongoDB Inc., MongoDB, 2023
4. Oracle company, MySQL, Oracle, 2023