Databases Introduction MySQL

How do RDBMS work?

Mariyan Apostolov

mariyan.apostolov89@gmail.com

CODERDOJO <BULGARIA>

Table of Content

Data Management

When Do We Need a Database?

CODERDOJO
<BULGARIA>

Data Management

Teacher Diary

+ Date: 09.02.2019
+ Class: 4A
+ Student Name: Ivolyan
+ Subject: Mathematics
+ Term Mark: 6.00

	A	B	C	D	E	F
1	Teacher Diary					
2	Date	Class	Student Name	Subject	Term Mark	
3	09.02.2019	4A	Ivolyan	Mathematics	6.00	
4						

CODERDOJO
<BULGARIA>

Data Management

- Storing data is not the primary reason to use a database
- Excel storage runs into issues with
  - Size
  - Ease of updating
  - Access

CODERDOJO <BULGARIA>

FR

Data Management

- A database is an organized collection of related information
  - It uses rules on the contained data
  - Access to data is usually provided by a "system" (DBMS) database management

CODERDOJO <BULGARIA>

FR

RDBMS

- Relational Data Base Management System
  - Database management
  - It parses requests from the user and takes the appropriate action
  - Data is presented by relations – collection of tables related by common fields
  - MS SQL Server, DB2, Oracle and MySQL

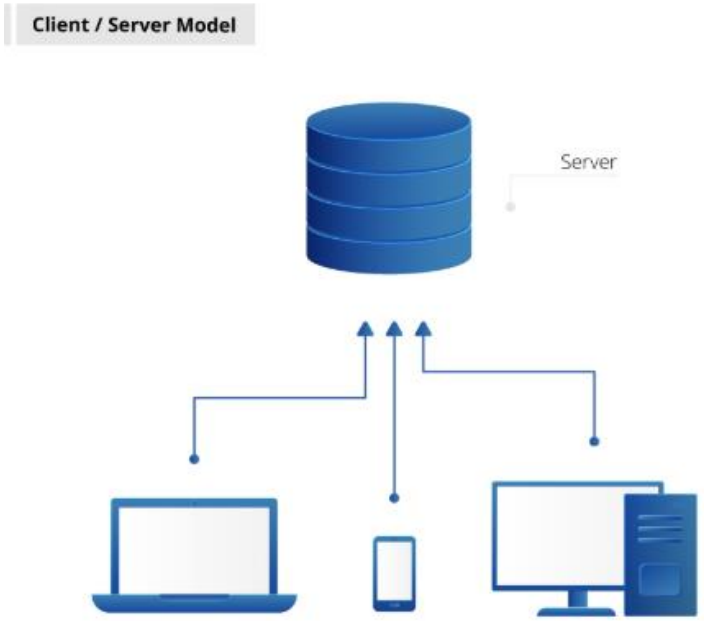
CODERDOJO <BULGARIA>

Database Engine

Client-Server Model

CODERDOJO
<BULGARIA>

Database Engine Flow



CODERDOJO
<BULGARIA>

Top Database Engines

Rank			DBMS	Database Model	Score		
Feb 2019	Jan 2019	Feb 2018			Feb 2019	Jan 2019	Feb 2018
1.	1.	1.	Oracle	Relational DBMS	1264.02	-4.82	-39.26
2.	2.	2.	MySQL	Relational DBMS	1167.29	+13.02	-85.18
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1040.05	-0.21	-81.98
4.	4.	4.	PostgreSQL	Relational DBMS	473.56	+7.45	+85.18
5.	5.	5.	MongoDB	Document store	395.09	+7.91	+58.67
6.	6.	6.	IBM Db2	Relational DBMS	179.42	-0.43	-10.55
7.	7.	8.	Redis	Key-value store	149.45	+0.43	+22.43
8.	8.	9.	Elasticsearch	Search engine	145.25	+1.81	+19.93
9.	9.	7.	Microsoft Access	Relational DBMS	144.02	+2.41	+13.95
10.	10.	11.	SQLite	Relational DBMS	126.17	-0.63	+8.89

Structured Query Language

Query Components

CODERDOJO
<BULGARIA>

Structured Query Language

- Programming language designed for managing data in a relational database
- Developed at IBM in the early 1970s
- To communicate with the Engine we use SQL

CODERDOJO <BULGARIA>

Structured Query Language

- Subdivided into several language elements
  - Queries
  - Clauses
  - Expressions
  - Predicates
  - Statements

```
UPDATE clause  UPDATE country
SET clause     SET population = population + 1
WHERE clause   WHERE name = 'USA';
```

Expression

Statement

Expression

Predicate

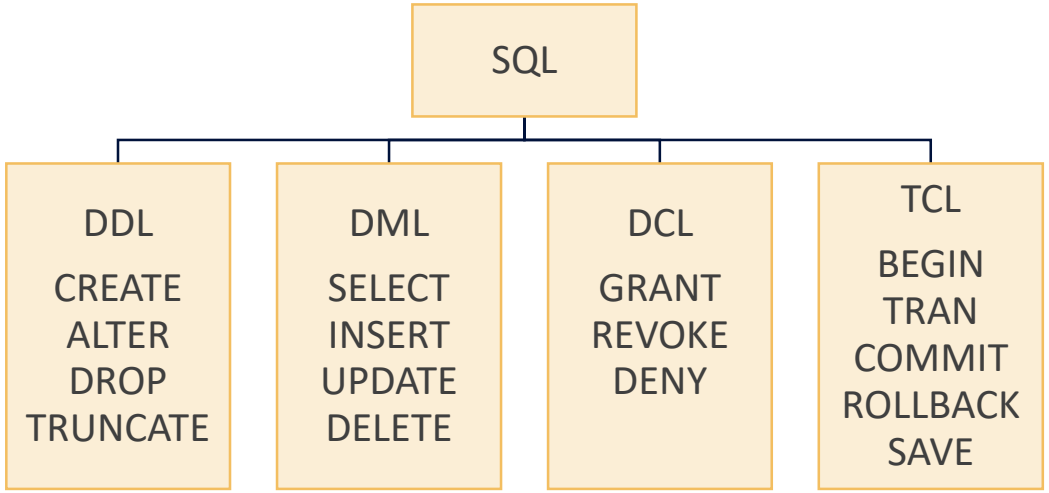
CODERDOJO <BULGARIA>

Structured Query Language

- Logically divided in four sections
  - Data Definition – describe the structure of our data
  - Data Manipulation – store and retrieve data
  - Data Control – define who can access the data
  - Transaction Control – bundle operations and allow rollback

CODERDOJO <BULGARIA>

Structured Query Language



MySQL

Relational DB Management

CODERDOJO
<BULGARIA>

MySQL

- Open-source relational database management system
- Used in many large-scale websites like Google, Facebook, YouTube etc.
- Works on many system platforms – Windows, Linux, MAC OS
- Download MySQL Server
  - Windows: https://dev.mysql.com/downloads/windows/installer/
  - Ubuntu/Debian: https://dev.mysql.com/downloads/repo/apt/

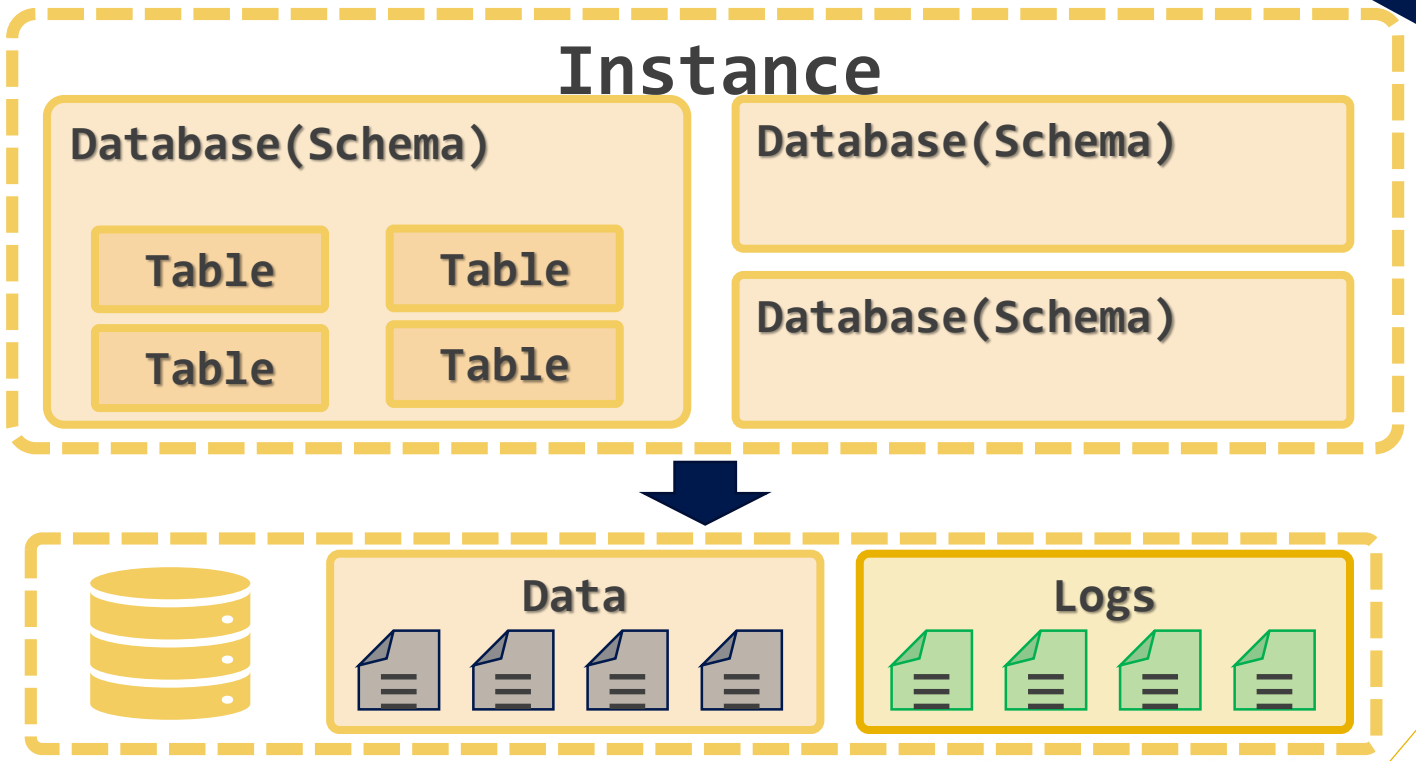
CODERDOJO <BULGARIA>

MySQL Server Architecture

- Logical Storage
  - Instance
  - Database/Schema
  - Table
- Physical Storage
  - Data files and Log files
  - Data pages

CODERDOJO <BULGARIA>

MySQL Server Architecture



CODERDOJO
<BULGARIA>

Database Table Elements

- The table is the main building block of any database
- Each row is called a record or entity
- Columns (fields) define the type of data they contain

CODERDOJO <BULGARIA>

Database Table Elements

Column

Row

Date	Class	Student Name	Subject	Term Mark
09.02.2019	4A	Ivolyan	Mathematics	6.00
09.02.2019	4A	Mariya	BEL	6.00
09.02.2019	4A	Atanas	Physics	6.00

Cell

CODERDOJO <BULGARIA>

Table Relationships

CODERDOJO
<BULGARIA>

Why Split Related Data?

Redundant Information

Date	Class	Student Name	Subject	Term Mark
09.02.2019	4A	Ivolyan	Mathematics	6.00
09.02.2019	4A	Mariya	BEL	6.00
09.02.2019	4A	Atanas	Mathematics	6.00

CODERDOJO <BULGARIA>

Related Tables

- We split the data and introduce relationships between the tables to avoid repeating information
- Connection via Foreign Key in one table pointing to the Primary Key in another

CODERDOJO
<BULGARIA>

Related Tables

Id	Date	Class Id	Student Name	Subject Id	Term Mark
1	09.02.2019	1	Ivolyan	1	6.00
2	09.02.2019	1	Mariya	2	6.00
3	09.02.2019	1	Atanas	1	6.00

Primary Key

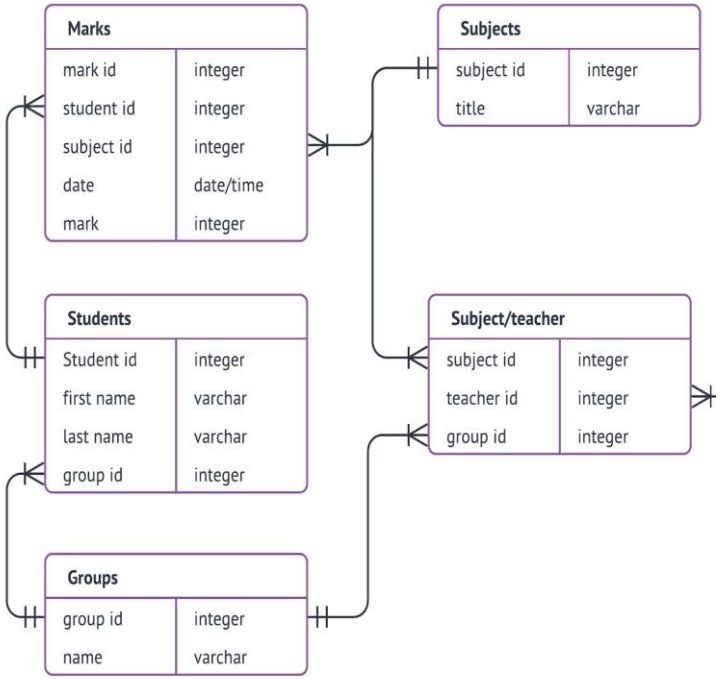
Foreign Key

Id	Class Name
1	4A
2	4B

Id	Subject Name
1	Mathematics
2	BEL

CODERDOJO <BULGARIA>

Table Relationships



CODERDOJO
<BULGARIA>

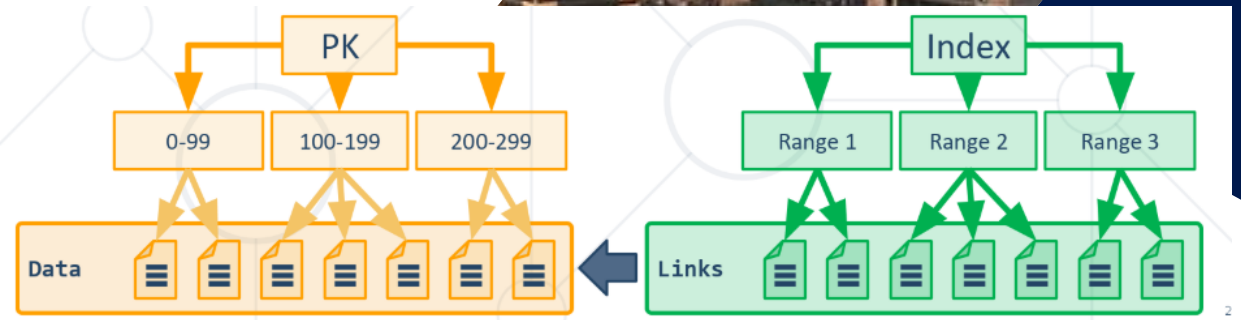
Programmability

Customizing Database Behavior

CODERDOJO
<BULGARIA>

Indices

- Indices make data lookup faster
  - Clustered – bound to the primary key, physically sorts data
  - Non-Clustered – can be any field, references the primary index
- Structured as an ordered tree



CODERDOJO <BULGARIA>

Views

- Views are prepared queries for displaying sections of our data
- Evaluated at run time – they do not increase performance

```
CREATE VIEW v_employee_names AS
SELECT e.employee_id,
       e.first_name,
       e.last_name
FROM soft_uni.employees AS e
```

```
SELECT * FROM v_employee_names
```

CODERDOJO <BULGARIA>

Procedures, Functions and Triggers

- A database can further be customized with reusable code
- Procedures – carry out a predetermined action
  - E.g. get all employees with salary above 35000
- Functions – receive parameters and return a result
  - E.g. get the age of a person using their birthdate and current date
- Triggers – watch for activity in the database and react to it
  - E.g. when a record is deleted, write it to an archive

CODERDOJO <BULGARIA>

Procedures

```
CREATE PROCEDURE udp_get_employees_salary_above_35000()
BEGIN
        SELECT first_name
             , last_name
         FROM employees
        WHERE salary > 35000;
END

CALL udp_get_employees_salary_above_35000
```

CODERDOJO <BULGARIA>

Functions

```
CREATE FUNCTION udf_get_age (dateValue DATE)
RETURNS INT
        BEGIN
        DECLARE result INT;
        SET result = TIMESTAMPDIFF(YEAR, dateValue, NOW());
        RETURN result;
        END

SELECT udf_get_age('1988-12-21');
```

CODERDOJO <BULGARIA>

Thank You.

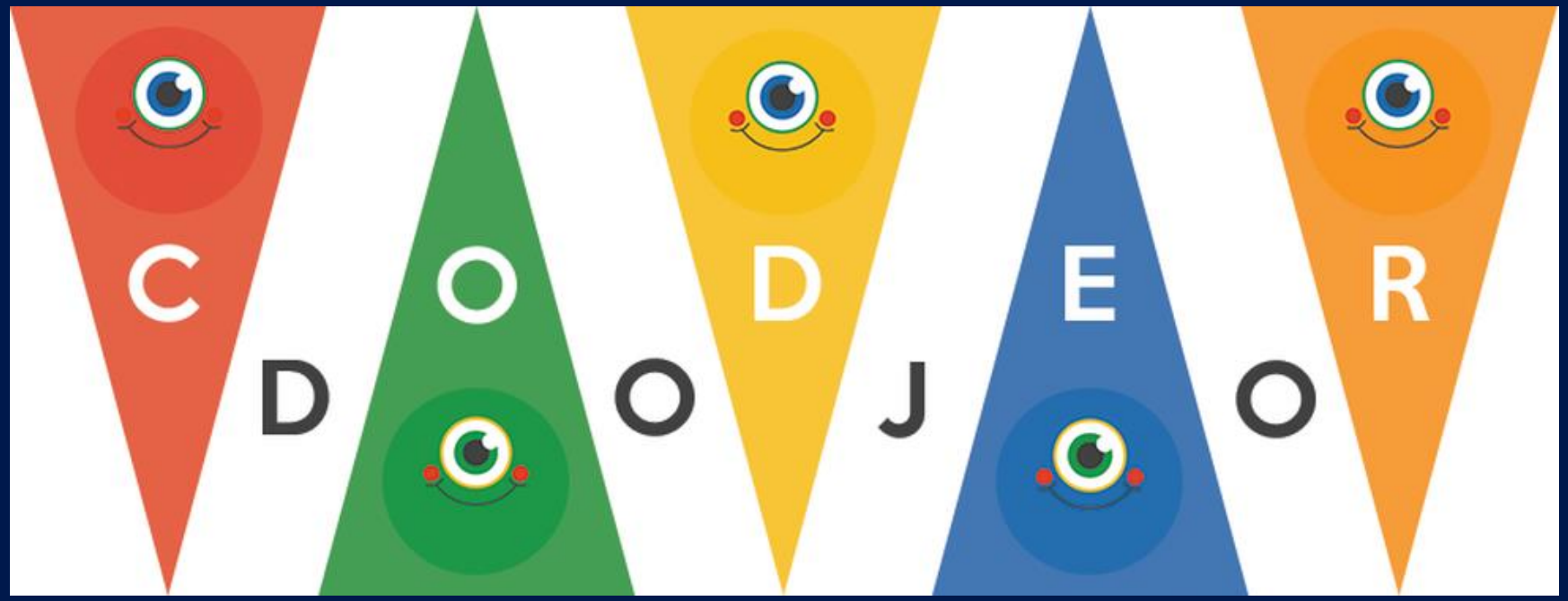
Mariyan Apostolov

mariyan.apostolov89@gmail.com

https://www.coderdojo.bg/

CODERDOJO
<BULGARIA>

FR



CODERDOJO
<BULGARIA>