

Practical Course Data Science: Database Management Systems (DBMS)

Assignments and Organisational Aspects

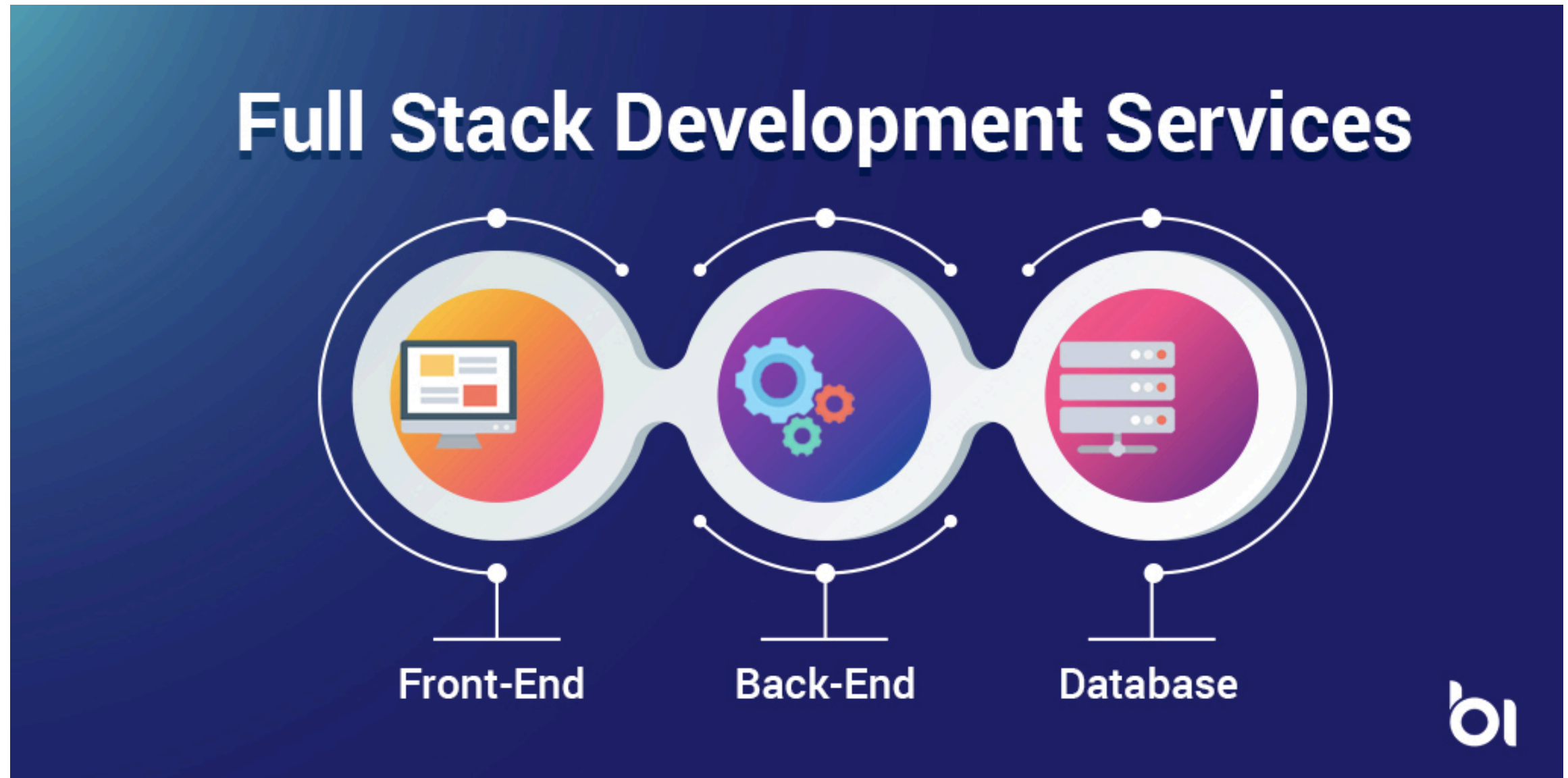
Prof. Dr. Lena Wiese
Dr. Ahmed Al-Ghezi

Database Technologies und Data Analytics Goethe-Universität Frankfurt a. M.

Agenda

- Overview
- Data set
- DBMS
- Assignments
- Self-study project
- Schedule

Full Stack Development



<https://binaryinformatics.com/>

Overview

- The course offers hands-on experience and a direct comparison of 4 DBMS.
- Online assignments: 4 tasks sheets.
- On site assignments: 2 tasks sheets.
- Project: code+on site presentation.

Enron Email Dataset

- By the CALO Project (A Cognitive Assistant that Learns and Organizes).
- Data from about 150 users, mostly senior management of Enron.
- A total of about 0.5M messages.
- Made public as result of government investigations.
- <https://www.cs.cmu.edu/~enron/>

DBMS

- PostgreSQL 
- Cassandra 
- Neo4J 
- MongoDB 

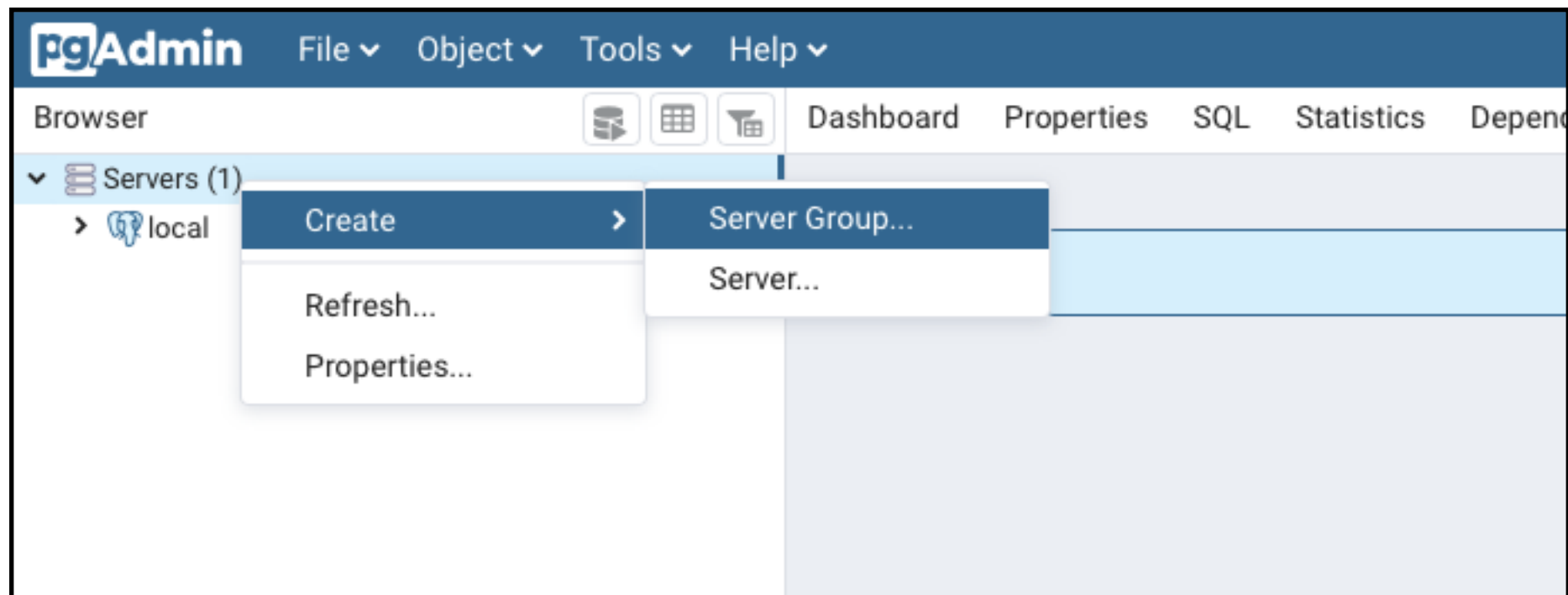
PostgreSQL



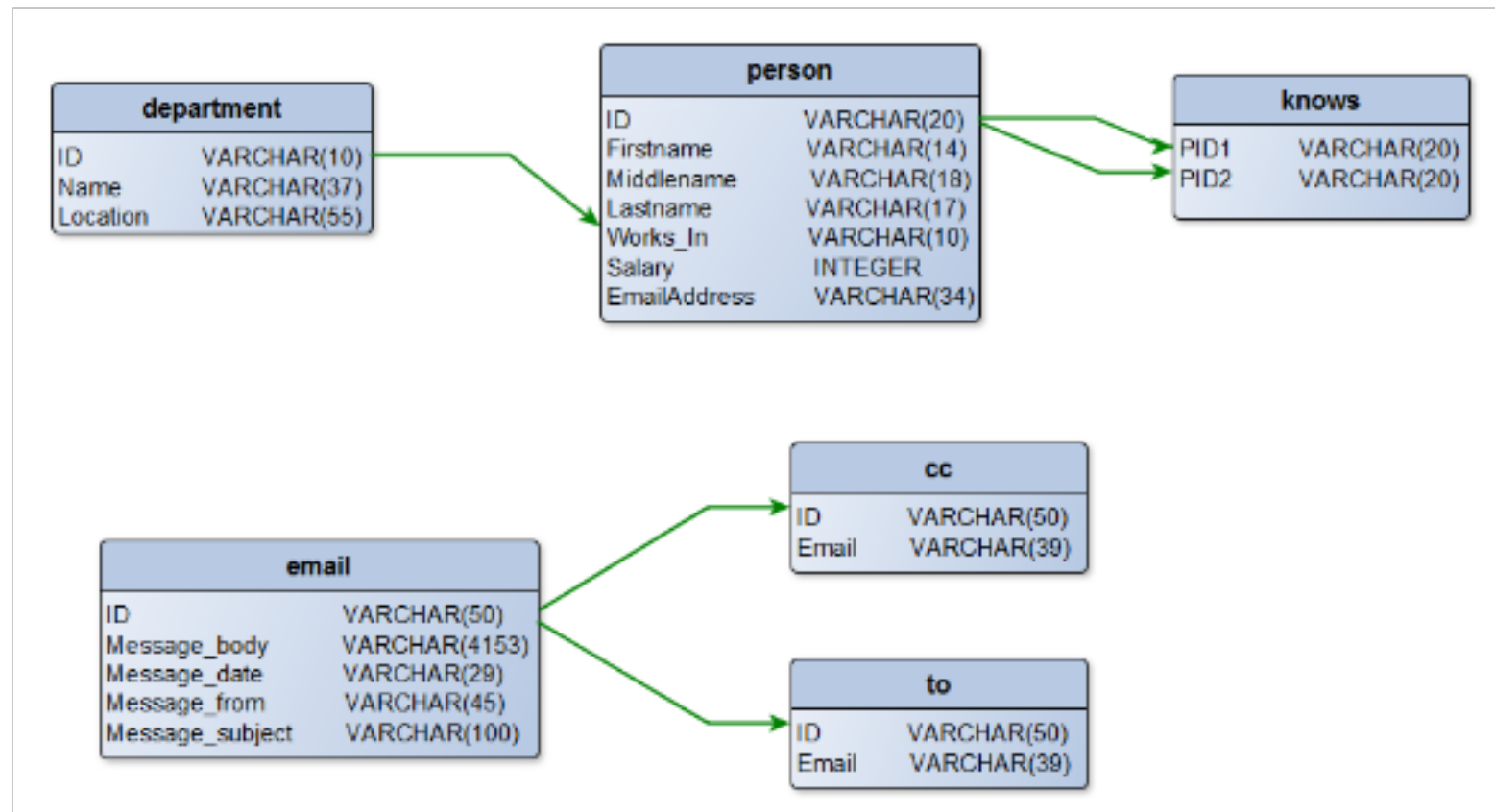
- Powerful relational database management system.
- Popular web-based client: pgAdmin.

PostgreSQL

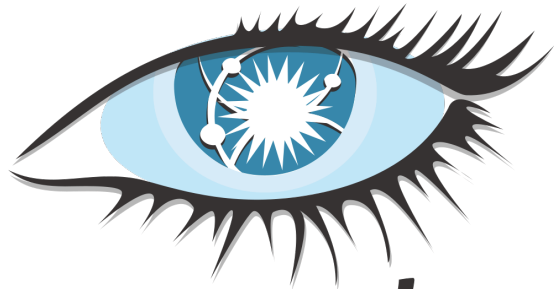
- You receive your credentials per Moodle messages
- Get access to the assignment sheet on Moodle
- Log in to our PgAdmin on: [http://\[REDACTED\]](http://[REDACTED])



Enron Relational Scheme



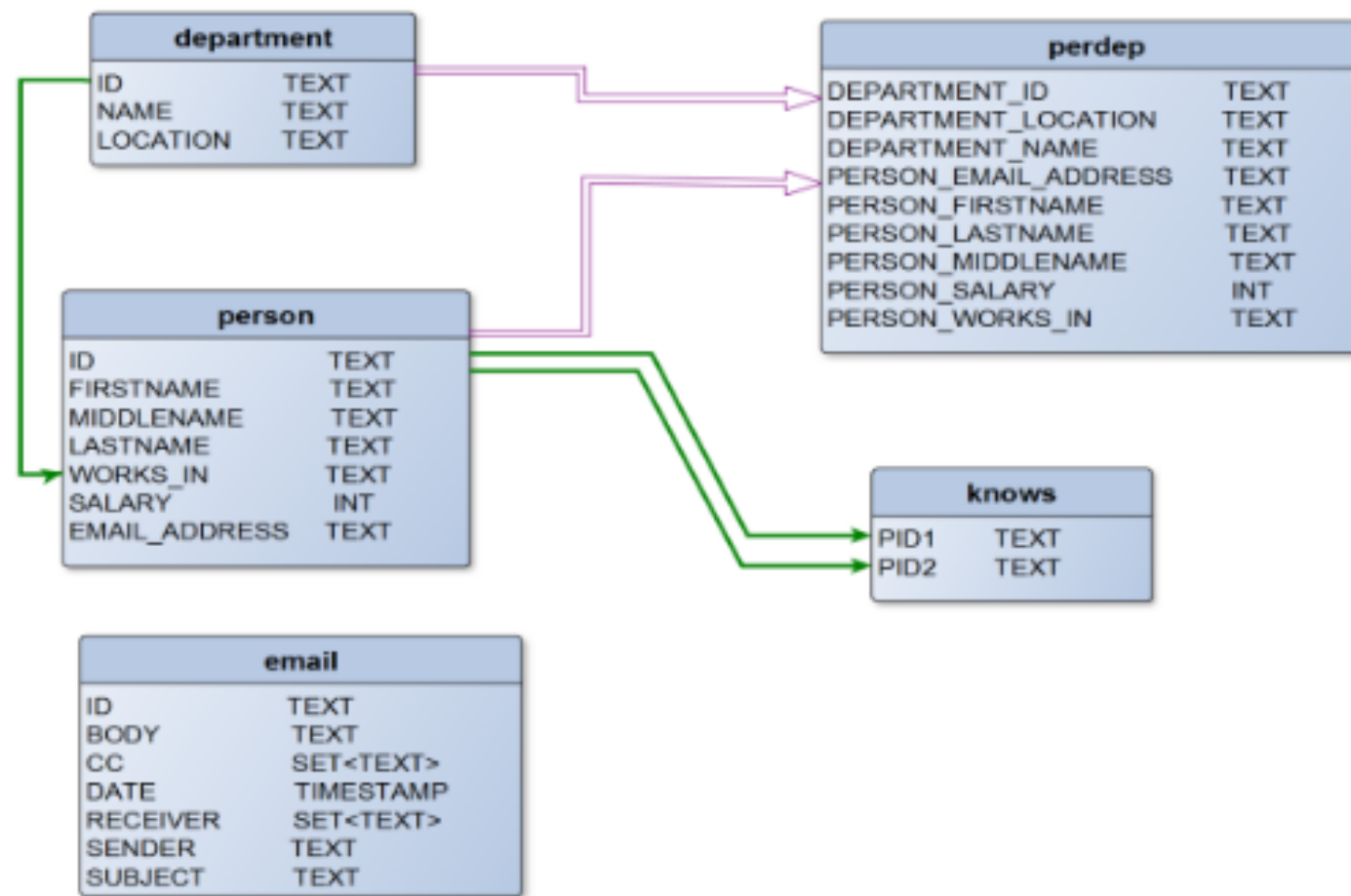
Apache Cassandra



cassandra

- NoSQL database management system.
- Distributed and easy to configure.
- Wide-column store (extensible record).
- Query Language: CQL
- No native support for join

Enron Keyspace



Accessing Cassandra

- Web client at: [http://\[REDACTED\]](http://[REDACTED])

The screenshot shows the Cassandra Web interface. At the top, there's a dark bar with 'Cassandra Web' on the left and 'Execute' with a refresh icon on the right. A red arrow points to the 'Execute' button. Below this, the 'Keyspaces' section on the left lists several keyspace names with their respective table counts in circles. The 'enron' keyspace is highlighted in blue, and a red arrow points to it. Below the keyspace list is the 'Hosts' section, showing a single host '138.201.68.27 (up)' in a green box. The main area displays 'Keyspace: enron' with two tabs: 'Tables' (selected) and 'Definition'. Below the tabs is a table listing the tables in the 'enron' keyspace. The table has two columns: 'Name' and 'Actions'. The tables listed are 'department', 'email', 'knows', 'person', and 'perdep'. Each table has a '- Delete' button in the 'Actions' column. At the bottom right of the table, there is a '+ Create' button. The text 'Total: 5 tables' is displayed below the table.

Name	Actions
department	- Delete
email	- Delete
knows	- Delete
person	- Delete
perdep	- Delete

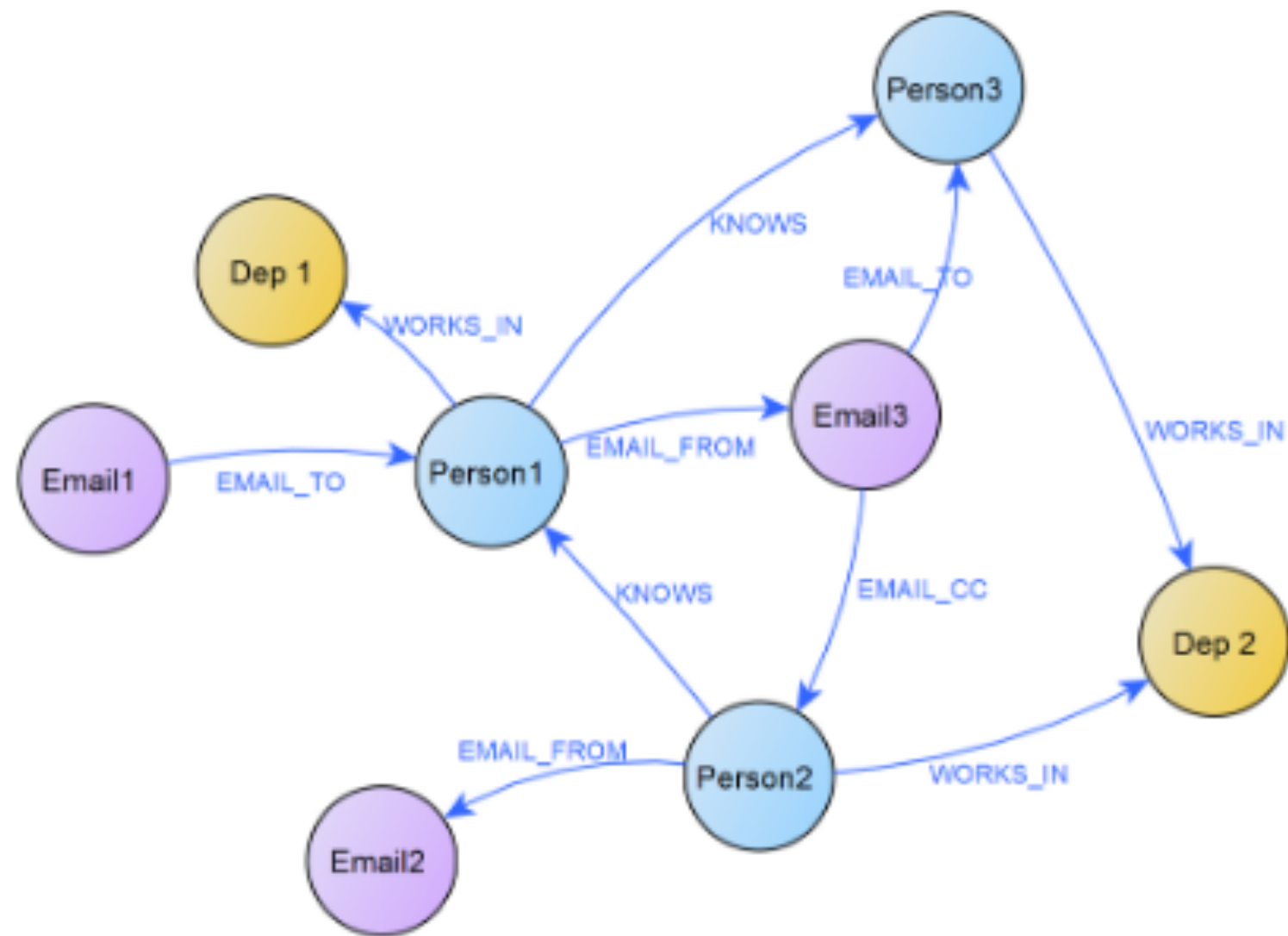
Total: 5 tables

Neo4J



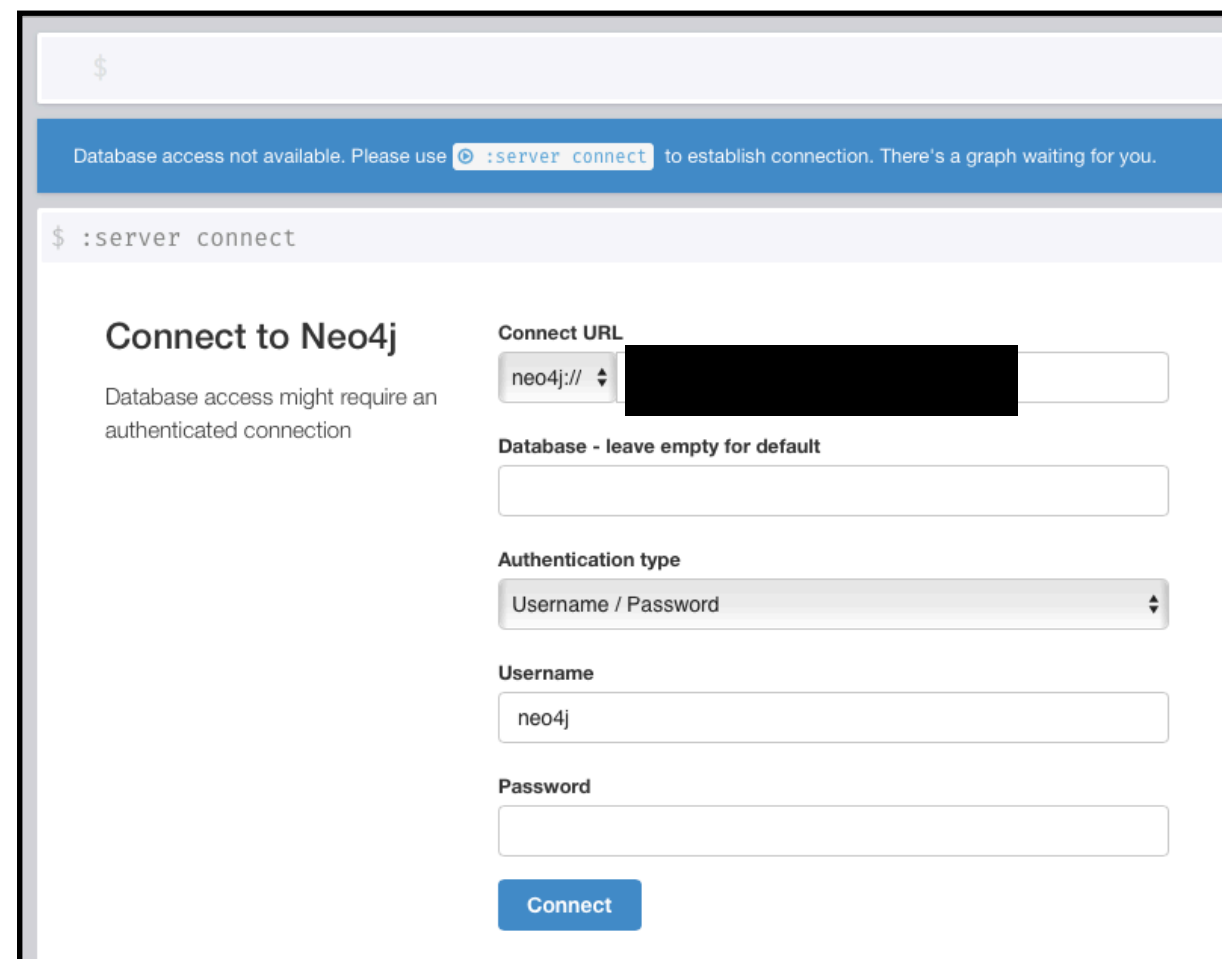
- Graph database management system.
- Everything is stored in the form of an edge, node, or attribute.
- Supports Cypher query language.
- Key words: MATCH, RETURN, WHERE, WITH, DISTINCT

Enron Graph



Accessing Neo4J

- Go to: [http://\[redacted\]/browser/](http://[redacted]/browser/)



The screenshot shows the 'Connect to Neo4j' dialog box in the Neo4j Browser. At the top, a blue banner states: 'Database access not available. Please use `:server connect` to establish connection. There's a graph waiting for you.' Below this, the terminal prompt '\$:server connect' is visible. The main section is titled 'Connect to Neo4j' and includes a note: 'Database access might require an authenticated connection'. The form contains the following fields:

- Connect URL:** A dropdown menu showing 'neo4j://' followed by a redacted black box.
- Database - leave empty for default:** An empty text input field.
- Authentication type:** A dropdown menu with 'Username / Password' selected.
- Username:** A text input field containing 'neo4j'.
- Password:** An empty text input field.

A blue 'Connect' button is located at the bottom right of the form.

MongoDB



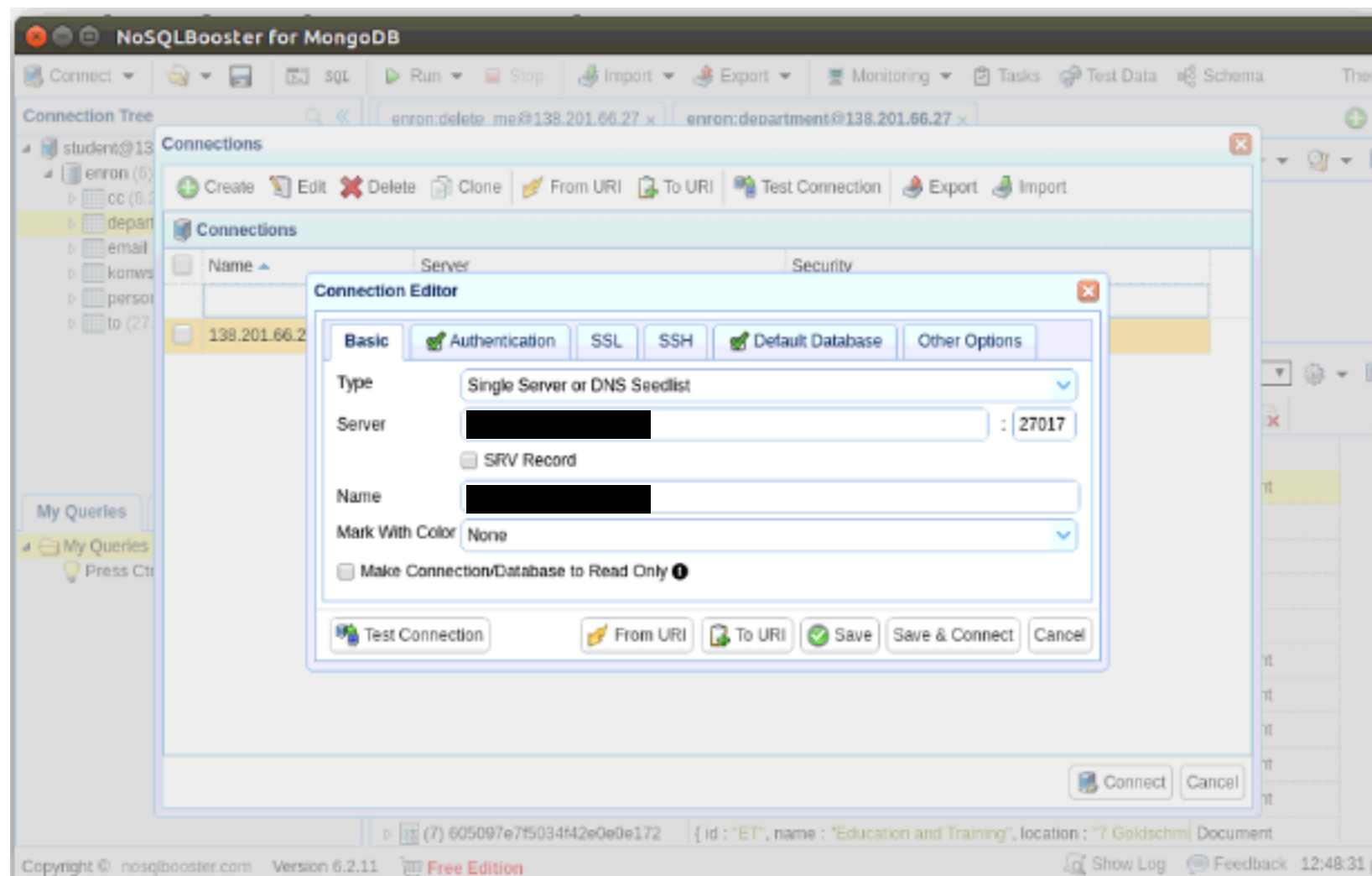
- NoSQL database system
- Document-oriented
- JSON-like documents with optional schemas
- Distributed
- It uses MongoDB Query Language (MQL)

Enron Collections

<p>email</p> <p>(3) ObjectId("5b0d270a") { 6 fields }</p> <table> <tr> <td> _id</td> <td>ObjectId("5b0d270a9587ce1de41e5c00")</td> </tr> <tr> <td> ID</td> <td><39889.1075846790345.JavaMail.evans@t</td> </tr> <tr> <td> MESSAGE_BODY</td> <td>Here are a few valuable documents for Edu</td> </tr> <tr> <td> MESSAGE_DATE</td> <td>2000-02-01 02:38:00+0000</td> </tr> <tr> <td> MESSAGE_FROM</td> <td>jennifer.rudolph@enron.com</td> </tr> <tr> <td> MESSAGE_SUBJECT</td> <td>To better serve</td> </tr> </table>	_id	ObjectId("5b0d270a9587ce1de41e5c00")	ID	<39889.1075846790345.JavaMail.evans@t	MESSAGE_BODY	Here are a few valuable documents for Edu	MESSAGE_DATE	2000-02-01 02:38:00+0000	MESSAGE_FROM	jennifer.rudolph@enron.com	MESSAGE_SUBJECT	To better serve	<p>person</p> <p>(2) ObjectId("5b0d") { 8 fields }</p> <table> <tr> <td> _id</td> <td>ObjectId("5b0d26b89587ce1de41dce14")</td> </tr> <tr> <td> id</td> <td>arnold-j</td> </tr> <tr> <td> firstname</td> <td>Ben</td> </tr> <tr> <td> middlename</td> <td>Luca</td> </tr> <tr> <td> lastname</td> <td>Arnold</td> </tr> <tr> <td> works_in</td> <td>HS</td> </tr> <tr> <td> salary</td> <td>65.000 (65.0K)</td> </tr> <tr> <td> email_address</td> <td>arnold@enron.com</td> </tr> </table>	_id	ObjectId("5b0d26b89587ce1de41dce14")	id	arnold-j	firstname	Ben	middlename	Luca	lastname	Arnold	works_in	HS	salary	65.000 (65.0K)	email_address	arnold@enron.com
_id	ObjectId("5b0d270a9587ce1de41e5c00")																												
ID	<39889.1075846790345.JavaMail.evans@t																												
MESSAGE_BODY	Here are a few valuable documents for Edu																												
MESSAGE_DATE	2000-02-01 02:38:00+0000																												
MESSAGE_FROM	jennifer.rudolph@enron.com																												
MESSAGE_SUBJECT	To better serve																												
_id	ObjectId("5b0d26b89587ce1de41dce14")																												
id	arnold-j																												
firstname	Ben																												
middlename	Luca																												
lastname	Arnold																												
works_in	HS																												
salary	65.000 (65.0K)																												
email_address	arnold@enron.com																												
<p>department</p> <p>(3) ObjectId { 4 fields }</p> <table> <tr> <td> _id</td> <td>ObjectId("5b0d26fe9587ce1de41e5bf4")</td> </tr> <tr> <td> id</td> <td>CA</td> </tr> <tr> <td> name</td> <td>Communications and Arts</td> </tr> <tr> <td> location</td> <td>1253 McGill College, Montreal, Quebec, H3B 2Y5</td> </tr> </table>	_id	ObjectId("5b0d26fe9587ce1de41e5bf4")	id	CA	name	Communications and Arts	location	1253 McGill College, Montreal, Quebec, H3B 2Y5	<p>knows</p> <p>(2) ObjectId { 3 fields }</p> <table> <tr> <td> _id</td> <td>ObjectId("5b0d26e29587ce1de41e3826")</td> </tr> <tr> <td> pid1</td> <td>allen-p</td> </tr> <tr> <td> pid2</td> <td>arnold-j</td> </tr> </table>	_id	ObjectId("5b0d26e29587ce1de41e3826")	pid1	allen-p	pid2	arnold-j														
_id	ObjectId("5b0d26fe9587ce1de41e5bf4")																												
id	CA																												
name	Communications and Arts																												
location	1253 McGill College, Montreal, Quebec, H3B 2Y5																												
_id	ObjectId("5b0d26e29587ce1de41e3826")																												
pid1	allen-p																												
pid2	arnold-j																												
<p>to</p> <p>(2) ObjectId { 3 fields }</p> <table> <tr> <td> _id</td> <td>ObjectId("5b0d26cb9587ce1de41dceac")</td> </tr> <tr> <td> id</td> <td><11703025.1075855236918.JavaMail.evans@thyme></td> </tr> <tr> <td> email</td> <td>kay.mann@enron.com</td> </tr> </table>	_id	ObjectId("5b0d26cb9587ce1de41dceac")	id	<11703025.1075855236918.JavaMail.evans@thyme>	email	kay.mann@enron.com	<p>cc</p> <p>(2) ObjectId { 3 fields }</p> <table> <tr> <td> _id</td> <td>ObjectId("5b0d26ef9587ce1de41e3c0f")</td> </tr> <tr> <td> id</td> <td><11703025.1075855236918.JavaMail.evans@thyme></td> </tr> <tr> <td> email</td> <td>hallb@gtlaw.com</td> </tr> </table>	_id	ObjectId("5b0d26ef9587ce1de41e3c0f")	id	<11703025.1075855236918.JavaMail.evans@thyme>	email	hallb@gtlaw.com																
_id	ObjectId("5b0d26cb9587ce1de41dceac")																												
id	<11703025.1075855236918.JavaMail.evans@thyme>																												
email	kay.mann@enron.com																												
_id	ObjectId("5b0d26ef9587ce1de41e3c0f")																												
id	<11703025.1075855236918.JavaMail.evans@thyme>																												
email	hallb@gtlaw.com																												

MongoDB Client

- To connect to our MongoDB server you can use “NoSQLBooster”.
- <https://nosqlbooster.com/downloads>



Online Assignments

- Four assignments for the four database systems.
- Download assignment description files from Moodle.
 - <https://moodle.studiumdigitale.uni-frankfurt.de/moodle/course/view.php?id=1333>
 - Submit to Moodle before an assignment's deadline.
- Fill out a copy of the CSV template file with your solution.
- **Don't alter the file's structure or the columns' names.**
- Save it as **CSV** format with the name as your **first.last** names.

Task Timing



- Not all tasks are necessary solvable within the given system!
- For each task, there is a maximum working time.
- If you think you will not finish in time, stop working 10 minutes before time end. Provide explanation with answers to:
 - ◆ Is the task solvable within the current system at all, and why?
 - ◆ If it is solvable with more time, what approach you would try out next?

Onsite Assignments

- Two assignments for two database systems
- Instructions given on the lab
- Limited access to the internet

CSV File Template

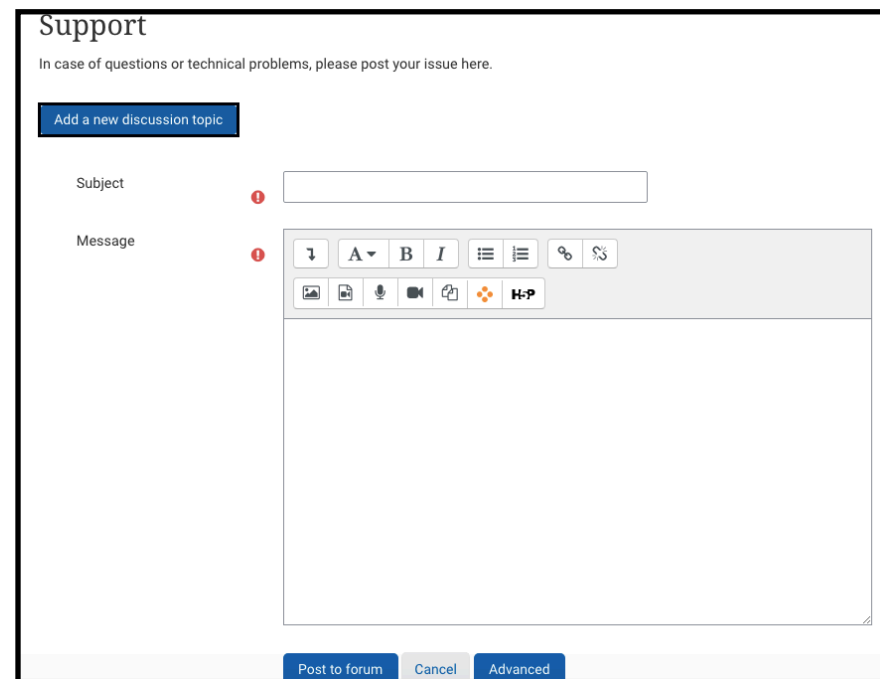
	Task	Executable query	Does the query return correct results?	Results size	Partial solution/Description	SpentTime (minutes)	How difficult is this task for you?
1							
2							
3							
4							
5							
6							
7							

Fair Usage

- The database systems are shared by all students
- Do not change the original data and its structure
- Create copies whenever applicable
- Delete anything you have created (tables, functions...) before you end your session
- Most of the tasks' queries are executed in a fraction of second
- In case of badly designed queries or user-defined functions with loops, the system may hangout or respond with a timeout error
 - ❖ Don't repeat by following trial and error approach, as this could make the system unresponsive to others

Support

- In case of technical issues, use the support forum in the Moodle.



The screenshot shows a Moodle forum interface titled "Support". Below the title is a subtitle: "In case of questions or technical problems, please post your issue here." There is a blue button labeled "Add a new discussion topic". Below this, there are two input fields: "Subject" and "Message". The "Subject" field has a red exclamation mark icon to its right. The "Message" field has a red exclamation mark icon to its right and a rich text editor toolbar above it. The toolbar includes icons for bold, italic, underline, list, link, unlink, and other formatting options. At the bottom of the form, there are three buttons: "Post to forum", "Cancel", and "Advanced".

- Moodle messages
- You could reach me by email: alghezi@uni-frankfurt.de

Overview

- Data set
- DBMS
- Assignments
- Self-study project
- Schedule

Selfstudy Assignment

- You apply your knowledge in implementing a small web-based application.
- Frontend has up to 4 interfaces.
- Backend provides the interface to a selected DB.

Selfstudy Assignment

- You can freely chose a technology (typically a database and at least one other programming/scripting language) to fulfil the requirements.
- Submit a PDF report
- Give a 15 minutes presentation of your work. Date and time to be set.

Schedule

Acceptance via Moodle messages	16.10.2023
You confirm via Moodle messages	17.10.2023
PostgreSQL:	18.10. - 30.10.2023
Cassandra:	01.11 - 15.11.2023
Neo4J:	16.11 - 30.11.2023
MongoDB:	01.12.- 15.12.2023
On site 1:	31.10.2023
On site 2:	28.11.2023
selfstudy assignment:	
Opening:	16.12.2023
Submission of code+report:	28.01.2024
Presentations:	01.02.2024

“Questions?”