

Database Management Systems Assignment 3

Published: 18.12.2025
Due: 18.1.26, 23:59

Introduction

The goal of this assignment is to design a web application centered on **movies**, with features tailored to a specific target audience.

For example, movie producers might be interested in analyzing the characteristics of blockbuster films, while fans of specific genres might want to explore their favorite actors' roles across different movies. This assignment encourages creativity in exploring a wide range of possibilities.

You are not required to implement the web application's interface for this assignment. Instead, the focus will be on designing and developing the database server. The frontend design will be documented using PowerPoint or Word, detailing the application's GUI and available features, specifying the backend queries for each page. No frontend coding is needed.

General Guidelines and Objectives

- Develop a demo web application related to movies, supported by a MySQL database schema.
- Source data from at least one of the following:
 - API.
 - Any other publicly available information (e.g., a .csv file).
- Populate the database with a minimum of 5,000 records across at least 5 tables.
- Implement at least 5 different queries, including:
 - 2 full-text queries.
 - 3 complex queries (e.g., nested queries, group by, aggregations, EXISTS).
- Utilize the MySQL server "mysqlsrv1.cs.tau.ac.il."
- Use Python version 3.14

Collaboration and DB Access

- Complete tasks in pairs.
- Follow [instructions](#) to create and link to a MySQL account, with one person from each team responsible for account creation.
- Perform all database interactions using the designated user.

Coding Guidelines

- Use Python to create the database and handle data insertion and updates.
- Python and SQL queries should be readable and documented. Use meaningful names and elegant solutions while avoiding code duplication.
- Write your SQL queries in a Python file. Each query should be in a separate function named *query_NUM*
- Treat the input parameters for your *query_NUM* functions as inputs provided by the user.
- If you use any external libraries, make sure they don't automatically create a schema or generate SQL queries.
- Validate that your source code is executable.
- Populate your database with data using Python code, you may not use manual insertions or updates. We will not execute the entire data insertion script, but we will check if it is valid.
- Handle errors appropriately.

Database Design

Design the database based on principles covered in class:

- Use meaningful names for tables, columns, indices, keys etc.
- Implement foreign keys when applicable.
- Leverage indices to optimize your queries.

Documentation

1. User manual:
 - Provide an overview of the application's functionality.
 - Present the designed application's features using PowerPoint or Word.
 - Make sure you clearly show a mockup of your application, with all its features.
2. System documentation:
 - Describe the database schema structure.
 - Provide reasoning for the chosen database design, considering efficiency and drawbacks of alternative designs.
 - Explain why you chose to create your indices, and which queries they are designed to help.
 - Detail the five main queries, their purpose and how the database design supports them.
 - Outline code structure and API usage.

Submission Instructions

When you submit your assignment, your DB should be populated with data. We will not run the Python code that populates your DB, but the code should be included in the submissions files, and it will be graded.

Submit source code and documentation in a single ID1-ID2.zip file with the following structure:

1. src/
 - create_db_script.py - Contains code responsible for creating the database.
 - api_data_retrieve.py - Handles API retrieval (if any) and data insertion.
 - queries_db_script.py - Includes functions for your DB queries (*query_NUM*).
 - queries_execution.py - Includes the main function and provides example usages of your queries from queries_db_script.py with invocation parameters.
2. /documentation/
 - name_and_id.txt - Team members IDs and names.
 - user_manual.pdf - See item [1](#).
 - system_docs.pdf - See item [2](#).
 - mysql_and_user_password.txt - The MySQL user and password you were assigned.
3. requirements.txt - Python requirements file.