# DSCI 551 – Fall 2025

## Homework 3: Data modeling and SQL

**Due: 11:59pm, October 20, 2025, Monday**

**100 points**

**NO Late Submissions will be accepted!**

**Huggingface** (<https://huggingface.co/>) is a popular Github-like repository for sharing machine learning (ML) models (as well as datasets and apps). In this homework, we consider a database “**551face**” for storing ML models, users who contribute the models, files for the models, and history of downloads (which users downloaded which files).

**A screenshot of a computer

AI-generated content may be incorrect.**

Specifically, you are provided with an SQL script “551face.sql” that creates and populates the tables in the database. The tables are:

* **User**: stores use id, name and email address.
* **Model**: stores model id, name, task the model is used for (e.g., language understanding), the number of parameters (in millions) in the model, and the id of users who contributed and owned the model.
* **File**: stores file id, path, size (in MB), and the id of the model this file belongs to.
* **Download**: tracks which users downloaded which files.

**Tasks:**

1. **ER model (20 points):** According to the tables provided to you, draw a corresponding ER model (using software, not by hand please). The ER model should capture the entities and relationships, attributes, keys, and the multiplicity of relationships. The model should be one that can be converted into the relations (i.e., tables given to you) by following the conversion process described in class.
2. **SQL (80 points):** For each of the following questions, write an SQL query to answer the question. For each question, include the query and query execution result in your submission. 10 points per question.
3. For each task, find out how many files there are for the task. Order the task in the descending order of the file counts. If there are multiple tasks with the same file count, show these tasks in the ascending order of task names.
4. Find out (names of) tasks which have at least two models available in the repository. Output **only** the task names in the ascending order.
5. **Without** using group by and aggregation, find out tasks which have at least two models available in the repository that were contributed by different users. Return (unique) task names.
6. Compute the average size of the files which are for the 'speech-recognition' task and have a path that ends with ".safetensors".
7. Using **CTE** (common table expression), find ids of files that have been downloaded most frequently. Note there may be multiple such files.
8. Find users who downloaded their own files. Return the user ids and file ids. Order the results first by user id and then by file id, and return only the **first 5** results.
9. Find (unique) ids of users who downloaded both files “f1” and “f2”.
10. Using **CTE**, find names of "language generation" models with the largest number of parameters.

**Submission**: Submit a docx/pdf file that contains your ER diagram, queries, and their results. Note that queries need to be shown in texts, not screenshots (so that we can copy extract your queries for execution if needed). But query results should be shown in screenshots. For example,

A screenshot of a computer

AI-generated content may be incorrect.