Data-Intensive Systems

Spring 2024

Lab 1 Description

DATA-INTENSIVE APPLICATIONS AND SYSTEMS

EPFL

SQL Developer – SQL IDE

- If not already set up in previous lab session

- Java client for the Oracle RDBMS
  - Available for download, ideally via the download link: https://www.oracle.com/tools/downloads/sqldev-downloads.html

- Most SQL IDEs work (that has JDBC connection –this should be standard):
  - DBeaver (open source): https://dbeaver.io/
  - JetBrains DataGrip (free for students): https://www.jetbrains.com/datagrip/
  - You may use other software if you prefer so

SQL Developer + Oracle DBMS

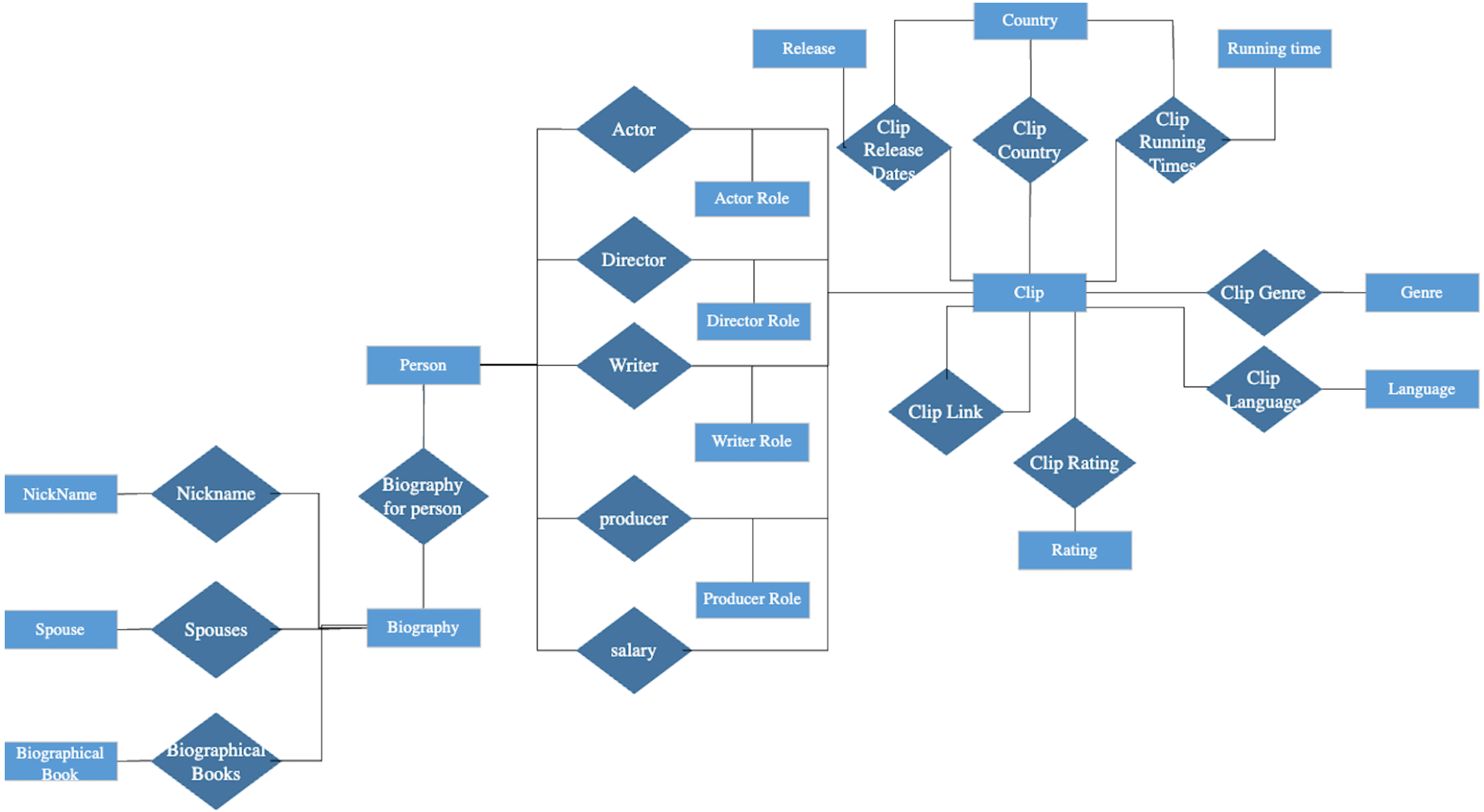
- Create a new connection with parameters:
  - username: C##DB2024_ST03
  - password: DB2024Lab1
  - hostname: cs322-db.epfl.ch
  - port: 1521
  - SID: ORCLCDB
  - Make sure you are connected to the EPFL network, or that you are using a VPN!

The IMBD Dataset

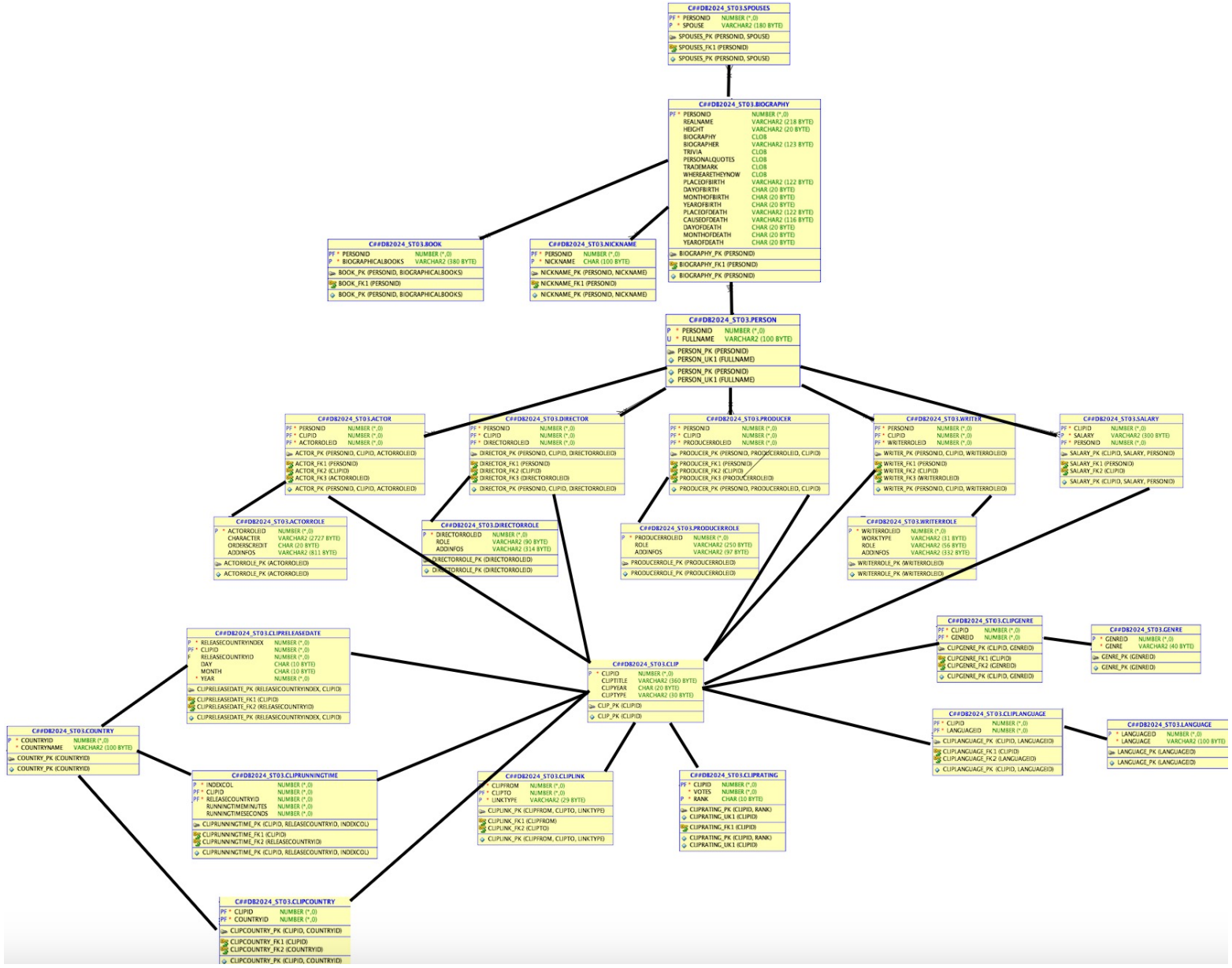
- Actors, Directors, Writers, Producers
- Clips:
  - rating, link, genre, language, country, running times, release dates

- Check lab 1 description for more details on the ER, attributes and values

The ER


Country
Release
Running time
Actor
Clip Release Dates
Clip Country
Clip Running Times
Actor Role
Director
Director Role
Clip
Clip Genre
Genre
Person
Writer
Clip Link
Clip Language
Language
Writer Role
Clip Rating
NickName
Nickname
Biography for person
producer
Rating
Producer Role
Spouse
Spouses
Biography
salary
Biographical Book
Biographical Books

The Database tables



Lab 1 Parts and Methodology

- Analyze the tables
- Understand the entities and the ER model
- Get familiar with the database

- Write a series of queries

The queries

- **Calculate the maximum number of clips any director has directed. Group by PERSONID from table DIRECTOR and print the PERSONID from table DIRECTOR of the director with the maximum number of clips that you calculated, and this maximum number of clips. Output should be in the format PERSONID, NUMBER_OF_CLIPS.**

- **Print the 10 most common clip languages. Order by LANGUAGE in a descending order. Output should be in the format LANGUAGE.**

- **Find the FULLNAME from table ACTOR of the actor/actress who has acted in more clips than anyone else. Calculate the number of clips he/she has acted in. Group by FULLNAME and print the FULLNAME from table ACTOR and the calculation. Output should be in the format FULLNAME, YOUR_CALCULATION.**

The queries

- **Print the CLIPTITLE from table CLIP, RUNNINGTIMEMINUTES from table CLIPRUNNINGTIME, and RUNNINGTIMESECONDS from table CLIPRUNNINGTIME, of the 10 clips with the longest duration (consider both the minutes and seconds of the clip) that were released in France. Use the RELEASECOUNTRYID from CLIPRUNNINGTIME table. Order by descending order according to duration. Output should be in the format CLIPTITLE, RUNNINGTIMEMINUTES RUNNINGTIMESECONDS.**

- **Compute the number of clips released per country. Print the 10 first rows grouped by countryname from table COUNTRY. Printed output should include the COUNTRYNAME from table COUNTRY, and NUMBEROFCLIPS that you will calculate. Order by COUNTRYNAME from table COUNTRY in descending order. Output should be in the format COUNTRYNAME, NUMBEROFCLIPS**

The queries

- **Compute the numbers of clips per genre released in the USA. Print the 10 first rows grouped by genre from table GENRE. Order by GENRE from table GENRE. Output should be in the format GENRE, YOUR_CALCULATION**

- **Compute the average number of votes of an actor's clips (for each actor) when she/he has a leading role (actorroleid is strictly smaller than 4). Group by PERSONID, print the PERSONID and the average votes for the first 10 rows. Order by PERSONID in a descending order. Output should be in the format PERSONID, AVERAGE_VOTES**

- **Print the FULLNAME from table PERSON of screenplay story writers who have worked with more than 2 producers. Group by FULLNAME and print the first 10 rows. Order by FULLNAME in a descending order. There should be no duplicates in the output. Output should be in the format FULLNAME**

The queries

- **Find the FULLNAME from table PERSON of the actors that are not married or have not been married in the past and have participated in more than 2 clips as “co-director”. Calculate the number of films they have participated in as “co-director”. Group by FULLNAME and print the fullname and the calculation for the first 10 rows. Order by FULLNAME in a descending order. Output should be in the format FULLNAME, YOUR_CALCULATION**

- **Compute the average votes from table CLIPRATING for the clips whose genre is the most popular genre. Print the CLIPID and the average votes for the first 10 rows. Order by CLIPID in a descending order. Output should be in the format CLIPID, AVERAGE_VOTES**

Individual Assignment

- The lab assignment is individual.
- The SQL code you write should be your own and should not be copied from someone else, or the internet.
- We will run plagiarism checks.

Lab 1 Deadline

Mon	Tue	Wed	Thur	Fri	Sat	Sun
Feb 19 **LEC 1:** Intro, overview: ER & Relational Model	Feb 20	Feb 21 **TUT 1:** SQL 1 **Lab session 1**	Feb 22	Feb 23	Feb 24	Feb 25
Feb 26 **LEC 2:** Relational Algebra & SQL **Lab 1**: : RDBMS use with SQL	Feb 27	Feb 28 **TUT 2:** SQL 2 **Lab session 2**	Feb 29	Mar 1	Mar 2	Mar 3
Mar 4 **LEC 3:** File Systems & File Layouts (DSM/NSM/PAX)	Mar 5	Mar 6 **TUT 3:** ER - Relational model (translation) **Lab session 3**	Mar 7	Mar 8	Mar 9	Mar 10
Mar 11 **LEC 4:** Storage hierarchy	Mar 12	Mar 13 **Lab session 4**	Mar 14	Mar 15	Mar 16	Mar 17
Mar 18 **LEC 5:** Indexes & Memory	Mar 19	Mar 20 **Lab session 5**	Mar 21	Mar 22	Mar 23	Mar 24
Mar 25 **LEC 6:** Midterm **DUE:** Lab1 **Lab 2**: : Buffer pool	Mar 26	Mar 27 **Lab session 6**	Mar 28	Mar 29	Mar 30	Mar 31
Apr 1 **LEC 7:** Spring break	Apr 2	Apr 3 **Lab session 7**	Apr 4	Apr 5	Apr 6	Apr 7
Apr 8 **LEC 8:** Hashing and Sorting & storage hierarchy	Apr 9	Apr 10 **Lab session 8**	Apr 11	Apr 12	Apr 13	Apr 14
Apr 15 **LEC 9:** Query Operators I	Apr 16	Apr 17 **Lab session 9**	Apr 18	Apr 19	Apr 20	Apr 21
Apr 22 **LEC 10:** Query Operators II	Apr 23	Apr 24 **Lab session 10**	Apr 25	Apr 26	Apr 27	Apr 28
Apr 29 **LEC 11:** Query Optimization **DUE:** Lab2 **Lab 3**: : Index	Apr 30	May 1 **Lab session 11**	May 2	May 3	May 4	May 5
May 6 **LEC 12:** Transactions and Concurrency Control & Concurrency I	May 7	May 8 **Lab session 12**	May 9	May 10	May 11	May 12
May 13 **LEC 13:** Concurrency Control and Eventual Consistency & Concurrency II	May 14	May 15 **Lab session 13**	May 16	May 17	May 18	May 19
May 20 **LEC 14:** No lecture	May 21	May 22 **Lab session 14**	May 23	May 24	May 25	May 26
May 27 **LEC 15:** Parallel and Distributed data systems **DUE:** Lab3	May 28	May 29 **Lab session 15**	May 30	May 31	Jun 1	Jun 2

Lab 1 Submission **25/3**

Lab 1 submission

- Write each query in a separate .sql file.
- For each query, use the prefix Q, for example, Q_1.sql is the 1st query.
- Then, save these files in a folder named S[sciper], e.g. S369567.

Lab 1 Grading Scheme

- Grades will be released ~3-4 weeks after the deadline

- Full points are awarded for a correct response in the queries:

Task	Number of points / 100
Points per query (x 10)	10
Total points for all queries	100

Questions

- Lab1 published – start from understanding the ER model
- Your frequent questions will be added to Ed
- Lab session on Wednesdays 17:15-19:00