DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Databases Project – Spring 2019

Contents

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Introduction

In this project the students will get a set of data files. Based on that data, they will i) design a database schema, ii) parse, clean, and load the data into a DBMS, iii) write and optimize queries, and, finally, iv) implement an interface that will access the database and offer an interactive experience querying a given dataset.

IMPORTANT: Read the whole document before starting doing any work.

Short Description of Project

The dataset contains data about Airbnb, which is a private, online marketplace and hospitality service company. The project is done in teams of 3 people. The project is separated into 3 milestones, which follow the material taught in the lectures. We have synchronized each milestone with the material of the lectures for your convenience.

The first milestone requires you to analyze the dataset and extract the E-R (Entity-Relationship) schema as well as getting acquainted with a DBMS. The second milestone requires you to express a simple set of queries on top of the loaded database. The goal of this part is to familiarize with data loading and the challenging task of data cleaning. You will also get to apply your SQL skills, and get a first intuition about how query performance is directly dependent on i) the way you formulate a query and ii) the logical and physical design of your database. Simultaneously, during this milestone you have to design a first version of the interface to query the dataset. Finally, in the third part of the project you will express a set of more sophisticated SQL queries, which you will also analyze to come up with a detailed description of the execution. In addition, during this milestone you will have to **fully implement** the interface.

For each of these milestones the students should prepare a document following the provided template which describes the completed work. The grading will be done based on the final report as well on a presentation and short discussion with the TAs. The final report should contain material about all the work done for the 3 milestones combined into one document**. The reports after the first two milestones are optional, while the final (3rd) deliverable is mandatory and gradable. However, only the teams that submit the intermediate reports will get feedback on their progress.**

IMPORTANT: Only the teams that deliver the intermediate milestone deliverables within deadline will receive feedback!

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Deliverable 1: Create ER model, Design & Create Schema

Deadline (to get feedback): 25/03/2019

The students will use the data from the following data files:

- berlin_calendar.csv
- berlin_listings.csv
- berlin_reviews.csv
- barcelona_calendar.csv
- barcelona_listings.csv
- barcelona_reviews.csv
- madrid_calendar.csv
- madrid_listings.csv
- madrid_reviews.csv

The goal of this deliverable is to design an ER model and a corresponding relational schema, and create the database tables in a database system. The organization of the data in files and the given description ***DOES NOT IMPLY*** an ER model or a relational schema. It is given to help the student understand the format of the data faster. Finally, a discussion about constraints and removing redundant information should be included in the project report.

In the 1st deliverable the students should:

1. Create an ER model for the provided data.
2. Design the database and the constraints needed to maintain the database consistent.
3. Provide the SQL commands to create the tables in a relational database system.
4. Describe their work in the form of a report which should contain an ER diagram, SQL DDL code for table creation, description of the data constraints, and justification of the design choices (in a few paragraphs). The report should be submitted as a single pdf file (**one pdf per group**).

Important Note: Before designing an E-R schema, understand the data and read carefully the notes given in the form of **FAQ** at the end of the project description. If you need any clarifications, ask the TAs during the project session or office hours.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Deliverable 2: Import Data. Basic SQL queries

Deadline (to get feedback): 29/04/2019

In this phase, students have to import the provided raw data into their database. Besides this initial loading phase, the students should accommodate the insertion of new data into any table using a user interface. Through this interface, users should also be able to perform simple queries over the data, e.g., search for a keyword in any table.

Besides this insert and search functionality, students have to implement the following queries in SQL:

1. What is the average price for a listing with 8 bedrooms?
2. What is the average cleaning review score for listings with TV?
3. Print all the hosts who have an available property between date 03.2019 and 09.2019.
4. Print how many listing items exist that are posted by two different hosts but the hosts have the same name.
5. Print all the dates that 'Viajes Eco' has available accommodations for rent.
6. Find all the hosts (host_ids, host_names) that have only one listing.
7. What is the difference in the average price of listings with and without Wifi.
8. How much more (or less) costly to rent a room with 8 beds in Berlin compared to Madrid on average?
9. Find the top-10 (in terms of the number of listings) hosts (host_ids, host_names) in Spain.
10. Find the top-10 rated (review_score_rating) apartments (id,name) in Barcelona.

In summary, in the 2nd deliverable the students should:

1. Parse the given data and import them in the created database as described in your 1st deliverable.
2. Implement (using SQL) the queries described above.
    a. Provide the SQL code as well as the first 5 rows (when applicable) of the result for each query.
    b. Note: Consider the use of indexes to accelerate long-running queries.
3. Start working on the interface to access and visualize the data. A website or a java application are good choices, but students are free to choose any technology they want.
    a. More information can be found on the “Interface” section of this document
    b. For this deliverable, some mockup screenshots of an interface and a first version of the search /insert queries it triggers in the backend suffice.
4. Extend the project report from the first deliverable with the description of the work done for the second deliverable and an explanation for the design choices. Include any changes to the design covered in the first deliverable, with justification of the changes. Include the screenshot of the interface and the description of the way search functionality is implemented in the application. The report should be submitted as a single pdf file.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Deliverable 3: Interesting SQL queries

Deadline: 31/05/2019

A series of more interesting queries should be implemented with SQL. In addition, performance of chosen three queries should be optimized by using indexes based on the query plans. Lastly, the interface should be implemented and finalized.

The queries to be implemented are:

1. Print how many hosts in each city have declared the area of their property in square feet. Sort the output based on the city name in ascending order.
2. The quality of a neighborhood is defined based on the number of listings and the review score of these listings, one way for computing that is using the median of the review scores, as medians are more robust to outliers. Find the top-5 neighborhoods using median review scores (review_scores_rating) of listings in Madrid. Note: Implement the median operator on your own, and do not use the available built-in operator.
3. Find all the hosts (host_ids, host_names) with the highest number of listings.
4. Find the 5 most cheapest Apartments (based on average price within the available dates) in Berlin available for at least one day between 01-03-2019 and 30-04-2019 having at least 2 beds, a location review score of at least 8, flexible cancellation, and listed by a host with a verifiable government id.
5. Each property can accommodate different number of people (1 to 16). Find the top-5 rated (review_score_rating) listings for each distinct category based on number of accommodated guests with at least two of these facilities: Wifi, Internet, TV, and Free street parking.
6. What are top three busiest listings per host? The more reviews a listing has, the busier the listing is.
7. What are the three most frequently used amenities at each neighborhood in Berlin for the listings with “Private Room” room type?
8. What is the difference in the average communication review score of the host who has the most diverse way of verifications and of the host who has the least diverse way of verifications. In case of a multiple number of the most or the least diverse verifying hosts, pick a host one from the most and one from the least verifying hosts.
9. What is the city who has the highest number of reviews for the room types whose average number of accommodates are greater than 3.
10. Print all the neighborhoods in Madrid which have at least 50 percent of their listings occupied at some date in year 2019 and their host has joined airbnb before 01.06.2017
11. Print all the countries that had at least 20% of their listings available at some date in year 2018.
12. Print all the neighborhoods in Barcelona where more than 5 percent of their accommodation’s cancelation policy is strict with grace period.

In total, in the 3rd deliverable the students should:

1. Accommodate all above queries by giving the corresponding SQL code.
    a. Note: Consider the use of indexes to accelerate your long-running queries.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

2. Select 3 queries from Deliverable 3, and accelerate them by using indexes. Explain the necessities of indexes based on the queries and the query plans that you can find from the system (you are free to select any 3 queries you like from the queries of the 3rd deliverable).
3. After the introduced optimizations, report the runtime of all queries in milliseconds and explain the distribution of the cost (based again on the plans) for the 3 queries selected in part 2.
4. Present the results of the queries.
5. Build an interface to run queries/insert data/delete data giving as parameters the details of the queries. **Read the "Interface" section of this document for more details.**
6. Complete the project report written for the previous deliverables by adding description of the queries and the interfaces, explanation for the design choices, analysis of the chosen queries, as well as the changes compared to the work described in the previous deliverables. The report should be submitted as a single pdf file.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

"Airbnb" data description

In this section, we present the data on which the project is based. Read carefully the data description, the FAQ and if in doubt ask the TAs for clarification. The data is stored in CSV (comma separated values) files.

berlin_listings.csv, barcelona_listings.csv, madrid_listings.csv

These files outlines the real estate listings for Berlin, Barcelona and Madrid.

1. id
   The unique listing identifier.
2. listing_url
   The URL of the listing.
3. name
   The name of the listing.
4. summary
   A small description of the listing.
5. space
   A small description of the space of the listing.
6. description
   A large description of the listing.
7. neighborhood_overview
   Description of the neighbourhood of the listing.
8. notes
   An extra note about the listing.
9. transit
   Description of the transportation to the listing.
10. access
    Specification of the accessibilities of household stuff, such as kitchen facilities.
11. interaction
    Description of whom/how to interact regarding the listing.
12. house_rules
    House rule specifications.
13. picture_url
    The URL to the picture of the listing.
14. host_id
    The unique host identifier.
15. host_url
    The URL of the host.
16. host_name
    The name of the host.
17. host_since
    The date that the host has started working with Airbnb.
18. host_about

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

A small description of the host.

19. host_response_time
    The amount of time within which the host responses.
20. host_response_rate
    The rate at which the host replies the messages.
21. host_thumbnail_url
    The URL to a thumbnail profile photo of the host.
22. host_picture_url
    The URL to a profile photo of the host.
23. host_neighbourhood
    The neighbourhood the host lives in.
24. host_verifications
    The way with which the host can be verified.
25. neighbourhood
    The neighbourhood where the listing is in.
26. city
    The city where the listing is in.
27. country_code
    The code of the country where the listing is in.
28. country
    The country where the listing is in.
29. latitude
    The latitude of the listing.
30. longitude
    The longitude of the listing.
31. property_type
    The type of the property.
32. room_type
    The type of the room.
33. accommodates
    The number of people that the listing can accommodate.
34. bathrooms
    The number of bathrooms that the listing has.
35. bedrooms
    The number of bedrooms that the listing has.
36. beds
    The number of beds that the listing has.
37. bed_type
    The type of the beds.
38. amenities
    The set of amenities that listing features.
39. square_feet

The area of the listings in square feet.

40. price
    The daily price of the listing. It is the price for the day when the data is collected. For the price for a particular date, please see the *_calendar.csv files.
41. weekly_price
    The weekly price of the listing.
42. monthly_price
    The monthly price of the listing.
43. security_deposit
    The amount of money for security deposit.
44. cleaning_fee
    The fee for cleaning.
45. guests_included
    The number of guests that the daily price covers.
46. extra_people
    The additional price to be paid for every extra guest in addition to the number of guests specified by the guests_included attribute.
47. minimum_nights
    The minimum number of nights to rent.
48. maximum_nights
    The maximum number of nights to rent.
49. review_scores_rating
    The rating score of the listing.
50. review_scores_accuracy
    The accuracy score of the listing.
51. review_scores_cleanliness
    The cleanliness score of the listing.
52. review_scores_checkin
    The checkin score of the listing (to quantify how easy the checkin is).
53. review_scores_communication
    The communication score of the host.
54. review_scores_location
    The location score of the listing.
55. review_scores_value
    The score on the value that the listing provides for the price.
56. is_business_travel_ready
    Whether the listing can be used for business travels.
57. cancellation_policy
    The cancellation policy of the listing.
58. require_guest_profile_picture
    Whether the listing requires a guest profile picture.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

59. require_guest_phone_verification
    Whether the listing requires guest phone verification.

berlin_reviews.csv, barcelona_reviews.csv, madrid_reviews.csv

These files outlines the reviews associated with the listings for Berlin, Barcelona and Madrid.

1. listing_id
   The identifier of the listing that is reviewed.
2. id
   The unqiue review identified.
3. date
   The date that the review has been written.
4. reviewer_id
   The uniquer reviewer identified
5. reviewer_name
   The name of the reviewer
6. comments
   The review.

berlin_calendar.csv, barcelona_calendar.csv, madrid_calendar.csv

These files outlines the day-by-day availabilities and prices of the listings for Berlin, Barcelona and Madrid.

1. listing_id
   The identifier of the listing whose availability and price information is given.
2. date
   The date on which the listing is available or not.
3. available
   Whether the listing is available or not.
4. price
   The price of the listing for the particular date.

You can find the data here: http://diaswww.epfl.ch/courses/db2019/project/airbnb.zip.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Interface

Here we describe the requirements for the interface. The goal of implementing this interface is to get hands-on experience with a technology used to connect and query a database from an application.

Functionality

In the context of this project we require the following functionality:

- Implement a query submission interface. This should not be a text box where SQL is typed. It should be a set of boxes and drop-down menus.
- The query submission interface should be able to accommodate all the queries requested in the context of this project. Make sure to accommodate a way to input parameters without displaying SQL code to the user.
- The results of the queries should be printed in a user-friendly manner -- not just a console printout of the results. Some options would be either visualizing results in a page-based format (e.g., like the biography box in a wiki page) or clean column printouts (clear column separation and easily legible text).
- The search box. Add a word-based search box, which you can use to perform keyword-based search without necessarily knowing the schema of the data (e.g., search for 'Frank'). The follow-up search functionality should allow the user to choose one of the results (e.g., by clicking on it or selecting a box next to it) and get more information about it.

Design

The design of the interface is left entirely up to you. As long as it covers the full functionality described above, the colors/size of windows, etc. does not play any role.

Implementation

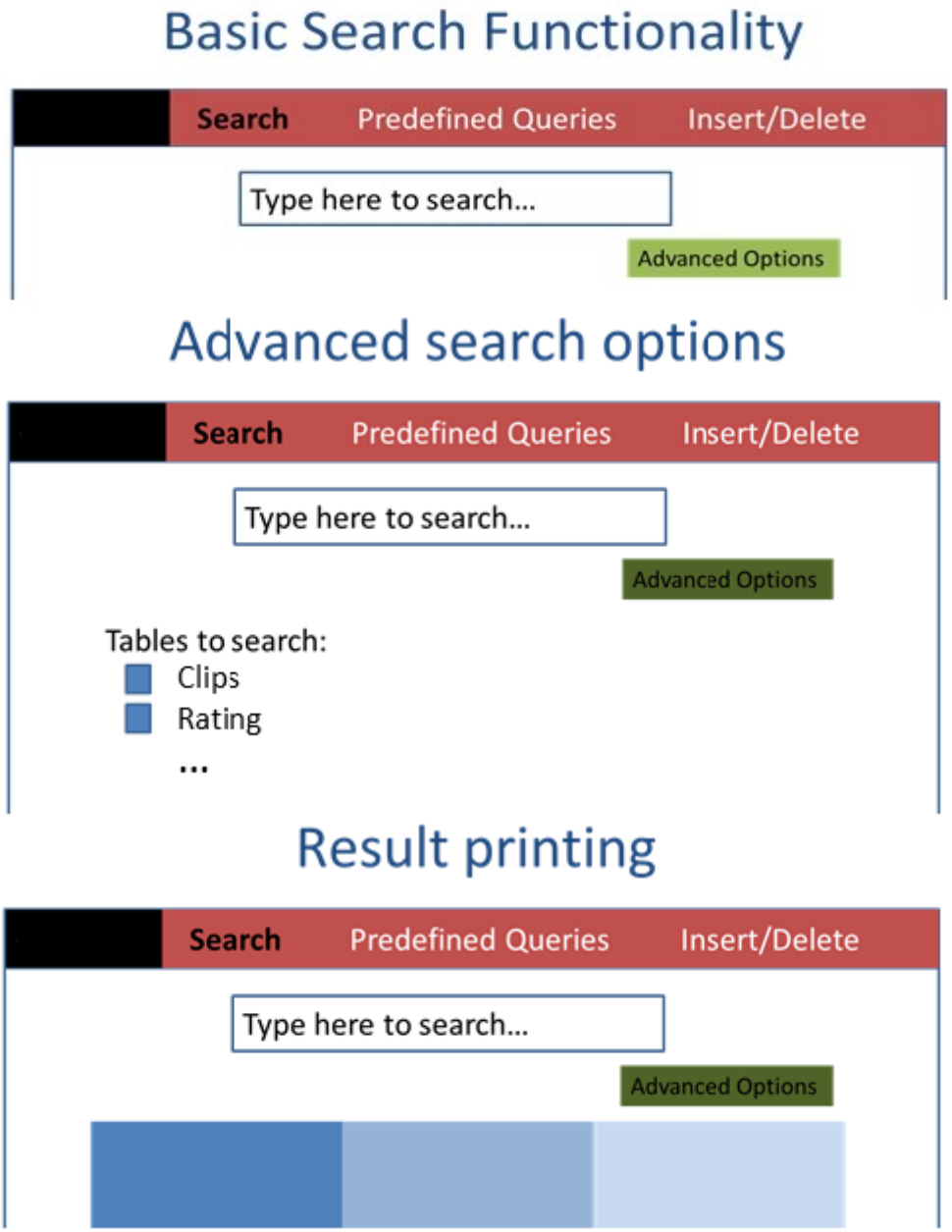
Students are free to choose any technology they want. In the past, your colleagues usually opted for Java-based applications or a website.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Interface Example

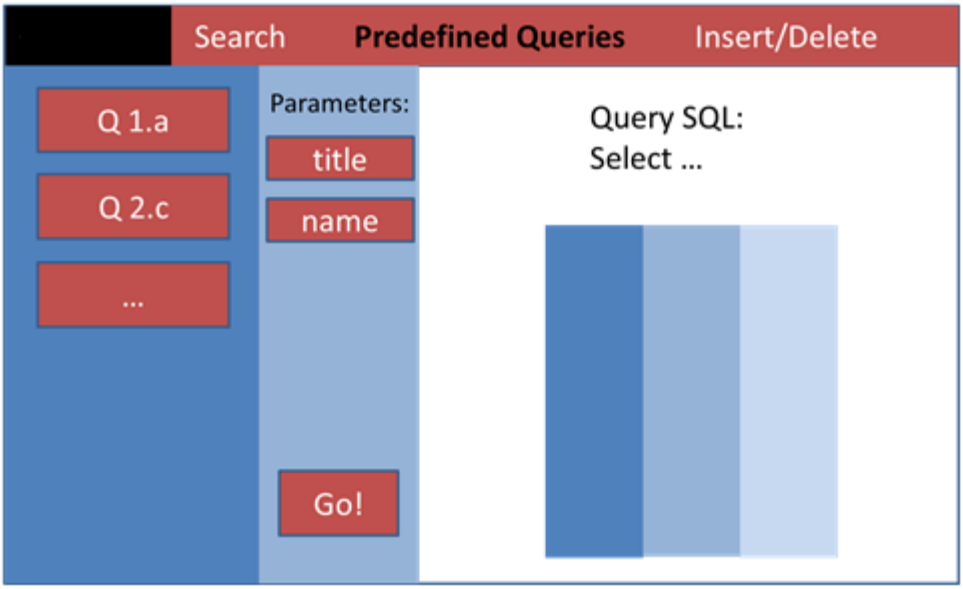
We now provide templates of the functionality expected from your interface. This template is meant as a mock-up of the three basic user interactions expected: i) search, ii) deliverable queries, iii) insertion/deletion. You are free to implement this functionality in any way you want.



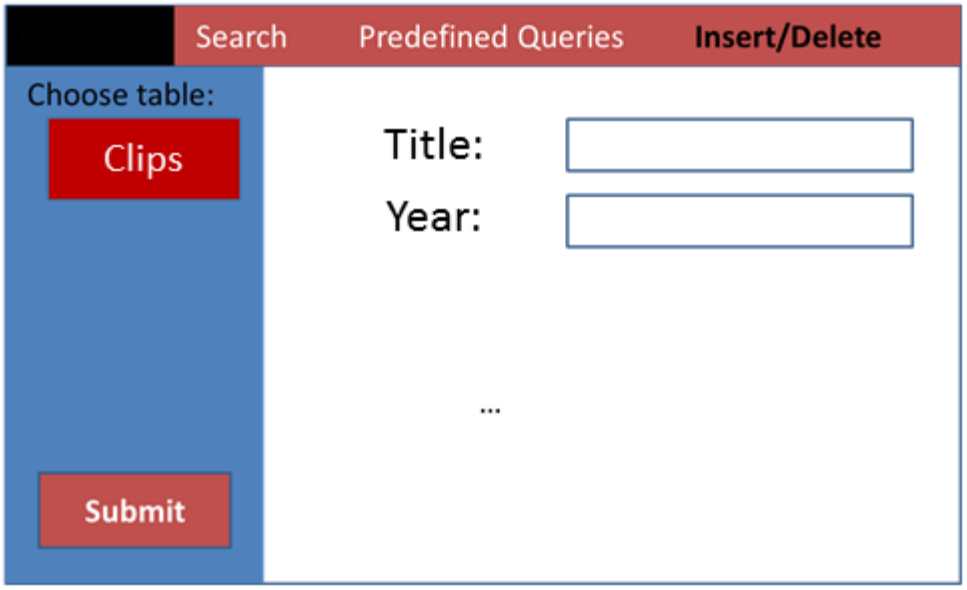
DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Predefined Queries



Insert Functionality



DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Frequently Asked Questions

How does one browse the data?

The dataset size is substantial, so it is hard to open most files using a notepad or text editor.
Applications such as Notepad++ and Sublime Text do a better job, but may still have issues with bigger files.
We thus also propose using Unix commands such as:

1. head: prints the first 50 lines of the file
2. less: allows backward movement in the file as well as forward movement
3. vi text editor: this editor does not open the whole file but only the part that is displayed

Which is the format of the given data?

The given data is CSV files (Comma Separated Values) which are values separated with comma (,). Each column represents a specific attribute. Usually in CSV files the name of the attribute is given in the first line of the file.

Why are the datasets "dirty"?

Real-world data is almost always dirty; missing values are commonplace; users abuse DBMS datatypes and store values based on their arbitrary, ad-hoc rules. We consider data cleaning to be a part of your project. Regarding how to perform data cleaning, there is more than one correct solution. Some possible ways are the following:

- Use Unix commands such as sed, grep, awk.
- Use your favorite scripting language, or a typical program that handles data inconsistencies. For example, Python, Java, and Scala all feature CSV parsers which you can use to read and transform the data.
- Load the data in a DBMS and then use DBMS functions to transform them based on your requirements.

Which database system should I use?

You are free to use any DBMS you want. Typical open-source examples are MySQL and PostgreSQL. We will also grant you access to an Oracle installation located on a server of the DIAS lab, which you can use. We will do our best to troubleshoot any issues with the Oracle server and help with issues related to your own installations.

Which character encoding should I set?

All files use utf-8 encoding. Take care of initializing your database using the correct encoding before creating tables or loading the data.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

What should I do if it takes too long to load the data?

The two most common reasons for a slow data loading process are the following:

1. Defining too many indexes/foreign key relations in your tables can delay loading significantly. **We therefore propose that you first create simple tables with only primary key properties, or without any constraints specified at all**. Once data is loaded, add the more complex table relations and indexes.
2. If you are using the database system provided by us, make sure that you are connected to the epfl network via cable (i.e., use a machine from the laboratory). If you connect via wi-fi or from home via vpn, it takes a long time to upload the data files, thus leading to large loading times.

An additional scenario is that you have set up your own database server (e.g., PostgreSQL or MySQL), and that the default resources allocated to it are very few. In that case, some useful links are the following:

- https://dev.mysql.com/doc/refman/5.5/en/innodb-buffer-pool.html
- http://www.rathishkumar.in/2017/01/how-to-allocate-innodb-buffer-pool-size-in-mysql.html
- https://wiki.postgresql.org/wiki/Tuning_Your_PostgreSQL_Server

What should I pay attention to?

1. **There is no intermediate grading**
   a. We still urge you to complete the milestones on time, so that you will not be overwhelmed at the end of the semester.
   b. The parts of the project are created in a way so that you will use the things you learn in the course and the exercise session and have hands-on experience.
   c. Every one of your deliverables should include the text from the previous ones too, explicitly updated to reflect the changes you made to address the feedback we gave you.
2. **Collaboration**
   a. We want you to collaborate
   b. We DO NOT CARE how you will split the work -> As long as you do equal parts of the work
   c. Writing the queries can (and should!) be done by everyone!
      i. You can solve the queries in multiple ways to find the optimal one!
3. The only important deadline on which you are graded is the last one **BUT** if you want feedback make sure to send us the milestone deliverables!

What is more important? The user interface or the actual "database work"?

The course concentrates on data management, not on HCI or web-based development. Therefore, it is by far more important to us that you concentrate on writing efficient queries, tuning your system, and deciding on which indexing structures are the appropriate ones for your needs. We will give a full grade to any user interface that covers the basic functionality we request.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Can I discard some data?

Dropping some erroneous values is acceptable. Under no circumstances, however, should you drop a significant chunk of the data. Whenever you drop some data, you should include the description of the dropped data and the reason for doing so.

How long should the deliverables be?

There is no strict page limit, as long as the deliverables report on the points we requested and are informative.

How should I choose my team?

Putting teams together is entirely up to you. Our advice is that every team member should be exposed equally to every task of the project. While, for example, it might appear tempting to a good frontend developer to focus on the user interface and quickly finish her assigned task, she will then be disadvantaged in the course midterm and final, because her SQL and query optimization experience will be limited.

What should I do if one of my teammates does not work?

We advise that you address the issue early on, before you encounter high load due to a deadline. We cannot be more lenient to such teams as a whole for fairness reasons. During the final project presentation, however, it becomes obvious whether a team member did not place equal effort; this student will get a lower grade.

When can I ask questions about the project?

The weekly project session is the intended place for questions. Otherwise, please use the moodle forum for questions that are of interest to your colleagues, too. Finally, every TA has specified office hours that you can use for further clarifications.