Databases Project – Spring 2019

Team No: 32

Names: Samuel Chassot, Daniel Nunes Silva, Sophie Ammann

Contents

Contents 1

Deliverable 1 2

Assumptions 2

Entity Relationship Schema 2

Schema 2

Description 2

Relational Schema 2

ER schema to Relational schema 2

DDL 3

General Comments 3

Deliverable 2 4

Assumptions 4

Data Loading 4

Query Implementation 4

Query a: 4

Description of logic: 4

SQL statement 4

Interface 4

Design logic Description 4

Screenshots 4

General Comments 4

Deliverable 3 5

Assumptions 5

Query Implementation 5

Query a: 5

Description of logic: 5

SQL statement 5

Query Analysis 5

Selected Queries (and why) 5

Query 1 5

Query 2 5

Query 3 5

Interface 6

Design logic Description 6

Screenshots 6

General Comments 6

# Deliverable 1

## Assumptions

1. We suppose that cities in this database have unique name inside their country. Hence we can use as primary key for the City entity.
2. A listing always has review scores, even if they are empty or *null*.

## Entity Relationship Schema

|  |  |
| --- | --- |
| **Entities** | **Description** |
| Listing | Represents a listing in an AirBnb service. |
| Host | Person that hosts a listing. |
| Neighbourhood | Part of a city. |
| City | City. |
| House\_properties | Properties of the accommodation. |
| Economic\_ properties | Costs related to the rent of the accommodation. |
| Administrative\_ properties | Rules related to the rent of the accommodation. |
| Review | Review in the Airbnb system of a listing. |
| Reviewer | Person who writes a review about a listing. |
| Review\_scores | Scores related to a listing of a review in different domains. |
| Calender | Availabilites of a listing. |
| Location | Location on a map of a listing. |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity** | **Relation** | **Entity** | **Constraints** |
| a Listing | has | House\_properties | one-to-one relationship (Listing‘s weak entity) |
| a Listing | has | Economic\_properties | one-to-one relationship (Listing‘s weak entity) |
| a Listing | has | Administrative\_properties | one-to-one relationship (Listing‘s weak entity) |
| a Listing | has | Review\_scores | one-to-one relationship (Listing‘s weak entity) |
| a Host | owns | a Listing | each listing has exatcly one host |
| a Listing | occupies | a Calender | a listing has availabilities in time. Each date has the corresponding listing's availability |
| a Review | reviews | a Listing | a review reviews exaxtly one listing |
| a Reviewer | writes | a Review | a review has exactly one reviewer |
| a Listing | is in | a Neighbourdhood | a listing is in exactly one neigbourhood |
| a Listing | is in | a City | a listing is in exactly one city |
| a Neigbourhood | is in | a City | a neigborhood is in exactly one city |

### Schema

<Add the figure of the ER schema>

### Description

<Describe all the choices you made for Entities and Relationships>

## Relational Schema

### ER schema to Relational schema

<Describe the transition from ER schema to Relational schema>

### DDL

<Provide the DDL>

## General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>

# Deliverable 2

## Assumptions

<In this section write down the assumptions you made about the data. Write a sentence for each assumption you made>

## Data Loading

## Query Implementation

<For each query>

### Query a:

#### Description of logic:

<What does the query do and how do I decide to solve it>

#### SQL statement

<The SQL statement>

## Interface

### Design logic Description

<Describe the general logic of your design as well as the technology you decided to use>

### Screenshots

<Provide some initial screen shots of your interface>

## General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>

# Deliverable 3

# Assumptions

<In this section write down the assumptions you made about the data. Write a sentence for each assumption you made>

## Query Implementation

<For each query>

### Query a:

#### Description of logic:

<What does the query do and how do I decide to solve it>

#### SQL statement

<The SQL statement>

## Query Analysis

### Selected Queries (and why)

#### Query 1

<Initial Running time:

Optimized Running time:

Explain the improvement:

Initial plan

Improved plan>

#### Query 2

<Initial Running time:

Optimized Running time:

Explain the improvement:

Initial plan

Improved plan>

#### Query 3

<Initial Running time:

Optimized Running time:

Explain the improvement:

Initial plan

Improved plan>

# Interface

### Design logic Description

<Describe the general logic of your design as well as the technology you decided to use>

### Screenshots

<Provide some initial screen shots of your interface>

# General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>