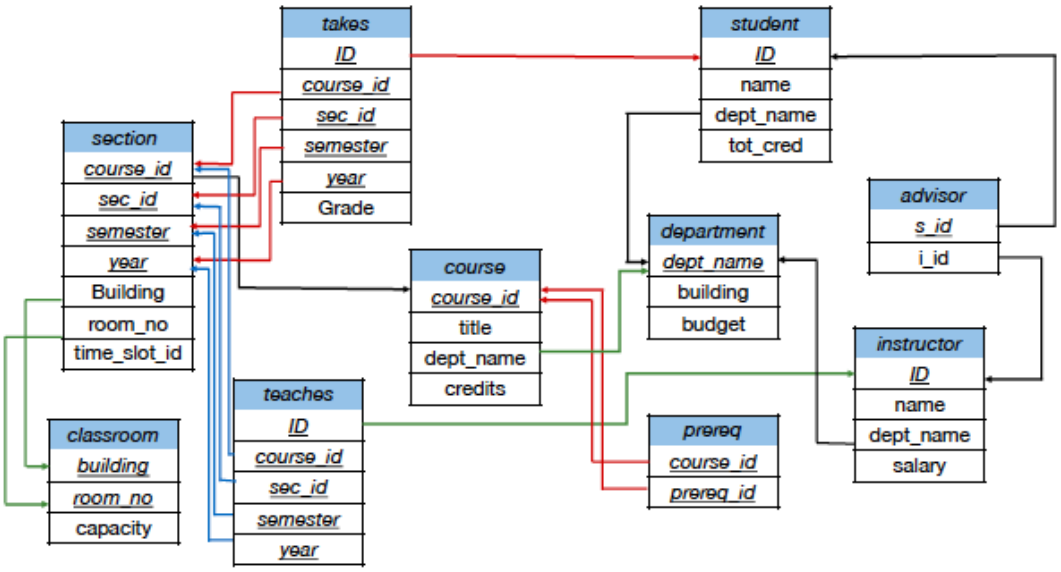
Assignment 2: AdvancedSQL, IC's & Data Integration

Group 6: Juan Moreno Díez (1840886), Leticia Marcal Russo (0664618), Luc Lubbers (1558501)



Question 1

Create the above database (with all speci ed primary and foreign keys), along with the following constraints:
(a) Values in the semester (attribute) should be one of ('Q1', 'Q2', 'Q3', 'Q4') and time slot id should be one of ('A','B','C','D')
(b) Values of the year should be greater than 2000.

SQL code can be found attached in the file: ***script.sql***

Question 2

Write queries to insert 4 records into the 'student','advisor' and 'instructor' tables. Then check your data by writing a query which returns the names of students and their advisors.

SQL code can be found attached in the file: ***script.sql***

```
SELECT S.name, I.name
FROM student S, instructor I, advisor A
WHERE A.s_id = S.ID AND A.i_id = I.ID;
```

The result of the query is the following:

Ana id = 1 and Karin id = 2
John id = 2 and Ivar id = 3
Hammed id = 3 and Matty id = 4
Maria id = 4 and Hans id = 1

So the query is showing correctly which advisor is linked with its student; PS: insertion queries can be found at the end of the ***script.sql***.

Question 3.
The Functional dependencies R(A,B,C,D,E,F,G) is given : F = ABD → EG, C→ DG, E → FG, AB → C, G → F. Find the candidate key for R.

Essential attributes: A, B (ones that are on the LHS that do not belong to the RHS)
Non-essential: C,D,E,F,G

= {A,B}
= {A,B,C} using AB -> C
= {A,B,C,D,G} using C -> DG
= {A,B,C,D,F,G} using G -> F
= {A,B,C,D,E,F,G} using ABD -> EG

AB is the possible candidate key because all attributes can be determined from them.

Question 4.
Relation schema r (A, B, C, D, E). Consider the table in csv ('fdExample.csv') file uploaded on the teams, if you know that (First name → Gender), write a python script that can check all the violations of such functional dependency.

52 violations were found, the code for finding those violations is included in the file: ***script_fd.ipynb.***

Question 5.
Find a tool (library or algorithm) that discovers functional dependencies in a given dataset, run the tool on a dataset and report the discovered FDs.

Source in which be based our answer ***https://github.com/nabihach/FD_CFD_extraction***
Steps:

- download the full repository
- make sure that the *fdExample.csv* file is in the same folder as where the *tane.py* is.
- Run the following command ***python3 tane.py fdExample.csv***
- The following output will appear

```
List of all FDs:  [['A', 'K'], ['A', 'G'], ['A', 'J'], ['A', 'I'], ['A', 'D'], ['A', 'B'], ['A', 'C
'], ['A', 'E'], ['A', 'F'], ['A', 'H'], ['G', 'K'], ['G', 'A'], ['G', 'J'], ['G', 'I'], ['G', 'D'],
 ['G', 'B'], ['G', 'C'], ['G', 'E'], ['G', 'F'], ['G', 'H'], ['B', 'C'], ['B', 'D'], ['E', 'B'], ['
B', 'E'], ['B', 'F'], ['B', 'I'], ['B', 'J'], ['B', 'K'], ['D', 'C'], ['C', 'D'], ['E', 'C'], ['F',
 'C'], ['C', 'F'], ['H', 'C'], ['I', 'C'], ['C', 'I'], ['J', 'C'], ['C', 'J'], ['K', 'C'], ['C', 'K
'], ['E', 'D'], ['F', 'D'], ['D', 'F'], ['H', 'D'], ['I', 'D'], ['D', 'I'], ['J', 'D'], ['D', 'J'],
 ['K', 'D'], ['D', 'K'], ['E', 'F'], ['E', 'I'], ['E', 'J'], ['E', 'K'], ['H', 'F'], ['I', 'F'], ['
F', 'I'], ['J', 'F'], ['F', 'J'], ['K', 'F'], ['F', 'K'], ['H', 'I'], ['H', 'J'], ['H', 'K'], ['J',
 'I'], ['I', 'J'], ['K', 'I'], ['I', 'K'], ['K', 'J'], ['J', 'K'], ['BH', 'A'], ['BH', 'G'], ['EH',
 'A'], ['EH', 'G']]
Total number of FDs found:  74
```

Question 6.

Compute the Levenshtein distance between \Computation" and \Completion" assuming the following costs of the operations:

1. each operation costs 1.
2. update (substitute) costs 2, other operations have cost of 1.
3. update (substitute) costs 3, other operations have cost of 1.

What do you expect if the cost of the update operation is > 3 while the cost of insert and delete remains 1? Explain your answer.

C	O	M	P	U	T	A	T	I	O	N
I	I	I	I	I	I	I	I	I	I	I
C	O	M	P	L	E	*	T	I	O	N
				S	S	D				

Each operation costs 1:
It would cost 3

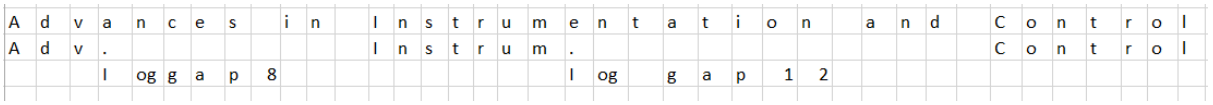
Update (substitute) costs 2, other operations have cost of 1:
It would cost 5

Update (substitute) costs 3, other operations have cost of 1:
It would cost 7

When the cost of the update (substitute) operation is >3, then it is not worth it to use. It would be convenient to insert and delete characters instead of substituting them.

Question 7.

Compute the gap distance between \Advances in Instrumentation and Control" and \Adv. Instrum. Control" assuming that (insertion cost = opengap cost = 1) and extend gap cost is 0.1.



1 + o + 8e + 1 + o + 12e
= 2 + 2o + 20e
= 2 + (2x1) + (20x0.1)
= 2 + 2 + 2
= 6

The gap distance in this case is 6.

Question 8.
Compute Jaccard Distance between each pair of the following three sets:
A = {0, 1, 2, 5, 6,}; B = {0, 2, 3, 5, 7, 9}; C = {2 ,3 ,5 ,6 }

A & B
Intersection = 3
Union = 8
Similarity = intersection / union = 3/8 = 0.375
Distance = 1 - (intersection / union) = 1 - 0.375 = 0.625

A & C
Intersection = 3
Union = 6
Similarity = intersection / union = 3/6 = 0.5
Distance = 1 - (intersection / union) = 1 - 0.5 = 0.5

B & C
Intersection = 3
Union = 7
Similarity = intersection / union = 3/7 = 0.43
Distance = 1 - (intersection / union) = 1 - 1.33 = 0.57

Question 9.
Compute Jaccard bag similarity and Jaccard Distance of each pair of the following sets:
A = {1,1,2,2,5}; B = {1,2,2,2,5,5}; C = {1,2,3,4,5}.

We talked to Hakim and he said we can consider Jaccard Distance using the concept of bag in this exercise. That is why we are considering the union to be the sum of all the elements.

A & B
intersection = 4

union = 11
Similarity = intersection / union = 4/11 = 0.36
Distance = 1 - (intersection / union) = 1 - 0.36 = 0.64

So the similarity for A & B will be 0.36 and the distance will be 0.64.

A & C
intersection = 3
union = 10
Similarity = intersection / union = 3/10 = 0.30
Distance = 1 - (intersection / union) = 1 - 0.30 = 0.70

So the similarity for A & C will be 0.30 and the distance will be 0.70.

B & C
intersection = 3
union = 11
Similarity = intersection / union = 3/11 = 0.27
Distance = 1 - (intersection / union) = 1 - 0.27 = 0.73

So the similarity for A & C will be 0.27 and the distance will be 0.73.

Question 10.
Compute the Jaro and Jaro-Winkler similarity between arnab and urban. What do you think is the reason for this result?

A R N A B
U R B A N

Jaro Similarity

$$\left\lfloor \frac{\max(|S_1|,|S_2|)}{2} \right\rfloor$$

(Max (5, 5)) / 2 = 5 / 2 = 2.5 - 1 = 1.5 = **1** (it will be 1, because it needs to be rounded down)
C = 2 (common character)
T = number transpositions/2 = 0 / 2 = 0
|S1| = 5
|S2| = 5

In this case, the transposition is 0, because we cannot move the letters more than 1 position.

Jaro Similarity is calculated by:

$$\frac{1}{3}\left(\frac{C}{|S_1|}+\frac{C}{|S_2|}+\frac{C-T}{C}\right)$$

Jaro Sim = ⅓ (2/5 + 2/5 + ((2-0) / 2)) = (1 / 3) * 1.8 = **0.6**

Jaro-Winkler
For Jaro-Winkler,we use this formula:

$$JaroSim + P * L * (1 - JaroSim)$$

where P is 0.1 by default and L is the length of the common prefix. In our case, we do not have common prefix, so L is zero. This means that Jaro-Winkler has the same value of Jaro, which is **0.6**.

Question 11.
What is the cardinality of the set that has 5-shingles of the following document \“Many problems can be expressed as finding similar sets"?

Many_, any_p, ny_pr, y_pro, _prob, probl, roble, oblem, blems, lems_, ems_c, ms_ca, s_can, _can_, can_b, an_be, n_be_, _be_e, be_ex, e_exp, _expr, expre, xpres, press, resse, essed, seed_, eed_a, ed_as, d_as_, _as_f, as_fi, s_fin, _find, findi, indin, nding, ding_, ing_s, ng_si, g_sim, _simi, simil, imila, milar, ilar_, lar_s, ar_se, r_set, _sets

Result = 50

Just to check for repetitions = n - k + 1 = 54 - 5 + 1 = 50
So it matches!

Question 12.
(a) Let C(D1) = {aa, bb, ab , ba}, C(D2) = {aa, ac, ca, ba}, C(D3) = {ab, ba, ca} be the 2-shingle representation of documents D1, D2, D3. Create the matrix representation of the shingles-documents relationship.

	D1	D2	D3
aa	1	1	0
bb	1	0	0
ab	1	0	1
ba	1	1	1
ac	0	1	0
ca	0	1	1

(b) Consider the following permutations:
p1 = {aa, bb, ab, ba, ac, ca},
p2 = {ca, ac, ba, ab, bb, aa},

p3 = {ac, ca, ab, ba, bb, aa}.
Create the signature matrix of the documents.

	D1	D2	D3
P1	aa	aa	ab
P2	ab	aa	aa
P3	ab	aa	aa

(c) Compare the Jaccard similarity of each pair of documents with the Jaccard similarity of their signatures.

Jaccard Similarity of the **documents**

C(D1) = {aa, bb, ab , ba}, C(D2) = {aa, ac, ca, ba}, C(D3) = {ab,ba, ca}
D1 & D2
Intersection = 2
Union = 6
Similarity = intersection / union = 2/6 = ⅓ = 0.33

D1 & D3
Intersection = 2
Union = 5
Similarity = intersection / union = 2/5 = 0.4

D2 & D3
Intersection = 2
Union = 5
Similarity = intersection / union = 2/5 = 0.4

Jaccard Similarity of the **signatures**

D1 & D2
Similarity = ⅓ = 0.33

D1 & D3
Similarity = 0/3 = 0

D2 & D3
Similarity = ⅔ = 0.66