

## Exercises for DW & DM

### Sheet 2

No solution is to be handed in for this exercise. You are only encouraged to familiarize yourself with some of the data cleaning tools as well as the OLAP operations.

#### Exercise 1

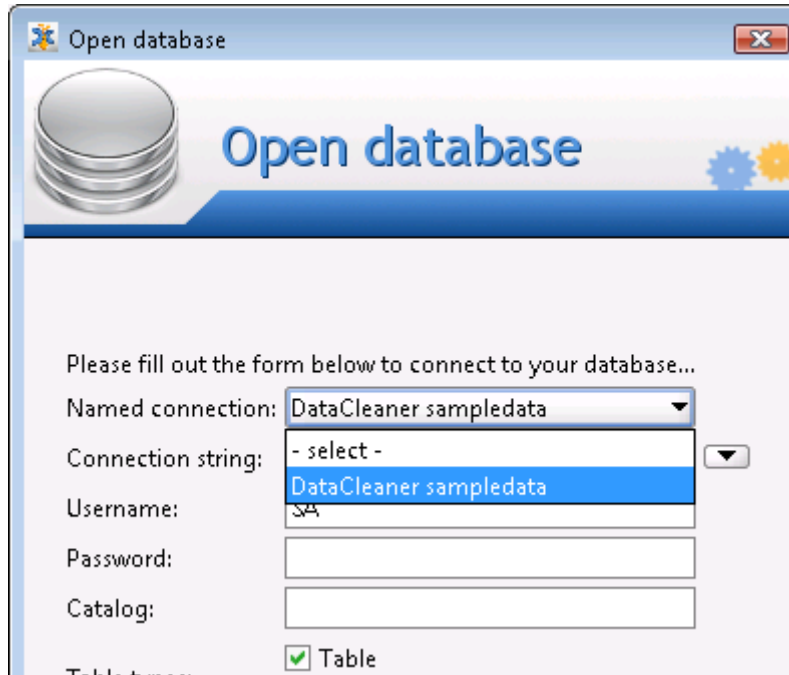
1. Install Eobjects Data Cleaner (<http://datacleaner.eobjects.org/downloads>). Perform the following tasks, by using the sample database provided with the software (by choosing it from the drop down menu as observed in the Annex1)
  - a. Compose a regular expression, which validates only strings that contain letters only (no spaces or other characters than letters). Start with only one capital letter, and continue with at least one, up to 20 small letters. See examples in Annex 2.
  - b. Use the regular expression from 2.a, and create a validation task, add as validation rule a “regex validation”. Choose as data selections the CUSTOMER table, and as data subset the CONTACTLASTNAME and CONTACTFIRSTNAME attributes. Write the lastname and firstname of the clients which did not pass the validation.
  - c. Give three examples (of different patterns) of strings which pass the validation of the following regular expression, and one that doesn't:

$(\backslash+\backslash\{1,2\})?(\backslash(\backslash\{1,4\}\backslash)|(\backslash\{3,5\}[-/?])(\backslash\{1,5\})$

#### Exercise 2

Simulate the functionality of the Multiple Minimum Supports mining algorithm on the transactions provided in Annex 3, presenting each of the 2 steps, as well as the initialization,  $k=2$  and generalization phases for step 1. Minimum support values are also provided in the Annex 3.  $\varphi = 20\%$  and  $\text{minconf} = 60\%$ .

## Annex 1



Open database

Please fill out the form below to connect to your database...

Named connection: DataCleaner sampledata

Connection string: - select -

Username: SA

Password:

Catalog:

Table type:  Table

## Annex 2

String	Evaluation
A	Bad
Aa	Good
AA	Bad
aa	Bad
A1	Very bad
Thomas	Good
Thomas Mann	Bad

### Annex 3

Item	MIS %	Transactions
1	70	1, 4, 6
2	17	1
3	15	1, 5, 6
4	30	1, 6
5	30	4, 6
6	35	1, 2, 3, 5
		1, 2, 3, 5
		6
		1
		1, 6