



LARUTAN CAP BADAK GROUP

Project Data Warehouse & BI

# Credit Card Applicant



LARUTAN CAP BADAK GROUP

# Member of Group

**Tasya Juli Chantika Gurning**

tasyajulichntka@gmail.com

**Muhammad Yusran**

@gmail.com

**Tri Putra Jaya**

triputrajaya92@gmail.com

**Fitri Zahrotul Wardah**

fitrizahrotulw@gmail.com



LARUTAN CAP BADAK GROUP

# Table of Content

---

## 1) Dataset Description

## 2) ETL Process

- Applicant Dimension
- CreditRecord Dimension
- Time Dimensoon
- CreditCard Fact

## 3) Star Schema

## 4) OLAP Output

## 5) Insights



# Dataset

## application\_record.csv

- **ID** : Client number
- **CODE\_GENDER** : Gender
- **FLAG\_OWN\_CAR** : Is there a car
- **FLAG\_OWN-REALTY** :: Is there a property
- **CNT\_CHILDREN** : Number of children
- **AMT\_INCOME\_TOTAL** : Annual income
- **NAME\_INCOME\_TYPE** : Income category
- **NAME\_EDUCATION\_TYPE** : Education level
- **NAME\_FAMILY\_STATUS** : Marital status
- **NAME\_HOUSING\_TYPE** : Way of living
- **DAYS\_BIRTH** : Birthday | *Count backwards from current day (0), -1 means yesterday*
- **DAYS\_EMPLOYED** : Start date of employment | *Count backwards from current day(0). If positive, it means the person currently unemployed.*
- **FLAG\_MOBIL** : Is there a mobile phone
- **FLAG\_WORK\_PHONE** : Is there a work phone
- **FLAG\_PHONE** : Is there a phone
- **FLAG\_EMAIL** : Is there an email
- **OCCUPATION\_TYPE** : Occupation
- **CNT\_FAM\_MEMBERS** : Family size







# Dataset

## credit\_record.csv

- **ID** : Client number
- **MONTHS\_BALANCE** : Record month | *The month of the extracted data is the starting point, backwards, 0 is the current month, -1 is the previous month, and so on.*
- **STATUS** : Status | 0 is 1-29 days past due, 1 is 30-59 days past due, 2 is 60-89 days overdue, 3 is 90-119 days overdue, 4 is 120-149 days overdue, 5 is Overdue or bad debts, write-offs for more than 150 days, C is paid off that month, X is No loan for the month.





LARUTAN CAP BADAK GROUP

# ETL Process



**LARUTAN CAP BADAK GROUP**

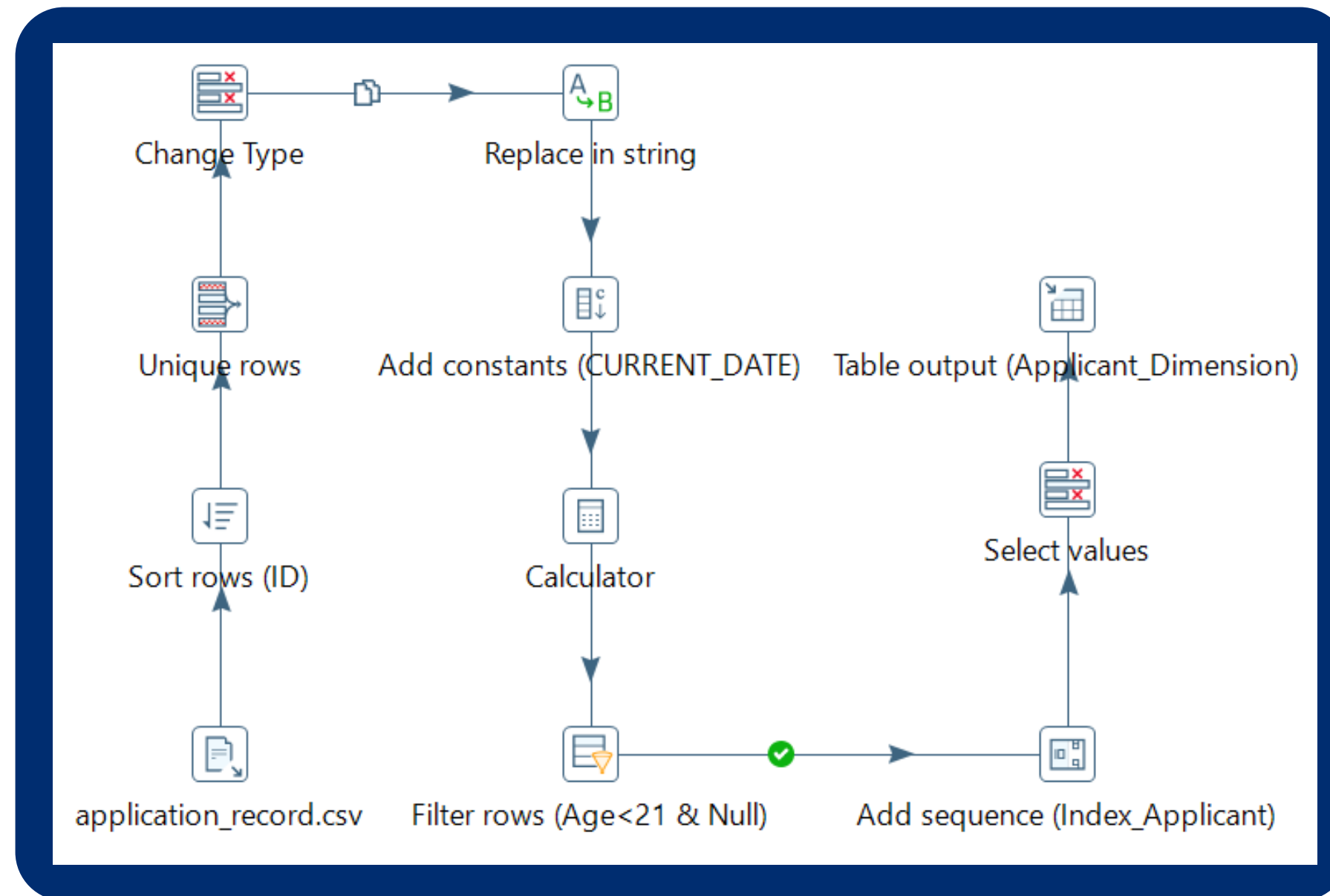
*ETL Process*

**01**

# **Applicant\_Dimension**



# Applicant\_Dimension





# Applicant\_Dimension (1)



Importing application table from  
your OLTP DB.

CSV file input

Step name: application\_record.csv

Filename: D:\\_KULIAH\_\SEM 7\MBKM\Big Data & Business Intelligence - Celerates\W8 - ETL Pentaho & Data Warehouse\application\_record.csv Browse...

Delimiter: , Insert TAB

Enclosure: "

NIO buffer size: 1000

Lazy conversion? ☒

Header row present? ☒

Add filename to result ☐

The row number field name (optional):

Running in parallel? ☐

New line possible in fields? ☐

Format: mixed

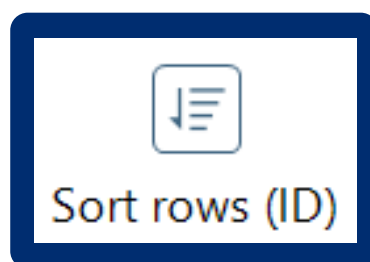
File encoding:

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	ID	Integer	#	15	0	IDR	.	,	none
2	CODE_GENDER	String		1		IDR	.	,	none
3	FLAG_OWN_CAR	Boolean				IDR	.	,	none
4	FLAG_OWN_REALTY	Boolean				IDR	.	,	none
5	CNT_CHILDREN	Integer	#	15	0	IDR	.	,	none
6	AMT_INCOME_TOTAL	Number	#,.	8	1	IDR	.	,	none
7	NAME_INCOME_TYPE	String		20		IDR	.	,	none
8	NAME_EDUCATION_TYPE	String		29		IDR	.	,	none
9	NAME_FAMILY_STATUS	String		20		IDR	.	,	none
10	NAME_HOUSING_TYPE	String		17		IDR	.	,	none
11	DAYS_BIRTH	Integer	#	15	0	IDR	.	,	none
12	DAYS_EMPLOYED	Integer	#	15	0	IDR	.	,	none
13	FLAG_MOBIL	Integer	#	15	0	IDR	.	,	none
14	FLAG_WORK_PHONE	Integer	#	15	0	IDR	.	,	none
15	FLAG_PHONE	Integer	#	15	0	IDR	.	,	none
16	FLAG_EMAIL	Integer	#	15	0	IDR	.	,	none
17	OCCUPATION_TYPE	String		14		IDR	.	,	none
18	CNT_FAM_MEMBERS	Number	#,.	3	1	IDR	.	,	none

Help OK Get Fields Preview Cancel



# Applicant\_Dimension (2)



Sort data based on applicant ID.

Sort rows

Step name

Sort rows (ID)

Sort directory

%%java.io.tmpdir%%

Browse...

TMP-file prefix

out

Sort size (rows in memory)

1000000

Free memory threshold (in %)

Compress TMP Files?

☐

Only pass unique rows? (verifies keys only)

☐

Fields :

#	Fieldname	Ascending	Case sensitive compare?	Sort based on current locale?	Collator Strength	Presorted?
1	ID	Y	N	N	0	N

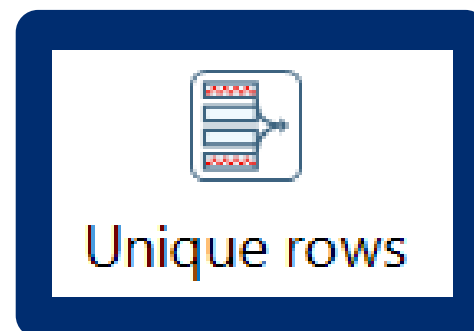
Help

OK

Cancel

Get Fields

# Applicant\_Dimension (3)



Filter duplicate applicant ID.

Unique rows

Step name

Settings

Add counter to output? ☐ Counter field

Redirect duplicate row ☐ Error description

Fields to compare on (no entries means: compare complete row)

#	Fieldname	Ignore case
1	ID	Y





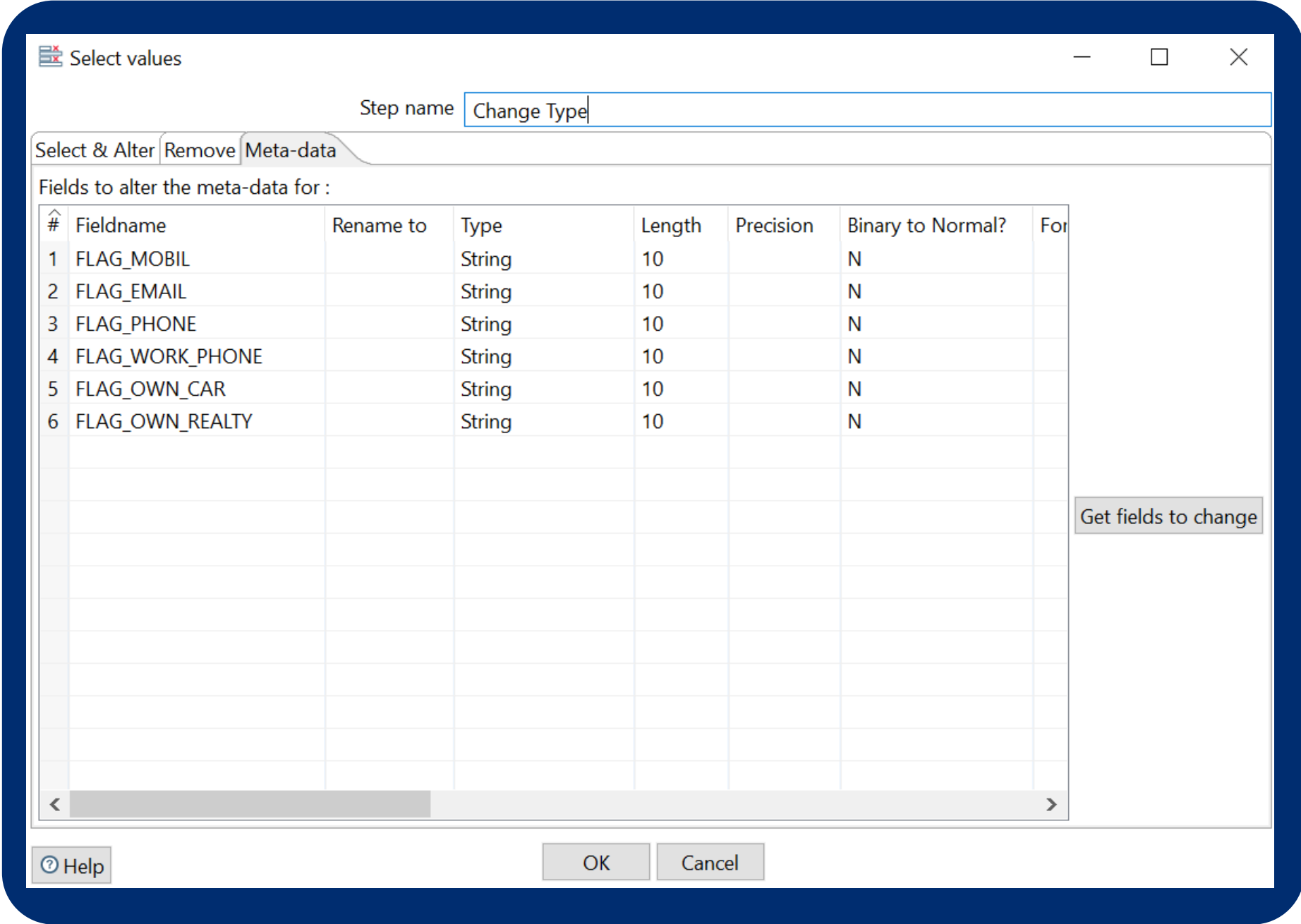






Replace some values from `Flag_*` field to make it easier to understand.

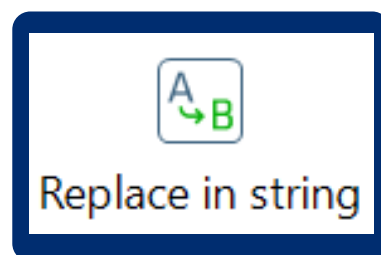
Field	Current Value	Replace With
Gender	M	Male
Gender	Y	Female
Owner_car, Owner_realty	N	Don't Have
Owner_car, Owner_realty	Y	Have
Mobil, Work_phone, Phone, Email	1	Have
Mobil, Work_phone, Phone, Email	0	Don't Have







# Applicant\_Dimension (5)



Replace some values from Flag\_\* field to make it easier to understand. (Cont.)

Field	Current Value	Replace With
Gender	M	Male
Gender	Y	Female
Owner_car, Owner_realty	N	Don't Have
Owner_car, Owner_realty	Y	Have
Mobil, Work_phone, Phone, Email	1	Have
Mobil, Work_phone, Phone, Email	0	Don't Have

Replace in string

Step nameReplace in string

Fields string

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set empty string?	Replace with field	Whole Word	Case sensitive	Is Unicode
1	CODE_GENDER		N	M	Male	N		Y	Y	N
2	CODE_GENDER		N	F	Female	N		Y	Y	N
3	FLAG_OWN_CAR		N	N	Don't Have	N		Y	Y	N
4	FLAG_OWN_REALTY		N	N	Don't Have	N		Y	Y	N
5	FLAG_OWN_CAR		N	Y	Have	N		Y	Y	N
6	FLAG_OWN_REALTY		N	Y	Have	N		Y	Y	N
7	FLAG_MOBIL		N	1	Have	N		Y	Y	N
8	FLAG_WORK_PHONE		N	1	Have	N		Y	Y	N
9	FLAG_PHONE		N	1	Have	N		Y	Y	N
10	FLAG_EMAIL		N	1	Have	N		Y	Y	N
11	FLAG_MOBIL		N	0	Don't Have	N		Y	Y	N
12	FLAG_WORK_PHONE		N	0	Don't Have	N		Y	Y	N
13	FLAG_PHONE		N	0	Don't Have	N		Y	Y	N
14	FLAG_EMAIL		N	0	Don't Have	N		Y	Y	N

<

>

Help

OK

Get fields

Cancel



• • • • •



Add new columns with constant date (October 1, 2021) as Current Date.

Add constants

Step name

Add constants (CURRENT\_DATE)

Fields :

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Value	Set empty string?
1	CURRENT_DATE	Date	yyyy-MM-dd	50					2021-10-01	N

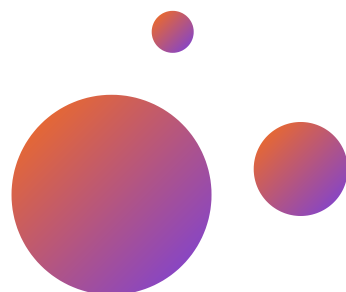
<

>

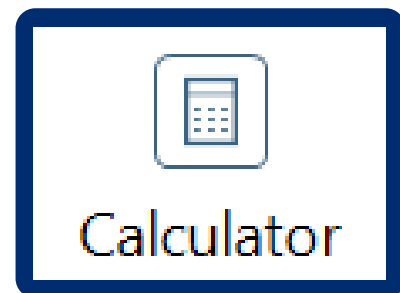
Help

OK

Cancel



# Applicant\_Dimension (7)



Calculate DOB and date of applicant start working based on current date (October 1, 2021).  
Hint : Date A + Date B

Calculate age of applicant based on current year (2021).  
Hint : Year A – Year B

Calculator

Step name

Calculator

☒ Throw an error on non existing files


Fields:

#	New field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Conversion mask	Decimal
1	DAYS_BIRTH_2	Date A + B Days	CURRENT_DATE	DAYS_BIRTH		Date	50		N		
2	DAYS_EMPLOYED_2	Date A + B Days	CURRENT_DATE	DAYS_EMPLOYED		Date	50		N		
3	AGE_1	Year of date A	DAYS_BIRTH_2			Integer	2		N		
4	AGE_2	Year of date A	CURRENT_DATE			Integer	2		N		
5	AGE_CALCULATION	A - B	AGE_2	AGE_1		Integer	2		N		

Help

OK Cancel

# Applicant\_Dimension (8)

  
Filter rows (Age<21 & Null)



Filter applicant data which has null values.

Filter applicant data who is less than 21 y.o.

Filter rows

Step name

Send 'true' data to

Send 'false' data to

The condition:

AGE\_CALCULATION >= [21]

AND

ID IS NOT NULL

AND

CODE\_GENDER IS NOT NULL

AND

FLAG\_OWN\_CAR IS NOT NULL

AND

FLAG\_OWN\_REALTY IS NOT NULL

AND

CNT\_CHILDREN IS NOT NULL

AND

AMT\_INCOME\_TOTAL IS NOT NULL

AND

NAME\_EDUCATION\_TYPE IS NOT NULL

AND

NAME\_FAMILY\_STATUS IS NOT NULL

AND

>>>

NAME\_HOUSING\_TYPE IS NOT NULL

AND

DAYS\_BIRTH IS NOT NULL

AND

FLAG\_WORK\_PHONE IS NOT NULL

AND

DAYS\_EMPLOYED IS NOT NULL

AND

FLAG\_MOBIL IS NOT NULL

AND

FLAG\_PHONE IS NOT NULL

AND

FLAG\_EMAIL IS NOT NULL

AND

OCCUPATION\_TYPE IS NOT NULL

AND

CNT\_FAM\_MEMBERS IS NOT NULL

Help

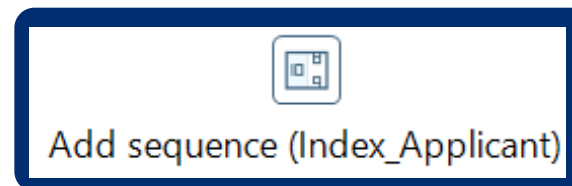
OK

Cancel





# Applicant\_Dimension (9)



Adding Index\_Applicant (to replace ID as primary key).

Add sequence

Step nameAdd sequence (Index\_Applicant)

Name of valueIndex\_Applicant

Use a database to generate the sequence

Use DB to get sequence?☐

ConnectionProject Data Warehouse & BI

Edit...New...Wizard...

Schema name

Schemas...

Sequence nameSEQ\_

Sequences...

Use a transformation counter to generate the sequence

Use counter to calculate sequence?☒

Counter name (optional)

Start at value1

Increment by1

Maximum value999999999

OKCancel

Help



# Applicant\_Dimension (10)



Select values



Select only selected Value.

Fields to alter the meta-data for :

#	Fieldname	Rename to	Type	Length	Precision
1	Index_Applicant		Integer	10	
2	ID	Applicant_ID	String	10	
3	CODE_GENDER	Applicant_Gender	String	7	
4	CNT_CHILDREN	Total_Children	Integer	2	
5	CNT_FAM_MEMBERS	Total_Family_Members	Integer	2	
6	AGE_CALC	Applicant_Age	Integer	2	
7	DAYS_BIRTH2	Birth_Date	Date	50	
8	DAYS_EMPLOYED2	Employed_Date	Date	50	
9	NAME_EDUCATION_TYPE	Education_Type	String	50	
10	NAME_FAMILY_STATUS	Family_Status	String	50	
11	NAME_HOUSING_TYPE	Housing_Type	String	50	
12	NAME_INCOME_TYPE	Income_Type	String	50	
13	OCCUPATION_TYPE	Job_Title	String	50	
14	AMT_INCOME_TOTAL	Total_Income	Integer	10	
15	FLAG_EMAIL	Owned_Email	String	10	
16	FLAG_MOBIL	Owned_Mobile_Phone	String	10	
17	FLAG_OWN_CAR	Owned_Car	String	10	
18	FLAG_OWN_REALTY	Owned_Realty	String	10	
19	FLAG_PHONE	Owned_Phone	String	10	
20	FLAG_WORK_PHONE	Owned_Work_Phone	String	10	

Select values

Step nameSelect values

Select & AlterRemoveMeta-data

Fields to alter the meta-data for :

#	Fieldname	Rename to	Type	Length	Precision	Binary to N
1	Index_Applicant		Integer	10		N
2	ID	Applicant_ID	String	10		N
3	CODE_GENDER	Applicant_Gender	String	7		N
4	CNT_CHILDREN	Total_Children	Integer	2		N
5	CNT_FAM_MEMBERS	Total_Family_Members	Integer	2		N
6	AGE_CALCULATION	Applicant_Age	Integer	2		N
7	DAYS_BIRTH_2	Birth_Date	Date	50		N
8	DAYS_EMPLOYED_2	Employed_Date	Date	50		N
9	NAME_EDUCATION_TYPE	Education_Type	String	50		N
10	NAME_FAMILY_STATUS	Family_Status	String	50		N
11	NAME_HOUSING_TYPE	Housing_Type	String	50		N
12	NAME_INCOME_TYPE	Income_Type	String	50		N
13	OCCUPATION_TYPE	Job_Title	String	50		N
14	AMT_INCOME_TOTAL	Total_Income	Integer	10		N
15	FLAG_EMAIL	Owned_Email	String	10		N
16	FLAG_MOBIL	Owned_Mobile_Phone	String	10		N
17	FLAG_OWN_CAR	Owned_Car	String	10		N
18	FLAG_OWN_REALTY	Owned_Realty	String	10		N
19	FLAG_PHONE	Owned_Phone	String	10		N
20	FLAG_WORK_PHONE	Owned_Work_Phone	String	10		N

Get fields to change

HelpOKCancel

Select & AlterRemoveMeta-data

Fields to remove :

#	Fieldname
1	DAYS_BIRTH
2	DAYS_EMPLOYED
3	CURRENT_DATE
4	AGE_1
5	AGE_2



# Applicant\_Dimension (11)

Export to OLAP DB on your RDBMS.

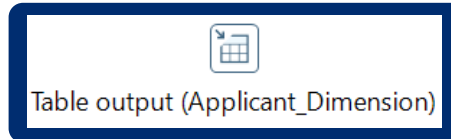


Table output

Step name: Table output (Applicant\_Dimension)

Connection: Project Data Warehouse & BI

Target schema:

Target table: Applicant\_Dimension

Commit size: 1000000

Truncate table: ☐

Ignore insert errors: ☐

Specify database fields: ☐

Main options

Database fields

Partition data over tables: ☐

Partitioning field:

Partition data per month: ☒

Partition data per day: ☐

Use batch update for inserts: ☒

Is the name of the table defined in: ☐

Field that contains name of table:

Store the tablename field: ☒

Return auto-generated key: ☐

Name of auto-generated key field:

Help OK Cancel SQL

Database Connection

General

Advanced

Options

Pooling

Clustering

Connection name: Project Data Warehouse & BI

Connection type: PostgreSQL

Settings

Host Name: localhost

Database Name: postgres

Port Number: 5432

Username: postgres

Password:

Access: Native (JDBC)

Test Feature List Explore

OK Cancel

Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE Applicant\_Dimension ( Applicant\_ID VARCHAR(10) , Applicant\_Gender VARCHAR(7) , Owned\_Car VARCHAR(10) , Owned\_Realty VARCHAR(10) , Total\_Children SMALLINT , Total\_Income BIGINT , Income\_Type VARCHAR(50) , Education\_Type VARCHAR(50) , Family\_Status VARCHAR(50) , Housing\_Type VARCHAR(50) , Owned\_Mobile\_Phone VARCHAR(10) , Owned\_Work\_Phone VARCHAR(10) , Owned\_Phone VARCHAR(10) , Owned\_Email VARCHAR(10) , Job\_Title VARCHAR(50) , Total\_Family\_Members SMALLINT , Birth\_Date TIMESTAMP , Employed\_Date TIMESTAMP , Applicant\_Age SMALLINT , Index\_Applicant BIGINT ) ;

Line 1 column 32

Execute Clear cache Close



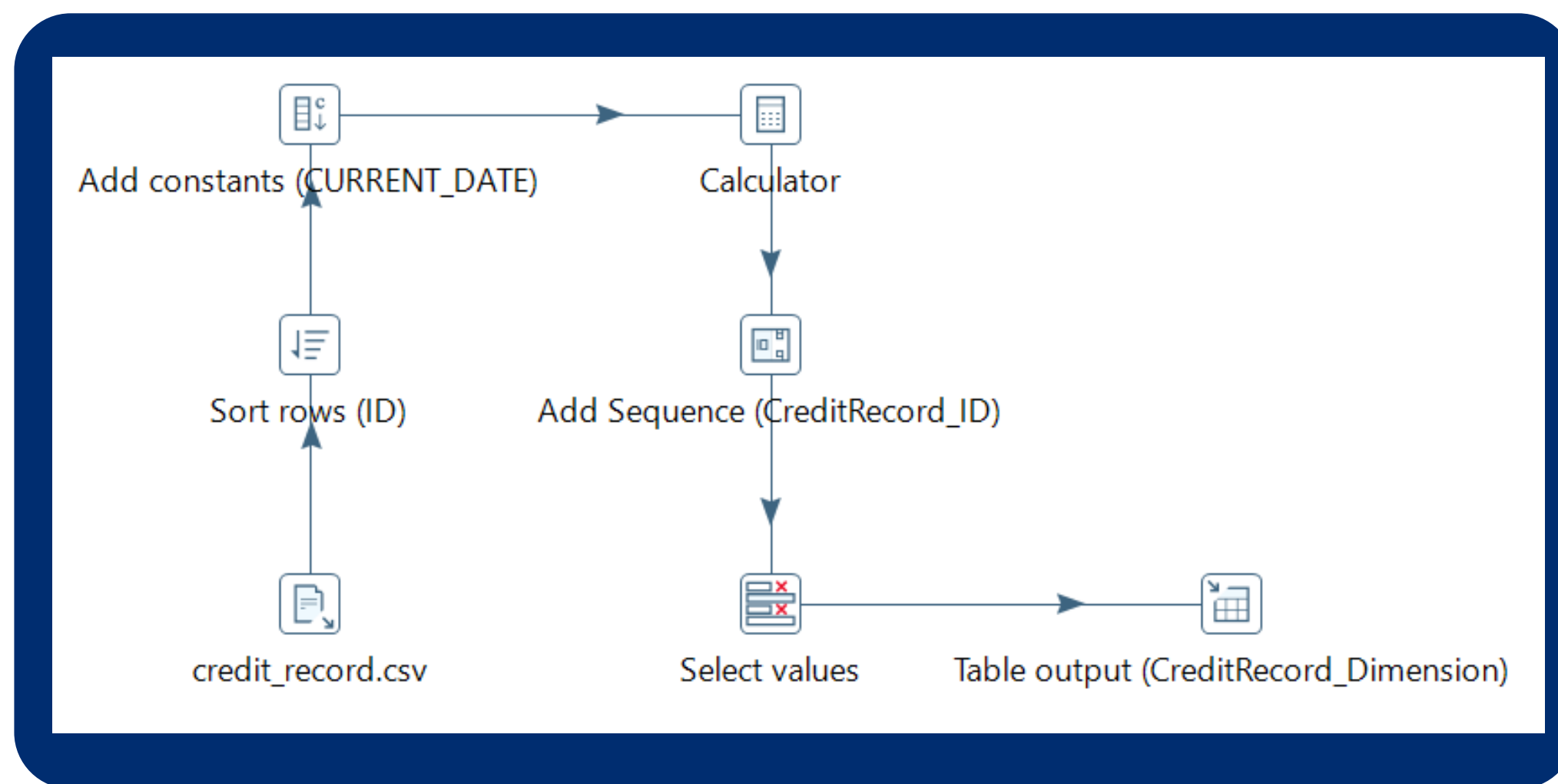
02

# CreditRecord\_Dimension

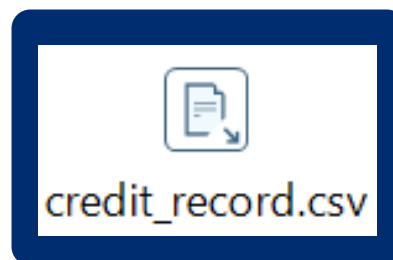




# CreditRecord\_Dimension



# CreditRecord\_Dimension (1)



Importing credit record table  
from your OLTP DB.

CSV file input

Step name: credit\_record.csv

Filename: D:\\_KULIAH\_\SEM 7\MBKM\Big Data & Business Intelligence - Celerat Browse...

Delimiter: , Insert TAB

Enclosure: "

NIO buffer size: 50000

Lazy conversion? ☒

Header row present? ☒

Add filename to result ☐

The row number field name (optional):

Running in parallel? ☐

New line possible in fields? ☐

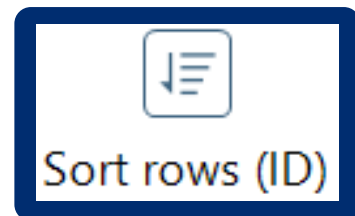
Format: mixed

File encoding:

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	ID	Integer	#	15	0	IDR	.	,	none
2	MONTHS_BALANCE	Integer	#	15	0	IDR	.	,	none
3	STATUS	String		1		IDR	.	,	none

Help OK Get Fields Preview Cancel

# CreditRecord\_Dimension (2)



Sort data based on applicant ID.

Sort rows

Step name

Sort rows (ID)

Sort directory

%%java.io.tmpdir%%

Browse...

TMP-file prefix

out

Sort size (rows in memory)

1000000

Free memory threshold (in %)

Compress TMP Files?

☐

Only pass unique rows? (verifies keys only)

☐

Fields :

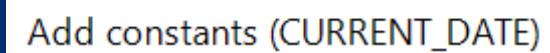
#	Fieldname	Ascending	Case sensitive compare?	Sort based on current locale?	Collator Strength	Presorted?
1	ID	Y	N	N	0	N

Help

OK

Cancel

Get Fields



Add new columns with constant date (October 1, 2021) as Current\_date.

**Add constants**

Step name:

Fields :

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Value	Set empty string?
1	CURRENT_DATE	Date	yyyy-MM-dd	50					2021-10-01	N

< >





Calculator

---

Step name

---

☒ Throw an error on non existing files

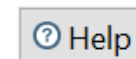
Fields:

#	New field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Convert
1	MONTHS_BALANCE_2	Date A + B Months	CURRENT_DATE	MONTHS_BALANCE		Date	50		N	

< \_\_\_\_\_ >

---

Help OK Cancel

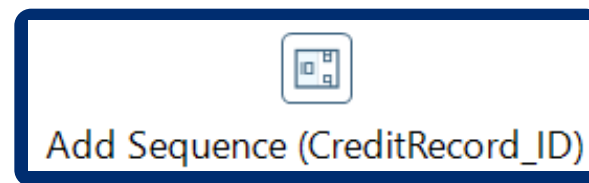


OK

Cancel



# CreditRecord\_Dimension (5)



Adding CreditRecord\_ID (to replace Applicant ID as primary key).

Add sequence

Step name

Add Sequence (CreditRecord\_ID)

Name of value

CreditRecord\_ID

Use a database to generate the sequence

Use DB to get sequence?

☐

Connection

Project Data Warehouse & BI (2)

Edit...

New...

Wizard...

Schema name

Schemas...

Sequence name

SEQ\_

Sequences...

Use a transformation counter to generate the sequence

Use counter to calculate sequence?

☒

Counter name (optional)

Start at value

1

Increment by

1

Maximum value

999999999

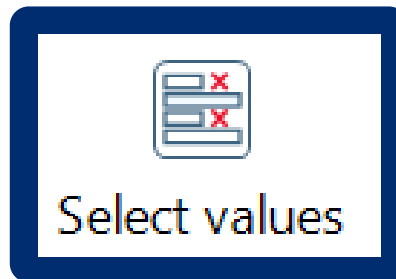
OK

Cancel

Help



# CreditRecord\_Dimension (6)



Select values



Select only selected Value.

Fields to alter the meta-data for :

#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?
1	CreditRecord_ID		String	15		N
2	ID	Applicant_ID	String	10		N
3	MONTHS_BALANCE2	Record_Date	Date	50		N
4	STATUS	Status	String	5		N

Select values

Step name

Select & Alter Remove Meta-data

Fields to alter the meta-data for :

#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?	Format	Date f
1	CreditRecord_ID		String	15		N		N
2	ID	Applicant_ID	String	10		N		N
3	MONTHS_BALANCE_2	Record_Date	Date	50		N		N
4	STATUS	Status	String	5		N		N

Get fields to change

Help OK Cancel

Select & Alter Remove Meta-data

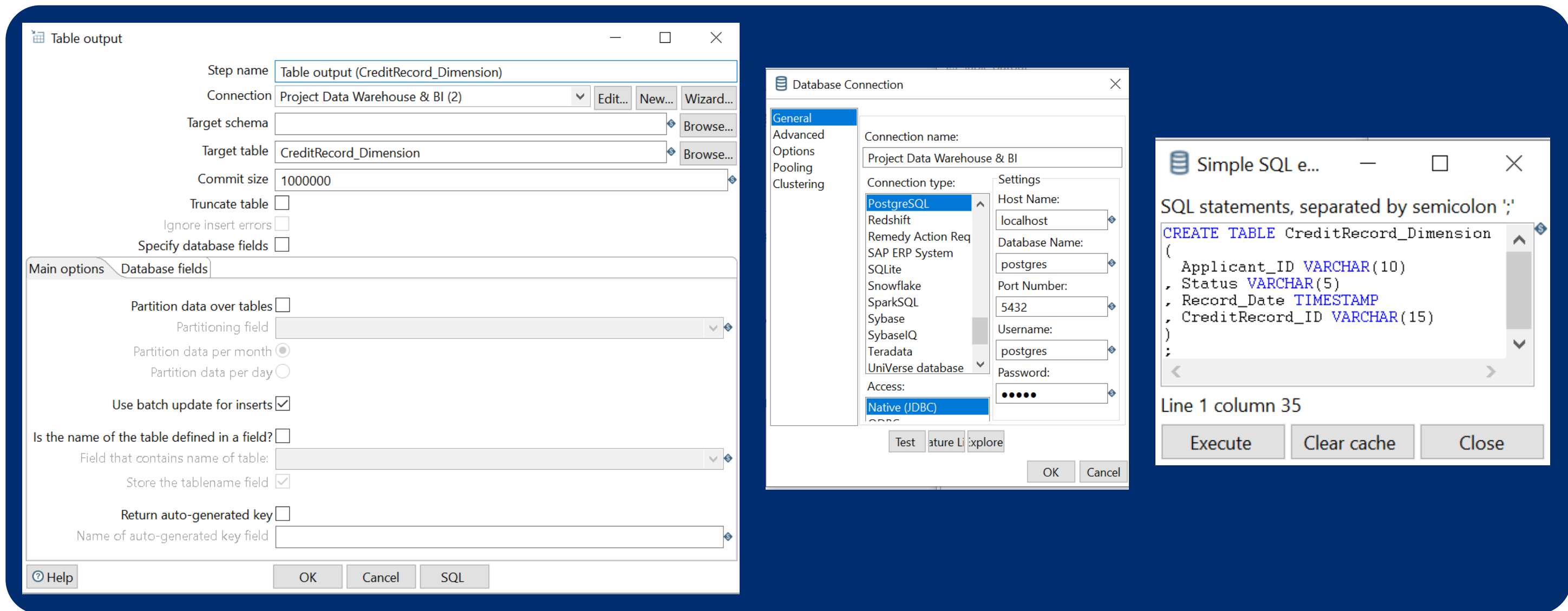
Fields to remove :

#	Fieldname
1	MONTHS_BALANCE
2	Current_Date

# CreditRecord\_Dimension (7)

Export to OLAP DB on your  
RDBMS.

Table output (CreditRecord\_Dimension)



The screenshot displays two windows from a data integration tool. The 'Table output' window on the left is configured with the following details:

- Step name:** Table output (CreditRecord\_Dimension)
- Connection:** Project Data Warehouse & BI (2)
- Target schema:** (empty)
- Target table:** CreditRecord\_Dimension
- Commit size:** 1000000
- Truncate table:** ☐
- Ignore insert errors:** ☐
- Specify database fields:** ☐
- Main options:**
  - Partition data over tables:** ☐
  - Partitioning field:** (empty)
  - Partition data per month:** ☒
  - Partition data per day:** ☐
  - Use batch update for inserts:** ☒
  - Is the name of the table defined in a field?:** ☐
  - Field that contains name of table:** (empty)
  - Store the tablename field:** ☒
  - Return auto-generated key:** ☐
  - Name of auto-generated key field:** (empty)

The 'Database Connection' window on the right shows the configuration for the 'Project Data Warehouse & BI' connection:

- Connection name:** Project Data Warehouse & BI
- Connection type:** PostgreSQL
- Settings:**
  - Host Name:** localhost
  - Database Name:** postgres
  - Port Number:** 5432
  - Username:** postgres
  - Password:** (masked with dots)
- Access:** Native (JDBC)

Below these windows, a 'Simple SQL editor' window is shown with the following SQL statement:

```
CREATE TABLE CreditRecord_Dimension
(
  Applicant_ID VARCHAR(10)
, Status VARCHAR(5)
, Record_Date TIMESTAMP
, CreditRecord_ID VARCHAR(15)
)
```

The status bar at the bottom of the SQL editor indicates 'Line 1 column 35'. Buttons for 'Execute', 'Clear cache', and 'Close' are visible at the bottom of the SQL editor window.

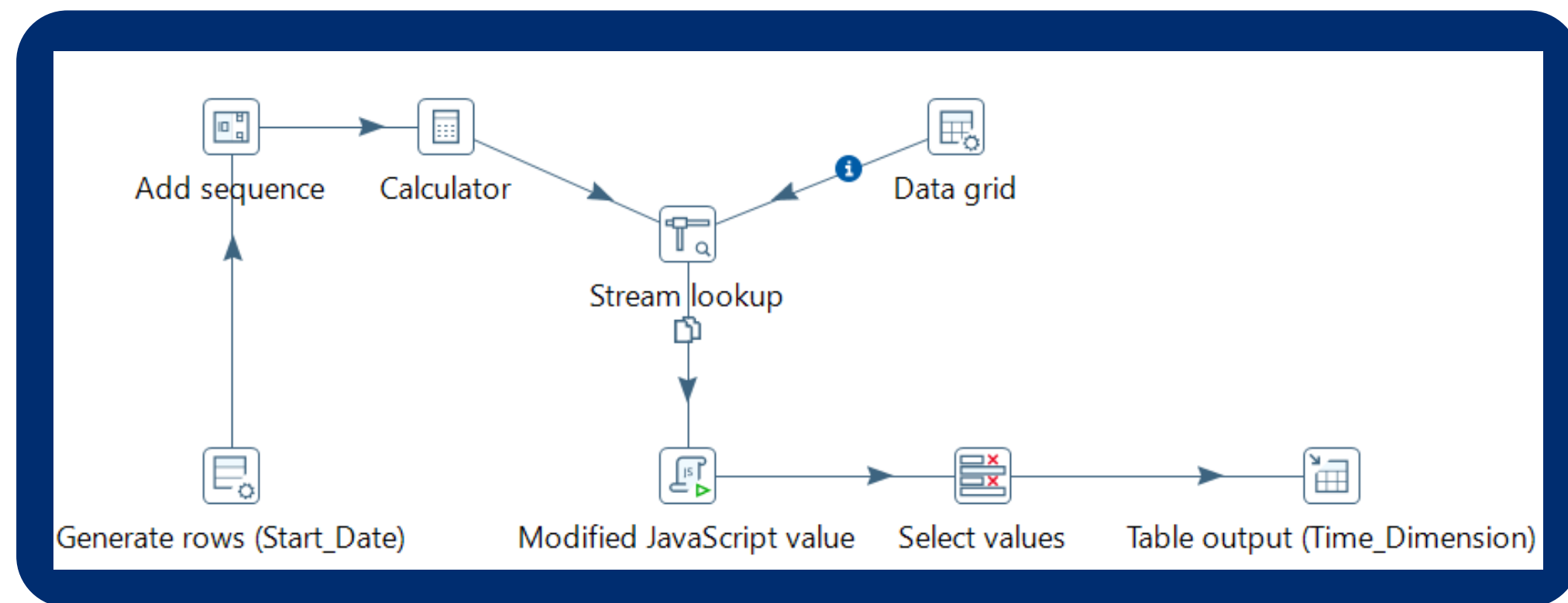


03

# Time\_Dimension



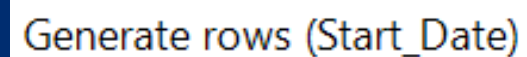
# Time\_Dimension





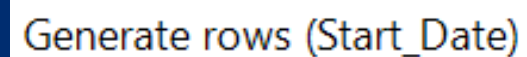


**LARUTAN CAP BADAK GROUP**



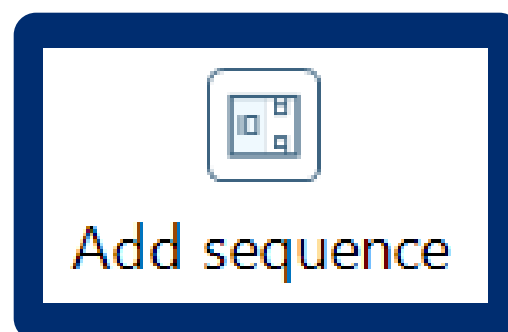
Generate rows (Start\_Date)

Generate rows (Start\_Date)





# Time\_Dimension (2)



Add row with sequence  
from 1 to 99999.

Add sequence

Step name

Add sequence

Name of value

Next\_Day

Use a database to generate the sequence

Use DB to get sequence?

☐

Connection

Project Data Warehouse & BI (3)

Edit...

New...

Wizard...

Schema name

Schemas...

Sequence name

SEQ\_

Sequences...

Use a transformation counter to generate the sequence

Use counter to calculate sequence?

☒

Counter name (optional)

Start at value

0

Increment by

1

Maximum value

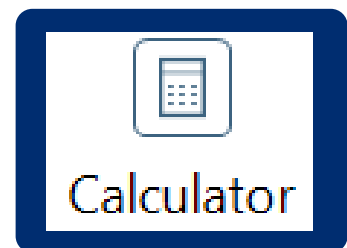
99999

OK

Cancel

Help

# Time\_Dimension (3)



Calculating start date with  
sequence data to make  
next\_date (ex: January 2,  
2016; January 3, 2016).

Calculator

Step name

Calculator

☒ Throw an error on non existing files

Fields:

#	New field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Conversion mask	De
1	Date	Date A + B Days	Start_Date	Next_Day		Date			N	yyyy-MM-dd	
2	Day	Day of month of date A	Date			None			N		
3	Month	Month of date A	Date			None			N		
4	Year	Year of date A	Date			None			N		

Help

OK Cancel

# Time\_Dimension (4)

**Stream lookup**

Step name: Stream lookup

Lookup step: Data grid

The key(s) to look up the value(s):

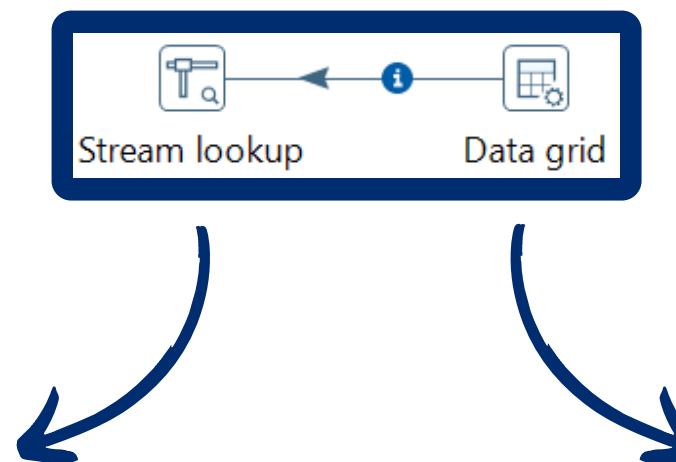
#	Field	LookupField
1	Month	Month_No

Specify the fields to retrieve :

#	Field	New name	Default	Type
1	Month_Name			String

☒ Preserve memory (costs CPU)  
☐ Key and value are exactly one integer  
☐ Use sorted list (i.s.o. hashtable)

? Help OK Cancel Get Fields Get lookup fields



Creating new columns (Day, Months, and Year).

Creating month number and month name.

1 for Januari

2 for Februari

.

.

12 for December

Combine 'Month' from Calculator node to 'No\_Month' from Data Grid node.

**Data grid**

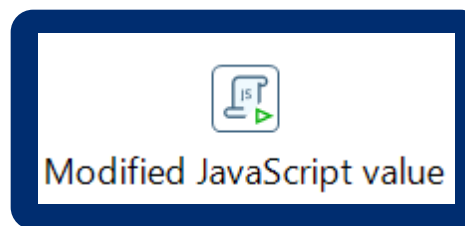
Step name: Data grid

Meta Data

#	Name	Type	Format	Length
1	Month_No	Integer		
2	Month_Name	String		

? Help OK Preview Cancel

# Time\_Dimension (5)



Creating time ID using  
JavaScript code.  
Expected output :  
20161001,20161002,...

Modified JavaScript value

Step name

Transform Script

Transform Cons

Transform Funct

Input fields

Start\_Date.get

Next\_Day.get

Date.getDate

Day.getIntege

Month.getInte

Year.getIntege

Script 1

//Script here

**var** Time\_ID = (Year.getInteger()\*10000)+(Month.getInteger()\*100)+Day.getInteger();

Linenr: 0

Compatibility moc ☒ Optimization level

Fields

#	Fieldname	Rename to	Type	Length	Precision	Replace value 'Fieldname' or 'Rename to'
1	Time_ID		Integer	16	2	N

Help

OK

Cancel

Get variables

Test script

☐ ☐ ☐ ☐ ☐



Select only selected Value.

#	Fieldname	Rename to	Length	Precision
1	Time_ID			
2	Day			
3	Month			
4	Year			
5	Date			

Select values

Step name

Select & Alter Remove Meta-data

Fields :

#	Fieldname	Rename to	Length	Precision
1	Time_ID			
2	Day			
3	Month			
4	Year			
5	Date			

Get fields to select

Edit Mapping

Include unspecified fields, ☐

Help OK Cancel



# Time\_Dimension (7)

Export to OLAP DB on  
your RDBMS.


  
Table output (Time\_Dimension)

Table output

Step name

Table output (Time\_Dimension)

Connection

Project Data Warehouse & BI (3)

Edit... New... Wizard...

Target schema

Browse...

Target table

Time\_Dimension

Browse...

Commit size

9999999

Truncate table

☐

Ignore insert errors

☐

Specify database fields

☐

Main options

Database fields

Partition data over tables

☐

Partitioning field

Partition data per month

☒

Partition data per day

☐

Use batch update for inserts

☒

Is the name of the table defined in a field?

☐

Field that contains name of table:

Store the tablename field

☒

Return auto-generated key

☐

Name of auto-generated key field

Database Connection

General

Advanced

Options

Pooling

Clustering

Connection name:

Project Data Warehouse & BI (3)

Connection type:

PostgreSQL

Redshift

Remedy Action Reque

SAP ERP System

SQLite

Snowflake

SparkSQL

Sybase

SybaseIQ

Teradata

UniVerse database

Settings

Host Name:

localhost

Database Name:

postgres

Port Number:

5432

Username:

postgres

Password:

.....

Access:

Native (JDBC)

ODBC

JNDI

Test

ature Li

Explore

Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE Time\_Dimension

(

Time\_ID BIGINT

, "Day" DOUBLE PRECISION

, "Month" DOUBLE PRECISION

, "Year" DOUBLE PRECISION

, "Date" TIMESTAMP

)

;

Line 1 column 27

Execute

Clear cache

Close

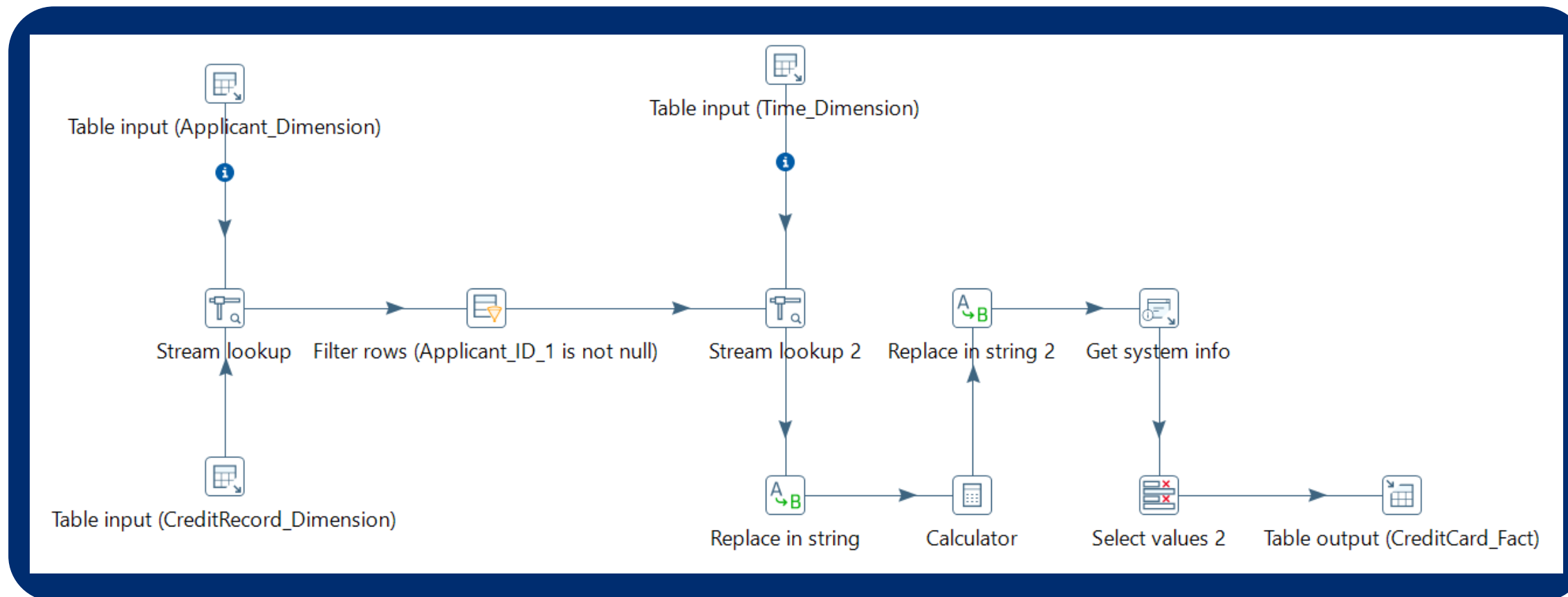


04

CreditCard\_Fact




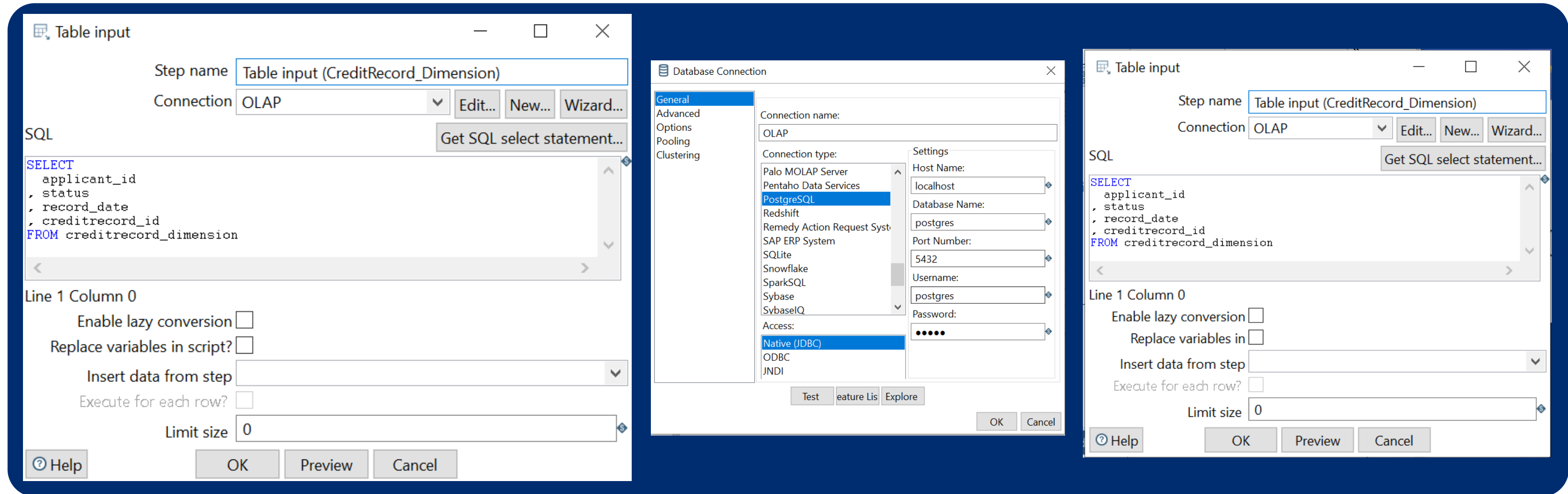
# CreditCard\_Fact



# CreditCard\_Fact (1)

Importing Credit Record dimension from OLAP.

  
Table input (CreditRecord\_Dimension)



The image displays the configuration of a 'Table input' step in Pentaho Data Integration (PDI). The main window shows the 'Table input' step with the following details:

- Step name:** Table input (CreditRecord\_Dimension)
- Connection:** OLAP
- SQL:**

```
SELECT
  applicant_id
, status
, record_date
, creditrecord_id
FROM creditrecord_dimension
```
- Line 1 Column 0:**
  - Enable lazy conversion ☐
  - Replace variables in script? ☐
  - Insert data from step
  - Execute for each row? ☐
  - Limit size

The 'Database Connection' dialog is also visible, showing the configuration for the 'OLAP' connection:

- Connection name:** OLAP
- Connection type:** PostgreSQL
- Settings:**
  - Host Name: localhost
  - Database Name: postgres
  - Port Number: 5432
  - Username: postgres
  - Password: \*\*\*\*
- Access:** Native (JDBC)

The 'Table input' step is also shown with the same configuration as the main window, including the SQL statement and the 'Line 1 Column 0' options.



# CreditCard\_Fact (2)

Stream lookup

Step name: Stream lookup

Lookup step: Table input (Applicant\_Dimension)

The key(s) to look up the value(s):

#	Field	LookupField
1	Applicant_ID	Applicant_ID

Specify the fields to retrieve :

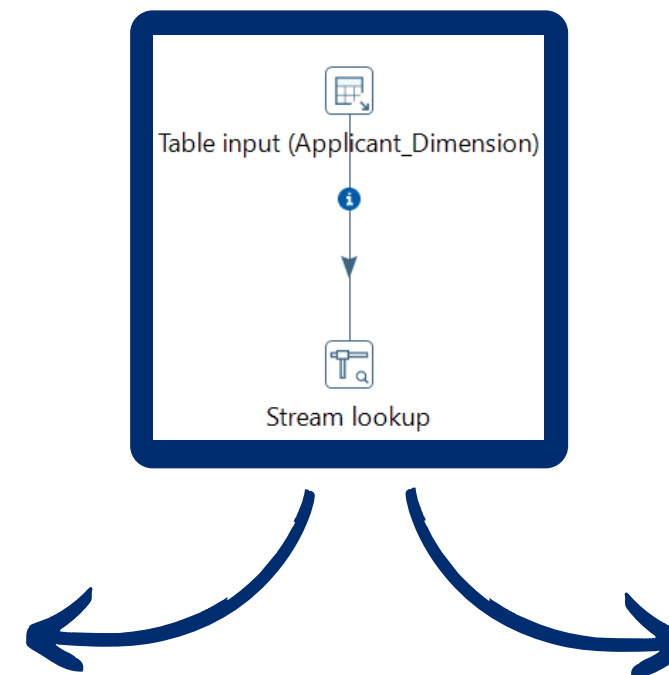
#	Field	New name	Default	Type
1	Applicant_ID			None

Preserve memory (costs CPU) ☒

Key and value are exactly one ☐

Use sorted list (i.s.o. hashtable) ☐

Buttons: ? Help, OK, Cancel, Get Fields, Get lookup fields



Importing Application dimension from OLAP.

Join both dimension tables based on applicant ID.

Table input

Step name: Table input (Applicant\_Dimension)

Connection: OLAP

SQL:

```
SELECT
  applicant_id
, applicant_gender
, owned_car
, owned_realty
, total_children
, total_income
, income_type
, education_type
, family_status
, housing_type
, owned_mobile_phone
, owned_work_phone
, owned_phone
, owned_email
, job_title
, total_family_members
, birth_date
, employed_date
, applicant_age
, index_applicant
FROM applicant_dimension
```

Line 1 Column 0

Enable lazy ☐

Replace variables in ☐

Insert data from

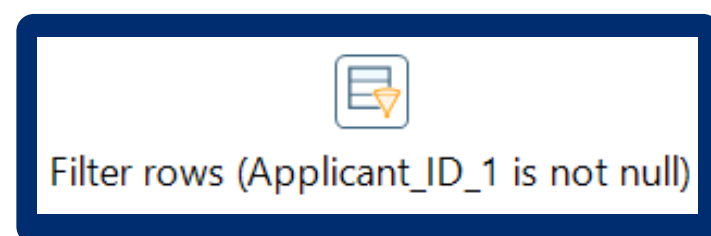
Execute for each ☐

Limit size: 0

Buttons: ? Help, OK, Preview, Cancel



# CreditCard\_Fact (3)



Filter applicant ID that doesn't exists in both tables.

Filter rows

Step name: Filter rows (Applicant\_ID\_1 is not null)

Send 'true' data to step: [dropdown]

Send 'false' data to: [dropdown]

The condition:

[+] Applicant\_ID\_1 IS NOT NULL [ ]

[?] Help OK Cancel





# CreditCard\_Fact (4)

Stream lookup

Step name: Stream lookup 2

Lookup step: Table input (Time\_Dimension)

The key(s) to look up the value(s):

#	Field	LookupField
1	Record_Date	Date

Specify the fields to retrieve :

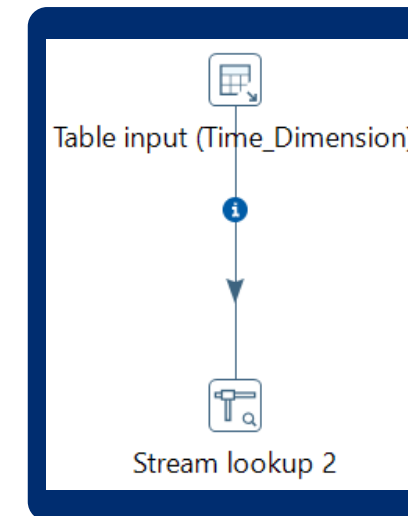
#	Field	New name	Default	Type
1	Time_ID			None

Preserve memory (costs CPU) ☒

Key and value are exactly one ☐

Use sorted list (i.s.o. hashtable) ☐

Help OK Cancel Get Fields Get lookup fields



Importing Time dimension from OLAP.

Join application & credit record dimension with time dimension.

Table input

Step name: Table input (Time\_Dimension)

Connection: OLAP

SQL:

```
SELECT
  time_id
  , "Day"
  , "Month"
  , "Year"
  , "Date"
FROM time_dimension
```

Line 1 Column 0

Enable lazy ☐

Replace variables in ☐

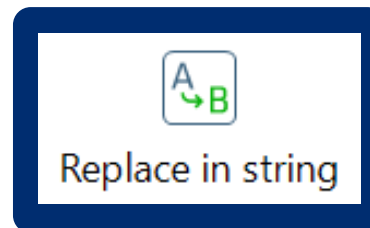
Insert data from step

Execute for each row? ☐

Limit size: 0

Help OK Preview Cancel

# CreditCard\_Fact (5)



Replace C, X, 0 with 'Good Debt'  
(C: loan for that month is already  
paid; X: no loan for that month; 0:  
loan is 1 to 29 days overdue).

Replace 1, 2, 3, 4, 5 with 'Bad  
Debt' (1: loan is 30 to 59 days  
overdue; 2: loan is 60 to 89 days  
overdue; 3: loan is 90 to 119 days  
overdue; 4: loan is 120 to 149  
days overdue; 5: loan is more  
than 150 days overdue).

**Replace in string**

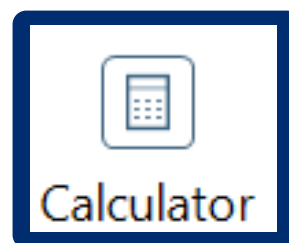
Step name:

Fields string

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set empty s
1	Status		N	C	Good Debt	N
2	Status		N	X	Good Debt	N
3	Status		N	0	Good Debt	N
4	Status		N	1	Bad Debt	N
5	Status		N	2	Bad Debt	N
6	Status		N	3	Bad Debt	N
7	Status		N	4	Bad Debt	N
8	Status		N	5	Bad Debt	N

<  >

# CreditCard\_Fact (6)



Creating 2 copies from  
'Status' column  
( 'Good\_Debt' and  
'Bad\_Debt' ).

Calculator

Step name  
Calculator

☒ Throw an error on non existing files

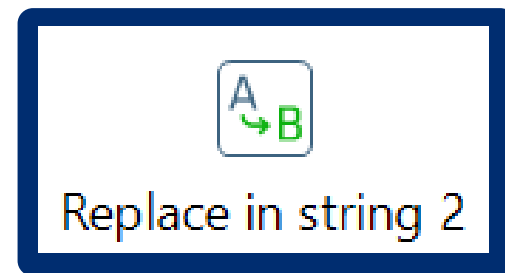
Fields:

#	New field	Calculation	Field A	Field B	Field C	Value type	Length	Pi
1	Good_Debt	Create a copy of field A	Status			String	10	
2	Bad_Debt	Create a copy of field A	Status			String	10	

< >

Help OK Cancel

# CreditCard\_Fact (7)



Good\_Debt: Good Debt will be change to 1, while Bad Debt will be change to 0.

Bad\_Debt: Good Debt will be change to 0, while Bad Debt will be change to 1.

Replace in string

Step name: Replace in string 2

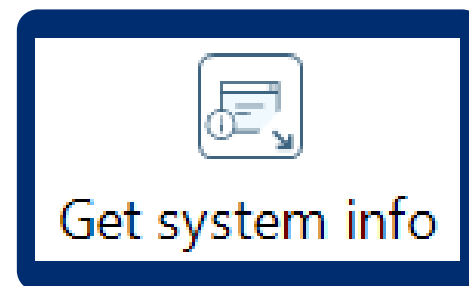
Fields string

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set emp
1	Good_Debt		N	Good Debt	1	N
2	Good_Debt		N	Bad Debt	0	N
3	Bad_Debt		N	Good Debt	0	N
4	Bad_Debt		N	Bad Debt	1	N

< >

Help OK Get fields Cancel

# CreditCard\_Fact (8)



To create date & time when ETL was performed.

Get system info

Step name

Fields:

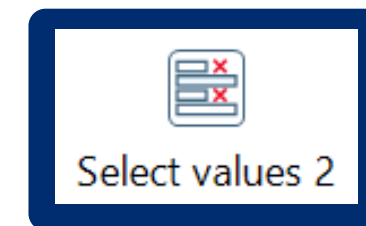
#	Name	Type	
1	Last_ETL	system date (variable)	



# CreditCard\_Fact (9)

Select only selected Value.

#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?
1	Time_ID		BigNumber			N
2	Applicant_ID		String			N
3	CreditRecord_ID		String			N
4	Good_Debt		Integer	1		N
5	Bad_Debt		Integer	1		N
6	Last_ETL		None			N



Select values

Step name Select values 2

Select & Alter Remove Meta-data

Fields to alter the meta-data for :

#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?	Forr
1	Time_ID		BigNumber			N	
2	Applicant_ID		String			N	
3	CreditRecord_ID		String			N	
4	Good_Debt		Integer	1		N	
5	Bad_Debt		Integer	1		N	
6	Last_ETL		None			N	

Get fields to change

Help OK Cancel

Select values

Step name Select values 2

Select & Alter Remove Meta-data

Fields to remove :

#	Fieldname
1	Applicant_ID_1
2	Record_Date
3	Status

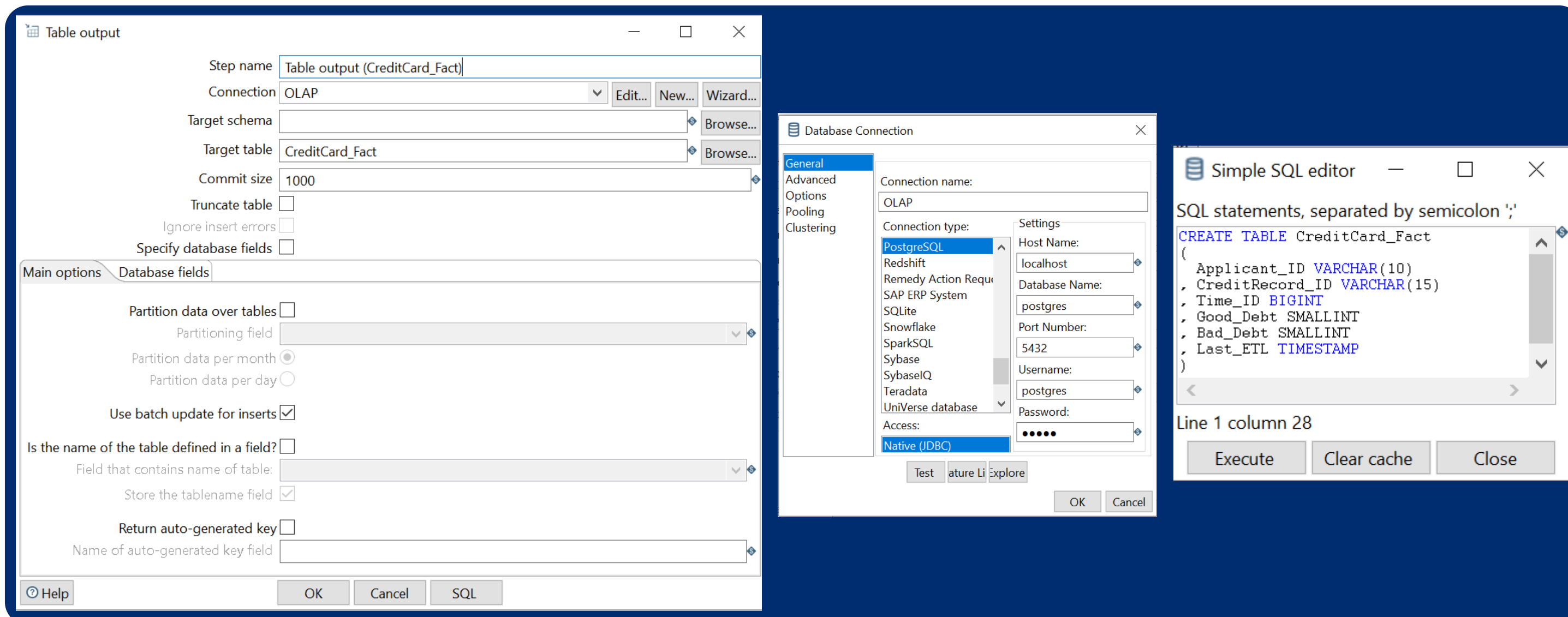
Get fields to remove

Help OK Cancel

# CreditCard\_Fact (10)

Export to OLAP DB on  
your RDBMS.

Table output (CreditCard\_Fact)



The screenshot displays the Tableau ETL process interface. On the left, the 'Table output' window is open, showing the 'Step name' as 'Table output (CreditCard\_Fact)', 'Connection' as 'OLAP', 'Target schema' as 'CreditCard\_Fact', and 'Commit size' as '1000'. The 'Database fields' tab is selected, showing options for partitioning and batch updates. On the right, the 'Database Connection' window is open, showing the 'Connection name' as 'OLAP', 'Connection type' as 'PostgreSQL', and 'Settings' for 'Host Name' (localhost), 'Database Name' (postgres), 'Port Number' (5432), 'Username' (postgres), and 'Password' (masked). Below these windows, a 'Simple SQL editor' window is open, displaying the SQL statement: 

```
CREATE TABLE CreditCard_Fact
(
  Applicant_ID VARCHAR(10)
, CreditRecord_ID VARCHAR(15)
, Time_ID BIGINT
, Good_Debt SMALLINT
, Bad_Debt SMALLINT
, Last_ETL TIMESTAMP
)
```

 The editor shows 'Line 1 column 28' and has buttons for 'Execute', 'Clear cache', and 'Close'.



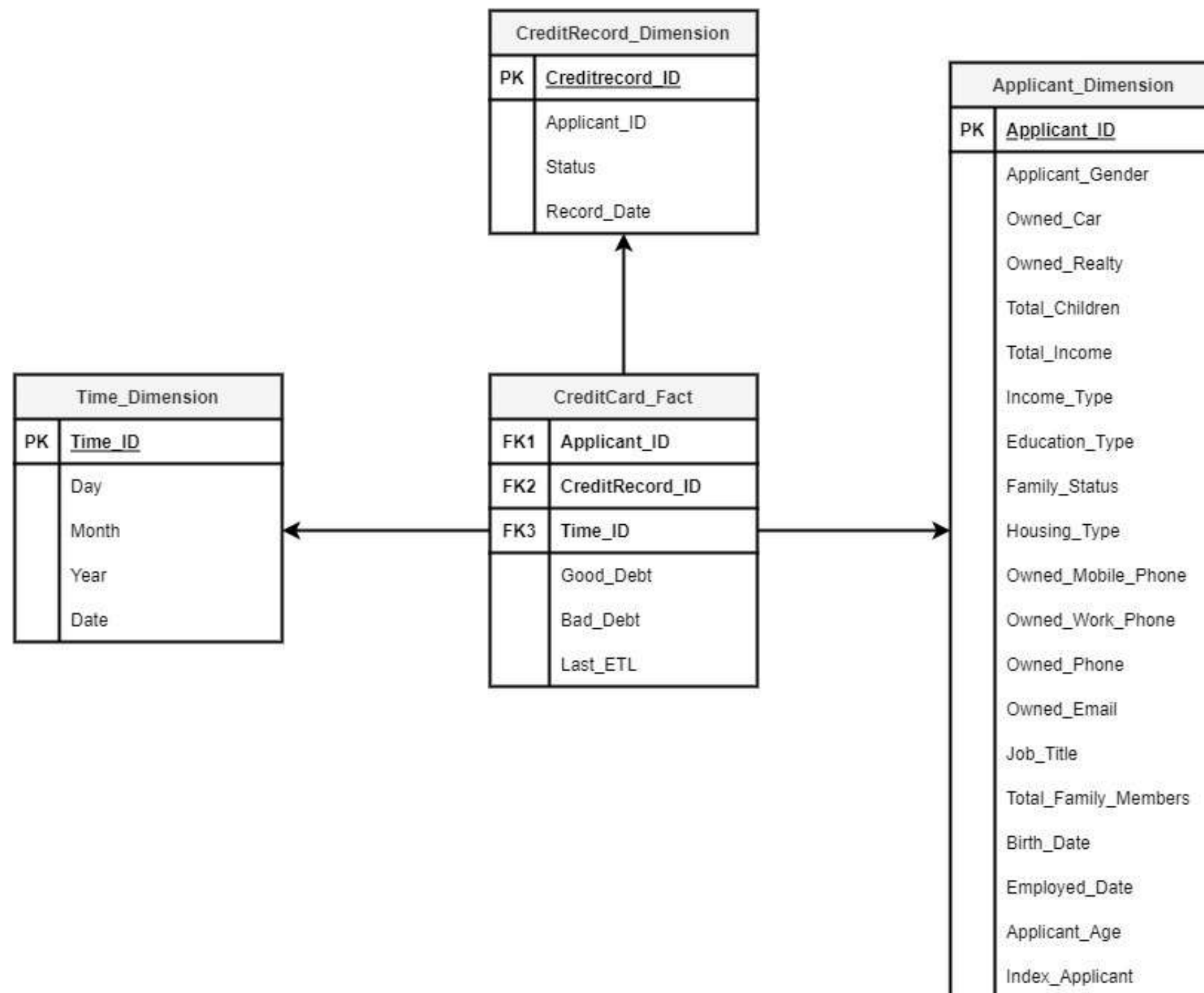


LARUTAN CAP BADAK GROUP

# Star Schema



# Star Schema







LARUTAN CAP BADAK GROUP

# OLAP Output



# Result from OLAP

## Output random testing

### • Credit Card Fact

- select \* from creditcard\_fact where applicant\_id='5008824';

	applicant_id	creditrecord_id	time_id	good_debt	bad_debt	last_etl
1	5008824	93173	20,211,001	1	0	22-10-11 11:47:49.608
2	5008824	93174	20,210,901	1	0	22-10-11 11:47:49.608
3	5008824	93175	20,210,801	1	0	22-10-11 11:47:49.608
4	5008824	93176	20,210,701	1	0	22-10-11 11:47:49.608

- select applicant\_id, sum(good\_debt) as good\_debt, sum(bad\_debt) as bad\_debt, last\_etl from creditcard\_fact where applicant\_id='5008824' group by applicant\_id, last\_etl;

	applicant_id	good_debt	bad_debt	last_etl
1	5008824	4	0	2022-10-11 11:47:49.608

### • Credit Record Information

- select \* from credit\_record where id='5008824';

ID	MONTHS_B/	STATUS
5008824	0 0	
5008824	-1 0	
5008824	-2 0	
5008824	-3 0	

- select \* from creditrecord\_dimension where applicant\_id='5008824';

	applicant_id	status	record_date	creditrecord_id
1	5008824	0	21-10-01 00:00:00.000	93173
2	5008824	0	21-09-01 00:00:00.000	93174
3	5008824	0	21-08-01 00:00:00.000	93175
4	5008824	0	21-07-01 00:00:00.000	93176

# Result from OLAP Output random testing

## • Applicant General Information

- select \* from applicant\_record where id='5008824';

ID	CODE_GEND	FLAG_OWN	FLAG_OWN	CNT_CHILDR	AMT_INCOM	NAME_INCO	NAME_EDUC	NAME_FAMI	NAME_HOU	DAYS_BIRTH
5008824	M	Y	Y	0	135000	Commercial as	Secondary / se	Married	House / apartm	-17778


DAYS_EMPL	FLAG_MOBII	FLAG_WORK	FLAG_PHON	FLAG_EMAIL	OCCUPATIO	CNT_FAM_M
-1194	1	0	0	0	Laborers	2

- select \* from applicant\_dimension where applicant\_id='5008824';

	ABC applicant_id 🔼🔼	ABC applicant_gender 🔼🔼	ABC owned_car 🔼🔼	ABC owned_realty 🔼🔼	123 total_children 🔼🔼	123 total_income 🔼🔼	ABC income_type 🔼🔼	ABC education_type 🔼🔼
1	5008824	Male	Have	Have	0	135,000	Commercial associate	Secondary / secondary special

ABC family_status 🔼🔼	ABC housing_type 🔼🔼	ABC owned_mobile_phone 🔼🔼	ABC owned_work_phon 🔼🔼	ABC owned_phone 🔼🔼	ABC owned_email 🔼🔼	ABC job_title 🔼🔼	123 total_family_member 🔼🔼
Married	House / apartment	Have	Don't Have	Don't Have	Don't Have	Laborers	2

🕒 birth_date 🔼🔼	🕒 employed_date 🔼🔼	123 applicant_age 🔼🔼	123 index_applicant 🔼🔼
1973-01-28 00:00:00.000	2018-06-25 00:00:00.000	48	12





# Result from OLAP Output random testing

- select applicant\_age, birth\_date, applicant\_gender, family\_status  
from applicant\_dimension where applicant\_id='5008824';

	123 applicant_age T↑	🕒 birth_date T↑	ABC applicant_gender T↑	ABC family_status T↑
1	48	1973-01-28 00:00:00.000	Male	Married





LARUTAN CAP BADAK GROUP

# Insights



# Insights

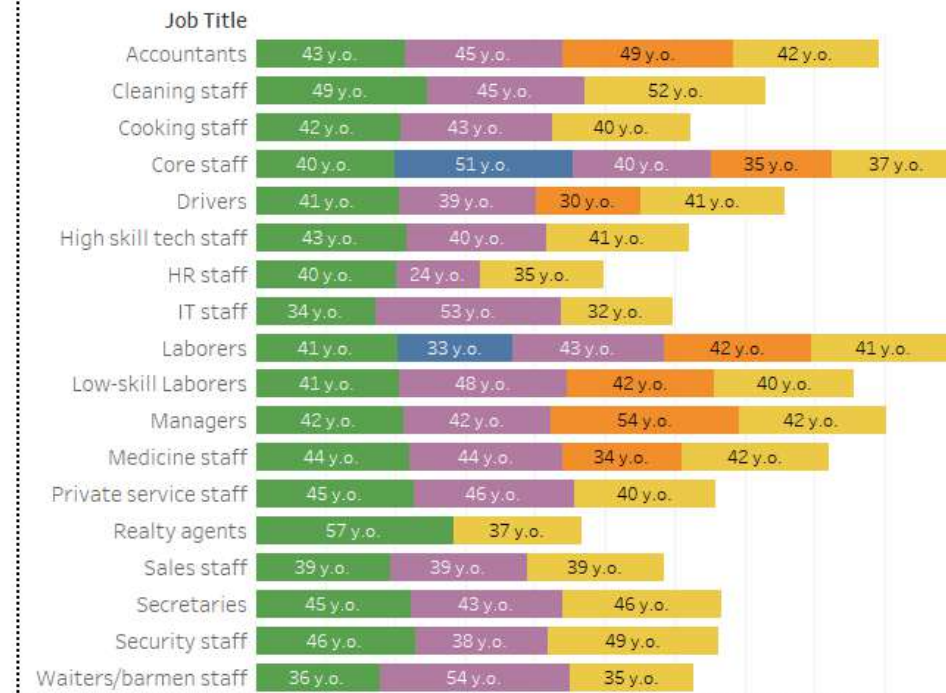
## Applicant Credit Record

Applicant  
**24,79K**

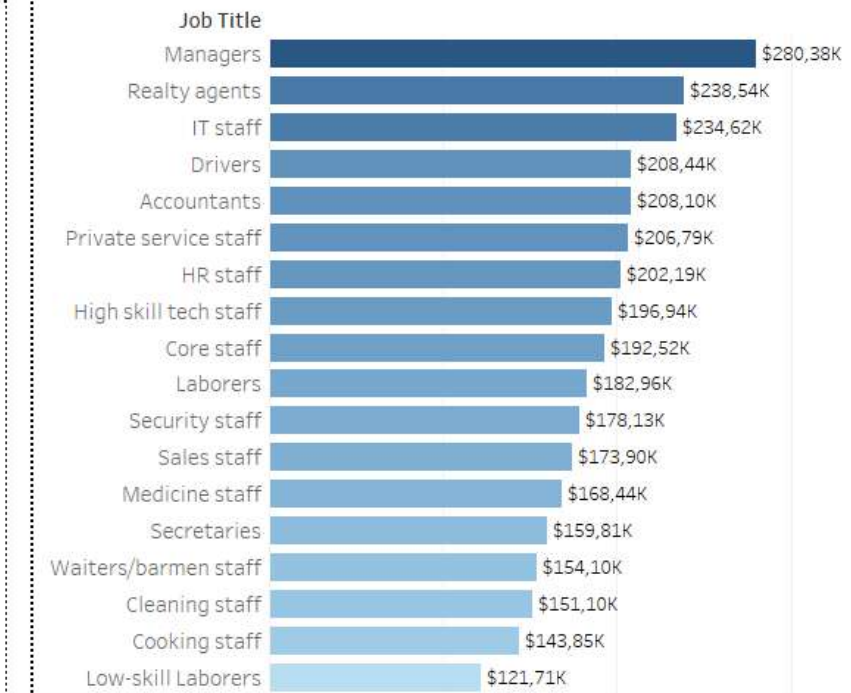
Credit Card  
**535,84K**

Credit Record  
**129,63M**

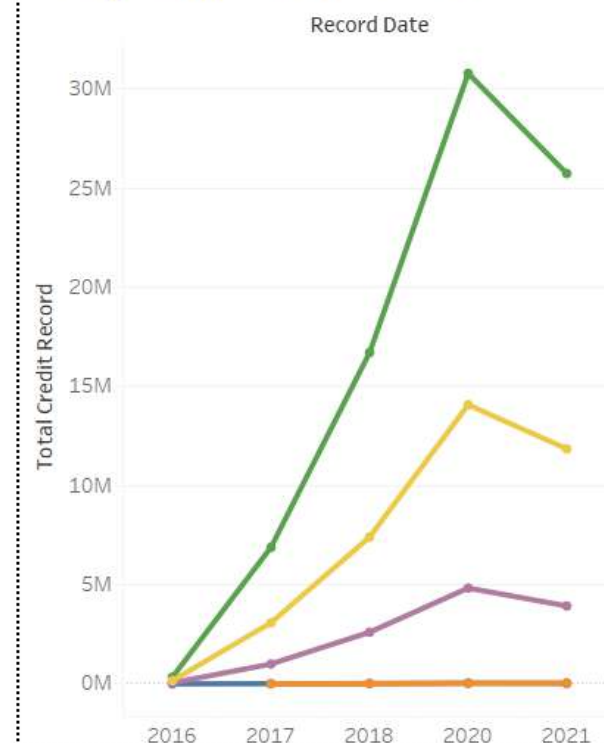
### Average Applicant Age by Job Title



### Average Income by Job Title



### Yearly Total Credit Records



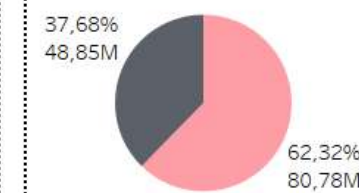
Income Type

- Commercial associate
- Pensioner
- State servant
- Student
- Working

Applicant Gender

- Female
- Male

### Total Credit Records by Gender



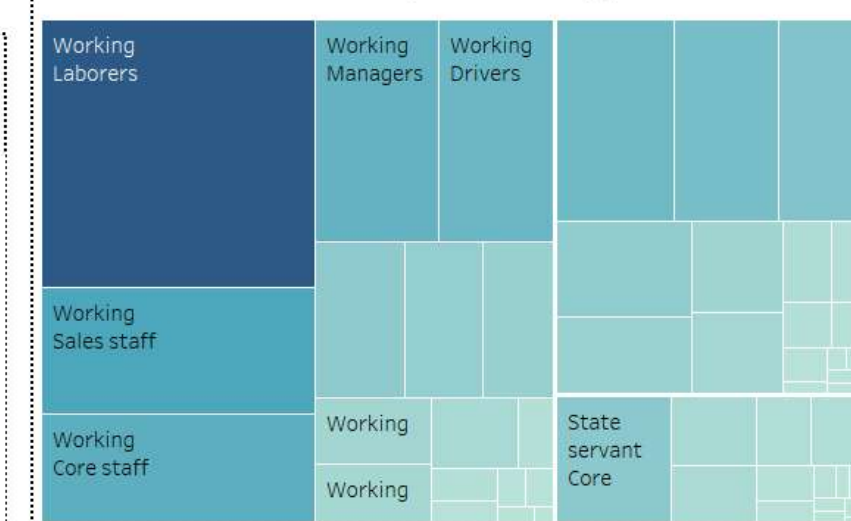
Avg. Total Income

\$121,71K

Total Credit Record

0,00M

### Total Credit Records by Income Type & Job Title







# Point of Insights

01

## Average Applicant Age by Job Title

Each job title is carried out by applicants of various ages. One of them is Accountants who are carried out applicants with an average age of 43 y.o. in the Working income type, 45 y.o. in the State servant income type, 49 y.o. in the Pensioner income type, and 42 y.o. in the Commercial associate income type.

02

## Average Income by Job Title

The largest average income by job title is Managers with an average total income of \$280,38K.

03

## Yearly Total Credit Records

Based on yearly total credit records, Working income is the highest total credit, but in 2020 the total credit record has decreased significantly.

04

## Total Credit Records by Gender

Females do more credit than males, with a total female credit record of 48,85M.

05

## Total Credit Records by Income Type and Job Title

The top 5 types of income with job titles from the applicants are as follows:

- |                         |                      |
|-------------------------|----------------------|
| 1) Working, Laborers    | 4) Working, Managers |
| 2) Working, Sales staff | 5) Working, Drivers  |
| 3) Working, Core staff  |                      |



LARUTAN CAP BADAK GROUP

# Thank You