

Projeto Final em Engenharia Informática

Projeto de Data Warehouse

Implementação e análise

Manual de utilização da implementação da Data Warehouse

Pedro Miguel Sequeira Narciso

Professor Luís Cavique

Lisboa 10 de Dezembro de 2020

Descreve-se neste manual a implementação das instruções de SQL realizadas durante a implementação da Datawarehouse resultante do Projeto Final de Engenharia Informática elaborado no âmbito da Licenciatura em Engenharia Informática da Universidade Aberta.

# Extração de tabelas

## 1.1 Tabela dim\_actor

Extração de dados da tabela ***actor*** da base de dados ***sakila***

INSERT INTO 101\_dim\_actor (

actor\_id,

first\_name,

last\_name,

last\_update,

extract\_date )

SELECT

actor.actor\_id,

actor.first\_name,

actor.last\_name,

actor.last\_update,

Now() AS extract\_date

FROM actor ;

## 1.2 Tabela dim\_customers

Extração de dados da tabela ***customers*** da base de dados ***sakila***

INSERT INTO 102\_dim\_customers (

customer\_id,

first\_name,

last\_name,

city,

store,

active,

create\_date,

last\_update,

extract\_date )

SELECT

CST.customer\_id,

CST.first\_name,

CST.last\_name,

CT1.city AS city,

CT.city AS store,

CST.active,

CST.create\_date,

CST.last\_update, Now() AS [extract]

FROM (city AS CT

INNER JOIN (address AS AD1

INNER JOIN store AS STR ON AD1.address\_id = STR.address\_id) ON CT.city\_id = AD1.city\_id)

INNER JOIN ((address AS ADR

INNER JOIN city AS CT1 ON ADR.city\_id = CT1.city\_id)

INNER JOIN customer AS CST ON ADR.address\_id = CST.address\_id) ON STR.store\_id = CST.store\_id ;

## 1.3 Tabela dim\_film

Extração de dados da tabela ***movies*** da base de dados ***suppliers***

INSERT INTO 103\_dim\_film (

movie\_id,

title,

release\_year,

origin,

last\_update,

extract\_date )

SELECT

movie\_id,

title,

release\_year,

'supp\_aquisitions' AS tabela,

last\_update,

Now() AS Extract\_date

FROM movies ;

Extração de dados das tabelas ***film e language*** da base de dados ***sakila***

INSERT INTO 103\_dim\_film (

film\_id,

title,

release\_year,

name,

origin,

last\_update,

extract\_date )

SELECT

F.film\_id,

F.title,

F.release\_year,

L.name,

'Sakila\_film' AS origin,

F.last\_update,

Now() AS extract\_date

FROM [language] AS L

INNER JOIN film AS F ON L.language\_id = F.language\_id ;

## 1.4 Tabela dim\_staff

Extração de dados das tabelas ***employees e department*** da base de dados ***employees***

INSERT INTO 104\_dim\_staff (

employee\_id,

first\_name,

last\_name,

hire\_date,

exit\_date,

department\_name,

last\_update,

extract\_date )

SELECT

E.employee\_id,

E.first\_name,

E.last\_name,

E.hire\_date,

E.exit\_date,

D.department\_name,

E.last\_update,

Now() AS extract\_date

FROM departments AS D

INNER JOIN (employees AS E

INNER JOIN department\_employee AS DE

ON E.employee\_id = DE.employee\_id)

ON D.department\_id = DE.department\_id ;

Extração de dados da tabela ***staff*** da base de dados ***sakila***

INSERT INTO 104\_dim\_staff (

staff\_id,

first\_name,

last\_name,

last\_update,

extract\_date )

SELECT

staff\_id,

first\_name,

last\_name,

last\_update,

Now() AS extract\_date;

FROM staff

## 1.5 Tabela dim\_suppliers

Extração de dados da tabela ***suppliers*** da base de dados ***suppliers***

INSERT INTO 105\_dim\_suppliers (

supplier\_id,

name,

supplier\_type,

last\_update,

extract\_date )

SELECT

supplier\_id,

name,

supplier\_type,

last\_update,

Now() AS extract\_date

FROM suppliers ;

## 1.6 Tabela dim\_rental

Extração de dados da tabela ***rental***  da base de dados ***sakila***

INSERT INTO 106\_dim\_rentals (

rental\_id,

rental\_date,

return\_date,

last\_update,

extract\_date )

SELECT

rental\_id,

rental\_date,

return\_date,

last\_update,

Now() AS extract\_date

FROM rental ;

## 1.7 Tabela dim\_category

Extração de dados da tabela ***category***  da base de dados ***sakila***

INSERT INTO 108\_dim\_category (

category\_id,

category\_name,

last\_update,

extract\_date )

SELECT

category\_id,

name,

last\_update,

Now() AS extract\_date

FROM category ;

## 1.8 Tabela fact\_aquisitions

Extração de dados da tabela ***aquisitions*** da base de dados ***suppliers***

INSERT INTO 120\_fact\_aquisitions (

aquisition\_id,

movie\_id,

supplier\_id,

staff\_id,

unit\_price,

quantity,

aquisition\_date,

extract\_date )

SELECT

aquisition\_id,

movie\_id,

supplier\_id,

staff\_id,

price AS unit\_price,

quantity,

aquisition\_date,

Now() AS [extract]

FROM aquisitions ;

## 1.9 Tabela fact\_communications

Extração de dados da tabela ***communication, contacts*** e ***suppliers*** da base de dados ***suppliers***

INSERT INTO 121\_fact\_communication (

comm\_id,

supplier\_id,

staff\_id, type,

comm\_date,

extract\_date )

SELECT

CM.comm\_id,

S.supplier\_id,

CM.staff\_id,

CM.type,

CM.comm\_date,

Now() AS [extract]

FROM suppliers AS S

INNER JOIN (contacts AS C

INNER JOIN communications AS CM

ON C.contact\_id = CM.contact\_id)

ON S.supplier\_id = C.supplier\_id ;

## 1.10 Tabela fact\_film\_actor

Extração de dados da tabela ***film\_actor*** da base de dados ***sakila***

INSERT INTO 122\_fact\_film\_actor (

actor\_id,

film\_id,

last\_update,

extract\_date )

SELECT

actor\_id,

film\_id,

last\_update,

Now() AS extract\_date

FROM film\_actor;

## 1.10 Tabela fact\_payment

Extração de dados das tabelas ***inventory, rental e payment*** da base de dados ***sakila***

INSERT INTO 123\_fact\_payment (

payment\_id,

staff\_id,

amount,

payment\_date,

last\_update,

customer\_id,

rental\_id,

film\_id,

extract\_date )

SELECT

P.payment\_id,

P.staff\_id,

P.amount,

P.payment\_date,

P.last\_update,

P.customer\_id,

P.rental\_id, I.film\_id,

Now() AS [extract]

FROM (inventory AS I

INNER JOIN rental AS R ON I.inventory\_id = R.inventory\_id)

INNER JOIN payment AS P ON R.rental\_id = P.rental\_id ;

## 1.11 Tabela fact\_relatives

Extração de dados das tabelas ***relatives*** da base de dados ***employee***

INSERT INTO 124\_fact\_relatives (

relative\_id,

employee\_id,

relantionship\_id,

first\_name,

last\_name,

gender,

birth\_date,

last\_update, extract\_date )

SELECT

relative\_id,

employee\_id,

relationship\_id,

first\_name,

last\_name,

gender,

birth\_date,

last\_update,

Now() AS [extract]

FROM relatives;

## 1.12 Tabela fact\_salaries

Extração de dados das tabelas ***titles e salaries*** da base de dados ***employee***

INSERT INTO 125\_fact\_salaries (

employee\_id,

title,

salary,

start\_date,

extract\_date )

SELECT

salaries.employee\_id,

titles.title,

salaries.salary,

salaries.start\_date,

Date() AS [extract]

FROM titles

INNER JOIN salaries ON titles.title\_id = salaries.title\_id;

## 1.13 Tabela film\_category

Extração de dados da tabela ***film\_category*** da base de dados ***sakila***

INSERT INTO 126\_fact\_film\_category (

film\_id,

category\_id,

last\_update,

extract\_date )

SELECT

film\_id,

category\_id,

last\_update,

Now() AS extract\_date

FROM film\_category ;

## 1.14 Tabela dim\_calendar

Extração de dados da tabela ***120\_fact\_aquisitions***

INSERT INTO 107\_dim\_calendar (

calendar,

tbl\_origem )

SELECT DISTINCT

aquisition\_date,

'120\_fact\_aquisitions' AS tabela

FROM 120\_fact\_aquisitions

GROUP BY aquisition\_date;

Extração de dados da tabela ***121\_fact\_communications***

INSERT INTO 107\_dim\_calendar (

calendar,

tbl\_origem )

SELECT

comm\_date AS calendar,

'ETL\_fact\_communications' AS tbl\_origem

FROM 121\_fact\_communication

GROUP BY comm\_date;

Extração de dados da tabela ***123\_fact\_payment***

INSERT INTO 107\_dim\_calendar (

calendar,

tbl\_origem )

SELECT DISTINCT

payment\_date AS calendar,

'123\_fact\_payment' AS tbl\_origem

FROM 123\_fact\_payment

GROUP BY payment\_date;

Extração de dados da tabela ***124\_fact\_relatives***

INSERT INTO 107\_dim\_calendar (

calendar,

tbl\_origem )

SELECT DISTINCT

birth\_date,

'124\_fact\_relatives' AS tbl\_origem

FROM 124\_fact\_relatives

GROUP BY birth\_date;

Extração de dados da tabela ***125\_fact\_salaries***

INSERT INTO 107\_dim\_calendar (

calendar,

tbl\_origem )

SELECT DISTINCT

start\_date AS calendar,

'125\_fact\_salaries' AS tbl\_origem

FROM 125\_fact\_salaries

GROUP BY start\_date;

# Transformações em tabelas

## Tabela dim\_actor

Transformar de dados da tabela ***101\_dim\_actor***

INSERT INTO 201\_dim\_actor (

actor\_id,

first\_name,

last\_name,

load\_date )

SELECT

actor\_id,

first\_name,

last\_name,

extract\_date

FROM 101\_dim\_actor ;

## Tabela dim\_customers

Transformar dados da tabela ***121\_dim\_customers***

Conversão do campo ***create\_date***, originalmente no formato *data/hora* para o formato de *data*.

INSERT INTO 202\_dim\_customers (

customer\_id,

first\_name,

last\_name,

city, store,

create\_date,

active,

load\_date )

SELECT

customer\_id,

first\_name,

last\_name,

city,

store,

DateSerial(Year([create\_date]),Month([create\_date]),Day([create\_date])) AS create\_dt,

active,

extract\_date

FROM 102\_dim\_customers ;

## Tabela dim\_films

Transformar dados da tabela ***103\_dim\_films***

Utilização de Group BY de forma a agrupar os filmes cujos títulos e anos de lançamento sejam idênticos;

Utilização da função agregadora MAX de forma a retirar um só valor dos campos cuja origem foi só uma tabela.

Utilização da função agregadora MIN no campo extract\_date.

INSERT INTO 203\_dim\_film (

film\_id,

movie\_id,

title,

release\_year,

film\_language,

load\_date )

SELECT

Max(film\_id) AS max\_film\_id,

Max(movie\_id) AS max\_movie\_id,

title,

release\_year,

Max(name) AS film\_language,

Min(extract\_date) AS load\_date

FROM

103\_dim\_Film

GROUP BY

title, release\_year;

Sobre a tabela resultante deve-se fazer algumas verificações de coerência, por exemplo verificar se o campo title possui um valor “null” ou se os campos movie\_id ou film\_id não têm referência nas duas tabelas importadas (se tiver valor “zero”). Nesta situação ou o filme só existe numa tabela, ou existe diferença na grafia.

SELECT

film\_key,

title,

movie\_id,

film\_id

FROM 203\_dim\_film

WHERE (((title) Is Null) OR ((movie\_id)=0)) OR (((film\_id)=0));

## Tabela dim\_staff

Transformar dados da tabela ***103\_dim\_staff***

Utilização de Group BY de forma a agrupar os funcionários como mesmo nome e apelido;

Utilização da função agregadora MAX de forma a retirar um só valor dos campos cuja origem foi só uma tabela.

Utilização da função agregadora MIN no campo extract\_date.

INSERT INTO 204\_dim\_staff (

staff\_id,

employee\_id,

first\_name,

last\_name,

hire\_date,

exit\_date,

department\_name,

load\_date )

SELECT

Max(staff\_id) AS staffid,

Max(employee\_id) AS employeeid,

first\_name,

last\_name,

Max(hire\_date) AS HrDt,

Max(exit\_date) AS ExtDt,

Max(department\_name) AS DepNa,

Min(extract\_date) AS Extract

FROM 104\_dim\_staff

GROUP BY

first\_name, last\_name;

## Tabela dim\_suppliers

Transformar dados da tabela ***105\_dim\_suppliers***

INSERT INTO 205\_dim\_suppliers (

supplier\_id,

supplier\_name,

supplier\_type,

load\_date )

SELECT

supplier\_id,

name,

supplier\_type,

extract\_date

FROM 105\_dim\_suppliers ;

## Tabela dim\_rental

Transformar dados da tabela ***106\_dim\_rental***

Conversão do campo ***rental\_date*** e ***return\_date***, originalmente no formato *data/hora* para o formato de *data*.

INSERT INTO 206\_dim\_rental (

rental\_id,

rental\_date,

return\_date,

load\_date )

SELECT

rental\_id,

DateSerial(Year(rental\_date),Month(rental\_date),Day(rental\_date)) AS RenDate,

DateSerial(Year(return\_date),Month(return\_date),Day(return\_date)) AS RetDate, extract\_date

FROM 106\_dim\_rental ;

## Tabela dim\_calendar

Transformar dados da tabela ***107\_dim\_calendar***

Efetua-se o Group\_by de forma a eliminar as datas repetidas

INSERT INTO 207\_dim\_calendar (

calendar,

nDay,

nWeekDay,

tWeekDay,

nYearDay,

nYearWeek,

nMonth,

tMonth,

nQuarter,

nYear )

SELECT

DateSerial(Year([calendar]),Month([calendar]),Day([calendar])) AS Calendario,

Day([calendar]) AS nDay,

Weekday([calendar]) AS nWeekDay,

WeekdayName(Weekday([calendar]),True,1) AS tWeekDay,

Format([calendar],'y') AS nYearDay,

Format([calendar],'ww') AS nWeekYear,

Month([calendar]) AS nMonth,

MonthName(Month([calendar]),True) AS tMonth,

Format([calendar],'q') AS nQuarter,

Year([calendar]) AS nYear

FROM 107\_dim\_calendar

GROUP BY

DateSerial(Year([calendar]),

Month([calendar]),

Day([calendar])),

Day([calendar]),

Weekday([calendar]),

WeekdayName(Weekday([calendar]),True,1),

Format([calendar],'y'),

Format([calendar],'ww'),

Month([calendar]),

MonthName(Month([calendar]),True),

Format([calendar],'q'),

Year([calendar]);

## Tabela dim\_category

Transformar dados da tabela ***108\_dim\_category***

INSERT INTO 208\_dim\_category (

category\_id,

category\_name,

load\_date )

SELECT

category\_id,

category\_name,

extract\_date

FROM 108\_dim\_category ;

## Tabela fact\_aquisitions

Transformar dados da tabela ***120\_fact\_aquisitions***

Conversão do campo ***aquisition\_date***, originalmente no formato *data/hora* para o formato de *data*.

Criado o campo ***total\_price***, resultante da multiplicação: [price]\*[quantity]

INSERT INTO 220\_fact\_aquisitions (

aquisition\_id,

movie\_id,

supplier\_id,

staff\_id,

unit\_price,

total\_price,

quantity,

aquisition\_date,

load\_date )

SELECT

aquisition\_id,

movie\_id,

supplier\_id,

staff\_id,

price,

[price]\*[quantity] AS total,

quantity,

DateSerial(Year([aquisition\_date]),Month([aquisition\_date]),Day([aquisition\_date])) AS AquDate,

extract\_date

FROM 120\_fact\_aquisitions ;

## Tabela fact\_communications

Transformar dados da tabela ***121\_fact\_communications***

INSERT INTO 221\_fact\_communications (

comm\_id,

supplier\_id,

staff\_id,

comm\_type,

comm\_date,

load\_date )

SELECT

comm\_id,

supplier\_id,

staff\_id,

type,

comm\_date,

extract\_date

FROM 121\_fact\_communications;

## Tabela fact\_film\_actor

Transformar dados da tabela ***122\_fact\_film\_actor***

INSERT INTO 222\_fact\_film\_actor (

actor\_id,

film\_id,

load\_date )

SELECT

actor\_id AS aid,

film\_id AS fid,

extract\_date AS load\_date

FROM 122\_fact\_film\_actor ;

## Tabela fact\_payment

Transformar dados da tabela ***123\_fact\_payment***

Conversão do campo ***payment\_date***, originalmente no formato *data/hora* para o formato de *data*.

INSERT INTO 223\_fact\_payment (

payment\_id,

customer\_id,

staff\_id,

rental\_id,

film\_id,

amount,

payment\_date,

load\_date )

SELECT

payment\_id,

customer\_id,

staff\_id,

rental\_id,

film\_id,

amount,

DateSerial(Year(payment\_date),Month(payment\_date),Day(payment\_date)) AS PaymDate,

extract\_date

FROM 123\_fact\_payment ;

## Tabela fact\_relatives

Transformar dados da tabela ***124\_fact\_relatives***

Só se efetua a transformação nos registos dos “filhos” e “filhas”.

INSERT INTO 224\_fact\_relatives (

relative\_id,

employee\_id,

child\_first\_name,

child\_last\_name,

child\_gender,

child\_birth\_date,

load\_date )

SELECT

relative\_id,

employee\_id,

first\_name,

last\_name,

gender,

birth\_date,

extract\_date

FROM 124\_fact\_relatives

WHERE (relantionship\_id)=3 Or (relantionship\_id)=4;

## Tabela fact\_salaries

Transformar dados da tabela ***125\_fact\_salaries***

INSERT INTO 225\_fact\_salaries (

employee\_id,

title,

salary,

start\_date,

load\_date )

SELECT

employee\_id,

title,

salary,

start\_date,

extract\_date

FROM 125\_fact\_salaries ;

## Tabela fact\_film\_category

Transformar dados da tabela ***126\_fact\_film\_category***

INSERT INTO 226\_fact\_film\_category (

film\_id,

category\_id,

load\_date )

SELECT

film\_id,

category\_id,

extract\_date

FROM 126\_fact\_film\_category;

# Transformações em tabelas

## Tabela dim\_actor

Carregar dados da tabela ***201\_dim\_actor***

INSERT INTO 301\_dim\_actor (

actor\_id,

first\_name,

last\_name,

load\_date )

SELECT

actor\_id,

first\_name,

last\_name,

load\_date

FROM 201\_dim\_actor ;

## Tabela dim\_customer

Carregar dados da tabela ***202\_dim\_customers***

INSERT INTO 302\_dim\_customer (

customer\_id,

first\_name,

last\_name,

city,

store,

create\_date,

active,

load\_date )

SELECT

customer\_id,

first\_name,

last\_name,

city,

store,

create\_date,

active,

load\_date

FROM 202\_dim\_customers ;

## Tabela dim\_film

Extração de dados da tabela ***203\_dim\_film***

INSERT INTO 303\_dim\_film (

film\_id,

movie\_id,

title,

release\_year,

film\_language,

load\_date )

SELECT

film\_id,

movie\_id,

title,

release\_year,

film\_language,

load\_date

FROM 203\_dim\_film ;

## Tabela dim\_staff

Carregar dados da tabela ***204\_dim\_staff***

INSERT INTO 304\_dim\_staff (

staff\_id,

employee\_id,

first\_name,

last\_name,

hire\_date,

exit\_date,

department\_name,

load\_date )

SELECT

staff\_id,

employee\_id,

first\_name,

last\_name,

hire\_date,

exit\_date,

department\_name,

load\_date

FROM 204\_dim\_staff ;

## Tabela dim\_suppliers

Carregar dados da tabela ***205\_dim\_suppliers***

INSERT INTO 305\_dim\_suppliers (

supplier\_id,

supplier\_name,

supplier\_type,

load\_date )

SELECT

supplier\_id,

supplier\_name,

supplier\_type,

load\_date

FROM 205\_dim\_suppliers ;

## Tabela dim\_rental

Carregar dados da tabela ***206\_dim\_rental***

INSERT INTO 306\_dim\_rental (

rental\_id,

rental\_date,

return\_date,

load\_date )

SELECT

rental\_id,

rental\_date,

return\_date,

load\_date

FROM 206\_dim\_rental ;

## Tabela dim\_category

Carregar dados da tabela ***208\_dim\_category***

INSERT INTO 308\_dim\_category (

category\_id,

category\_name,

load\_date )

SELECT

category\_id,

category\_name,

load\_date

FROM 208\_dim\_category ;

## Tabela dim\_calendar

Carregar dados da tabela ***207\_dim\_calendar***

INSERT INTO 307\_dim\_calendar (

calendar,

nDay,

nWeekday,

tWeekDay,

nYearDay,

nYearWeek,

nMonth,

tMonth,

nQuarter,

nyear )

SELECT

calendar,

nDay,

nWeekday,

tWeekDay,

nYearDay,

nYearWeek,

nMonth,

tMonth,

nQuarter,

nyear

FROM 207\_dim\_calendar ;

## Tabela fact\_aquisitions

Carregar dados da tabela ***220\_fact\_aquisitions***

Carregar dados da tabela ***303\_dim\_film***

Carregar dados da tabela ***304\_dim\_staff***

Carregar dados da tabela ***305\_dim\_suppliers***

Carregar dados da tabela ***307\_dim\_calendar***

INSERT INTO 320\_fact\_aquisitions (

aquisition\_id,

film\_key,

supplier\_key,

staff\_key,

unit\_price,

total\_price,

quantity,

aquisition\_date\_key,

load\_date )

SELECT

QFA.aquisition\_id,

DF.film\_key,

DS.supplier\_key,

DST.staff\_key,

QFA.unit\_price,

QFA.total\_price,

QFA.quantity,

DC.calendar\_key,

QFA.load\_date

FROM ((((SELECT

aquisition\_id,

movie\_id,

supplier\_id,

staff\_id,

unit\_price,

total\_price,

quantity,

aquisition\_date,

load\_date

FROM 220\_fact\_aquisitions) AS QFA

INNER JOIN 303\_dim\_film AS DF ON QFA.movie\_id = DF.movie\_id)

INNER JOIN 304\_dim\_staff AS DST ON QFA.staff\_id = DST.staff\_id)

INNER JOIN 305\_dim\_suppliers AS DS ON QFA.supplier\_id = DS.supplier\_id)

INNER JOIN 307\_dim\_calendar AS DC ON QFA.aquisition\_date = DC.calendar;

## Tabela fact\_communications

Carregar dados da tabela ***221\_fact\_communications***

Carregar dados da tabela ***305\_dim\_suppliers***

Carregar dados da tabela ***304\_dim\_staff***

Carregar dados da tabela ***307\_dim\_calendar***

INSERT INTO 321\_fact\_communication (

comm\_id,

supplier\_key,

staff\_key,

comm\_type,

comm\_date\_key,

load\_date )

SELECT

QFC.comm\_id,

SUP.supplier\_key,

STF.staff\_key,

QFC.comm\_type,

CAL.calendar\_key,

QFC.load\_date

FROM (((SELECT

comm\_id,

supplier\_id,

staff\_id,

comm\_type,

comm\_date,

load\_date FROM 221\_fact\_communications) AS QFC

INNER JOIN 305\_dim\_suppliers AS SUP ON QFC.supplier\_id = SUP.supplier\_id)

INNER JOIN 304\_dim\_staff AS STF ON QFC.staff\_id = STF.staff\_id)

INNER JOIN 307\_dim\_calendar AS CAL ON QFC.comm\_date = CAL.calendar;

## Tabels fact\_film\_actor

Carregar dados da tabela ***222\_fact\_film\_actor***

Carregar dados da tabela ***303\_dim\_film***

Carregar dados da tabela ***301\_dim\_actor***

INSERT INTO 322\_fact\_film\_actor (

actor\_key,

film\_key,

load\_date )

SELECT

ACT.actor\_key,

FLM.film\_key,

FFA.load\_date

FROM ((SELECT

actor\_id,

film\_id,

load\_date

FROM 222\_fact\_film\_actor IN) AS FFA

INNER JOIN 303\_dim\_film AS FLM ON FFA.film\_id = FLM.film\_id)

INNER JOIN 301\_dim\_actor AS ACT ON FFA.actor\_id = ACT.actor\_id;

## Tabela fact\_payment

Carregar dados da tabela ***223\_fact\_payment***

Carregar dados da tabela ***302\_dim\_customer***

Carregar dados da tabela ***304\_dim\_staff***

Carregar dados da tabela ***306\_dim\_rental***

Carregar dados da tabela ***303\_dim\_film***

Carregar dados da tabela ***307\_dim\_calendar***

INSERT INTO 323\_fact\_payment(

INSERT INTO 323\_fact\_payment (

payment\_id,

customer\_key,

staff\_key,

rental\_key,

film\_key,

amount,

payment\_date\_key,

load\_date )

SELECT

PYM.payment\_id,

CTM.customer\_key,

STF.staff\_key,

RTL.rental\_key,

FLM.film\_key,

PYM.amount,

CLD.calendar\_key,

PYM.load\_date

FROM (((((SELECT

payment\_id,

customer\_id,

staff\_id,

rental\_id,

film\_id,

amount,

payment\_date,

load\_date FROM 223\_fact\_payment ) AS PYM

INNER JOIN 302\_dim\_customer AS CTM ON PYM.customer\_id = CTM.customer\_id)

INNER JOIN 304\_dim\_staff AS STF ON PYM.staff\_id = STF.staff\_id)

INNER JOIN 306\_dim\_rental AS RTL ON PYM.rental\_id = RTL.rental\_id)

INNER JOIN 307\_dim\_calendar AS CLD ON PYM.payment\_date = CLD.calendar)

INNER JOIN 303\_dim\_film AS FLM ON PYM.film\_id = FLM.film\_id;

## Tabela fact\_relatives

Carregar dados da tabela ***224\_fact\_relatives***

Carregar dados da tabela ***304\_dim\_staff***

Carregar dados da tabela ***307\_dim\_calendar***

INSERT INTO 324\_fact\_relatives (

relative\_id,

staff\_key,

child\_first\_name,

child\_last\_name,

child\_gender,

child\_birth\_date\_key,

load\_date )

SELECT

RTL.relative\_id,

STF.staff\_key,

RTL.child\_first\_name,

RTL.child\_last\_name,

RTL.child\_gender,

CLD.calendar\_key,

RTL.load\_date

FROM ((SELECT

relative\_id,

employee\_id,

child\_first\_name,

child\_last\_name,

child\_gender,

child\_birth\_date,

load\_date FROM 224\_fact\_relatives) AS RTL

INNER JOIN 304\_dim\_staff AS STF ON RTL.employee\_id = STF.employee\_id)

INNER JOIN 307\_dim\_calendar AS CLD ON RTL.child\_birth\_date = CLD.calendar;

## Tabela fact\_salaries

Carregar dados da tabela ***225\_fact\_salaries***

Carregar dados da tabela ***304\_dim\_staff***

Carregar dados da tabela ***307\_dim\_calendar***

INSERT INTO 325\_fact\_salaries (

staff\_key,

title,

salary,

start\_date\_key,

load\_date )

SELECT

STF.staff\_key,

EMP.title, EMP.salary,

CLD.calendar\_key,

EMP.load\_date

FROM ((SELECT

employee\_id,

title,

salary,

start\_date,

load\_date

FROM 225\_fact\_salaries) AS EMP

INNER JOIN 304\_dim\_staff AS STF ON EMP.employee\_id = STF.employee\_id)

INNER JOIN 307\_dim\_calendar AS CLD ON EMP.start\_date = CLD.calendar;

## Tabela fact\_film\_category

Carregar dados da tabela ***226\_fact\_film\_category***

Carregar dados da tabela ***308\_dim\_category***

Carregar dados da tabela ***303\_dim\_film***

INSERT INTO 326\_fact\_film\_category (

film\_key,

category\_key,

load\_date )

SELECT

FLM.film\_key,

CAT.category\_key,

FFC.load\_date

FROM ((SELECT

film\_id,

category\_id,

load\_date FROM 226\_fact\_film\_category) AS FFC

INNER JOIN 308\_dim\_category AS CAT ON FFC.category\_id = CAT.category\_id)

INNER JOIN 303\_dim\_film AS FLM ON FFC.film\_id = FLM.film\_id;

# KPI’s

## Filmes com melhor saída

# R1 Filmes mais alugados (top mês 4)

SELECT TOP 10

FIL.title AS Titulo,

Count(REN.rental\_key) AS Alugueres

FROM 307\_dim\_calendar AS CAL

INNER JOIN (306\_dim\_rental AS REN

INNER JOIN (323\_fact\_payment AS PAY

INNER JOIN 303\_dim\_film AS FIL

ON PAY.film\_key = FIL.film\_key)

ON REN.rental\_key = PAY.rental\_key)

ON CAL.calendar\_key = PAY.payment\_date\_key

GROUP BY FIL.title, CAL.nMonth

HAVING (CAL.nMonth=4)

ORDER BY Count(REN.rental\_key) DESC;

## Relação semanal de filmes alugados

# R2 Relação semanal filmes alugados

TRANSFORM Count(REN.rental\_key) AS Contar

SELECT

CAL.nYearWeek AS Semana,

Count(REN.rental\_key) AS Contar1

FROM 307\_dim\_calendar AS CAL

INNER JOIN (306\_dim\_rental AS REN

INNER JOIN 323\_fact\_payment AS PAY

ON REN.rental\_key = PAY.rental\_key)

ON CAL.calendar\_key = PAY.payment\_date\_key

GROUP BY CAL.nYearWeek

PIVOT CAL.nWeekday;

## Preferência de atores alugados por mês

# R3 Relatório mensal atores e alugueres

TRANSFORM Count(REN.rental\_id) AS ContarDerental

SELECT

ACT.first\_name & ' ' & ACT.last\_name AS Actor,

Count(REN.rental\_id) AS QtyAlugueres

FROM (((303\_dim\_film AS FIL

INNER JOIN (301\_dim\_actor AS ACT

INNER JOIN 322\_fact\_film\_actor AS FAC

ON ACT.actor\_key = FAC.actor\_key)

ON FIL.film\_key = FAC.film\_key)

INNER JOIN 323\_fact\_payment AS PAY

ON FIL.film\_key = PAY.film\_key)

INNER JOIN 306\_dim\_rental AS REN

ON PAY.rental\_key = REN.rental\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON REN.rental\_date = CAL.calendar

GROUP BY

ACT.first\_name & ' ' & ACT.last\_name

PIVOT CAL.nMonth;

# R3 Relatório mensal atores e alugueres (ForExcel)

SELECT

ACT.first\_name & ' ' & ACT.last\_name AS Actor,

Count(REN.rental\_id) AS QtyAlugueres,

CAL.nMonth

FROM (((303\_dim\_film AS FIL

INNER JOIN (301\_dim\_actor AS ACT

INNER JOIN 322\_fact\_film\_actor AS FAC

ON ACT.actor\_key = FAC.actor\_key)

ON FIL.film\_key = FAC.film\_key)

INNER JOIN 323\_fact\_payment AS PAY

ON FIL.film\_key = PAY.film\_key)

INNER JOIN 306\_dim\_rental AS REN

ON PAY.rental\_key = REN.rental\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON REN.rental\_date = CAL.calendar

GROUP BY

ACT.first\_name & ' ' & ACT.last\_name,

CAL.nMonth

ORDER BY

Count(REN.rental\_id) DESC;

## Evolução semanal de alugueres por categoria

# R4 Evolução semanal de alugueres por categoria

TRANSFORM Count(CAT.category\_key) AS Soma

SELECT CAT.category\_name

FROM ((326\_fact\_film\_category AS FCA

INNER JOIN ((303\_dim\_film AS FIL

INNER JOIN 323\_fact\_payment AS PAY ON FIL.film\_key = PAY.film\_key)

INNER JOIN 306\_dim\_rental AS REN ON PAY.rental\_key = REN.rental\_key) ON FCA.film\_key = FIL.film\_key)

INNER JOIN 308\_dim\_category AS CAT ON FCA.category\_key = CAT.category\_key)

INNER JOIN 307\_dim\_calendar AS CAL ON REN.rental\_date = CAL.calendar

GROUP BY CAT.category\_name

PIVOT CAL.nYearWeek;

# R4 Evolução semanal de alugueres por categoria (ForExcel)

SELECT

CAT.category\_name,

Count(CAT.category\_key) AS Soma,

CAL.nYearWeek

FROM ((326\_fact\_film\_category AS FCA

INNER JOIN ((303\_dim\_film AS FIL

INNER JOIN 323\_fact\_payment AS PAY

ON FIL.film\_key = PAY.film\_key)

INNER JOIN 306\_dim\_rental REN

ON PAY.rental\_key = REN.rental\_key)

ON FCA.film\_key = FIL.film\_key)

INNER JOIN 308\_dim\_category AS CAT

ON FCA.category\_key = CAT.category\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON REN.rental\_date = CAL.calendar

GROUP BY

CAT.category\_name,

CAL.nYearWeek;

## Relação de funcionários e vencimentos

# R5 Relação Funcionários e vencimentos

SELECT

STA.first\_name & ' ' & STA.last\_name AS Nome,

SAL.title AS Categoria,

SAL.salary AS Vencimento

FROM 304\_dim\_staff AS STA

INNER JOIN 325\_fact\_salaries AS SAL

ON STA.staff\_key = SAL.staff\_key;

## Festa de natal crianças até 14 anos

# R6 Festa natal crianças idade até 14 anos

TRANSFORM Count(REL.relative\_key) AS numero

SELECT

REL.child\_gender

FROM 324\_fact\_relatives AS REL

INNER JOIN 307\_dim\_calendar AS CAL

ON REL.child\_birth\_date\_key = CAL.calendar\_key

WHERE

(((Year((Date()-[calendar]))-1900)<=14))

GROUP BY

REL.child\_gender

PIVOT Year((Date()-[calendar]))-1900;

## Relação de aquisições semanais por funcionário

# R7 Aquisições semanais por funcionário

TRANSFORM Count(AQU.aquisition\_id) AS Aquisições

SELECT

CAL.nYearWeek AS Semana,

Count(AQU.aquisition\_id) AS [Total Aquisições]

FROM ((320\_fact\_aquisition AS AQU

INNER JOIN 304\_dim\_staff AS STA

ON AQU.staff\_key = STA.staff\_key)

INNER JOIN 303\_dim\_film AS FIL

ON AQU.film\_key = FIL.film\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON AQU.aquisition\_date\_key = CAL.calendar\_key

GROUP BY

CAL.nYearWeek

PIVOT STA.first\_name & ' ' & STA.last\_name;

# R7 Aquisições semanais por funcionário (ForExcel)

SELECT

STA.first\_name & ' ' & STA.last\_name AS Funcionario,

Count(AQU.aquisition\_id) AS Aquisições,

CAL.nYearWeek AS Semana

FROM ((320\_fact\_aquisition AS AQU

INNER JOIN 304\_dim\_staff AS STA

ON AQU.staff\_key = STA.staff\_key)

INNER JOIN 303\_dim\_film AS FIL

ON AQU.film\_key = FIL.film\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON AQU.aquisition\_date\_key = CAL.calendar\_key

GROUP BY

STA.first\_name & ' ' & STA.last\_name,

CAL.nYearWeek;

## **Contactos mensais com fornecedores**

# R8 Contactos mensais com fornecedores

TRANSFORM

Count(COM.comm\_key) AS ContarDecomm\_key

SELECT

STA.first\_name & ' ' & STA.last\_name AS Nome,

SUP.supplier\_name, Count(COM.comm\_key) AS Total

FROM ((321\_fact\_communication AS COM

INNER JOIN 304\_dim\_staff AS STA

ON COM.staff\_key = STA.staff\_key)

INNER JOIN 305\_dim\_suppliers AS SUP

ON COM.supplier\_key = SUP.supplier\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON COM.comm\_date\_key = CAL.calendar\_key

GROUP

BY STA.first\_name & ' ' & STA.last\_name,

SUP.supplier\_name

PIVOT CAL.nMonth;

# R8 Contactos mensais com fornecedores (ForExcel)

SELECT

STA.first\_name & ' ' & STA.last\_name AS Nome,

COM.comm\_type,

SUP.supplier\_name,

Count(COM.comm\_key) AS ContarDecomm\_key,

CAL.nMonth

FROM ((321\_fact\_communication AS COM

INNER JOIN 304\_dim\_staff AS STA

ON COM.staff\_key = STA.staff\_key)

INNER JOIN 305\_dim\_suppliers AS SUP

ON COM.supplier\_key = SUP.supplier\_key)

INNER JOIN 307\_dim\_calendar AS CAL

ON COM.comm\_date\_key = CAL.calendar\_key

GROUP BY

STA.first\_name & ' ' & STA.last\_name,

COM.comm\_type,

SUP.supplier\_name,

CAL.nMonth;