DUT STID, Université de la Côte d'Azur

Bases de données avancées

Procédures stockées et triggers

Prof. Dario Malchiodi

UNIVERSITÀ DEGLI STUDI DI MILANO
DIPARTIMENTO DI INFORMATICA

UNIVERSITÉ CÔTE D'AZUR

Utiliser SQL directement dans jupyter

```
conda install -c conda-forge ipython-sql

conda install -c anaconda pymysql
```

In [1]:

```
try:
    import pymysql
    pymysql.install_as_MySQLdb()
except ImportError:
    pass
```

In [2]:

```
%load_ext sql
```

In [3]:

```
%sql mysql://superheroadmin:Passw0rd.@localhost/superheroes
```

```
/home/malchiodi/anaconda3/lib/python3.6/site-packages/pymysql/cursors.
py:170: Warning: (3719, "'utf8' is currently an alias for the characte
r set UTF8MB3, but will be an alias for UTF8MB4 in a future release. P
lease consider using UTF8MB4 in order to be unambiguous.")
  result = self._query(query)
```

Out[3]:

```
'Connected: superheroadmin@superheroes'
```

In [4]:

```
%sql SELECT * FROM heroes LIMIT 5;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

Out[4]:

id	name	identity	birth_place	publisher	height	weight	gender	first_appearance	eye_c
1	Wonder Woman	Diana Prince	Themyscira	DC Comics	183.13	74.74	F	1941	I
3	Wolverine	Logan	Alberta, Canada	Marvel Comics	160.7	135.21	M	None	I
4	Spider-Man	Peter Parker	New York, New York	Marvel Comics	178.28	74.25	M	None	H
5	Professor X	Charles Francis Xavier	New York, New York	Marvel Comics	183.74	86.89	M	1963	I
6	A-Bomb	Richard Milhouse Jones	Scarsdale, Arizona	Marvel Comics	203.21	441.95	M	2008	Ye

In [5]:

```
result = %sql SELECT * FROM heroes LIMIT 5;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

In [6]:

```
result
```

Out[6]:

id	name	identity	birth_place	publisher	height	weight	gender	first_appearance	eye_c
1	Wonder Woman	Diana Prince	Themyscira	DC Comics	183.13	74.74	F	1941	I
3	Wolverine	Logan	Alberta, Canada	Marvel Comics	160.7	135.21	M	None	I
4	Spider-Man	Peter Parker	New York, New York	Marvel Comics	178.28	74.25	M	None	H
5	Professor X	Charles Francis Xavier	New York, New York	Marvel Comics	183.74	86.89	M	1963	I
6	A-Bomb	Richard Milhouse Jones	Scarsdale, Arizona	Marvel Comics	203.21	441.95	M	2008	Ye

In [7]:

```
result[3]
```

Out[7]:

```
(5, 'Professor X', 'Charles Francis Xavier', 'New York, New York', 'Ma
rvel Comics', 183.74, 86.89, 'M', 1963, 'Blue', 'No Hair', 10.0, 'hig
h')
```

Procédures stockées

- Une procédure stockée est une série d'instructions SQL désignée par un nom.
- Les procédures stockées sont, comme leur nom l'indique, stockées de manière durable dans la BD dans laquelle elles sont enregistrées.
- Une fois qu'elle est créée, il est possible d'appeler une procédure par son nom. Les instructions de la procédure sont alors exécutées.

Géstion du délimiteur

Comme le corps de la procédure contiendra des instructions, il faut changer le délimiteur.

Donc un esemple de syntaxe complète pour une création de procédure est

```
DELIMITER |

CREATE PROCEDURE name()
BEGIN

  <instruction>;
  <instruction>;

END |

DELIMITER ;
```

Mais quand on utilise `%%sql` dans jupyter il ne faut pas modifier le delimiteur (en effet, ça déclencherait une erreur!).

In [83]:

```
%%sql

CREATE PROCEDURE find_millennial_heroes()
BEGIN
    select * from heroes where first_appearance between 1980 and 1990;
END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[83]:

```
[]
```

In [85]:

```
%sql CALL find_millennial_heroes()
```

```
 * mysql://superheroadmin:***@localhost/superheroes
62 rows affected.
```

In [87]:

```
import mysql.connector as mysql

try:
    mySQL_conn = mysql.connect(host='localhost',
                               database='superheroes',
                               user='superheroadmin',
                               password='Passw0rd.')
    cursor = mySQL_conn.cursor()
    cursor.callproc('find_millennial_heroes')
    # print out User details
    for result in cursor.stored_results():
        print(result.fetchall())
except mysql.connector.Error as error:
    print("Failed to execute stored procedure: {}".format(error))
finally:
    # closing database connection.
    if (mySQL_conn.is_connected()):
        cursor.close()
        mySQL_conn.close()
        print('connection is closed')
```

```
[(22, 'Lady Deathstrike', 'Yuriko Oyama', 'Osaka, Japan', 'Marvel Comi
cs', 175.85, 58.89, 'F', 1985, 'Brown', 'Black', 30.0, 'good'), (23,
'Yoda', 'Yoda', None, 'George Lucas', 66.29, 17.01, 'M', 1980, 'Brow
n', 'White', 55.0, 'high'), (32, 'Adam Strange', 'Adam Strange', 'Chic
ago, Illinois', 'DC Comics', 185.1, 88.92, 'M', 1986, 'Blue', 'Blond',
10.0, 'good'), (49, 'Warp', 'Emil LaSalle', None, 'DC Comics', 173.42,
67.42, 'M', 1981, 'Brown', 'Black', 10.0, 'moderate'), (50, 'Warpath',
'James Proudstar', 'Camp Verde, Arizona ', 'Marvel Comics', 218.05, 15
8.22, 'M', 1984, 'Brown', 'Black', 75.0, 'moderate'), (54, 'Walrus',
'Hubert Carpent', 'Brooklyn, New York', 'Marvel Comics', 183.7, 162.0
3, 'M', 1984, 'Blue', 'Black', 30.0, 'average'), (65, 'Vixen', 'Mari M
cCabe', 'Zambesi', 'DC Comics', 175.71, 63.57, 'F', 1981, 'Amber', 'Bl
ack', 40.0, 'average'), (75, 'Trigon', 'Trigon', None, 'DC Comics', No
ne, None, 'M', 1981, 'Yellow', 'Black', 100.0, 'high'), (90, 'The Come
dian', 'Edward Morgen Blake', None, 'DC Comics', 188.41, 101.45, 'M',
1986, 'Brown', 'Black', 15.0, 'good'), (99, 'T-800', 'Cyberdyne System
s Series 800 Terminator Model 101', None, 'Dark Horse Comics', None, 1
76.28, 'M', 1984, 'Red', None, 35.0, 'good'), (106, 'Superboy-Prime',
'Kal-El', 'Krypton (Earth-Prime) ', 'DC Comics', 180.55, 77.41, 'M', 1
985, 'Blue', 'Black / Bl', 100.0, 'high'), (110, 'Sunspot', 'Roberto D
aCosta', 'Rio de Janeiro, Brazil ', 'Marvel Comics', 173.58, 77.69,
'M', 1982, 'brown', 'black', 65.0, 'good'), (116, 'Starfire', "Korian
d'r", 'Tamaran', 'DC Comics', 193.77, 71.78, 'F', 1980, 'Green', 'Aubu
rn', 80.0, 'average'), (158, 'Shadowcat', 'Kitty Pryde', 'Deerfield, I
llinois', 'Marvel Comics', 168.63, 50.4, 'F', 1980, 'Hazel', 'Brown',
10.0, 'high'), (178, 'Rorschach', 'Walter Joseph Kovacs', None, 'DC Co
mics', 168.6, 63.06, 'M', 1986, 'Blue', 'Red', 10.0, 'good'), (180, 'R
ogue', 'Anna Marie', 'Caldecott County, Mississippi ', 'Marvel Comic
s', 173.48, 54.43, 'F', 1981, 'Green', 'Brown / Wh', 10.0, 'good'), (1
85, 'Robin III', 'Tim Drake', None, 'DC Comics', 165.94, 56.03, 'M', 1
989, 'Blue', 'Black', 15.0, 'high'), (196, 'Red Robin', 'Tim Drake',
'Gotham City', 'DC Comics', 165.39, 56.35, 'M', 1989, 'Blue', 'Black',
15.0, 'high'), (204, 'Raven', 'Rachel Roth', 'Azarath', 'DC Comics', 1
65.01, 50.41, 'F', 1980, 'Indigo', 'Black', 10.0, 'average'), (206, 'R
azor-Fist II', 'Douglas Scott', None, 'Marvel Comics', 191.11, 117.57,
'M', 1981, 'Blue', 'No Hair', None, None), (218, 'Psylocke', 'Elizabet
h Braddock', 'Braddock Manor, England', 'Marvel Comics', 180.88, 70.9
```

6, 'F', 1986, 'Blue', 'Purple', 35.0, 'good'), (219, 'Proto-Goblin', 'Nels Van Adder', None, 'Marvel Comics', None, None, 'M', 1990, 'Gree n', 'Blond', 40.0, 'good'), (220, 'Predator', 'Yautja', 'Yautja Prim e', 'Dark Horse Comics', 213.59, 234.19, 'M', 1987, None, None, 30.0, 'good'), (245, 'Oracle', 'Barbara Gordon', None, 'DC Comics', 178.12, 59.33, 'F', 1989, 'Blue', 'Red', 15.0, 'good'), (260, 'Nebula', 'Nebul a', None, 'Marvel Comics', 185.08, 83.19, 'F', 1985, 'Blue', 'No Hai r', 55.0, 'good'), (281, 'Monarch', 'Nathaniel Christopher Adam', Non e, 'DC Comics', 193.57, 90.72, 'M', 1987, 'Blue', 'White', 100.0, 'hig h'), (282, 'Mogo', 'Mogo', None, 'DC Comics', None, None, 'M', 1985, N one, None, None, None), (288, 'Mister Sinister', 'Nathaniel Essex', 'L ondon, England ', 'Marvel Comics', 196.89, 128.37, 'M', 1987, 'Red', 'Black', 50.0, 'high'), (303, 'Metallo', 'John Corben', None, 'DC Comi cs', 196.93, 90.94, 'M', 1987, 'Green', 'Brown', 55.0, 'good'), (319, 'Magus', 'Magus', 'Kvch (Technarchys home planet) ', 'Marvel Comics', 183.74, None, 'M', 1984, 'Black', None, 100.0, 'high'), (331, 'Longsho t', 'Longshot', None, 'Marvel Comics', 188.23, 36.71, 'M', 1985, 'Blu e', 'Blond', 10.0, 'average'), (341, 'Legion', 'David Haller', 'Israe l', 'Marvel Comics', 175.86, 59.83, 'M', 1985, 'Green / Bl', 'Black', 100.0, 'average'), (359, 'Kilowog', 'Kilowog', 'Bolovax Vik', 'DC Comi cs', 234.39, 324.1, 'M', 1986, 'Red', 'No Hair', 90.0, 'high'), (360, 'Killer Croc', 'Waylon Jones', None, 'DC Comics', 244.58, 356.55, 'M', 1983, 'Red', 'No Hair', 55.0, 'low'), (374, 'Jubilee', 'Jubilation Le e', 'Beverly Hills, California', 'Marvel Comics', 165.92, 52.34, 'F', 1989, 'Red', 'Black', 10.0, 'average'), (379, 'John Constantine', 'Joh n Constantine', None, 'DC Comics', 183.28, None, 'M', 1985, 'Blue', 'B lond', 10.0, 'good'), (388, 'Jean-Luc Picard', 'Jean-Luc Picard', 'La Barre, France, Earth', 'Star Trek', None, None, 'M', 1987, None, None, 10.0, 'good'), (405, 'Indiana Jones', 'Indiana Jones', 'Princeton, New Jersey', 'George Lucas', 183.78, 79.76, 'M', 1981, None, None, 10.0, 'good'), (410, 'Husk', 'Paige Elisabeth Guthrie', 'Cumberland, Kentuck y', 'Marvel Comics', 170.84, 58.07, 'F', 1984, 'Blue', 'Blond', 65.0, 'good'), (411, 'Huntress', 'Helena Rosa Bertinelli', None, 'DC Comic s', 180.44, 59.36, 'F', 1989, 'Blue', 'Black', 10.0, 'good'), (414, 'H ydro-Man', 'Morris Bench', 'Bronx, New York', 'Marvel Comics', 188.67, 119.09, 'M', 1981, 'Brown', 'Brown', 15.0, 'moderate'), (455, 'Gog', 'Gog', None, 'DC Comics', None, None, 'M', 1981, None, None, 35.0, 'av erage'), (485, 'Firestar', 'Angelica Jones', 'Place of birth unknown', 'Marvel Comics', 173.63, 56.41, 'F', 1985, 'Green', 'Red', 10.0, 'aver age'), (510, 'Dr Manhattan', 'Jonathan Osterman', None, 'DC Comics', N one, None, 'M', 1986, 'White', 'No Hair', 100.0, 'high'), (527, 'Death stroke', 'Slade Joseph Wilson', None, 'DC Comics', 193.87, 101.98, 'M', 1980, 'Blue', 'White', 30.0, 'good'), (534, 'Dazzler', 'Alison Bl aire', 'Gardendale, Long Island, New York ', 'Marvel Comics', 173.32, 52.67, 'F', 1980, 'Blue', 'Blond', 10.0, 'good'), (539, 'Darkman', 'Pe yton Westlake', None, 'Universal Studios', None, None, 'M', 1990, Non e, None, 15.0, 'good'), (547, 'Cyborg Superman', 'Henry Henshaw', Non e, 'DC Comics', None, None, 'M', 1990, 'Blue', 'Black', 95.0, 'good'), (553, 'Cyborg', 'Victor Stone', 'New York City, New York', 'DC Comic s', 198.12, 173.81, 'M', 1980, 'Brown', 'Black', 55.0, 'good'), (555, 'Cottonmouth', 'Burchell Clemens', None, 'Marvel Comics', 183.34, 99.1 7, 'M', 1985, 'Brown', 'Black', 10.0, 'average'), (567, 'Cheetah III', 'Barbara Minerva', None, 'DC Comics', 175.91, 54.75, 'F', 1987, 'Brow n', 'Brown', 100.0, 'high'), (570, 'Cheetah II', 'Deborah Domaine', No ne, 'DC Comics', 170.55, 55.24, 'F', 1980, 'Green', 'Brown', 20.0, 'mo derate'), (594, 'Cannonball', 'Samuel Zachery Guthrie', 'Cumberland Co unty, Kentucky ', 'Marvel Comics', 183.7, 81.7, 'M', 1982, 'Blue', 'Bl ond', 30.0, 'average'), (603, 'Brundlefly', 'Seth Brundle', None, Non e, 193.22, None, 'M', 1986, None, None, 35.0, 'good'), (620, 'Blizzard II', 'Donald Gill', 'Newark, Delaware', 'Marvel Comics', 175.15, 77.9 4, 'M', 1987, 'Brown', 'Brown', 10.0, 'moderate'), (631, 'Blackout',

```
'Blackout', None, 'Marvel Comics', 191.2, 104.09, 'M', 1990, 'Red', 'W
hite', 35.0, 'good'), (644, 'Black Abbott', 'Black Abbott', None, 'Mar
vel Comics', None, None, 'M', 1984, 'Red', 'Black', None, None), (649,
'Bird-Brain', 'Bird-Brain', None, 'Marvel Comics', None, None, None, 1
987, None, None, 10.0, 'moderate'), (667, 'Battlestar', 'Lemar Hoskins
', 'Chicago, Illinois ', 'Marvel Comics', 198.54, 133.04, 'M', 1986,
'Brown', 'Black', 55.0, 'average'), (691, 'Arclight', 'Philippa Sonta
g', 'Vietnam', 'Marvel Comics', 173.43, 57.25, 'F', 1986, 'Violet', 'P
urple', 65.0, 'moderate'), (697, 'Ariel', 'Ariel', None, 'Marvel Comic
s', 165.35, 59.17, 'F', 1987, 'Purple', 'Pink', 10.0, 'average'), (70
7, 'Ammo', 'Ammo', None, 'Marvel Comics', 188.93, 101.09, 'M', 1988,
'Brown', 'Black', None, None)]
connection is closed
```

In [91]:

```
%%sql

CREATE PROCEDURE get_age_of(IN p_name VARCHAR(100), OUT age INT)
BEGIN
    SELECT YEAR(NOW()) - first_appearance INTO age FROM heroes where p_name=name;
END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[91]:

```
[]
```

In [92]:

```
%%sql

CALL get_age_of('Wonder Woman',@age);
SELECT @age;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
1 rows affected.
```

Out[92]:

@age
78

In [98]:

```
try:
    mySQL_conn = mysql.connect(host='localhost',
                                   database='superheroes',
                                   user='superheroadmin',
                                   password='Passw0rd.')
    cursor = mySQL_conn.cursor()

    args = ['Wonder Woman', 0]
    result_args = cursor.callproc('get_age_of', args)

    cursor.callproc('find_millennial_heroes')
    # print out details
    print(result_args[1])
except mysql.connector.Error as error:
    print("Failed to execute stored procedure: {}".format(error))
finally:
    # closing database connection.
    if (mySQL_conn.is_connected()):
        cursor.close()
        mySQL_conn.close()
        print('connection is closed')
```

```
78
connection is closed
```

Suppression d'une procédure

In [99]:

```
%%sql

DROP PROCEDURE get_age_of;
DROP PROCEDURE find_millennial_heroes;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
0 rows affected.
```

Out[99]:

```
[]
```

Triggers

In [8]:

```
%%sql

SHOW TRIGGERS;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[8]:

Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	col

- Un *trigger* (ou *déclencheur*) est un objet stocké dans la base de données, à la manière d'une table ou d'une procédure stockée.
- Un trigger est lié à une table, donc en cas de suppression d'une table, les triggers liés à celle-ci sont supprimés également.
- Un trigger définit une ou plusieurs instructions dont l'exécution est déclenchée par une insertion, une modification ou une suppression de données dans la table à laquelle le trigger est lié.
- Les instructions du trigger peuvent être exécutées avant la requête ayant déclenché celui-ci, ou après. Ce comportement est à définir à la création du trigger.
- Une table ne peut posséder qu'un seul trigger par combinaison événement/moment (BEFORE UPDATE, AFTER DELETE…).

À selon de la version de mySQL, il est nécéssaire de allouer une permission differente à un usager afin qu'il puisse gérer des triggers. Certaines versions récentes de mySQL ont une permission `TRIGGER` qui est spécifique pour les triggers:

```
GRANT TRIGGER ON superheroes.* TO superheroadmin@localhost;
```

Les versions moins récentes font référence à la permission SUPER:

```
GRANT SUPER ON *.* TO superheroadmin@localhost;
```

La requête pour créer un trigger a la syntaxe suivante:

```
CREATE TRIGGER nom_trigger moment_trigger evenement_trigger
ON nom_table FOR EACH ROW
corps_trigger;
```

où:

- `nom_table` est le nom de la table associée au trigger,
- `evenement_trigger` indique la requête associée au trigger (`INSERT` , `UPDATE` ou `DELETE`),
- `moment_trigger` spécifie si le trigger doit être déclenché avant (`BEFORE`) ou après (`AFTER`) la requête,
- `corps_trigger` contient les instructions du trigger.

Usage des triggers:

- comme alternative aux contraintes
- pour vérifier l'integrité des données
- pour assurer l'integrité des données
- pour effectuer un archivage en place des suppressions

Convention pour le nom d'un trigger: `<moment>_<evenement>_<table>`

Par exemple: `before_delete_heroes` , ou `after_insert_heroes` .

Règles:

- un seul trigger par table+evenement+moment, donc un maximum de six triggers par table
- (en plus, il y des limitations spécifiques pour les triggers)

`OLD` et `NEW` répresentent les valeurs de la ligne de table traitee:

- `OLD` : avant la modification (et ses valeurs ne sont pas modifiables)
- `NEW` : après la modification (et ses valeurs sont modifiables)

	OLD	NEW
INSERT	NON	OUI
UPDATE	OUI	OUI
DELETE	OUI	NON

Géstion du délimiteur

Comme le corps du trigger contiendra des instructions, il faut changer le délimiteur exactement comme on a vu pour les procédures stockées.

Donc un esemple de syntaxe complète pour une création de trigger est

```
DELIMITER |

CREATE TRIGGER before_insert_hero BEFORE INSERT
ON hero FOR EACH ROW
BEGIN

  <instruction>;
  <instruction>;

END |

DELIMITER ;
```

Ici aussi, quand on utilise `%%sql` dans jupyter il ne faut pas modifier le delimiteur (en effet, ça déclencherait une erreur!).

Exemple: alternative aux contraintes

Vérifions que la force d'un héro soit toujours un nombre entre 0 et 100.

Problème: quoi faire s'il y a une valeur invalide?

- déclencher une erreur
- utiliser `NULL`

Trigger qui déclenche une erreur

In [32]:

```
%%sql

CREATE TRIGGER before_heroes_insert
         BEFORE INSERT ON heroes
         FOR EACH ROW
     BEGIN
      IF (NEW.strength < 0 or NEW.strength > 100) THEN
          SIGNAL SQLSTATE '45000'
          SET MESSAGE_TEXT = 'invalid strength';
      END IF;
     END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[32]:

```
[]
```

In [33]:

```
%sql SHOW TRIGGERS;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[33]:

Trigger	Event	Table	Statement	Timing	Created	
before_heroes_insert	INSERT	heroes	BEGIN IF (NEW.strength < 0 or NEW.strength > 100) THEN SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'invalid strength'; END IF; END	BEFORE	2019-03-20 10:20:01.120000	ONLY_FULL_G

In [34]:

```
%sql INSERT INTO heroes (id, strength) VALUES (9999, 340)
```

```
   1058             compiled_sql,
   1059             distilled_params,
-> 1060             compiled_sql, distilled_params
   1061         )
   1062         if self._has_events or self.engine._has_events:

~/anaconda3/lib/python3.6/site-packages/sqlalchemy/engine/base.py
 in _execute_context(self, dialect, constructor, statement, parame
ters, *args)
   1198                 parameters,
   1199                 cursor,
-> 1200                 context)
   1201
   1202         if self._has_events or self.engine._has_events:

~/anaconda3/lib/python3.6/site-packages/sqlalchemy/engine/base.py
 in _handle_dbapi_exception(self, e, statement, parameters, curso
r, context)
   1411                 util.raise_from_cause(
   1412                     sqlalchemy_exception,
```

Suppression d'un trigger

In [35]:

```
%sql drop trigger before_heroes_insert;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[35]:

```
[]
```

Trigger qui utilise `NULL`

In [36]:

```
%%sql

CREATE TRIGGER before_heroes_insert
         BEFORE INSERT ON heroes
         FOR EACH ROW
     BEGIN
      IF (NEW.strength < 0 or NEW.strength > 100) THEN
          SET NEW.strength = NULL;
      END IF;
     END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[36]:

```
[]
```

In [38]:

```
%sql INSERT INTO heroes (id, strength) VALUES (9999, 340)
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[38]:

```
[]
```

In [39]:

```
%sql SELECT * FROM heroes WHERE ID=9999;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[39]:

id	name	identity	birth_place	publisher	height	weight	gender	first_appearance	eye_colo
9999	None	None	None	None	None	None	None	None	Nor

In [40]:

```
%sql DELETE FROM heroes WHERE ID=9999;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[40]:

```
[]
```

Assurer l'integrité des données

In [41]:

```
%%sql

CREATE TABLE leagues (
    id INT,
    name VARCHAR(100)
);

INSERT INTO leagues VALUES (1, 'Justice league');
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
0 rows affected.
1 rows affected.
```

Out[41]:

```
[]
```

In [55]:

```
%%sql

CREATE TABLE heroes_leagues (
    hero_id INT,
    league_id INT
)
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[55]:

```
[]
```

In [54]:

```
%sql SELECT id, name FROM heroes WHERE name IN ('Superman', 'Batman', 'Wonder Woman
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

Out[54]:

id	name
1	Wonder Woman
113	Superman
316	Martian Manhunter
490	Flash
666	Batman

In [56]:

```
%%sql

INSERT INTO heroes_leagues VALUES
    (1, 1),
    (113, 1),
    (316, 1),
    (490, 1),
    (666, 1);
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

Out[56]:

```
[]
```

In [65]:

```
%%sql

INSERT INTO heroes_leagues (hero_id, league_id)
    (SELECT id, 1 FROM heroes WHERE name IN ('Superman', 'Wonder Woman', 'Martian M
;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

Out[65]:

```
[]
```

In [42]:

```
%%sql

CREATE TRIGGER after_update_heroes
    AFTER UPDATE ON heroes
    FOR EACH ROW
    BEGIN
      UPDATE heroes_leagues SET hero_id = NEW.id WHERE hero_id = OLD.id;
    END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[42]:

```
[]
```

In [66]:

```
%sql SELECT * from heroes_leagues;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

Out[66]:

hero_id	league_id
1	1
113	1
316	1
490	1
666	1

In [68]:

```
%sql UPDATE heroes SET id = 9999 WHERE name = 'Wonder Woman';
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[68]:

```
[]
```

In [69]:

```
%sql SELECT * from heroes_leagues;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
5 rows affected.
```

Out[69]:

hero_id	league_id
9999	1
113	1
316	1
490	1
666	1

Archivage en place de suppression

In [ ]:

```
%%sql

CREATE TABLE archived_heroes(
    id INTEGER UNIQUE NOT NULL,
    name VARCHAR(100),
    identity VARCHAR(100),
    birth_place VARCHAR(100),
    publisher VARCHAR(100),
    height FLOAT,
    weight FLOAT,
    gender CHAR(1),
    first_appearance INTEGER,
    eye_color VARCHAR(10),
    hair_color VARCHAR(10),
    strength FLOAT,
    intelligence VARCHAR(20),
    PRIMARY KEY(id)
);
```

In [79]:

```
%%sql

CREATE TRIGGER before_delete_heroes
    BEFORE DELETE ON heroes
    FOR EACH ROW
    BEGIN
      INSERT INTO archived_heroes SELECT * FROM heroes WHERE id = OLD.id;
    END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[79]:

```
[]
```

In [80]:

```
%%sql

DELETE FROM heroes WHERE name = 'Apocalypse';
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[80]:

```
[]
```

In [82]:

```
%sql SELECT * FROM archived_heroes;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[82]:

id	name	identity	birth_place	publisher	height	weight	gender	first_appearance	eye
700	Apocalypse	En Sabah Nur	Akkaba, Egypt	Marvel Comics	213.77	135.62	M	1986	

In [101]:

```
%%sql

CREATE PROCEDURE undo_delete(IN p_id INT)
BEGIN
    INSERT INTO heroes SELECT * FROM archived_heroes WHERE id = p_id;
    DELETE FROM archived_heroes WHERE id = p_id;
END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[101]:

```
[]
```

In [102]:

```
%sql CALL undo_delete(700);
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[102]:

```
[]
```

In [103]:

```
%sql SELECT * FROM archived_heroes;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[103]:

id	name	identity	birth_place	publisher	height	weight	gender	first_appearance	eye_color

In [104]:

```
%sql SELECT * FROM heroes WHERE id=700;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[104]:

id	name	identity	birth_place	publisher	height	weight	gender	first_appearance	eye
700	Apocalypse	En Sabah Nur	Akkaba, Egypt	Marvel Comics	213.77	135.62	M	1986	

Historisation

In [74]:

```
%%sql

CREATE TABLE heroes_history (
    date DATETIME,
    user VARCHAR(100),
    type VARCHAR(10)
);
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[74]:

```
[]
```

In [75]:

```
%%sql

CREATE TRIGGER after_heroes_insert
    AFTER INSERT ON heroes
    FOR EACH ROW
    BEGIN
      INSERT INTO heroes_history VALUES (NOW(), CURRENT_USER(), 'INSERT');
    END;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
0 rows affected.
```

Out[75]:

```
[]
```

In [76]:

```
%sql INSERT INTO heroes (id, name) VALUES (9999, 'new hero');
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[76]:

```
[]
```

In [100]:

```
%sql SELECT * FROM heroes_history;
```

```
 * mysql://superheroadmin:***@localhost/superheroes
1 rows affected.
```

Out[100]:

date	user	type
2019-03-20 11:31:51	superheroadmin@localhost	INSERT