BAUS Philippe

INHA DATABASE LAB WEEK 9

```
from sqlalchemy import create_engine
my_conn = create_engine("sqlite:////content/my_db.db/")
```

```
r_set = my_conn.execute('''select name from sqlite_master where type = 'table' ''')
for row in r_set:
  print(row)
```

```
    ('category',)
    ('subcategory',)
    ('student',)
```

```
my_conn.execute('''DROP TABLE student''')
```

```
    <sqlalchemy.engine.cursor.LegacyCursorResult at 0x7f932a622f10>
```

```
r_set = my_conn.execute('''select name from sqlite_master where type = 'table' ''')
for row in r_set:
  print(row)
```

```
    ('category',)
    ('subcategory',)
```

Table student hs been droped

```
with open ("student_tables.sql", "r") as file:
  query = file.read()
  my_conn.execute(query)
```

```
r_set = my_conn.execute('''select name from sqlite_master where type = 'table' ''')
for row in r_set:
  print(row)
```

```
    ('category',)
    ('subcategory',)
    ('student',)
```

```
with open ("student_insert.sql", "r") as file:
  query = file.read()
  my_conn.execute(query)
```

```
r_set = my_conn.execute('''select * from student''')
```

✓ 0 s terminée à 18:36

```
  print(row)
```

```
    (1, 'John Deo', 'Four', 75, 'female')
    (2, 'Max Ruin', 'Three', 85, 'male')
    (3, 'Arnold', 'Three', 55, 'male')
    (4, 'Krish Star', 'Four', 60, 'female')
    (5, 'John Mike', 'Four', 60, 'female')
    (6, 'Alex John', 'Four', 55, 'male')
    (7, 'My John Rob', 'Five', 78, 'male')
    (8, 'Asruid', 'Five', 85, 'male')
    (9, 'Tes Qry', 'Six', 78, 'male')
    (10, 'Big John', 'Four', 55, 'female')
    (11, 'Ronald', 'Six', 89, 'female')
    (27, 'Big Nose', 'Three', 81, 'female')
    (28, 'Rojj Base', 'Seven', 86, 'female')
    (29, 'Tess Played', 'Seven', 55, 'male')
    (30, 'Reppy Red', 'Six', 79, 'female')
    (31, 'Marry Toeey', 'Four', 88, 'male')
    (32, 'Binn Rott', 'Seven', 90, 'female')
    (33, 'Kenn Rein', 'Six', 96, 'female')
    (34, 'Gain Toe', 'Seven', 69, 'male')
    (35, 'Rows Noump', 'Six', 88, 'female')
```

Students have been inserted

```
from sqlalchemy.exc import SQLAlchemyError
q = "UPDATE student SET mark = mark+1"
try:
  r_set = my_conn.execute(q)
except:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record updated : ", r_set.rowcount)
```

```
    Nb record updated :  20
```

```
r_set = my_conn.execute('''select * from student''')
for row in r_set:
  print(row)
```

```
    (1, 'John Deo', 'Four', 76, 'female')
    (2, 'Max Ruin', 'Three', 86, 'male')
    (3, 'Arnold', 'Three', 56, 'male')
    (4, 'Krish Star', 'Four', 61, 'female')
    (5, 'John Mike', 'Four', 61, 'female')
    (6, 'Alex John', 'Four', 56, 'male')
    (7, 'My John Rob', 'Five', 79, 'male')
    (8, 'Asruid', 'Five', 86, 'male')
    (9, 'Tes Qry', 'Six', 79, 'male')
    (10, 'Big John', 'Four', 56, 'female')
    (11, 'Ronald', 'Six', 90, 'female')
    (27, 'Big Nose', 'Three', 82, 'female')
```

```
(27, 'Big Nose', 'Three', 82, 'female')
(28, 'Rojj Base', 'Seven', 87, 'female')
(29, 'Tess Played', 'Seven', 56, 'male')
(30, 'Reppy Red', 'Six', 80, 'female')
(31, 'Marry Toeey', 'Four', 89, 'male')
(32, 'Binn Rott', 'Seven', 91, 'female')
(33, 'Kenn Rein', 'Six', 97, 'female')
(34, 'Gain Toe', 'Seven', 70, 'male')
(35, 'Rows Noump', 'Six', 89, 'female')
```

We can see that grades have been updated

```
q = "UPDATE student SET mark = mark+1 WHERE id=7"
try:
  r_set = my_conn.execute(q)
except:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record updated : ", r_set.rowcount)
```

```
Nb record updated :  1
```

```
r_set = my_conn.execute('''select * from student''')
for row in r_set:
  print(row)
```

```
(1, 'John Deo', 'Four', 76, 'female')
(2, 'Max Ruin', 'Three', 86, 'male')
(3, 'Arnold', 'Three', 56, 'male')
(4, 'Krish Star', 'Four', 61, 'female')
(5, 'John Mike', 'Four', 61, 'female')
(6, 'Alex John', 'Four', 56, 'male')
(7, 'My John Rob', 'Five', 80, 'male')
(8, 'Asruid', 'Five', 86, 'male')
(9, 'Tes Qry', 'Six', 79, 'male')
(10, 'Big John', 'Four', 56, 'female')
(11, 'Ronald', 'Six', 90, 'female')
(27, 'Big Nose', 'Three', 82, 'female')
(28, 'Rojj Base', 'Seven', 87, 'female')
(29, 'Tess Played', 'Seven', 56, 'male')
(30, 'Reppy Red', 'Six', 80, 'female')
(31, 'Marry Toeey', 'Four', 89, 'male')
(32, 'Binn Rott', 'Seven', 91, 'female')
(33, 'Kenn Rein', 'Six', 97, 'female')
(34, 'Gain Toe', 'Seven', 70, 'male')
(35, 'Rows Noump', 'Six', 89, 'female')
```

We can see that student 7 grade has been updated 79 => 80

```
q = "UPDATE student SET mark = mark+1 WHERE class = 'Four' "
try:
  r_set = my_conn.execute(q)
except:
```

```
except:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record updated : ", r_set.rowcount)
```

```
Nb record updated :  6
```

```
r_set = my_conn.execute('''select * from student''')
for row in r_set:
  print(row)
```

```
(1, 'John Deo', 'Four', 77, 'female')
(2, 'Max Ruin', 'Three', 86, 'male')
(3, 'Arnold', 'Three', 56, 'male')
(4, 'Krish Star', 'Four', 62, 'female')
(5, 'John Mike', 'Four', 62, 'female')
(6, 'Alex John', 'Four', 57, 'male')
(7, 'My John Rob', 'Five', 80, 'male')
(8, 'Asruid', 'Five', 86, 'male')
(9, 'Tes Qry', 'Six', 79, 'male')
(10, 'Big John', 'Four', 57, 'female')
(11, 'Ronald', 'Six', 90, 'female')
(27, 'Big Nose', 'Three', 82, 'female')
(28, 'Rojj Base', 'Seven', 87, 'female')
(29, 'Tess Played', 'Seven', 56, 'male')
(30, 'Reppy Red', 'Six', 80, 'female')
(31, 'Marry Toeey', 'Four', 90, 'male')
(32, 'Binn Rott', 'Seven', 91, 'female')
(33, 'Kenn Rein', 'Six', 97, 'female')
(34, 'Gain Toe', 'Seven', 70, 'male')
(35, 'Rows Noump', 'Six', 89, 'female')
```

We updated the grade of every student in class Four

```
q = "UPDATE student SET mark = mark+1 WHERE class = 'Three' "
try:
  r_set = my_conn.execute(q)
except:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record updated : ", r_set.rowcount)
```

```
Nb record updated :  3
```

```
r_set = my_conn.execute('''select * from student''')
for row in r_set:
  print(row)
```

```
(1, 'John Deo', 'Four', 77, 'female')
(2, 'Max Ruin', 'Three', 87, 'male')
(3, 'Arnold', 'Three', 57, 'male')
(4, 'Krish Star', 'Four', 62, 'female')
```

```
(5, 'John Mike', 'Four', 62, 'female')
(6, 'Alex John', 'Four', 57, 'male')
(7, 'My John Rob', 'Five', 80, 'male')
(8, 'Asruid', 'Five', 86, 'male')
(9, 'Tes Qry', 'Six', 79, 'male')
(10, 'Big John', 'Four', 57, 'female')
(11, 'Ronald', 'Six', 90, 'female')
(27, 'Big Nose', 'Three', 83, 'female')
(28, 'Rojj Base', 'Seven', 87, 'female')
(29, 'Tess Played', 'Seven', 56, 'male')
(30, 'Reppy Red', 'Six', 80, 'female')
(31, 'Marry Toeey', 'Four', 90, 'male')
(32, 'Binn Rott', 'Seven', 91, 'female')
(33, 'Kenn Rein', 'Six', 97, 'female')
(34, 'Gain Toe', 'Seven', 70, 'male')
(35, 'Rows Noump', 'Six', 89, 'female')
```

```
id = input("ENTER student id : ")
my_data=(id,)
q = "UPDATE student SET mark = mark+5 WHERE id = ?"
try:
  r_set = my_conn.execute(q,my_data)
except:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record updated : ", r_set.rowcount)
```

```
ENTER student id : 6
Nb record updated :  1
```

Here we can see that Python input function is correctly working, and that we can ue the value into a parametric SQL request

```
my_data = ('Six', 'Seven')
q= "UPDATE student SET class=? WHERE class=?"
try:
  r_set = my_conn.execute(q,my_data)
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record updated : ", r_set.rowcount)
```

```
Nb record updated :  4
```

```
r_set = my_conn.execute('''select * from student''')
for row in r_set:
  print(row)
```

```
(1, 'John Deo', 'Four', 77, 'female')
(2, 'Max Ruin', 'Three', 87, 'male')
(3, 'Arnold', 'Three', 57, 'male')
```

```
(3, 'Arnold', 'Three', 97, 'male')
(4, 'Krish Star', 'Four', 62, 'female')
(5, 'John Mike', 'Four', 62, 'female')
(6, 'Alex John', 'Four', 62, 'male')
(7, 'My John Rob', 'Five', 80, 'male')
(8, 'Asruid', 'Five', 86, 'male')
(9, 'Tes Qry', 'Six', 79, 'male')
(10, 'Big John', 'Four', 57, 'female')
(11, 'Ronald', 'Six', 90, 'female')
(27, 'Big Nose', 'Three', 83, 'female')
(28, 'Rojj Base', 'Six', 87, 'female')
(29, 'Tess Played', 'Six', 56, 'male')
(30, 'Reppy Red', 'Six', 80, 'female')
(31, 'Marry Toeey', 'Four', 90, 'male')
(32, 'Binn Rott', 'Six', 91, 'female')
(33, 'Kenn Rein', 'Six', 97, 'female')
(34, 'Gain Toe', 'Six', 70, 'male')
(35, 'Rows Noump', 'Six', 89, 'female')
```

```
q = "SELECT DISTINCT(class) FROM student"
try:
  r_set = my_conn.execute(q)
  data=r_set.fetchall()
  for row in data:
    print(row[0])
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record displayed : ", len(data))
```

```
Four
Three
Five
Six
Nb record displayed :  4
```

```
q = "SELECT DISTINCT(class) FROM student WHERE mark >= 80"
try:
  r_set = my_conn.execute(q)
  data=r_set.fetchall()
  for row in data:
    print(row[0])
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record displayed : ", len(data))
```

```
Three
Five
Six
Four
Nb record displayed :  4
```

```
q = "SELECT DISTINCT(class) FROM student ORDER BY class"
try:
  r_set = my_conn.execute(q)
  data=r_set.fetchall()
  for row in data:
    print(row[0])
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record displayed : ", len(data))
```

```
Five
Four
Six
Three
Nb record displayed :  4
```

Looks ordered by Alphabetical order

```
q = "SELECT DISTINCT class,sex FROM student ORDER BY class"
try:
  r_set = my_conn.execute(q)
  data=r_set.fetchall()
  for row in data:
    print(row[0], row[1])
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("Nb record displayed : ", len(data))
```

```
Five male
Four female
Four male
Six male
Six female
Three male
Three female
Nb record displayed :  7
```

```
q = "DELETE From Student WHERE id = 5"
try:
  r_set = my_conn.execute(q)
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("No record deleted: ", r_set.rowcount)
```

```
No record deleted:  1
```

```
q = "DELETE From Student WHERE class = 'Four' "
```

```
q = "DELETE FROM student WHERE class = 'four'
try:
  r_set = my_conn.execute(q)
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("No record deleted: ", r_set.rowcount)
```

```
    No record deleted:  5
```

```
my_data=("Five",)
q="DELETE FROM student WHERE class=?"
try:
   r_set = my_conn.execute(q, my_data)
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("No record deleted: ", r_set.rowcount)
```

```
    No record deleted:  2
```

```
my_data=("Six","Seven")
q="DELETE FROM student WHERE class=? OR class=?"
try:
   r_set = my_conn.execute(q, my_data)
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("No record deleted: ", r_set.rowcount)
```

```
    No record deleted:  9
```

```
q="DELETE FROM student"
try:
   r_set = my_conn.execute(q)
except SQLAlchemyError as e:
  error = str(e.__dict__['orig'])
  print(error)
else:
  print ("No record deleted: ", r_set.rowcount)
```

```
    No record deleted:  3
```

There were still rows because I had student in my class 'Three'

Produits payants Colab - Résilier les contrats ici