

NOM DIALLO  
Prénom Alpha  
Promo 2018  
Date 13/04/17



175



DIALLO Alpha Oumar  
M1 - 2016

13,5

## MATIÈRE Advanced Databases T12-IFM

2 - JDBC : 5  
List < Double > listMarks(Integer sid) throws SQLException

We Initialize the connection first:

```
Connection mycon = DriverManager.getConnection(url,  
name, pw) ✓
```

List < Double > listMarks(Integer sid) throws SQLException

called

```
Statement s = mycon.createStatement();  
String q = "SELECT mark FROM courses WHERE  
sid = " + sid; enrolled ✓  
ResultSet r = s.executeQuery(q);  
List<Double> l = new ArrayList<>();  
while (r.next()) {  
    l.add(r.getDouble());  
}  
return l;
```

⚠ If sid = 0 the query change a little  
please look the second copy to see the case in  
which the id is null. I forgot

## 2.2 - Using prepared statements :

We suppose that the connection is already established:

```
PreparedStatement p = mycon.prepareStatement("SELECT  
mark FROM courses WHERE cid = ?");  
// enrolled
```

```
p.setInt(1, sid);
```

```
ResultSet r = p.executeQuery(); List<Double> l = new ArrayList();
```

```
while (r.next()) {
```

```
l.add(r.getDouble()); }
```

```
return l; }
```

2.3 The prepared statement must be used because it helps to save CPU and because it already has an statement ready to execute.

## 3 - stored Procedures

2 - sets to 0 all marks of a course, given its cid:

```
Declare Procedure sets to zero (cid)  
Declare new  
Declare
```

```
Declare Proc
```

```
CREATE PROCEDURE SetstoZero (IN cid)
  DECLARE marknew / DECLARE sidnew
  DECLARE Not Found Exception
  SQL STATE '02100'
```

```
Cursor c For
  ( SELECT marksid FROM enrolled WHERE cid=cid)
```

Begin

Open c

IteruLoop: Loop

Fetch c INTO marknew sidnew

IF Not Found

Return Not Found *leave*

Update Enrollment mark = '0' <sup>sid = sidnew</sup> WHERE cid=cid

2 - SQL function that returns the average mark of student given his/her sid:

```
CREATE FUNCTION AveMark (IN sid)
  SET Ave = AVERAGE mark FROM enrolled
  WHERE sid = sid
  RETURN Ave
```

# 1 - Triggers: 4,5

```
1. CREATE TRIGGER mark_zero_to_twenty (
  BEFORE
  BEFORE INSERT UPDATE mark on enrolled
  FOR EACH ROW
  REFERENCING NEW ROW AS NEW ROW
  BEFORE AFTER INSERT UPDATE mark on enrolled
  REFERENCING NEW ROW AS NEW ROW
  NEW_ROW.mark > upper (NEW_ROW.mark)
```

```
2. CREATE TRIGGER upper_case (
  BEFORE / AFTER INSERT UPDATE ✓ ename on enro
  REFERENCING NEW ROW AS NEW ROW FER
  NEW_ROW.ename > upper (NEW_ROW.ename)
```

```
1. CREATE TRIGGER mark_zero_to_twenty (
  BEFORE INSERT UPDATE ✓ mark on enrollment
  REFERENCING NEW ROW AS NEW ROW ✓ FER
  IF (NEW_ROW.mark < 0)
    NEW_ROW.mark = 0
  ELSE IF (NEW_ROW.mark > 20)
    NEW_ROW.mark = 20
  ELSE
    NEW_ROW.mark = NEW_ROW.mark
```

NOM DIALLO  
Prénom Alpha  
Promo 2018  
Date 13/04/17

## MATIÈRE DE ADVANCED DATABASE

TRIGGERS: suite

CREATE TRIGGER studentHist ()

AFTER INSERT student

REFERENCING NEW ROW AS NEW ROW

FOR

INSERT INTO studenthistory VALUES (NEWROW.sid,  
currentdate, NEWROW.address, NEWROW.year)

2. JDBC

IN the case in which sid is null: ✓

2.1 If sid is null:

If (sid = 0) {

statement s = mycon.createStatement();

string q = "SELECT mark FROM enrolled";

The the procedure is the same than if there  
were a student id different to 0.

2.2 For prepared statement:

If (sid = 0) {

PreparedStatement p = mycon.prepareStatement("SELECT  
mark FROM enrolled");

✓ ok

Then the end is the same.





