JDBC API

provides Java developers with a standard API that is used to access databases, regardless of the driver and database product.

Afnan Mousa 15

Enas Morsy 20

Sara Mohammed 31

Shaima kamal 34

Overview

Java Database Connectivity (JDBC) provides Java developers with a standard API that is used to access

databases, regardless of the driver and database product. JDBC presents a uniform interface to databases

- change vendors and your applications only need to change their driver.

Goals

1. Application program, controls the organization, storage, management,and retrieval of data in a database.
2. Any other method in the interfaces not mentioned above will have an empty implementation that throws ..java.lang.UnsupportedOperationException.
3. Complete log of the operations done on the database through the DBMS should be shown. Operations include initiating DB connections, query execution, errors and warnings, connections closing using import java.util.logging.*.
4. Operations timestamp by `java.util.concurrent.TimeUnit` .
5. Implement the JDBC main interfaces to access the tables' data.
6. o java.sql.Driver o java.sql.ResultSetMetaData
7. o java.sql.Connection
8. o java.sql.Statement
9. o java.sql.Resultset

User Guide

1.First enter the path if you want to open a database that already exists,then you choose which interface you want to deal with or if you want to exist.

```
enter the path:C:\Users\Ena\Desktop\tt
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
```

2.if you choose the Statement one,its options will appear

```
enter the path:C:\Users\Ena\Desktop\tt
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
1
1.add to Batch
2.clear the Batch
3.execute Batch
4.create/drop DB or Table
5.Select the table
6.know the query timeout
7.close
```

3.if you choose the Result one,its options will appear.

```
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
2
enter the query
Select * from tab1
1.Set the cursor after last row
2.Set the cursor before first row
3.Set the cursor at the last row
4.Set the cursor at the first row
5.Set the position of the cursor
6.Move to the next row
7.Move to the previous row
8.Get the index of a column
9.Get an integer value by column's index
10.Get an integer value by column's name
11.Get a String value by column's index
12.Get a String value by column's name
13.Get a Object by column's index
14.Check if the cursor is after last row
15.Check if the cursor is before first row
16.Check if the result set is closed
17.Check if the cursor is at the first row
18.Check if the cursor is at the last row
19.Close
```

4.if you choose the MetaResult one,its operations will appear.

```
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
3
enter the query
select * from tab1
1.getColumnCount
2.getColumnName
3.getColumnType
4.getTableName
5.getColumnLabel
```

Design Decisions

I. Design Patterns

- Façade.
- Factory.
- Strategy.
- Singleton.
- pool.

II. JDK logger

Every step the user takes is recorded in a file called MY LOGGING

In addition to warnings resulting from user errors.

Bonus

In Time for java.sql.Statement :

- getQueryTimeout()
- setQueryTimeout(int seconds)

UML Diagram

In The Attached PDF which file has name “UML”

Sample Run

```
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
1
1.add to Batch
2.clear the Batch
3.execute Batch
4.create/drop DB or Table
5.Select the table
6.know the query timeout
7.close
1
enter the query
insert into tab1 values(88)
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
1
1.add to Batch
2.clear the Batch
3.execute Batch
4.create/drop DB or Table
5.Select the table
6.know the query timeout
7.close
3
1
```

```
enter the path:
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
1
1.add to Batch
2.clear the Batch
3.execute Batch
4.create/drop DB or Table
5.Select the table
6.know the query timeout
7.close
4
enter the query
create database h
succeeded
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
1
1.add to Batch
2.clear the Batch
3.execute Batch
4.create/drop DB or Table
5.Select the table
6.know the query timeout
7.close
4
enter the query
create table tab1 (c1 int)
succeeded
```

```
1.Statement
2.RESULTSEt
3.ResultSetMetaData
4.exit
3
enter the query
Select * from tab1
1.getColumnCount
2.getColumnName
3.getColumnType
4.getTableName
5.getColumnLabel
1
the number of columns: 1
```