Table of Contents

Table or pipelined functions are a cool thing. Use them to dynamically create records directly – without a physical table – in the database with all available database features. Not all database systems support table functions but Oracle and PostgreSql do. Here's an example for a table function with Oracle.

First, we need the table function:

```
-- the type
DROP TYPE type_testtable;
DROP TYPE type_testrow;

CREATE TYPE type_testrow AS OBJECT (
  id          NUMBER,
  description VARCHAR2(50)
);
/

CREATE TYPE test_testtable IS TABLE OF type_testrow;
/

-- the table function
CREATE OR REPLACE FUNCTION get_tablefunction(pRowCount IN NUMBER) RETURN
type_testtable AS
  result type_testtable := type_testtable();
BEGIN
  FOR i IN 1 .. pRowCount LOOP
    result.extend;
    result(result.LAST) := type_testrow(i, 'Description for record ' || i);
  END LOOP;
  RETURN result;
END;
/
```

To use the table function with a **DBStorage**, try following:

```
DBStorage dbs = new DBStorage();
dbs.setDBAccess(dba);
dbs.setFromClause("TABLE(get_tablefunction(:COUNT))");
dbs.open();

DirectObjectConnection con = new DirectObjectConnection();
con.put("dbs", dbs);

MasterConnection macon = new MasterConnection(con);
macon.open();

RemoteDataSource rds = new RemoteDataSource(macon);
rds.open();

RemoteDataBook rdb = new RemoteDataBook();
rdb.setDataSource(rds);
rdb.setName("dbs");
rdb.setFilter(new Equals("COUNT", BigDecimal.valueOf(20)));
rdb.open();

rdb.fetchAll();
```

From:
http://doc.sibvisions.com/ - **Documentation**

Permanent link:
http://doc.sibvisions.com/jvx/server/db/table_functions

Last update: **2020/06/29 13:00**