

CSE315 –Spring 2007

Experiment 1 – Chaining, Savepoints and Nesting

Due date: Feb 21, 2007

Goal:

Chaining, savepoints and nested transactions are all mechanisms that can be used to divide a task into parts that can be committed or aborted separately. The purpose of this experiment is to investigate how these mechanisms work in Sybase.

Description:

The task is to implement a transaction that raises the salary of faculty members in three different ways: using chaining, savepoints and nesting. The purpose of the transaction is to give all faculty members the same percentage raise, but the total salary after the raise has been given must not exceed \$100000. The transaction iterates a basic subtask that raises all faculty salaries by one percent (from their current value, hence compound interest) and then checks to see if the total salary exceeds \$100000. If so, the last iteration is undone. If not, another iteration of the subtask is initiated. Create an instance of the Faculty table for this experiment using the script

```
CREATE TABLE Faculty (  
    FacId      char(9)  NOT NULL,  
    Name       char(50) NOT NULL,  
    Password   char(20) NOT NULL,  
    Validity   char(1)  NOT NULL,  
    Town       char(20),  
    salary     float   NULL,  
    PRIMARY KEY (FacId)  
)
```

```
Insert into Faculty values ('200000001','200000001', '200000001','V', 'city41', 500.0)
```

```
Insert into Faculty values ('200000002','200000002', '200000002','V', 'city9', 500.0)
```

```
Insert into Faculty values ('200000003','200000003', '200000003','V', 'city14', 500.0)
```

```
Insert into Faculty values ('200000004','200000004', '200000004','V', 'city9', 500.0)
```

```
Insert into Faculty values ('200000005','200000005', '200000005','V', 'city38', 500.0)
```

Steps:

1. Create and initialize the Faculty table. This can be done using the Sybase Jisql tool.
2. Write a stored procedure for the body of the subtask (a one percent increment to all faculty). This can be done using the Sybase Jisql tool. Note that the stored procedure should not contain any transaction keyword (begin transaction, commit transaction, rollback transaction). All transaction control should be done in your java program.
3. Write the Java/JDBC programs SavepointClass.java, NestedClass.java, ChainingClass.java for each of the three cases. Each program invokes the stored procedure in a different way.
4. Run the JDBC code and get results.

Hand In:

1. Source code of the stored procedure.
2. Source code of the java programs in the three cases.
3. Final state of the Faculty table in the three cases.
4. An analysis of what happened in the three cases. You will have trouble with Sybase's implementation of nested transactions. Explain what Sybase does and what it should do.