

B561 Assignment 4
Due date: Monday, November 2nd 2007

Dirk Van Gucht & Mathias Niepert

October 23, 2007

Dynamic SQL & PL/SQL

- (a) Write procedures or functions in dynamic SQL (without cursors) for the following tasks. The difference between procedures is that functions return data while procedures do not. Also, note that the procedures/functions are supposed to be schema independent. Hence, you can use the tables from assignment 3 to test them.

- (1) Write a procedure `exec_SQL(statement)` that will execute any SQL statement. Example:

```
exec_SQL ('CREATE INDEX Sindex ON projects (JINDEX)');
```

creates the index `Sindex` on the column `JINDEX`.

- (2) Given a `WHERE` condition, write a function

```
rowCount(tab, where, schema)
```

that obtains the count of rows in any table, in any schema, for the specified condition. For example, assume the schema from assignment 3 is named "SPJSHIP", then

```
rowCount ('S', 'CITY = London', 'SPJSHIP')
```

returns the number of suppliers which are located in London.

- (3) Write a function

```
updateNumVal (  
  table IN VARCHAR,  
  column IN VARCHAR,  
  value IN NUMBER,
```

```
where IN VARCHAR := NULL,  
schema IN VARCHAR := NULL)
```

that lets you update the value of any numeric column in any table and which returns the number of rows that have been updated (i.e., return value is of type INTEGER). For example, for the database from assignment 3,

```
updateNumVal (  
  'P',  
  'WEIGHT',  
  '20',  
  'COLOR=Red',  
  'SPJSHIP')
```

would set the weight of all red parts to 20 and return the value 3, since 3 rows have been updated.

- (4) Write a function

```
EqualValCol (  
  value IN VARCHAR,  
  column IN VARCHAR := NULL,  
  schema IN VARCHAR := NULL)
```

that returns the number of tables in schema *schema* that contain the string *value* in some row for column *column*. If column is NULL, it returns the number of tables in schema *schema* that contain a row which contains the string *value* for some column. For example, for the database from assignment 3,

```
EqualValCol (  
  'London',  
  'CITY',  
  'SPJSHIP')
```

would return the value 3, since there are 3 tables which contain the string 'London' for the column 'CITY', whereas

```
EqualValCol (  
  'Screw',  
  NULL,  
  'SPJSHIP')
```

would return the value 1, since there is one table which contains the value 'Screw' for some column.

- (5) Write a function

```
SumVal (  
  schema IN VARCHAR := NULL)
```

that returns the sum of all numerical values in the tables of the schema *schema*. For example, for the database from assignment 3,

```
SumVal (  
  'SPJSHIP')
```

would return the value 8701, since this is the sum of (i) all the values for column STATUS in table S; (ii) all the values for column WEIGHT in table P; and (iii) all the values for column QTY in table SPJ.

- (6) Write a function

```
getMaxTab (  
  column IN VARCHAR,  
  schema IN VARCHAR := NULL )
```

that returns the name of the table for which the maximum numerical value for column *column* is greater than the maximum numerical value for column *column* for all the other tables. Of course, you can assume that there are at least two tables in schema *schema* with numerical column *column* (otherwise the function returns an error message). Example: Assume there are three tables T1, T2, and T3, all having a numerical column named *salary*, where 120,000 is the maximum value for *salary* in T1, 90,000 in T2, and 130,000 in T3. Then the function call

```
getMaxTab (  
  'salary',  
  'someschema')
```

would return 'T3'.

- (b) Write procedures or functions in dynamic SQL **now with cursors** for the previously given exercises (1)-(6).