

Exercise #1: Package specification only

1. Define a package named *Matrix*, whose specification includes the following items:

- a collection type (TABLE) of integers named *TheColumns* indexed by a binary integer (INDEX BY BINARY_INTEGER);
- a collection type (TABLE) of *TheColumns* named *TheRows* indexed by a binary integer;
- a variable *m* of type *TheRows*.

2. In an anonymous PL/SQL block:

- initialize each element *Matrix.m(i)(j)* with $i \times j$ (to keep it small, consider a matrix of 2 rows by 3 columns); no declaration is required;
- display *Matrix.m*.

Test! Is a global variable defined in a package a good idea?

Exercise #2: Package specification and body

We want to implement a package to manage table DEMO_STATES (you may copy it from table DARMONT.DEMO_STATES onto your account). The objective is to propose procedures and functions to:

- display the table's contents under format ST: STATE_NAME;
- add a state;
- modify a state's name (knowing its ST code);
- delete a state (knowing its ST code);
- count states.

1. Define the specification of a package named *statepack* containing:

- a record type named *statetuple* with the fields: *code* and *name* bearing the same types than that of table DEMO_STATES' attributes;
- a cursor named *statelist* that returns a *statetuple*.

2. Define the body of package *statepack* by fully defining cursor *statelist*. Test the whole package creation.

3. Add into package *statepack*'s specification and body a procedure named *display* (no parameter) that displays all states on screen in the desired format. In the procedure's code, use cursor *statelist* and a local variable of type *statetuple*. Test the procedure (EXECUTE statepack.display OR BEGIN statepack.display; END:).

4. Add into package *statepack*'s specification and body a procedure named *add* that inserts into table DEMO_STATES an new state whose ST code and name are passed as parameters. Test the procedure.

5. Add into package *statepack*'s specification and body a procedure named *mod* that modifies into table DEMO_STATES the state whose ST code and new name are passed as parameters. Test the procedure.

6. Add into package *statepack*'s specification and body a procedure named *del* that suppresses from table DEMO_STATES the state whose ST code is passed as parameter. Test the procedure.

7. Add into package *statepack*'s body only a function named *statecount* that returns the number of states in table DEMO_STATES. Test the function from outside the package. Does it work?

8. Call function *statecount* within procedure *display* and output the result as "NN state(s)" below the existing list of states. Where must the *statecount* function be defined in the package body? How could the *statecount* function be qualified?