```
// Illustrating ref cursor for a function without a parameter
import java.sql.*;
import oracle.jdbc.*;
import java.math.*;
import java.io.*;
import java.awt.*;
import oracle.jdbc.pool.OracleDataSource;

//    This program shows how to get a record set from an Oracle stored
//    procedure or function.
//    Suppose we have a table students2(sid, firstname, lastname, status,
//    gpa, email), and a stored function called getstudents in a package,
//    which returns all students records.
//    To return a result set from a stored procedure, you must use a
//    cursor variable. Cursor variables are basically pointers to cursors,
//    and you use them to pass around references to cursors, such as a
//    parameter to a stored procedure. This PL/SQL type of a cursor variable
//    is REF CURSOR.
//
//    create a type of ref cursor and a function that returns a
//      result of ref cursor:
//
//    create or replace package refcursor_jdbc1 as
//    type ref_cursor is ref cursor;
//    function getstudents
//    return ref_cursor;
//    end;
//
//    show errors
//
//    create or replace package body refcursor_jdbc1 as
//    function getstudents
//    return ref_cursor is
//    rc ref_cursor;
//    begin
//    open rc for
//    select * from students2;
//    return rc;
//    end;
//    end;
//    /
//    show errors

public class mydemo3 {

   public static void main (String args []) throws SQLException {
    try
    {

        //Connecting to Oracle server. Need to replace username and
        //password by your username and your password. For security
        //consideration, it's better to read them in from keyboard.
        OracleDataSource ds = new oracle.jdbc.pool.OracleDataSource();
        ds.setURL("jdbc:oracle:thin:@castor.cc.binghamton.edu:1521:acad111");
        Connection conn = ds.getConnection("username", "password");

        //Prepare to call stored procedure:
        CallableStatement cs = conn.prepareCall("begin ? := refcursor_jdbc1.getstudents();
end;");

           //register the out parameter (the first parameter)
        cs.registerOutParameter(1, OracleTypes.CURSOR);
```

```
         // execute and retrieve the result set
         cs.execute();
         ResultSet rs = (ResultSet)cs.getObject(1);

         // print the results
         while (rs.next()) {
             System.out.println(rs.getString(1) + "\t" +
                 rs.getString(2) + "\t" + rs.getString(3) +
                 rs.getString(4) +
                 "\t" + rs.getDouble(5) + "\t" +
                 rs.getString(6));
         }

         //close the result set, statement, and the connection
         cs.close();
         conn.close();
    }
    catch (SQLException ex) { System.out.println ("\n*** SQLException caught ***\n" +
 ex.getMessage());}
    catch (Exception e) {System.out.println ("\n*** other Exception caught ***\n");}
   }
 }
```