

HOW TO MAINTAIN REPORT HEADINGS WHEN “No Data Found”

LESSON 9

**A step by step guide to building a TOAD®
Reports Manager report that caters for
instances where there’s “No Data Found”**

**By
Gary Piper**

1 Introduction

TOAD Reports Manager reports work really well when the report has data to display. The same cannot be said however when there is no data to display – i.e. “No Data found”, especially if you have included dynamic data in the report heading. In this lesson we will demonstrate how to write TOAD reports that cater for the “No Data found” scenario.

Before getting started you need to have read and understood **Lesson 2 - How to Create TOAD Report Manager reports that call PLSQL packages** available from our websites [TOAD Report Lessons](#) page as the ability to generate TOAD reports using PLSQL packages is a pre-requisite for this lesson.

Also, ensure you have downloaded and installed the lesson 9 package and installed the two example TOAD reports (TRD format) available from our websites [TOAD Reports Lessons](#) Page.

2 Example Report

To demonstrate this lesson, we will use the two (2) TOAD reports “Lesson 9 - No data found A” and “Lesson 9 - No data found B”. Both reports are functionally identical in that they both list all those database tables for a selected **schema** that have not been analysed in the past **X** days. Both the schema and the overdue days are entered at report run time.

Example Report Parameters

Name	Type	Value (Literal)	Value (Expression)
last_analysed_overdue_a	Cursor		
DAYS_OVERDUE	Number	32	
SCHEMA_OWNER	String	applsys	

Example Report

Owner	Table Name	Rows	LastAnalysed	Age (YYMM:DD)	Partitioned	Temporary
APPLSYS	AD_APPLIED_PATCHES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_APPL_TOPS	1	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_APPS_DEPENDENCIES	27910	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_BUGS	105,050	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_CHECK_FILES	77,960	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_COMPRISING_PATCHES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_FEATURE_VERSIONS	4	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_FILES	278,080	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_FILE_VERSIONS	447,810	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_INSTALL_PROCESSES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_INVOKER_TASKS	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_JOBS_DEFERRED	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_MERGED_TABLES	1,327	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_MERGE_ACTIONS	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_MERGE_ACTION_TABLES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_PARALLEL_COMPILE	0	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_COMPILE_ERRORS	0	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_UPDATES	150	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_UPDATE_UNITS	13,058	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_WORKERS	614	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PATCH_COMMON_ACTIONS	133,180	20-Apr-12 19:01	00:01:03	No	No

Your Footer information goes here

To add further useful information, in both reports the “*Last Analysed*” date will be displayed in red if the analysed process was completed during the working day (Mon - Fri between 08:00 and 18:00). You should not be collecting stats during the working day..... ☺

Both reports call the same stored procedure [reports_lesson_nine](#), which contains the two package bodies one for each of the two (2) reports:

2.1 Report A

The first example TOAD report “*Lesson 9 - No data found A*” calls the stored procedure element *last_analysed_overdue_a*. This report lists all those database tables for a selected **schema** that have not been analysed in the past **X** days. It does **not** preserve headings when “No Data found”.

2.2 Report B

The second example report “*Lesson 9 - No data found B*” calls the stored procedure element *last_analysed_overdue_b*. This report lists all those database tables for a selected **schema** that have not been analysed in the past **X** days. However, this report does preserve headings when “No Data found”.

3 The Report Code

In the procedure [reports_lesson_nine](#), we have used global variables to hold the company name and report footer information, that way it is available for all reports in the procedure.

```
gv_your_co_name  varchar2(100) := 'YOUR COMPANY NAME';
gv_report_footer varchar2(100) := 'Your Footer information goes here';
```

We also created a “common” package to obtain much of the reports header information.

```
REPORTS_LESSON_NINE.get_instance_details (gv_instance_name, gv_host_name, gv_oracle_version );
```

The SQL for both the reports A and B to list the schema tables that are overdue for stats collection is as follows:

```
SELECT v_report_header report_header,
       gv_your_co_name your_co_name,
       gv_report_footer footer,
       upper(i_schema_owner) schema_owner,
       gv_instance_name instance_name,
       v_host_name host_name,
       v_oracle_version oracle_version,
       i_days_overdue entered_days_overdue,
       to_char(sysdate, 'DD-Mon-YY HH24:MI') report_date,
       owner,
       table name,
       num rows,
       nvl(to_char(last_analyzed, 'DD-Mon-YY HH24:MI'), 'Not Analysed') last_analyzed,
       lpad(trunc((months_between(sysdate , last_analyzed)) /12 ),2,'0') ||':'||
       lpad(trunc((months_between(sysdate , last_analyzed)) -
               (trunc((months_between(sysdate , last_analyzed)) /12 ) *12 ),2,'0') ||':'||
```

← Report Header information

```

        lpad(trunc(((months between(sysdate , last_analyzed)) -
        (trunc(months_between(sysdate , last_analyzed) /12 ) *12 )
        _
        (trunc(months_between(sysdate , last_analyzed)) -
        (trunc(months_between(sysdate , last_analyzed) /12 ) *12 )))
        * to char(last day(sysdate), 'DD'),2,'0') age,
initcap(partitioned) partitioned,
decode(temporary, 'Y', 'Yes', 'No') temporary_table,
decode(to_char(last_analyzed, 'd' ),
        1, 0, -- Sunday
        7, 0, -- Saturday
        1) +
        decode( to_number(to_char(last_analyzed, 'hh24' )),
        8, 1, 9, 1, 10, 1, 11, 1, 12, 1,
        13, 1, 14, 1, 15, 1, 16, 1, 17, 1,
        0 ) status_code
FROM dba_tables
WHERE (sysdate - last_analyzed) > i_days_overdue
      and owner = upper(i_schema_owner)
ORDER by owner,
        table_name;
    
```

Note: The “status code” column is used by the TAOD report to colour code the “Last Analysed” value if the “Last Analysed” date is within a working day.

4 No Data Found

For report “Lesson 9 - No data found A” where data exists the report behaves as expected:

Example report

YOUR COMPANY NAME						
Last Analysed - Overdue by more than 31 days						
Schema: APPLSYS						
Host: MY HOST						
Instance: vis						
Version: 10.2						
Report Date: 24-May-12 15:03						
Owner	Table Name	Rows	LastAnalysed	Age (YYMM:DD)	Partitioned	Temporary
APPLSYS	AD_APPLIED_PATCHES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_APPL_TOPS	1	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_APPS_DEPENDENCIES	27910	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_BUGS	105050	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_CHECK_FILES	77960	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_COMPRISING_PATCHES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_FEATURE_VERSIONS	4	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_FILES	278080	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_FILE_VERSIONS	447810	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_INSTALL_PROCESSES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_INVOKER_TASKS	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_JOBS_DEFERRED	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_MERGED_TABLES	1327	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_MERGE_ACTIONS	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_MERGE_ACTION_TABLES	0	20-Apr-12 19:00	00:01:03	No	No
APPLSYS	AD_PARALLEL_COMPILE	0	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_COMPILE_ERRORS	0	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_UPDATES	150	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_UPDATE_UNITS	13058	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PARALLEL_WORKERS	614	20-Apr-12 19:01	00:01:03	No	No
APPLSYS	AD_PATCH_COMMON_ACTIONS	133180	20-Apr-12 19:01	00:01:03	No	No

Your Footer information goes here

However, if we now increase the number of overdue days from one month (31 days) to two months (62 days) all tables are “Up-to-date” so no data is returned. As shown below, the result in our example is a report that lacks working headings etc. and is not suitable for distribution.

Example Report Parameters

+ -		Par	
Name	Type	Value (Literal)	Value (Expre... /
last_analysed_overdue_a	Cursor		
▶ DAYS_OVERDUE	Number	62	
SCHEMA_OWNER	String	applsys	

Example “No data found” report

Owner	Table Name	Rows	LastAnalysed	Age (YYMM-DD)	Partitioned	Temporary
by more than 0 days Schema: Host: Instance: Version: Report Date:						
The "Last Analysed" date will be displayed in red if the analysed process was executed during the working day (Mon - Fri between 08:00 and 18:00)						

As all the header and footer details were part of the SQL, the information is lost when the SQL returns no rows. As a result the report is useless as it contains no useful information.

5 Preserved Header Information Example

Using “Lesson 9 - No data found B” we show what occurs when the report is structured to preserve headings and provide a useful message when “No Data found”:

Example Report Parameters

Name	Type	Value (Literal)	Value (Expression)
last_analysed_overdue_b	Cursor		
DAYS_OVERDUE	Number	62	
SCHEMA_OWNER	String	applsys	

Example “No data found” report

Owner	Table Name	Rows	Last Analysed	Age (YY:MM:DD)	Partitioned	Temporary
	No Tables Found - All up-to-date					

The “Last Analysed” date will be displayed in red if the analysed process was executed during the working day (Mon - Fri between 08:00 and 18:00)

With this report we have retained all the header information and provided a “user friendly” message...

6 So how did we preserve the header information?

It’s really a simple task of checking if there are any rows to display:

```
SELECT count(*)
  INTO v_row_count
 FROM dba_tables
 WHERE (sysdate - last_analyzed) > i_days_overdue
        and owner = upper(i_schema_owner)
 ORDER by owner,
        table_name;
```

If there are rows execute SQL A

```
IF ( v_row_count > 0 ) THEN -- If Rows exist

  OPEN last_analysed_overdue_b_rc FOR
  SELECT v_report_header report_header,
         gv_your_co_name your_co_name,
         gv_report_footer footer,
         ..... * *
```

If not run SQL B

```
ELSE -- If Rows exist

    OPEN last_analysed_overdue_b_rc FOR
    SELECT v_report_header report_header,
           gv_your_co_name your_co_name,
           gv_report_footer footer,
           .....
    FROM dual;
```

Running a “select from dual” forces one record to be returned, that way the header and footer information is preserved.

It really is that simple....

6.1 The second SQL statement – “No Tables found” report message

In the second SQL statement:

```
    OPEN last_analysed_overdue_b_rc FOR
    SELECT v_report_header report_header,
           gv_your_co_name your_co_name,
           gv_report_footer footer,
           upper(i_schema_owner) schema_owner,
           gv_instance_name instance_name,
           gv_host_name host_name,
           gv_oracle_version oracle_version,
           i_days_overdue entered_days_overdue,
           to_char(sysdate, 'DD-Mon-YY HH24:MI') report_date,
           null owner,
           'No Tables Found - All up-to-date' table_name,
           null num_rows,
           null last_analyzed,
           null age,
           null partitioned,
           null temporary_table,
           0 status_code
    FROM dual;
```

...all the values are set to null except a column that is big enough to handle a user friendly message.

Note: All columns must be in the same order and aliased the same as the main SQL statement.

7 Source Code

All the source for this lesson, the procedure and the two TOAD reports in TRD format are available from the www.PIPER-Rx.com website [TOAD Report Lessons](#) page.

8 Want to know more?

There is loads more **FREE** information on TOAD Reports Manager reports and also on all aspects of OEBS Application Administration at the [PIPER-Rx](http://www.PIPER-Rx.com) website. I have been working with TOAD Reports Manager since its inception and have had over 20+ years working with Oracle (the product, not the Company) and Oracle E-Business Suite (since Release 5). Since the late 1990's I have spent more time sharing these learnings and the most popular papers and case studies I have presented are available at www.PIPER-Rx.com as well as over 250 TOAD Reports Manager reports and a whole host of Tips and Reports I have used throughout my career.

All information is at the [PIPER-Rx.com](http://www.PIPER-Rx.com) website **FREE** so why not check it out....I hope you find it useful! – **40,000+ downloaders to date can't be wrong!**

9 Disclaimer

The material contained in this document is provided by the author "as is" and any express or implied warranties, including, but not limited to, any implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the author be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of any content or information, even if advised of the possibility of such damage. It is always recommended that you seek independent, professional advice before implementing any ideas or changes to ensure that they are appropriate.

Oracle®, Oracle Applications® & Oracle E-Business Suite® are registered trademarks of Oracle Corporation

TOAD® is a registered trademark of Quest Software