

在Oracle中使用PL\_SQL操作COM对象（2）PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/143/2021\\_2022\\_\\_E5\\_9C\\_A8](https://www.100test.com/kao_ti2020/143/2021_2022__E5_9C_A8)

Oracle\_E4\_c102\_143667.htm 5) 操作Excel示例 本节介绍如何操作COM对象（以Excel对象为例），将数据库里的表记录通过PL/SQL语句输出成Excel文件。通过对象浏览器可以查看Excel对象提供的属性和方法。（如：在Excel菜单，工具 -> 宏 -> 打开Visual Basic编辑器，在编辑器里，视图 -> 对象浏览器）。使用Excel对象一般针对单元格进行操作（如，设置第一个单元格字体大小的代码为：Range("A1").Font.Size = 20）。对应的COM外部例程操作的步骤为：1、获取程序句柄；2、获取工作簿句柄；3、获取工作表句柄；4、获取Range区句柄；5、获取Font类句柄；6、设置Size属性。如下为操作Excel对象的例子，首先建立示例表，然后建立操作包（package），最后是使用包函数的过程（procedure）。创建一个测试用表：-- 销售表（脚本）

```
CREATE TABLE
IT_SALE_TAB(ITS_ID VARCHAR2(5),ITS_DATE
DATE,ITS_TOTAL NUMBER). INSERT INTO
IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL) VALUES( ' 1
',TO_DATE( ' 2004-01-01 ' , ' YYYY-MM-DD ' ),250).
INSERT INTO IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL)
VALUES( ' 2 ' ,TO_DATE( ' 2004-02-01 ' , ' YYYY-MM-DD
' ),150). INSERT INTO
IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL) VALUES( ' 3
',TO_DATE( ' 2004-03-01 ' , ' YYYY-MM-DD ' ),80).
INSERT INTO IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL)
```

```
VALUES( ' 4 ' ,TO_DATE( ' 2004-04-01 ' , ' YYYY-MM-DD
' ),96). INSERT INTO
IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL) VALUES( ' 5
' ,TO_DATE( ' 2004-05-01 ' , ' YYYY-MM-DD ' ),300).
INSERT INTO IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL)
VALUES( ' 6 ' ,TO_DATE( ' 2004-06-01 ' , ' YYYY-MM-DD
' ),210). INSERT INTO
IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL) VALUES( ' 7
' ,TO_DATE( ' 2004-07-01 ' , ' YYYY-MM-DD ' ),320).
INSERT INTO IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL)
VALUES( ' 8 ' ,TO_DATE( ' 2004-08-01 ' , ' YYYY-MM-DD
' ),280). INSERT INTO
IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL) VALUES( ' 9
' ,TO_DATE( ' 2004-09-01 ' , ' YYYY-MM-DD ' ),276).
INSERT INTO IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL)
VALUES( ' 10 ' ,TO_DATE( ' 2004-10-01 ' , ' YYYY-MM-DD
' ),368). INSERT INTO
IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL) VALUES( ' 11
' ,TO_DATE( ' 2004-11-01 ' , ' YYYY-MM-DD ' ),163).
INSERT INTO IT_SALE_TAB(ITS_ID,ITS_DATE,ITS_TOTAL)
VALUES( ' 12 ' ,TO_DATE( ' 2004-12-01 ' , ' YYYY-MM-DD
' ),305). COMMIT. 操作Excel对象包 ( Package )。 CREATE
OR REPLACE PACKAGE oraExcel IS xIthin BINARY_INTEGER
DEFAULT 2. DummyToken BINARY_INTEGER.
applicationToken BINARY_INTEGER:=-1. -- Excel对象句柄
WorkBooksToken BINARY_INTEGER:=-1. -- 工作簿句柄
```

```
WorkBookToken BINARY_INTEGER:=-1. WorkSheetToken
BINARY_INTEGER:=-1. -- 工作表句柄 WorkSheetToken1
BINARY_INTEGER:=-1. RangeToken BINARY_INTEGER:=-1.
-- Range区句柄 ChartObjectToken BINARY_INTEGER:=-1. -- 图
表对象句柄 ChartObject1 BINARY_INTEGER:=-1. Chart1Token
BINARY_INTEGER:=-1. hLines BINARY_INTEGER:=-1. i
BINARY_INTEGER. err_src VARCHAR2(255). err_desc
VARCHAR2(255). err_hpf VARCHAR2(255). err_hpID
BINARY_INTEGER. -- 创建Excel对象 FUNCTION
CreateExcelWorkSheet(servername VARCHAR2) RETURN
BINARY_INTEGER. -- 插入数据(字符型) FUNCTION
setCellValues(RANGE VARCHAR2,data VARCHAR2,TYPE
VARCHAR2) RETURN BINARY_INTEGER. -- 插入数据(日期
型) FUNCTION setCellValues(RANGE VARCHAR2,data
DATE,TYPE VARCHAR2) RETURN BINARY_INTEGER. -- 插
入数据(整型) FUNCTION setCellValues(RANGE
VARCHAR2,data BINARY_INTEGER,TYPE VARCHAR2)
RETURN BINARY_INTEGER. -- 插入数据(实型) FUNCTION
setCellValuesReal(RANGE VARCHAR2,data DOUBLE
PRECISION,TYPE VARCHAR2) RETURN BINARY_INTEGER.
-- 设置列宽 FUNCTION setCellColWidth(RANGE
VARCHAR2,width DOUBLE PRECISION,TYPE VARCHAR2)
RETURN BINARY_INTEGER. -- 设置表格 FUNCTION
setCellLines(RANGE VARCHAR2,BordersIndex
BINARY_INTEGER,weight BINARY_INTEGER DEFAULT
xlThin,TYPE VARCHAR2) RETURN BINARY_INTEGER. -- 合
```

```

并单元格 FUNCTION setCellMerge(RANGE
VARCHAR2,bValues BOOLEAN,TYPE VARCHAR2) RETURN
BINARY_INTEGER. -- 设置字体属性 FUNCTION
setCellFont(RANGE VARCHAR2,Property VARCHAR2,
fontValues BINARY_INTEGER,TYPE VARCHAR2) RETURN
BINARY_INTEGER. -- 调用方法 FUNCTION
callMethod(RANGE VARCHAR2,MethodName VARCHAR2)
RETURN BINARY_INTEGER. -- 插入图表 FUNCTION
InsertChart(xpos BINARY_INTEGER,ypos BINARY_INTEGER,
width BINARY_INTEGER,height BINARY_INTEGER, RANGE
VARCHAR2,TYPE VARCHAR2) RETURN BINARY_INTEGER.
-- 保存文件 FUNCTION SaveToFile(filename VARCHAR2)
RETURN BINARY_INTEGER. -- 关闭Excel对象 FUNCTION
CloseExcel RETURN BINARY_INTEGER. END oraExcel. 数据包
体内容 ( PACKAGE BODY ) CREATE OR REPLACE
PACKAGE BODY oraExcel IS FUNCTION
CreateExcelWorkSheet(servername VARCHAR2) RETURN
BINARY_INTEGER IS BEGIN -- 创建Excel对象
i:=ordcom.CreateObject( ' Excel.Application ' , 0,
servername,applicationToken). IF (i!=0) THEN -- 创建失败,提示
返回的错误信息 ordcom.GetLastError(err_src, err_desc, err_hpf,
err_hpID). raise_application_error(-20000,err_src || err_desc ||
err_hpf || err_hpID). END IF. -- 通过程序对象句柄获取工作簿
句柄 i:=ordcom.GetProperty(applicationToken, ' WorkBooks ' ,
0, WorkBooksToken). ordcom.InitArg(). ordcom.SetArg(-4167,
' 14 ' ). i:=ordcom.Invoke(WorkBooksToken, ' Add ' , 1,

```

```

WorkBookToken). ordcom.InitArg(). ordcom.SetArg( ' Sheet 1 ' ,
' BSTR ' ). -- 获取工作表句柄
i:=ordcom.GetProperty(applicationToken, ' Worksheets ' , 0,
WorkSheetToken1). i:=ordcom.Invoke(WorkSheetToken1, ' Add
' , 0, WorkSheetToken). RETURN i. END CreateExcelWorkSheet.
FUNCTION setCellValues(RANGE VARCHAR2,data
VARCHAR2, TYPE VARCHAR2) RETURN BINARY_INTEGER
IS BEGIN ordcom.InitArg(). ordcom.SetArg(RANGE, ' BSTR ' ).
-- 获取Range区句柄,之后将数据写入到指定的单元格
i:=ordcom.GetProperty(WorkSheetToken, ' Range ' , 1,
RangeToken). i:=ordcom.SetProperty(RangeToken, ' Value ' ,
data, TYPE). i:=ordcom.DestroyObject(RangeToken). RETURN i.
END setCellValues. FUNCTION setCellValues(RANGE
VARCHAR2,data BINARY_INTEGER,TYPE VARCHAR2)
RETURN BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range ' , 1,
RangeToken). i:=ordcom.SetProperty(RangeToken, ' Value ' ,
data, type). i:=ordcom.DestroyObject(RangeToken). RETURN i.
END setCellValues. FUNCTION setCellValuesReal(RANGE
VARCHAR2,data DOUBLE PRECISION,TYPE VARCHAR2)
RETURN BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range ' , 1,
RangeToken). i:=ordcom.SetProperty(RangeToken, ' Value ' ,
data, type). i:=ordcom.DestroyObject(RangeToken). RETURN i.

```

```

END setCellValuesReal. FUNCTION setCellValues(RANGE
VARCHAR2,data DATE,TYPE VARCHAR2) RETURN
BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range ' ,1,
RangeToken). i:=ordcom.SetProperty(RangeToken, ' Value ' ,
data, TYPE). i:=ordcom.DestroyObject(RangeToken). RETURN i.
END setCellColWidth. FUNCTION setCellMerge(RANGE
VARCHAR2,bValues BOOLEAN,TYPE VARCHAR2) RETURN
i. END setCellValues. FUNCTION setCellColWidth(RANGE
VARCHAR2,width DOUBLE PRECISION,TYPE VARCHAR2)
RETURN BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range
' ,1,RangeToken). i:=ordcom.SetProperty(RangeToken,
' ColumnWidth ' ,width,TYPE).
i:=ordcom.DestroyObject(RangeToken). RETURN
BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range
' ,1,RangeToken). i:=ordcom.SetProperty(RangeToken,
' MergeCells ' ,bValues, ' BOOLEAN ' ).
i:=ordcom.DestroyObject(RangeToken). RETURN i. END
setCellMerge. FUNCTION setCellLines(RANGE
VARCHAR2,BordersIndex BINARY_INTEGER, weight
BINARY_INTEGER DEFAULT xIThin,TYPE VARCHAR2)

```

```

RETURN BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range
' ,1,RangeToken). ordcom.InitArg().
ordcom.SetArg(BordersIndex,TYPE). -- 画表格的具体载入参数
i:=ordcom.GetProperty(RangeToken, ' Borders ' ,1,hLines).
i:=ordcom.SetProperty(hLines, ' weight ' ,weight,TYPE).
i:=ordcom.DestroyObject(hLines).
i:=ordcom.DestroyObject(RangeToken). RETURN i. END
setCellLines. FUNCTION setCellFont(RANGE
VARCHAR2,Property VARCHAR2,fontValues
BINARY_INTEGER,TYPE VARCHAR2) RETURN
BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range
' ,1,RangeToken). ordcom.InitArg().
ordcom.SetArg(Property,TYPE). ordcom.InitArg().
i:=ordcom.GetProperty(WorkSheetToken, ' ChartObjects ' ,0,
ChartObjectToken). ordcom.InitArg(). ordcom.SetArg(xpos, ' I2
' ). -- 载入图表对象位置参数 ordcom.SetArg(ypos, ' I2 ' ).
ordcom.SetArg(width, ' I2 ' ). ordcom.SetArg(height, ' I2 ' ).
i:=ordcom.Invoke(ChartObjectToken, ' Add ' ,4, ChartObject1).
-- 添加图表 i:=ordcom.GetProperty(ChartObject1, ' Chart ' ,
0,Chart1Token). ordcom.InitArg(). ordcom.SetArg(RANGE,
' BSTR ' ). i:=ordcom.GetProperty(WorkSheetToken, ' Range ' ,
1, RangeToken). -- 选取区域 ordcom.InitArg().

```

```

ordcom.SetArg(RangeToken, ' DISPATCH ' ). IF TYPE= ' xIPie
' THEN charttype := -4102. ELSIF TYPE= ' xl3DBar ' THEN
charttype := -4099. ELSIF TYPE= ' xlBar ' THEN charttype := 2.
ELSIF TYPE= ' xl3dLine ' THEN
i:=ordcom.GetProperty(RangeToken, ' Font ' ,0,hLines). -- 获取
字体对象
i:=ordcom.SetProperty(hLines,Property,fontValues,TYPE).
i:=ordcom.DestroyObject(hLines).
i:=ordcom.DestroyObject(RangeToken). RETURN i. END.
FUNCTION callMethod(RANGE VARCHAR2,MethodName
VARCHAR2) RETURN BINARY_INTEGER IS reti
BINARY_INTEGER := -1. BEGIN ordcom.InitArg().
ordcom.SetArg(RANGE, ' BSTR ' ).
i:=ordcom.GetProperty(WorkSheetToken, ' Range
' ,1,RangeToken). ordcom.InitArg().
i:=ordcom.Invoke(RangeToken,MethodName,0,reti). -- 调用对象
的方法 i:=ordcom.DestroyObject(RangeToken). RETURN reti.
END. FUNCTION InsertChart(xpos BINARY_INTEGER, ypos
BINARY_INTEGER, width BINARY_INTEGER, height
BINARY_INTEGER, RANGE VARCHAR2, TYPE VARCHAR2)
RETURN BINARY_INTEGER IS charttype BINARY_INTEGER:=
-4099. BEGIN charttype:= -4101. END IF.
ordcom.SetArg(charttype, ' I4 ' ).
i:=ordcom.Invoke(Chart1Token, ' ChartWizard ' , 2,
DummyToken). i:=ordcom.DestroyObject(RangeToken).
i:=ordcom.DestroyObject(ChartObjectToken).

```



```
i:=ordcom.DestroyObject(ChartObject1).
i:=ordcom.DestroyObject(Chart1Token). RETURN i. END
InsertChart. FUNCTION SaveToFile(filename VARCHAR2)
RETURN BINARY_INTEGER IS BEGIN ordcom.InitArg().
ordcom.SetArg(filename, ' BSTR ' ).
i:=ordcom.Invoke(WorkBookToken, ' SaveAs ' , 1,
DummyToken). -- 保存文件 IF (i!=0) THEN
ordcom.GetLastError(err_src, err_desc, err_hpf, err_hpID).
raise_application_error(-20000,err_src || err_desc || err_hpf ||
err_hpID). END IF. RETURN i. END SaveToFile. FUNCTION
CloseExcel RETURN BINARY_INTEGER IS BEGIN
ordcom.InitArg(). ordcom.InitArg(). ordcom.SetArg(FALSE,
' BOOL ' ). i:=ordcom.Invoke(WorkBookToken, ' Close ' , 0,
DummyToken). i:=ordcom.DestroyObject(WorkBookToken).
ordcom.InitArg(). i:=ordcom.Invoke(WorkBooksToken, ' Close
' , 0, DummyToken).
i:=ordcom.DestroyObject(WorkBooksToken).
i:=ordcom.Invoke(applicationToken, ' Quit ' , 0, DummyToken).
-- 关闭所有句柄 i:=ordcom.DestroyObject(WorkSheetToken).
i:=ordcom.DestroyObject(WorkSheetToken1).
i:=ordcom.DestroyObject(applicationToken).
i:=ordcom.DestroyObject(ChartObjectToken).
i:=ordcom.DestroyObject(Chart1Token).
i:=ordcom.DestroyObject(hLines).
i:=ordcom.DestroyObject(ChartObject1).
i:=ordcom.DestroyObject(dummyToken). RETURN i. END
```

CloseExcel. END oraExcel. 100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)