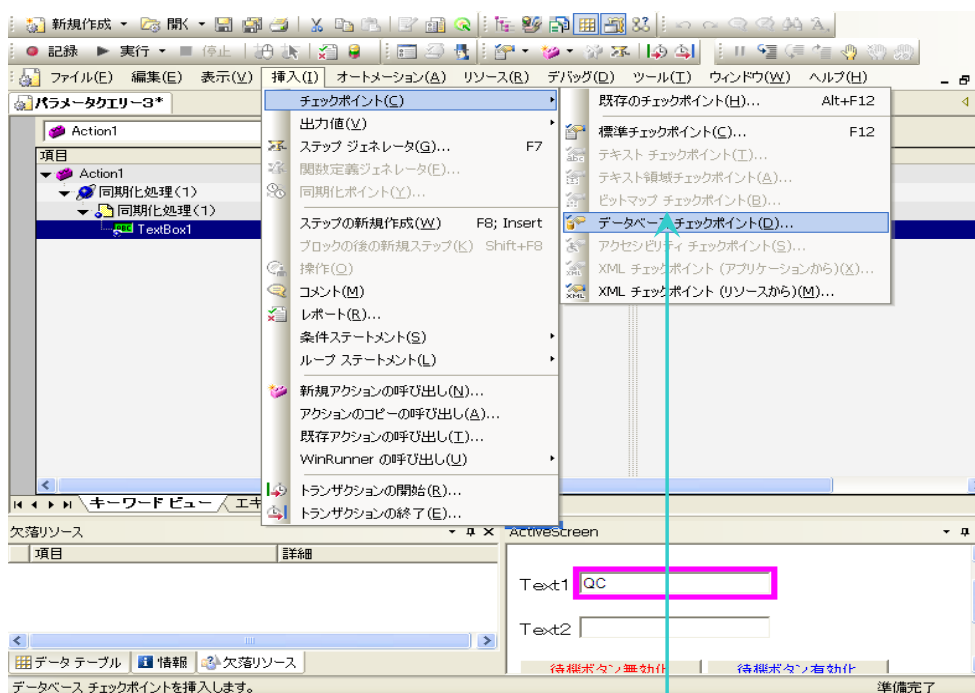


## 第4書 データベース操作

## 4. 1 データベースへの接続

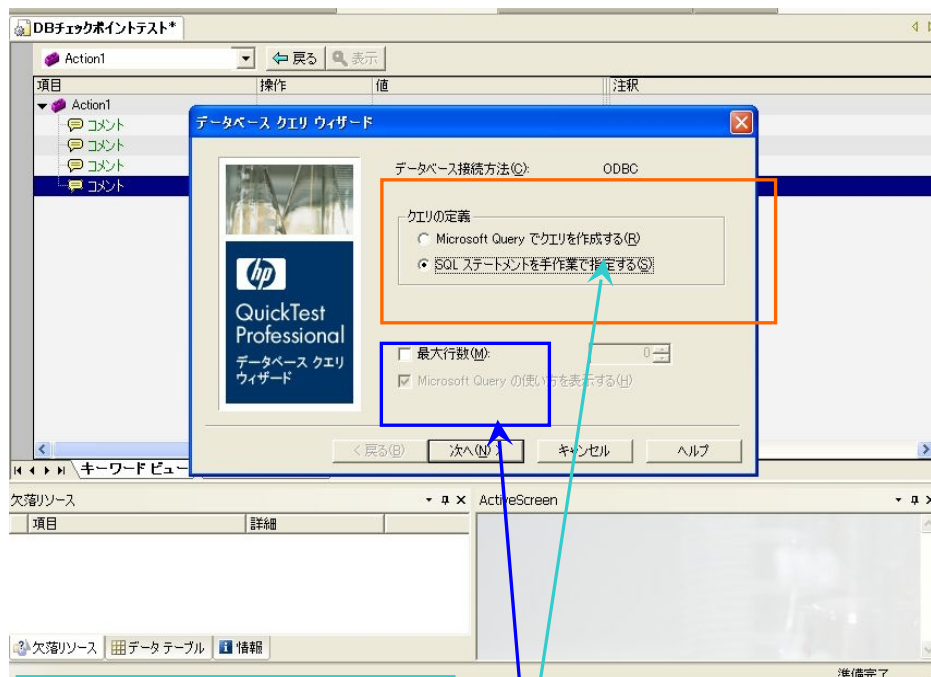
### (1) データベースチェックポイントの追加



データベースチェックポイントを追加します。

### (2) ODBCによる接続

- ・ ODBCを使用してデータベースへ接続します。

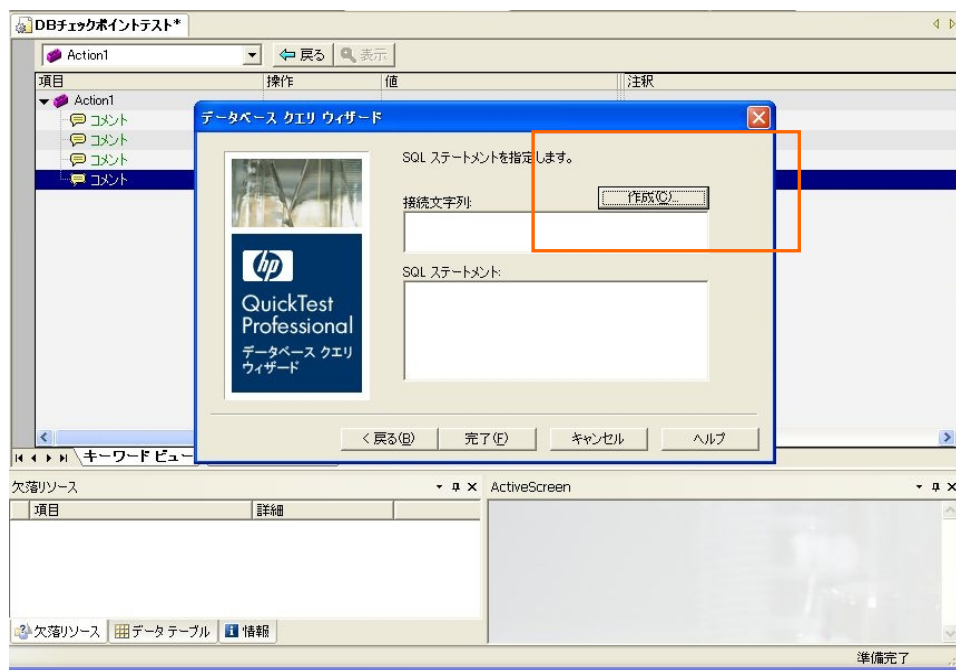


SQL文を手作業で指定する場合

最大フェッチ行数を指定する場合はここで最大行数を指定します。

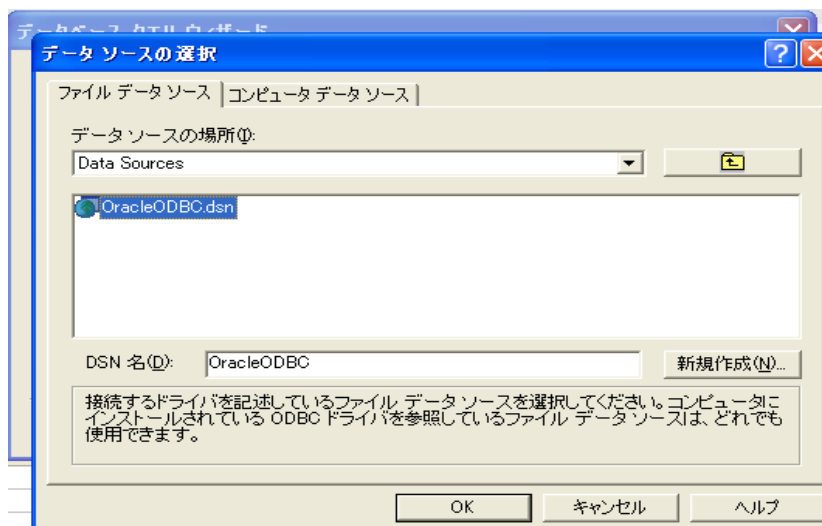
- ・ 接続文字列を作成します。

【作成ボタンクリック】 > 【データソース選択】 > 【データベース接続】により作成された文字列が入ります。

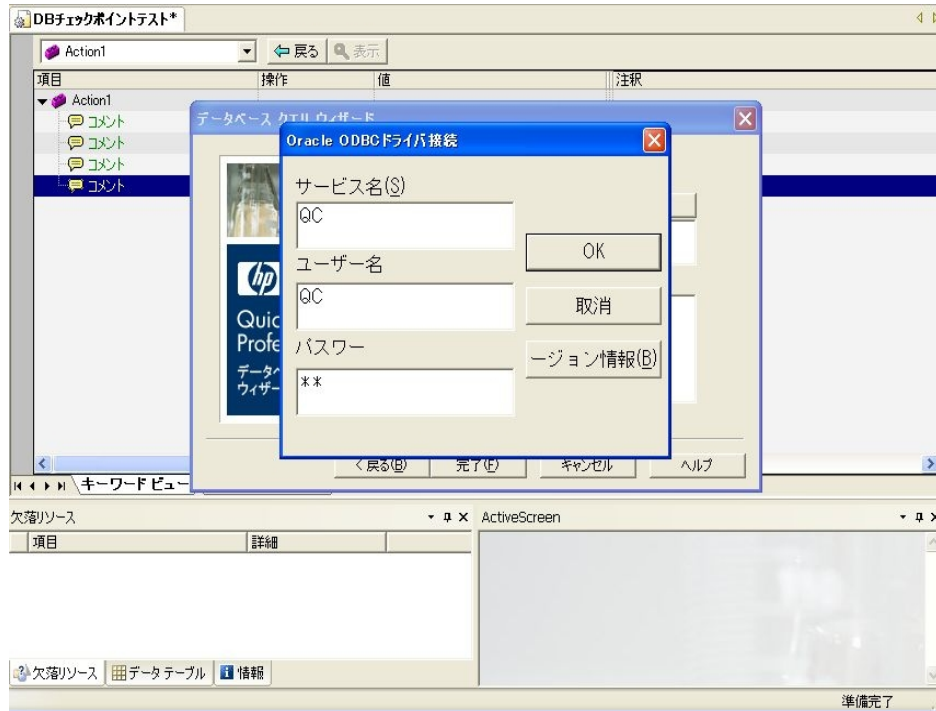


- ・ データソースを選択します。

ODBCドライバがインストールされていない場合は、ODBCドライバを先にインストールして下さい。



- ・データベースへ接続します。



### 4. 3 サンプルコード

```

' =====
' データベース関連ライブラリ (DataBase Access Script for Oracle)
' =====

Set objConn = CreateObject("ADODB.Connection")
Set objRs = CreateObject("ADODB.Recordset")

String1 = "依頼者2" 'テスト
Seq = 2 'テスト

' CommandText = "SELECT * FROM サポート内容 WHERE SEQ =" & Seq ' NUMBER 型の場合
CommandText= "SELECT * FROM サポート内容 WHERE 依頼者 =" & "" & String1 & "" ' VARCHAR2 型の場合
ConnectionString = "DRIVER={Oracle in OraDb10g_home1};"&_
    "SERVER=QC;UID=QC;PWD=QC"&_
    ";DBQ=QC;DBA=W;APA=T;EXC=F;XSM=Default;FEN=T;QTO=T;FRC=10;FDL=10;"&_
    "LOB=T;RST=T;GDE=F;FRL=Lo;BAM=IfAllSuccessful;NUM=NLS;DPM=F;MTS=T;"&_
    "MDI=Me;CSR=F;FWC=F;FBS=60000;TLO=0;"

MsgBox "SQL=" & CommandText 'テスト

objConn.Open ConnectionString
Set objRs = objConn.Execute(CommandText)

' =====
' テスト
' =====

rowcount = 0
Do While Not objRs.EOF
    rowcount = rowcount + 1
    MsgBox "Row " & rowcount & vbCrLf & vbCrLf & objRs(0)
    objRs.MoveNext
Loop

```

#### 4. 4 DBアクセスライブラリ (Oracle)

```
'=====
'@Author      : M.Fujishima
'@Description: Function Library (for Oracle DB Access)
'=====

'@Calling Sequence:
'   session_name_string (output):
'   connection_string  (input) : ODBC connection string
'=====

' DataBase connection  (データベースとの接続を確立する)
'=====

Function db_connect (byRef curSession, connection_string)

    Dim connection
    On Error Resume Next
    Set connection = CreateObject("ADODB.Connection")
    If Err.Number <> 0 then
        db_connect= "Error # " & CStr(Err.Number) & " " & Err.Description
        err.clear
        Exit Function
    End If

    connection.Open connection_string
```

```
' =====  
' @Calling Sequence:  
' session_name(input): session_name_string  
' CountSQL(output) : record count  
' =====  
' -----  
' executes an SQL statement.  
' -----  
  
Function db_get_rows_count (byRef recordSet)  
    Dim rows  
    rows = 0  
    recordSet.MoveFirst  
    Do Until recordSet.EOF  
        rows = rows+1  
        recordSet.MoveNext  
    Loop  
    db_get_rows_count = rows  
End Function
```

#### 4. 5 DBアクセスライブラリ (Oracle) 使用例

```

' =====
' サンプルシナリオ for DB Access Library
' =====

Set recordSet = CreateObject("ADODB.Recordset")

connection_string = "DRIVER={Oracle} in
OraDb10g_home1};SERVER=QC;UID=QC;PWD=QC;DBQ=QC;DBA=W;APA=T;EXC=F;XSM=Default;FEN=T;QTO=T;FRC=10;FDL=10;LOB=T;
RST=T;GDE=F;FRL=Lo;BAM=IfAllSuccessful;NUM=NLS;DPM=F;MTS=T;MDI=Me;CSR=F;FWC=F;FBS=60000;TLO=0;"

' =====

' セッション確立
' =====

rtnCode = db_connect(session, connection_string)
If rtnCode <> 0 Then
    MsgBox "セッション確立でエラーが発生しました"
else
    MsgBox "セッション確立 正常終了しました。"
end If

' =====

' SQL文実行 (SELECT)
' =====

SQL = "select * from 掲示板データ"
rtnCode = db_execute_query(recordSet, session, SQL)
rowCount = 0
Do While Not recordSet.EOF
    rowCount = rowCount + 1
    For i = 0 to 6
        ' MsgBox "掲示板データ " & rowcount & vbCrLf & vbCrLf & recordSet(i)
    Next
    recordSet.MoveNext
Loop

' =====

SQL = "select * from サポート内容"
rtnCode = db_execute_query(recordSet, session, SQL)
rowCount = 0
Do While Not recordSet.EOF
    rowCount = rowCount + 1
    For i = 0 to 6
        ' MsgBox "Row " & rowcount & vbCrLf & vbCrLf & recordSet(i)
    Next
    recordSet.MoveNext
Loop

```



```

MsgBox "レコード数 =" & rowCount
' -----
' レコード数取得
' -----

SQL = "SELECT COUNT(*) FROM 掲示板データ"
' rtnCode = db_execute_query (recordSet , session, SQL)
' rowCount = 0
' Do While Not recordSet.EOF
'   ' rowCount = rowCount + 1
'   ' recordSet.MoveNext
' Loop
MsgBox "レコード数 =" & rowCount
' -----
' SQL 文実行 (DELETE)
' -----

Set recordSet = CreateObject("ADODB.Recordset")
SQL = "delete from 掲示板データ where 見出し = '掲示板データ見出し3'"
' rtnCode = db_execute_query (recordSet , connection, SQL)
' -----
' SQL 文実行 (INSERT)
' -----

Set recordSet = CreateObject("ADODB.Recordset")
SQL = "INSERT INTO 掲示板データ VALUES (' Q T P ', '2009/06/18', '掲示板データ見出し3', '投稿者3', '所属3', '担
当システム3', '掲示板データ3')"
' rtnCode = db_execute_query (recordSet , connection, SQL)
' -----
' SQL 文実行 (UPDATE)
' -----

Set recordSet = CreateObject("ADODB.Recordset")
SQL = "UPDATE 掲示板データ SET 分類=' Q C ' WHERE 見出し='掲示板データ見出し3'"
' rtnCode = db_execute_query (recordSet , connection, SQL)
' -----
' DataBade disconnection (確立している接続を切る)
' -----

rtnCode = db_disconnect (session) 'session_name:input
If rtnCode <> 0 Then
    MsgBox "エラーが発生しました"
else
    MsgBox "正常終了しました。"
End If

```

## 第6章 データテーブル操作

## 6. 5 実行時にデータテーブルを作成する方法

Actionに関係なく、実行時にデータテーブルを作成することが出来ます。

The screenshot shows the U-Forex application window. The script editor contains the following code:

```

1:  '-----
2:  'U-Forex(定額版) サイクルテスト用スクリプト
3:  ' By M.Fujishima 2010.12.21
4:  '-----
5:
6:  path1 = ".¥データ¥前提条件データ¥"
7:  path2 = ".¥データ¥配信レートデータ¥"
8:  path3 = ".¥データ¥依頼内容データ¥"
9:
10:
11:  '使用データファイル名を取得する
12:
13:  'テスト日とケースNoよりデータを読み込む
14:  caseNo = 11 'Test No.となる
15:
16:  zenteiDir = path1 & caseNo & "." & "前提データ.xls"
17:  haishinDir = path2 & caseNo & "." & "配信データ.xls"
18:  iraiDir = path3 & caseNo & "." & "依頼内容データ.xls"
19:
20:
21:  '前提条件データをLocal Tableへ読み込む
22:
23:  '実行時にLocal Sheetを作成し、データを格納する
24:  var = DataTable.AddSheet("前提データ").AddParameter("有効比率","")
25:  DataTable.GetSheet("前提データ").AddParameter("充当金","")
26:  DataTable.GetSheet("前提データ").AddParameter("不足金","")

```

The data table below the script editor is empty, with columns A through I and rows 1 through 4. A red box highlights the data table area, and a blue arrow points to a text box explaining that the data table is not created before execution.

実行前に、「前提データ」、「配信レート」、および「依頼内容」のデータテーブルは作成されていません。

## ・ サンプルスクリプト

```
'=====
' 実行時にデータテーブル作成サンプルスクリプト
'
'                By M.Fujishima    2010.12.21
'=====

path1 = "..¥データ¥前提条件データ¥"
path2 = "..¥データ¥配信レートデータ¥"
path3 = "..¥データ¥依頼内容データ¥"
'=====

' 使用データファイル名を取得する
'=====

' テスト日とケース No よりデータを読み込む
CaseNo      = 11 ' Test No. となる
zenteiDir   = path1 & caseNo & "." & "前提データ.xls"
haishinDir  = path2 & caseNo & "." & "配信データ.xls"
iraiDir     = path3 & caseNo & "." & "依頼内容データ.xls"
'=====

' 前提条件データを Local Table へ読み込む
'=====

' 実行時に Local Sheet を作成し、データを格納する
var = DataTable.AddSheet ("前提データ").AddParameter ("有効比率", "")
```