

PL/SQL Web Toolkit Tutorial

Ammar Sajdi
PALCO/REALSOFT

Example1

```
create or replace procedure hello is
begin
  http.print('<h1>Hello Eyad</h1>');
end;
/
```

Example 2

```
Create or replace procedure first is
begin
  http.print('Hello World');
end;
/
```

Example3

```
create or replace procedure get_name is
  x emp.ename%type;
begin
  select ename into x from emp where empno=7499;
  http.print(x);
end;
/
```

Example 4

```
create or replace procedure get_name is
  x emp.ename%type;
  y emp.sal%type;
begin
  -- <h1> and </h1> are standard HTML tags
  -- to print more than one value , you must use ||
  select ename , sal into x, y from emp where empno=7499;
  http.print('<h1>' ||x||' ||y || '</h1>');
end;
/
```

Example 5

```
create or replace procedure get_name is
  x emp.ename%type;
  y emp.sal%type;
begin
  -- the following package will produce <html>
  http.htmlopen;
  -- adding background color
  http.print('<body bgcolor = "#00FFFF">');
  -- <h1> and </h1> are standard HTML tags
  -- to print more than one value , you must use ||
```

```

select ename , sal into x, y from emp where empno=7499;
  http.print('<h1>' ||x||' ||y || '</h1>');
end;
/

```

Example 6

```

create or replace procedure get_All is
cursor c1 is select * from emp;
rec c1%rowtype;
begin
  open c1;
  loop
    fetch c1 into rec;
    exit when c1%notfound;
-- This will print all values after each other, there is no new line
    http.print (rec.ename ||' || rec.sal);
  end loop;
end ;
/

```

Example 7

```

create or replace procedure get_All is
cursor c1 is select * from emp;
rec c1%rowtype;
begin
  open c1;
  loop
    fetch c1 into rec;
    exit when c1%notfound;
-- This will print all values after each other, there is no new line
    http.print (rec.ename ||' || rec.sal);
    http.print ('<Br>');
    http.nl;
  end loop;
end ;
/

```

Example 8

```

create or replace procedure new_Table is
  x boolean;
begin
-- OWA_UTIL is a package that is installed when OAS is installed
-- and the procedure tableprint will print a table based on the following
-- first column .. table name
-- second column .. BORDER (if you want the table to have borders_
-- NULL is you do not want a border
-- third column owa_util.pre_table or owa_util.html_table
-- this option will affect how the borders look like
-- fourth column The list of column you want to display seperated by
-- commas, and contained within single quotes
-- fifth column You can add an order by and a where condition
-- sixth column you can supply your own column headers (alias)
  x:= owa_util.tableprint('EMP','border',owa_util.pre_table,'ename, sal, job');
end;

```

/

Example 9

```
CREATE OR REPLACE PROCEDURE new_proc (p1 number) AS
-- Parameter p1 is expected to be passed by the user
-- to choose a specific deptno
-- Very IMPORTANT NOTE
-- To call this procedure from the Browser and pass this parameter
-- you must follow the standard for that which is
-- http://.../.../new_proc?P1=30   (P1 is the parameter name)
--                               (30 is the deptno )
cursor c1 is select * from emp where deptno = p1;
rec c1%rowtype ;
BEGIN
-- Generated by WEB ALCHEMY
http.htmlOpen;
http.headOpen;
http.title( 'table2');
http.headClose;
http.bodyOpen( cattributes => ' text="#000000" bgcolor="#CC66CC" link="#0000EE"
vlink="#551A8B"');
http.header( 1, 'Our converted program');
open c1;
loop
  fetch c1 into rec;
  exit when c1%notfound;
http.tableOpen( cborder => 'TRUE', cattributes => ' COLS=2 WIDTH="100% "' );
http.tableRowOpen;
http.tableData( '<FONT color="#3333FF">' ||
rec.ename ||
'</FONT>');
http.tableData( '<FONT color="#3333FF">' ||
rec.deptno ||
'</FONT>');
http.tableRowClose;
http.tableClose;
http.para;
-- http.print( 'End of page ' );
http.bodyClose;
http.htmlClose;
end loop;
END;
/
```

Example 10

```
create or replace procedure testing (p1 number default 0) is
  x boolean;
begin
  x:= owa_util.tableprint('DEPT','BORDER',owa_util.html_table,'*',
  'where deptno ='||p1);
end;
/
```

Example 11

```

CREATE OR REPLACE PROCEDURE testing1 AS
BEGIN
  http.htmlOpen;
  http.headOpen;
  http.title( 'table3');
  http.headClose;
  http.bodyOpen;
  http.tableOpen( cborder => 'TRUE', cattributes => ' COLS=3 WIDTH="50%" BGCOLOR="#33CCFF" );
  for rec in (select * from dept) loop
    http.tableRowOpen;
    http.tableData( rec.deptno);
    http.tableData( rec.dname);
    http.tableData( rec.loc);
    http.tableRowClose;
  end loop;
  http.tableClose;
  http.bodyClose;
  http.htmlClose;
END;
/

```

example 12

create or replace procedure calendar is

begin

```

-- This package expects a select statement whose first column is a date
-- The second column (if provided) will appear as a text on the date
-- specified by the first column.
-- The third column (if specifiiec) will act as a hyperlink for the second
-- column. Oracle expects a name of an existing procedure stored in the
-- third column
-- the following two records are inserted in the database to explain
/*

```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7839	KING	NEW_PROC?p1=10		17-NOV-0081	5000		10
7698	BLAKE	new_proc?p1=30	7839	01-MAY-0081	2850		30

*/

```

owa_util.calendarprint('select hiredate , ename, job from emp');
-- note that in the above statement the third field (job) must contain
-- procedure name
end;
/

```

example 13

create or replace procedure check_box is

begin

```

  http.formOpen (owa_util.get_owa_service_path||'new_proc');
  -- The above procedure will call get_name when the submit button is pressed
  http.formcheckbox('P1',10);
  http.formhidden ('P1','0');
  -- Note that when the SUBMIT button is pressed, the name of the CheckBox
  -- and itsvalue will be passed to the procedure defined in the http.FORMOPEN
  -- Therefore, the name of the checkbox should match the name of the parameter in

```

```

-- the procedure.
-- http.formcheckbox will create a check box with a name of B1 and a value of 10
-- effective when the check box is pressed
-- http.formhidden will take effect if the checkbox is not touched
-- http.nl
  http.formsubmit(", 'GO...'); -- Go here is the caption
owa_util.showsouce('CHECK_BOX') ;-- to display the source code on the browser
end;
/

```

example 14

```

create or replace procedure check_box is
begin
  http.formOpen (owa_util.get_owa_service_path||'new_proc');
  -- The above procedure will call get_name when the submit button is pressed
-- the following cursor will print checkboxes as many times as there
-- are record in the dept table
For rec in (select * from dept) loop
  http.formcheckbox('P1',rec.deptno);
  http.print (rec.dname);
  http.nl;
  http.formhidden ('P1','0');
-- Note that when the SUBMIT button is pressed, the name of the CheckBox
-- and itsvalue will be passed to the procedure defined in the http.FORMOPEN
-- Therefore, the name of the checkbox should match the name of the parameter in
-- the procedure.
-- http.formcheckbox will create a check box with a name of B1 and a value of 10
-- effective when the check box is pressed
-- http.formhidden will take effect if the checkbox is not touched
end loop;
  http.formsubmit(", 'GO...'); -- Go here is the caption
owa_util.showsouce('CHECK_BOX') ;-- to display the source code on the browser
end;
/

```

example 15

```

create or replace procedure check_box is
begin
  http.formOpen (owa_util.get_owa_service_path||'new_proc');
  -- The above procedure will call get_name when the submit button is pressed
-- the following cursor will print checkboxes as many times as there
-- are record in the dept table
-- the following will display three pop list to let you
-- choose a day.
owa_util.choose_date('date_name',sysdate);
For rec in (select * from dept) loop
  http.formcheckbox('P1',rec.deptno);
  http.print (rec.dname);
  http.nl;
  http.formhidden ('P1','0');
-- Note that when the SUBMIT button is pressed, the name of the CheckBox
-- and itsvalue will be passed to the procedure defined in the http.FORMOPEN
-- Therefore, the name of the checkbox should match the name of the parameter in

```

```

-- the procedure.
end loop;
  http.formsubmit('GO...'); -- Go here is the caption
owa_util.showsouce('CHECK_BOX') ;-- to display the source code on the browser
end;
/

```

Example 16

```

create or replace procedure pop_list is
begin
  http.formOpen (owa_util.get_owa_service_path||'new_proc');
  -- the following will create a pick list with a name P1
  http.formselectopen('P1','Pick a choice');
  for rec in (select deptno, dname from dept) loop
  -- The following will populate the pick list with dname and deptno
  -- Note that dname will appear in the list
  -- upon submitting, deptno will be assigned to P1 and sent to the procedure
  -- whose name is specified in the http.formopen
  http.formselectoption(rec.dname,"value="||rec.deptno);
  end loop;
  -- If you do not include the following statement, no output will appear
  http.formselectclose;
  http.formsubmit('CALL DEPT');
end;
/

```

Example 17

```

create or replace procedure list is
  p1 number;
begin
  http.formOpen (owa_util.get_owa_service_path||'new_proc');
  -- the first section in the listprint procedure is the select statement
  -- then the parameter name that will be assigned a value when the user
  -- presses on the submit button
  -- the number of row that will appear in the list (its hight)
  owa_util.listprint('select deptno, dname, null from dept','P1',2);
  http.formsubmit('submit');
end;
/

```

Example 18

```

create or replace procedure radio is
begin
  http.formOpen (owa_util.get_owa_service_path||'new_proc');
  -- The above procedure will call new_proc when the submit button is pressed
  -- the following cursor will print Radio Elements as many times as there
  -- are record in the dept table
  -- the following will display three pop list to let you
  -- choose a day.
  -- owa_util.choose_date('date_name',sysdate);
  For rec in (select * from dept) loop
  --

```

```

-- The next statement will define a radio group called P1.
-- The value which will be passed when any element is selected
-- is rec.deptno
    http.formradio('P1',rec.deptno);
    http.print (rec.dname);
    http.nl;
-- Note that when the SUBMIT button is pressed, the name of the RADIO
-- group (P1) and its value (rec.deptno) will be passed to the procedure defined in the http.FORMOPEN
-- Therefore, the name of the RADIO should match the name of the parameter in
-- the procedure (p1 in our case)
--
end loop;
    http.formsubmit(",GO..."); -- Go here is the caption
owa_util.showsouce('RADIO') ;-- to display the source code on the browser
end;
/

```

Example 19

```

create or replace procedure fixed_poplist is
begin
http.formOpen (owa_util.get_owa_service_path||'new_proc');
    -- the following will create a pick list with a name P1
http.formselectopen('P1','Pick a choice');
--
-- This example shows how to fill the poplist by static values
http.formselectoption('ÇáĬÇÆÑÉ ÇáÇæâi','value=10');
http.formselectoption('ÇáĬÇÆÑÉ ÇáËÇäiÉ','value=20');
http.formselectoption('ÇáĬÇÆÑÉ ÇáËÇäËÉ','value=30');
http.formselectclose;
http.print('</p><p>');
http.formsubmit(",report");
end;
/

```

Example 20

```

-- The first procedure will call the data entry screen

create or replace procedure data_Entry
is
begin
    http.formOpen (owa_util.get_owa_service_path||'dept_ins');
    http.print ('Please enter the deptno ');
    --
    -- This create a text item on the form
    -- The name of the item is Dno, its width is 2 and accepts 2 char
    http.formtext ('dno',2,2);
    http.nl;
    http.print (' Please enter Dept Name');
    http.formtext('name',30,30);
    http.nl;
    -- When you submit, you are supposed to prepare a procedure
    -- Than will insert the entered values into the database.
    -- In this case, we will create a procedure called 'dept_ins'

```

```

    http.formsubmit(",INSERT");
    http.Formclose;
-- Please NOTE that FORMCLOSE should appear as a last statement in
-- Form processing after the submit
end;
/

```

Example 21

```

create or replace procedure data_passwd
is
begin
http.formOpen (owa_util.get_owa_service_path||'dept_ins');
http.print ('Please enter the deptno ');
--
-- This create a text item on the form
-- The name of the item is Dno, its width is 2 and accepts 2 char
http.formtext ('dno',2,2);
http.nl;
http.print (' Please enter Dept Name');
-- This will create a text item, that will display ****
-- to protect user entry information
http.formpassword('name',30,30);
http.nl;
-- When you submit, you are supposed to prepare a procedure
-- Than will insert the entered values into the database.
-- In this case, we will create a procedure called 'dept_ins'
http.formsubmit(",INSERT");
http.Formclose;
-- Please NOTE that FORMCLOSE should appear as a last statement in
-- Form processing after the submit
end;
/

```

Example 22

/* The first procedure will display all column for the table name you specify when you call the procedure , the user should select some column from a check box and call another procedure that will display a table containing values of these column from the database */

```

create or replace procedure dynamic_table (the_table in char) is
cursor cols is select column_name from user_tab_columns
where table_name = upper (the_table);
begin
    http.htmlopen;
    http.headopen;
    http.htitle('Query the'||the_table||'table');
    http.headclose;
    http.bodyopen;
    http.formopen(owa_util.get_owa_service_path||'do_query');
-- assigning the parameter the_table to a variable called the_Table
http.formhidden('the_table',the_table);
-- assinging the value dummy to a new variable called COLS
http.formhidden('COLS','dummy');
--

```



```

-- creating check box group
for crec in cols loop
htp.formcheckbox('COLS',crec.column_name);
htp.print(crec.column_name);
htp.nl;
end loop;
htp.formsubmit('Execute Query');
-- when the submit is executed, the variables The_table and COL
-- are passed to the DO_QUERY procedure
owa_util.showsouce('DYNAMIC_TABLE');
htp.htmlclose;
htp.bodyclose;
htp.htmlclose;
end;
/

/* This is the second procedure, it will accept the table name as a parameter
and the selected Columns from the previous procedure. The selected column
will appear as a string in the COLS array

*/
create or replace procedure do_query (the_table in varchar2,cols in owa_util.ident_arr) is
column_list varchar2(32000);
col_counter integer;
ignore boolean;
begin
col_counter :=2;
loop

-- This loop will extract the columns from the COLS area and add a comma
-- between each column name
column_list := column_list||cols(col_counter)||',';
col_counter := col_counter+1;
end loop;
exception

-- When the array can no longer fetch any new value, it will raise
-- No data found. TO us it means that it fetched all column
when no_data_found then
--
-- This substr is important to remove the last comma
column_list := substr(column_list,1,length(column_list)-1);

-- This is a standard OWA_UTIL to simply display a list for the chosen columns
-- and their values from the TABLE specified in the first procedure
ignore := owa_util.tableprint(the_table,'BORDER',owa_util.HTML_table,
column_list);
end;

-- The second procedure is responsible for inserting the values

Create or replace procedure dept_ins (name in varchar2, dno in number) is
begin
htp.formOpen (owa_util.get_owa_service_path||'data_Entry');
-- The following statement will cause the parameter that were passed

```

```
-- by the previous procedure (Data entry) to be inserted into the database
insert into dept (deptno, dname)
  values (dno,name);
  http.formsubmit('BACK');
end;
/
```

Example 23

```
create or replace procedure the_list is
The_Table varchar2(200);
begin
http.formOpen (owa_util.get_owa_service_path||'dynamic_table');
owa_util.listprint('select table_name,table_name, null from user_tables','the_table',5);
-- The select statement show above will get table_names for existing
-- tables in the database
http.formsubmit('COLS');
end;
/
```

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.