

Oracle APEX Stuff (Oracle Application Express - APEX)

Lets dive into the divine world of #orclapex #Freelancer

Wednesday, 18 July 2018

Show values in right side of shuttle

While working with select list and shuttle, when we want to display values into right side of shuttle depending upon selection from select list, using numerous methods we can achieve this, today i'll show the simplest way it can be done without any hassle.

After trying different things which all worked according to the requirement, I found a simple solution which can be more effective though preventive measures must be taken before using it :)

We will create a select list (item(p1_dept) + lov) based on department table, which shows how many departments we have.

```
SELECT
    dname d,
    deptno r
FROM dept
```

Now, we will create a shuttle based on value employees table and put a deptno as filter in this sql, so whenever we select deptno from select list it will pass those values into shuttle sql, make sure to fill cascading lov parent items.(p1_dept)

```
SELECT
    ename ,
```

[pranav shah](#)

Search This Blog

• [Home](#)

Followers

Followers (3)



Follow by Email

[Blog Archive](#)

```

empno
FROM emp
WHERE deptno = :p1_dept --p1_dept is page item

```

List of Values

Type: SQL Query

SQL Query

```

SELECT
  empno
FROM emp
WHERE
  deptno = :p1_dept --p1_dept is page item

```

Display Extra Values: Yes

Cascading LOV Parent Item(s): P1_DEPT

Items to Submit:

Optimize Refresh: Yes

After finishing basic stuff we will create a dynamic action - pure client side to wrap it up.

Create action :

When : After refresh

Selection Type : Item(s)

Item(s) : Shuttle Item -- In may case it is P1_SHUTTLE

True action:

Execute Javascript Code :

```

var emps = document.getElementById('P1_SHUTTLE_LEFT').options;
var allValues = "";

```

```

for (index = 0, len = emps.length; index < len; ++index) {
  allValues += emps[index].value.concat(":");
}

```

`$s('P1_SHUTTLE',allValues);` //here we can use apex_item setValue to set values.

Other ways...

we can use Set value as in true action

Settings -> **set type** -> **sql statement**

```

select listagg( empno, ',' ) within group (order by upper( ename ) ) ret
from emp
where deptno = :P1_DEPT

```

▼ 2018 (1)

▼ July (1)

[Show values in right side of shuttle](#)

► 2017 (8)

► 2016 (20)

► 2015 (22)

Labels

- [adf](#)
- [apex oracl](#)
- [apexoracle5.0](#)
- [barcode](#)
- [collection](#)
- [css](#)
- [Csv](#)
- [dialogpage](#)
- [interactive grid](#)
- [javascript](#)
- [jquery](#)
- [modalwindow](#)
- [oracle apex](#)
- [oracle apex 5.0](#)
- [oracle apex 5.0.](#)
- [oracle apex 5.1](#)
- [oracleapex](#)
- [pl/sql](#)
- [plsql](#)
- [print](#)
- [printing](#)
- [shuttle](#)
- [SQL developer](#)

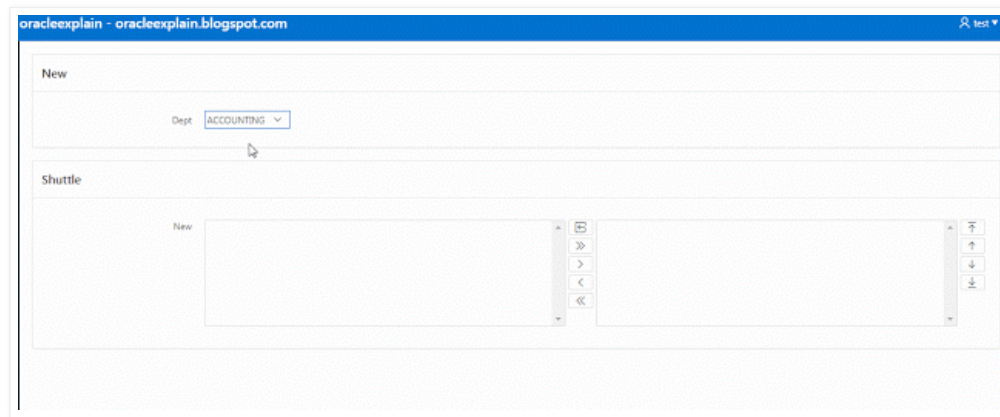
Items to Submit : P1_DEPT

Affected Elements :

Selection Type : Item(s)

Item(s) : My_Shuttle

Final output :



Also you can have a look here <https://community.oracle.com/thread/3591836> which is another way of setting values.

test/test

[demo](#)

Hope you like it!!

By [Pranav Shah - July 18, 2018](#) No comments:



Labels: [oracle apex](#), [oracle apex 5.1](#), [shuttle](#)

Tuesday, 12 December 2017

Preserve checked checkbox in collection - Interactive Grid

The latest interactive grid of oracle APEX comes with so many different possibilities to play with.

Thanks to Oracle Application Express developer team we can use this "Fabulous" feature. For someone who does not know what is Interactive Grid [here is the link](#) to explorer more.

Here we are going to collect selected data from interactive grid and add them into the collection using our custom made button.

To do so first we need to create a Interactive grid based on EMP table (or any other table - PK empno in this example) which do have EDIT option selected to yes. After creating the grid give it a static id (in my case i have given "pg_10_ig_emp" which describes (pg10(page_10),ig(interactive grid),emp(based on emp table)).

For some one who does not know how to work with javascript please go through this links which are very helpful.

[orclapex-ig-cheat-sheet](#)

[how to hack apex interactive grid](#)

Task to be accomplished.

- 1) Check how many records we have checked.
- 2) On click of our custom button add records in collection using ajax call.
- 3) Decorate custom button

Okay, so now first we will get checked records from our interactive grid and loop through it and on click of a button we are going to add it into the collection using ajax call.



```
var i, records, record, $EmpNo, model, $EmpName,
    view = apex.region("pg_10_ig_emp").widget().interactiveGrid("getCurrentView");

// editable view
if ( view.supports.edit )
{
    model = view.model;
    records = view.getSelectedRecords();
    if ( records.length > 0 )
    {
        for ( i = 0; i < records.length; i++ )
        {
            // Logic for ajax
        }
    }
}
```

That will get us the selected (checked) records now here we are checking if the grid is editable if it is we are going in. Again we are checking if count of selected record(s) is more than 0 then we will process further else none records are selected no need for further process.

We will create a Ajax call to add records in collection first we make sure if collection exists or not if exists then truncate it if not then create it so we get accurate data.



```
begin

IF NOT apex_collection.collection_exists(p_collection_name => 'BUCKET') THEN
    APEX_COLLECTION.CREATE_OR_TRUNCATE_COLLECTION('BUCKET');
END IF;

apex_collection.add_member (p_collection_name => 'BUCKET',
                           p_generate_md5    => 'YES',
                           p_c001           => apex_application.g_x01,
                           p_c002           => apex_application.g_x02
                           );
end;
```

Our collection name is "**BUCKET**" and added some member attributes([apex_collection](#))
(If we are using numbers then we can use **p_n001** instead of **p_c001**)

Lets click on "**attributes**" of interactive grid of our page and scroll down till you find "**Javascript Code**" in **properties** and add below code.

```

function(config) {
    var $ = apex.$Query,
        toolbarData = $.apex.interactiveGrid.copyDefaultToolbar(),
        toolbarGroup = toolbarData.toolbarFind("actions3");

    // Lets add new button with our custom action
    toolbarGroup.controls.push( {
        type: "BUTTON",
        action: "collect-action", //name of the action
        icon: "fa fa-bath", //button icon
        iconBeforeLabel: true, //icon position
        hot: true, //hot button
    });

    config.toolbarData = toolbarData;

    config.initActions = function( actions )
    {
        actions.add({
            name: "my-action",
            label: "Collect record(s)", //button label
            action: function(event, focusElement)
            {
                var i, records, record, $EmpNo, model, $EmpName,
                    view = apex.region("pg_10_ig_emp").widget().interactiveGrid("getCurrentView");

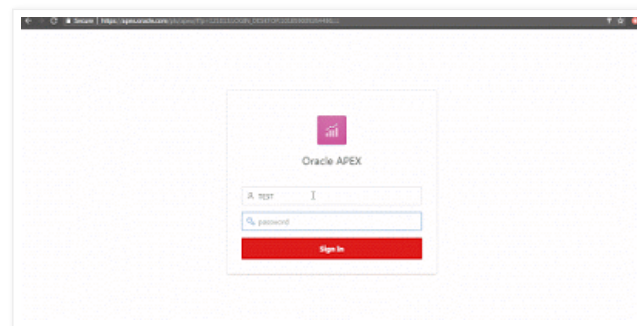
                // editable view
                if ( view.supports.edit )
                {
                    model = view.model;
                    records = view.getSelectedRecords();
                    if ( records.length > 0 )
                    {
                        for ( i = 0; i < records.length; i++ )
                        {
                            record = records[i];
                            $EmpNo = parseFloat(model.getValue(record, "EMPNO"));
                            $EmpName = model.getValue(record, "ENAME");
                            apex.server.process ("my_ajax_process",
                                {
                                    x01:$EmpNo,
                                    x02:$EmpName
                                },
                                {type: 'GET', dataType: 'text', success: function( text )
                                {
                                    apex.message.showPageSuccess("Information collected in our bucket!");
                                }});
                        }
                    }
                }
            }
        });
    }
    return config;
}

```

Let's go muddy we are adding custom action in our interactive grid tool bar with custom action name if you read carefully we have applied icon, texture, name. Then we are simply adding our ajax call to define action which will fire when we click on our custom button. Lastly we are showing success message when checked records are added into the collection.



On click of a button we are checking how many records checked then one by one we are passing it to our ajax call (x01,x02) when it gets done show the success message.



[Demo link](#)

credentials : test/test

YouTube : <https://youtu.be/BkQVrjXPj74>

By [Pranav Shah](#) - [December 12, 2017](#) 1 comment:



Labels: [collection](#), [interactive grid](#), [javascript](#), [oracle apex 5.1](#), [plsql](#)

Monday, 16 October 2017

Change loading icon in oracle APEX with just CSS

Okay so how can we change the loading(indicator) icon which is given by Oracle Application Express(5.0,5.1.*) with just CSS only yeah you heard me right with just a simple CSS.

Let's dig into it a bit more. Assume we already have loading indicator icon on some of the page of our Oracle APEX application if we inspect it it will have a class **u-Processing** we will grab that class and modify css based on our requirement.

1) Here we are going to hide the Loading icon with transparent color

```
.u-Processing { content: "<div class='\"Name2\"'></div>" !important;
height: 10% !important;
width: 30% !important;
cursor: text !important;
border-radius: 0% !important;
background-color: transparent !important;
z-index: 1000 !important;
}
```

2)Then in second step we actually going to append our text before loading icon comes

```
.u-Processing:before{ content: 'Please wait...' !important;
height: 10% !important;
width: 50% !important;
cursor: text !important;
border-radius: 0% !important;
font-size: 200% !important;
background-color: White !important;
z-index: 1000 !important;
}
```

Here i have created a DA which fires on page load which will refresh our region on same page.

So lets see how it looks....

The screenshot shows the Oracle APEX 'Play Ground (BluePrint)' interface. On the left is a dark blue sidebar with navigation links: 'Home', 'Datatable with js', and 'Not removable values'. The main area has a title bar 'Play Ground (BluePrint)' with a user dropdown 'nobody'. Below the title bar is a 'How To !!' link. A search bar with a magnifying glass icon and a 'Go' button is present. Below the search bar is a table with the following data:

Empno	Ename	Job	Mgr	Hiredate	Sal	Comm	Deptno
7839	KING	PRESIDENT	-	17-NOV-1981	5000	-	10
7698	BLAKE	MANAGER	7839	01-MAY-1981	2850	-	30
7782	CLARK	MANAGER	7839	06-JUN-1981	2450	-	10
7566	JONES	MANAGER	7839	02-APR-1981	2975	-	20
7788	SCOTT	ANALYST	7566	09-DEC-1982	3000	-	20
7602	FORD	ANALYST	7566	03-DEC-1981	3000	-	20
7369	SMITH	CLERK	7902	17-DEC-1980	800	-	20
7499	ALLEN	SALESMAN	7698	20-FEB-1981	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981	1250	500	30
7654	MARTIN	SALESMAN	7698	28-SEP-1981	1250	1400	30
7844	TURNER	SALESMAN	7698	08-SEP-1981	1500	0	30

Note* : For someone who does not know why i have used !important please continue reading.

It means, essentially, what it says; that 'this is important, ignore subsequent rules, and any usual specificity issues, apply this rule!'

In normal use a rule defined in an external stylesheet is overruled by a style defined in the head of the document, which, in turn, is overruled by an in-line style within the element itself (assuming equal specificity of the selectors). Defining a rule with the !important 'attribute' (?) discards the normal concerns as regards the 'later' rule overriding the 'earlier' ones.

Also, ordinarily, a more specific rule will override a less-specific rule. So:

```
a {
  /* CSS */
}
```

Is normally overruled by:

```
body div #elementID ul li a {
  /* CSS */
}
```

As the latter selector is more specific (and it doesn't, normally, matter where the more-specific selector is found (in the head or the external stylesheet) it will still override the less-specific selector (in-line style attributes will always override the 'more-', or the 'less-', specific selector as it's always more specific.

If, however, you add !important to the less-specific selector's CSS declaration, it will have priority.

P.S. : Use !important only when necessary.

YouTube : <https://youtu.be/BkQVrjXPj74>

By Pranav Shah - [October 16, 2017](#) 4 comments:

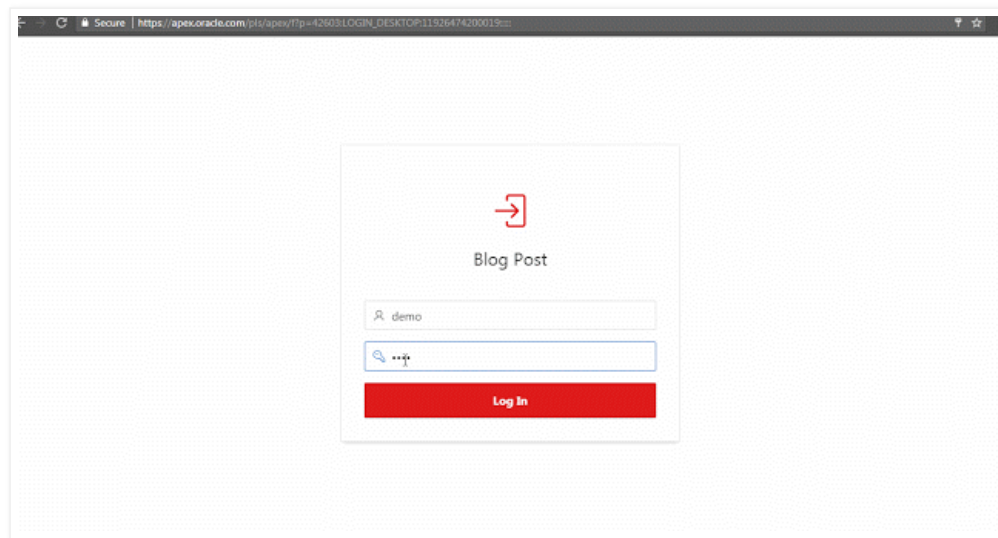
Sunday, 10 September 2017

Download line wise(record) csv using Oracle Application express

We are going to discuss about how we can download the CSV file for line wise records in oracle apex.

Let's Create a interactive(IR)/classical report on our page(based on emp table).

Region Query :



```
SELECT empno,
       ename,
       job,
       sal,
       '<a class="btn btn-large btn-primary logout" href="#"> <span class="pi_barcode fa fa-download" onclick="download_csv('||empno||','||empno||')"'
       style="align-items: center"></a>' AS "Dwnd_Csv"
FROM emp
```

What's this ?

```
<a class="btn btn-large btn-primary logout" href="#"> <span class="pi_barcode fa fa-download" onclick="download_csv('||empno||','||empno||')"'
style="align-items: center"></a>
```

Here we are defining what class we want and what should it look like. Oracle application express has inbuilt classes defined so we just need to reference them to get the desired output.

Onclick: It's an event will fire whenever user clicks on Icon.(Our function download_csv(param1,prama2);

So far so good but where is our actual function !

We need to create a function in Function and Global Variable Declaration and call it using onclick event.

```
function download_csv(eno,eno) { // here you can specify you're params for demo purpose i'd pass more params
apex.server.process ("download_csv",{ x01:eno, x02:eno},
    {
        type: 'GET',
        dataType: 'json',
        success: function(l_json_str)
    {
        // first we make sure it's a object or not then deal with it accordingly.
        var arrData = typeof l_json_str != 'object' ? JSON.parse(l_json_str) : l_json_str;
        var CSV = "";
        var row = "";
        for (var index in arrData[0]) {
            row += index + ',';
        }
        row = row.slice(0, -1);
        CSV += row + "\r\n";

        for (var i = 0; i < arrData.length; i++) {
            var row = "";
            for (var index in arrData[i]) {
                var arrValue = arrData[i][index] == null ? "" : arrData[i][index] ;
                row += arrValue + ',';
            }
            row.slice(0, row.length - 1);
            CSV += row + "\r\n";
        }
        var fileName = "EmployeeDetails"; // This will be our CSV file name.
        /*if(msieversion()){
            var IEwindow = window.open();
            IEwindow.document.write('sep=\r\n' + CSV);
            IEwindow.document.close();
            IEwindow.document.execCommand("SaveAs", true, fileName"barcode.csv");
            IEwindow.close();
        } else {*/
        var uri = 'data:application/csv;charset=utf-8,' + escape(CSV);
        var link = document.createElement("a");
        link.href = uri;
        link.style = "visibility:hidden";
        link.download = fileName + ".csv";
        document.body.appendChild(link);
        link.click();
        document.body.removeChild(link);
        //}
    }
});
```

```
}
```

Javascript function is calling our ajax server process which is...

Create Ajax callback with the name: download_csv

DECLARE

l_json_str VARCHAR2(1000); -- based on data we are fetching we need to change this

L_eno VARCHAR2(50):= apex_application.g_x02;

BEGIN

l_json_str:='[';

for x in (SELECT * FROM emp

WHERE empno = L_eno)

LOOP

l_json_str:=l_json_str|| '{"EmpNo":"' || x.empno || ',' "EmpName":"' || x.ename || ',' "Job":"' || x.job || ',' "Salary":"' || x.sal || '"}';

END LOOP;

l_json_str:=substr(l_json_str,1,length(l_json_str)-1);

l_json_str:=l_json_str||']';

sys.http.p(l_json_str);

END;

Assuming we know how ajax server process works with-in oracle application express.

Demo : [Download_csv](#)

Twitter : [Follow](#)

Linked-In : [Connect](#)

By [Pranav Shah](#) - [September 10, 2017](#) No comments:



Labels: [css](#), [Csv](#), [javascript](#), [oracle apex 5.0](#), [oracle apex 5.1](#), [pl/sql](#), [SQL developer](#)

Friday, 1 September 2017

Display formatted console with Oracle apex 5.1(Oracle Application Express)

Let's say we want to display object as a table when we console. Normally we just console everything and displaying as it is without formatting our object. It would be nice and beautiful if we just format it a bit. But question is HOW?

Ummm.... May be a structured way is way to go!! WAIT someone just said structured way then we must use table(That's what I think).

We can test this in console by pressing F12(Most of the browser support this using windows)

F12 ~> Console(Chrome)

F12 ~> Debugger(Firfox)

Code Snippet :

```
var Apps = [
  {"Application":"Sample Database", "Status":"Activated"},
  {"Application":"Interactive Grid", "Status":"Activated"},
  {"Application":"Oracle Stuff", "Status":"De-activated"}
];
console.table(Apps);
Output :
```

(index)	Application	Status
0	Sample Database	Activated
1	Interactive Grid	Activated
2	Oracle Stuff	De-activated

Array(3)

Cool right!! Just check how much cleaner it looks and it's easy to verify all the values which we are getting.

Now, let's try this in our apex application:

Here i did use a blueprint application we will just create a simple page and static region in it.(this will show case you how we can achieve the same in complex scenario when we have to deal with Ajax and Json)

Region will have :

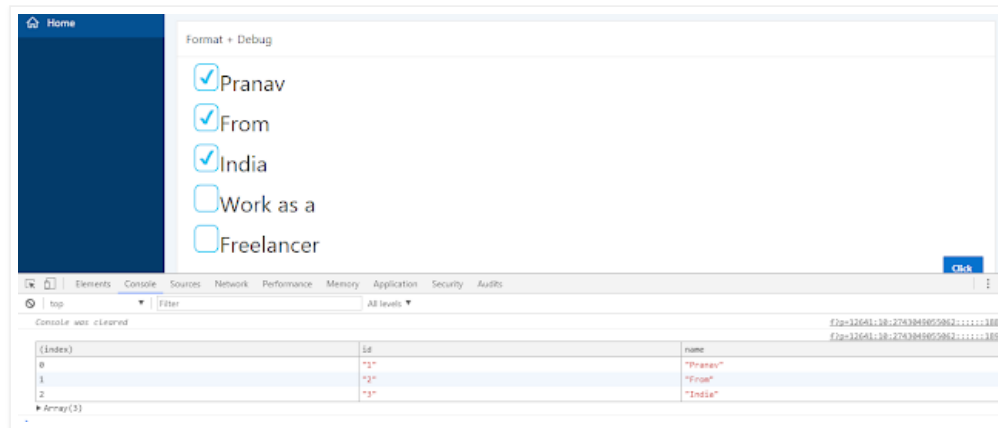
```
<label><input type="checkbox" class="chkCompare" title='Pranav' id='1' />
  <span style="font-size: 30px;">Pranav</span>
</label><br><br>
<label><input type="checkbox" class="chkCompare" title='From' id='2' /> <span style="font-size: 30px;">From</span></label><br><br>
<label><input type="checkbox" class="chkCompare" title='India' id='3' /> <span style="font-size: 30px;">India</span></label><br><br>
<label><input type="checkbox" class="chkCompare" title='Work as a' id='4' /> <span style="font-size: 30px;">Work as a</span></label><br><br>
<label><input type="checkbox" class="chkCompare" title='Freelancer' id='5' /> <span style="font-size: 30px;">Freelancer</span></label>
```

Now add some javascript code:

```
function test1(){
  var checkedElemets = $(''.chkCompare:checked');
  var values = checkedElemets.map(function() {
    return {
      id: this.id,
      name: this.title
    };
  }).get();
  console.clear();
  console.table(values);
}
```

Of course we have a button on a page which will call this function and kindly check in DOM window what output message it displays in console.

Output:



Demo : [Formatted Console](#)

Again this is very neat and simple example just try to use this in complex scenario it will help.

Some of the very awesome console API we can find here:

[Console API Reference](#)

[Debugging Asynchronous JavaScript with Chrome DevTools](#)

By [Pranav Shah](#) - [September 01, 2017](#) 3 comments:



Labels: [css](#), [javascript](#), [jquery](#), [oracle apex 5.1](#), [oracleapex](#)

Monday, 28 August 2017

Known Issues - Application Express 5.1(know bugs oracle 5.1)

25373158 - COOKIE NOT SET IN APEX 5.1

This bug can cause unsent cookies and Location redirect headers during login processing. It affects applications with customized login processes that write headers (e.g. a "Remember Me" cookie) or calls to LOGIN/POST_LOGIN in custom authentication sentry functions that stop the APEX engine.

Solution: There is a patchset exception for this available on My Oracle Support - search by bug number.

25430809 - INVALID BUILD OPTION MAPPINGS AFTER APP IMPORT

The default export option for "Exports with Original IDs" was set to "Yes" in APEX 5.1. This bug fix resets the default back to "No". The bug fix also corrects issues with build options, where export/import caused broken references.

Solution: There is a patchset exception for this available on My Oracle Support - search by bug number.

25303884 - CLIENT SIDE MESSAGING FAILS TO SUBSTITUTE #IMAGE_PREFIX# IN PAGE-LEVEL ERRORS, CAN AFFECT OLD OR CUSTOM THEMES

If you have a page with page attribute Reload on Submit = 'Only for Success' then page-level error notifications will be displayed using client side messaging. In old or custom themes it fails to substitute the #IMAGE_PREFIX# substitution string, which can lead to incorrect image references and therefore users perceiving missing images. This does not affect the Universal Theme, because no such images references exist for page-level error messages.

Workaround: You can set the page attribute 'Reload on Submit' back to 'Always', however, this will preclude you from being able to use an Interactive Grid on that page.

25303970 - INLINE ITEM ERRORS FOR OLDER THEMES ARE NOT ALWAYS CLEARED WHEN A PAGE HAS 'RELOAD ON SUBMIT' SET TO 'ONLY FOR SUCCESS'

If an application uses an older theme (not the Universal Theme) and has a page that uses the 'Reload on Submit' attribute setting of 'Only for Success', then Application Express will use the new client-side messaging to display error messages without a full page reload. However, the label templates in older themes are not compliant with the new client-side messaging logic, which in turn results in some fallback logic being used to display the inline item errors, causing issues with clearing the message. The issue occurs if a page has more than one item error displayed, and then subsequently if only one of those errors is rectified before the next attempt to save, the inline item error message for the rectified error will not be removed as it should be. It is removed from the page error notification, but not from the inline item notification.

Note: Applications using older themes, generally have the page attribute 'Reload on Submit' set to 'Always' so will not face this issue. However, new pages added to existing applications will have the page attribute set to 'Reload on Submit' by default.

Workaround: You can set the page attribute 'Reload on Submit' back to 'Always', however, this will preclude you from being able to use an Interactive Grid on that page.

25305292 - CLIENT SIDE INLINE ERROR MESSAGE FOR RICH TEXT ITEM GIVES JAVASCRIPT ERROR

If you have a Rich Text Editor item that has validations on a page with page attribute Reload on Submit = 'Only for Success' then if there are validation errors to report there will be a JavaScript error that keeps the inline validation message from displaying. This may also affect reporting other errors on the page, and will keep the page from being submitted again.

Workaround: You can set the page attribute 'Reload on Submit' back to 'Always', however, this will preclude you from being able to use an Interactive Grid on that page.

N/A - THE TIMEZONE APIS AND CONSTANTS AVAILABLE IN APEX_JSON SHOULD NOT BE USED

APEX_JSON contains new, undocumented APIs and constants for reading and writing TIMESTAMP, TIMESTAMP WITH TIME ZONE and TIMESTAMP WITH LOCAL TIME ZONE values. These APIs and constants are subject to change in later releases, which will likely break any code that utilizes them at this point in time.

Workaround: Do not use the undocumented timezone APIs and constants until they have been documented.

Changed Behavior:

25291599 - REGIONS ON GLOBAL PAGES CAN NOW PARTICIPATE IN PAGE-LEVEL REGION DISPLAY SELECTORS

Prior to 5.1, the 'Region Display Selector' attribute of regions on global pages was effectively ignored. In 5.1, this attribute is now used to determine if the region from the global page is displayed in region display selector regions on other pages for the same user interface. The default setting for 'Region Display Selector' is 'Yes', therefore, existing pages that include a Region Display Selector are very likely to display additional regions defined on the global page after upgrading. If the global page region can not be visually

displayed, for example if it is purely CSS code, then the application will appear broken to end users as clicking on this new selection will result in a blank display.

Workaround: Set the 'Region Display Selector' attribute to 'No' for all regions on the global page that you do not want displayed in region display selectors on other pages.

N/A - CLIENT-SIDE MESSAGING ISSUES IN APPS USING OLD, OR CUSTOM THEMES

If you create a new application using an old, or custom theme, or add new pages to an existing application which is not using the Universal Theme, then you may encounter minor differences in how inline errors are displayed. This can affect you if either:

1. The page attribute 'Reload on Submit' is set to 'Only for Success' (the default for new pages) and an item validation occurs.
2. The application 'Compatibility Mode' is set to '5.1' and the submit button is set to 'Execute Validations' = 'Yes', and any items have client-side validation such as 'Value Required' = 'Yes', which fail

An inline error is shown but it may not look exactly the same as on other existing pages because it isn't using the templates from the theme.

Workaround: These differences can be avoided by setting 'Reload on Submit' to 'Always', (Note: Interactive Grid requires this to be set to 'Only for Success') and not using client-side validation by either setting application compatibility mode < '5.1' or Setting Execute Validations = No. The differences can also be avoided by updating the old theme to use the new style of error templates, or by updating the application to use Universal Theme. For further details please review the [APEX 5.1 - Client side messaging and apps not using Universal Theme](#) blog post.

25247070 - TIE SWITCH TO ASYNCHRONOUS DYNAMIC ACTIONS TO COMPATIBILITY MODE 5.1

In Application Express 5.1 the compatibility mode controls how Ajax-based Dynamic Actions operate, specifically a setting of '5.1' will make all Ajax calls asynchronous. Prior to 5.1, Ajax-based Dynamic Actions that also had the 'Wait for Result' attribute set to 'Yes' would issue synchronous Ajax calls. Synchronous calls can have a negative impact by blocking all page interactions until the call has been completed. Dynamic Actions with 'Wait for Result' set to 'No' previously performed asynchronous Ajax calls, therefore, there is no change in behavior for these actions.

This change has been tied to compatibility mode to protect against possible regressions in existing applications, where applications may have relied upon the synchronous Dynamic Actions nature for certain functionality.

Workaround: Remove the reliance on synchronous-based Ajax calls before upgrading the compatibility mode for the application. For more information please see [Sync or Async? Dynamic Actions in APEX 5.1](#).

You may Find the Original Source here : [know bugs oracle 5.1](#)

By [Pranav Shah](#) - [August 28, 2017](#)

No comments:



Labels: [oracle apex 5.1](#), [pl/sql](#)

Tuesday, 25 April 2017

Tabular Form With Ajax specially for beginners

Sometime it's necessary to use tabular in oracle application express with ajax considering new comer who wants to explore more as they join in this article might help them to keep it going.

Steps to follow :

- 1) Create a tabular form based on table and save and run you're page you can see page is properly created and do not show any errors.
- 2) Now if you inspect you're tabular form checkbox (or cells) it will show name="f01" and Id = "f01_0001" just have a look at it for every cell you have on you're page.
- 3) Edit you're page and add Page Process(pl-sql) and target point will be Ajax callback lets add some pl/sql block in it

```
declare
```

```
l_id NUMBER;
l_json_str varchar2(500);
begin
select id
into
l_id
from my_table where column = apex_application.g_x01;

--x01 is what we gonna send from js code on some event like click,change,onselect
```

```
and many more...
l_json_str := '{"id":"' || l_id || '"}';
sys.http.p(l_json_str);
```

```
--press f12 network response
```

and you will see what is the response we are getting if there are some errors it will also show.

```
end;
```

If you have observed properly we are sending some values from javascript code which is used as filter in where clause. Lets assume we wanted to show those data whose id = 100 then we are sending 100 from js (x01) and based on that ajax callback function fires and show data.

Debug Lets say we wanted to check what is going on under the hood !!!

Press f12 in chrome and click on network.

Now lets say we have select list on page we changed the value of select list and get the id of value we have just selected that id will be passed as x01 to ajaxcall back.

- 4) Lets took a look at javascript code :

//Tabular form will have on lov cell based on change of it ajaxcall will fire. also add onchang="get_item_details(pThis)" in lov's html attribute

```
function get_item_details(pThis){
  var item_id = pThis.value;
  var row_id = pThis.id.substr(4);
  var my_id = $('#f02_'+row_id);
  apex.server.process ("ajax_callback_process_name",
  {
    x01:my_id
  },
```

```
{
  type: 'GET',
  dataType: 'json',
  success: function(l_json_str)
  {
    $(apex_item).val(l_json_str.uom_id);
  }
});
}
```

5) We need to make sure on which action we need to fire this action that is the most important thing to remember which is already shown above.

Disclaimer

Oracle application express 5.0+

Javascript

Jquery

Oracle database 11g

By [Pranav Shah](#) - [April 25, 2017](#) No comments:



Labels: [javascript](#), [jquery](#), [oracle apex 5.0](#)

[Home](#)

[Older Posts](#)

Subscribe to: [Posts \(Atom\)](#)

Show values in right side of shuttle

While working with select list and shuttle, when we want to display values into right side of shuttle depending upon selection from select ...

▼

List of Values

TypeSQL Query▼

SQL Query

✎

```
SELECT
  ename,
  empno
FROM emp
WHERE
  deptno = :p1_dept --p1_dept is page item
```

Display Extra Values

YesNo

Cascading LOV Parent Item(s)P1_DEPT

^

Items to Submit

^

Optimize Refresh

YesNo

[How to change logo in apex oracle?](#)

Change logo(image) in apex oracle 5.0(to display in every pages of application) Application(Working)->Shared components -> Static...



Remove X from Modal Page

Sometimes our clients have some good requirements where by we need to do some stuff according to their requirements as we all know "C...



Close Dialog Window and Refresh Parent Page:

Close Dialog Window and Refresh Parent Page: As per client need our team need to create some functionality which would be very interestin...

Pranav Shah . Simple theme. Powered by [Blogger](#).