

Great APEX Grids: Editable Essentials

Karen Cannell

kcannell@thtechnology.com

TH TECHNOLOGY

@thtechnology



hroug '22

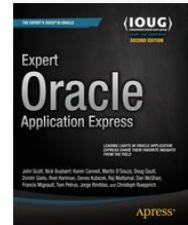
Autumn | Jesen

11. - 14.10.2022., Rovinj



About Me ...

- TH TECHNOLOGY – Oracle Consulting Services, APEX Focus
- Mechanical/SW Engineer - Analyzed, designed, developed, converted, upgraded, enhanced legacy & Oracle database applications for 30+ Years
- Web/APEX applications for Govt, Medical, Engineering, Fisheries since HTMLDB
- ODTUG Vice President, Kscope Conference Chair, Editor Emeritus, Technical Journal
- APress Author
- Oracle ACE Director 





500+ technical experts helping peers globally

The **Oracle ACE Program** recognizes and rewards community members for their technical and community contributions to the Oracle community



3 membership tiers



For more details on Oracle ACE Program:
ace.oracle.com



Nominate
yourself or someone you know:

ace.oracle.com/nominate

Connect:  aceprogram_ww@oracle.com

 Facebook.com/OracleACEs

 [@oracleace](https://twitter.com/oracleace)







#StaySafe
#StayVigilant

#Vaccinate ~ #Boost
#stayhealthy

ODTUG
Kscope23
aurora, co june 25-29



Kscope23.odtug.com


GAYLORD ROCKIES
RESORT & CONVENTION CENTER
Colorado

About You ...

- APEX Experience?
- APEX Versions?
- Grid Strategy?
- JavaScript Experts?



For Demo Application:

kcannell@thtechnology.com
@thtechnology

Editable Grid Essentials

- Default is Not Enough
- Edit Processing
- Grid Events
- Validations
 - Submitted Rows vs Validate All
- Toolbar Configurations
 - Global Config File

STANDARDIZE







Know Your Tool Use Your Tool

- Select -

▶ This month, is the primary site where you dock or launch this vessel still located at CAPE ANN MARINA in , MA?

▶ During the last week (Monday through Sunday), how many saltwater fishing trips targeting finfish did the TUNA HUNTER take? Please remember to include ALL trips, not just those where you fished for large pelagic species.

▶ How many of these trips were with paying customers?

▶ How many of these trips consisted of more than one day of fishing?

▶ On how many of these trips did you target large pelagic species such as tunas, sharks, billfish, dolphin, wahoo, amberjack or similar offshore species?

▶ During the last week (Monday through Sunday) how many additional non-fishing trips did your boat make? Please include any trips taken for fuel, bait or other recreational activities.

▶ On what days did each of the saltwater fishing trips taken by the TUNA HUNTER occur? (Multiday Trips: Record date the trip ended)

▼ How many separate fishing trips did you take on each day?

Day	# Fishing Trips	# Non-Fishing Trips
02-Oct-2022 - Sunday	1	NA
30-Sep-2022 - Friday	1	NA
29-Sep-2022 - Thursday	1	NA

✓ Save # Trips/Day

× Cancel

Next >

Interactive Grid

(Interactive Report or Tabular Form)

+ <All the Features You Ever Wanted>

(Thank You APEX Team!)

Interactive Grid

IGr

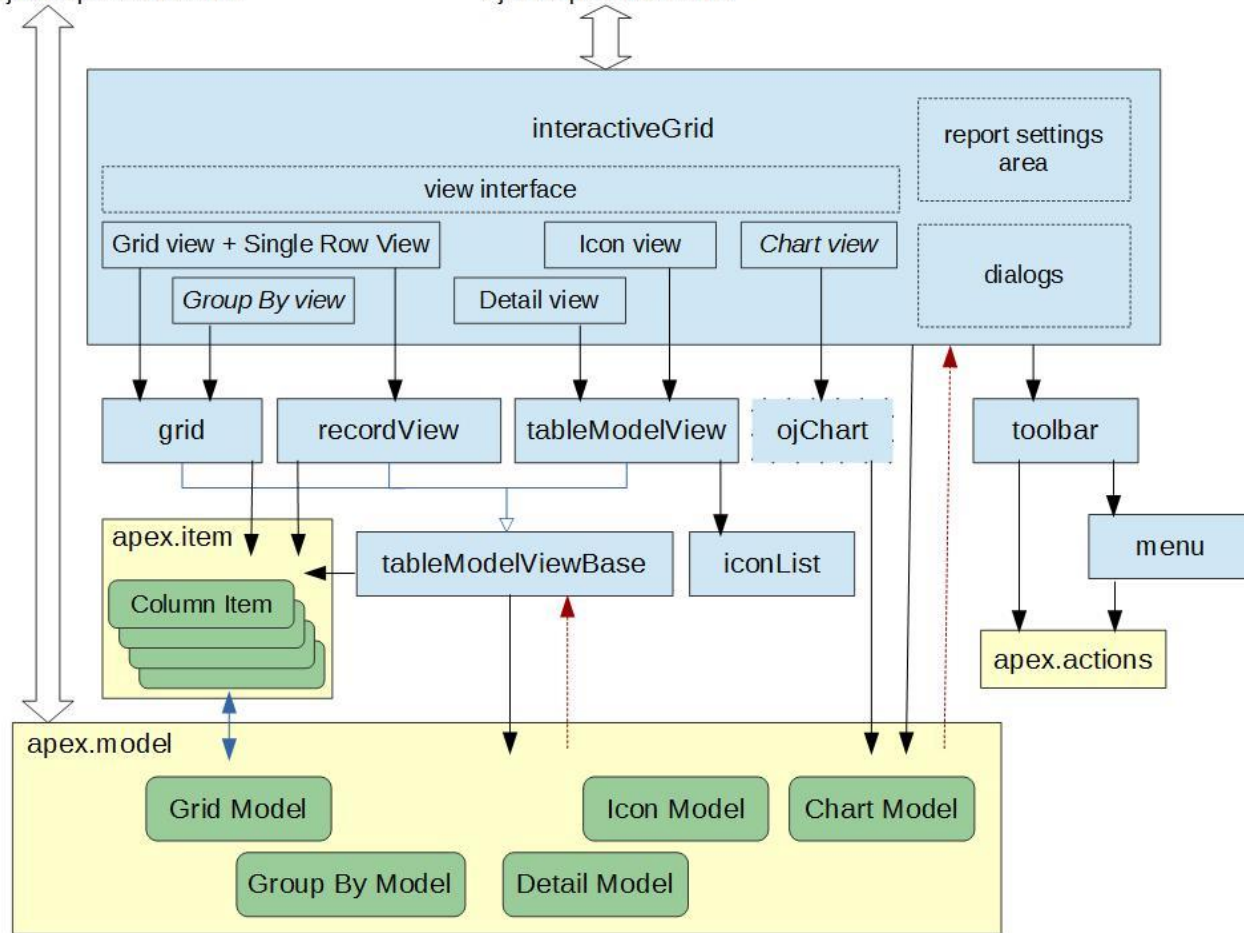
Inter

J Sny

<http://08/ir>

ajax requests to server

ajax requests to server



Widget

Module

inst of module

Topics ...

- [Basics](#) - Use Declarative Options First
- [Edit Processing](#)
- [Grid Events](#)
- Conditional Disable Cells, Always Edit
- [Validations](#)
 - [All Submitted Rows](#)
- [Custom Toolbar Config](#)
- [Standardize](#)

Basics

Background stuff you should
already be aware of

Interactive Report

- **Generates HTML on Server, part of Page**
- **Data Returned w UI Markup**
- **One Big JS Module**

Interactive Grid

- **HTML Renders on Client**
- **Data/Edits from Server via JSON**
- **Collection of Widgets**

Interactive Report

- **Collections
(pre 5.0)**
- **Browser Decides
Column Width**

```
table-layout: auto;
```

Interactive Grid

- **Model**
- **Each Column is an
"Item"**
- **Fixed Table
Layout**

E

Interactive Grid

Add Row

	Name	Job	Hiredate	Sal	Comm	Dept	Comm elig	Raise elig
<input type="checkbox"/>	WARD	Sales						
<input type="checkbox"/>	TURNER	Sales	09-JUN-81	2450	-	ACCOUNT...	No	Yes
<input checked="" type="checkbox"/>	ADAMS	Presi	23-JAN-82	1300	100	ACCOUNT...	Yes	No
<input type="checkbox"/>	KING	Presi	28-SEP-81	1250	1400	OPERATIO...	Yes	No
<input type="checkbox"/>	MENES		09-DEC-82	3000	-	RESEARCH	Yes	Yes
<input type="checkbox"/>	DETTIE		12-JAN-83	1100	-	RESEARCH	No	Yes
<input type="checkbox"/>	BLAKE		02-APR-81	2975	-	RESEARCH	No	Yes
<input type="checkbox"/>	BLAKE		03-DEC-81	950	-	SALES	Yes	Yes

Single Row View

+ Add Row

Duplicate Row

Delete Row

Refresh Row

Revert Changes

Edit

Save

Add Employee

Reset

Editable

4: Interactive Report


e-Rendering

Regions

Content Body

 Interactive Grid

▶  Columns

▼  Attributes


 Column Groups

 Interactive Report - Legacy

▶  Columns

▶  Attributes

 Interactive Report - Converted to I-C

 Filter Properties

▼ Edit

Enabled

Yes

No

Allowed
Operations

Add Row

Update Row

Delete Row

Allowed Row
Operations
Column

- Select -



Lost Update Type

Row Values



Add Row If Empty

Yes

No

▼ Edit Authorization

```
select EMPNO,  
       ENAME,  
       JOB,  
       MGR,
```

Type

Database Column

Database
Column

EDITABLE



Data Type

VARCHAR2



Query Only

Yes

No

Primary Key

Yes

No



Lost Update Type

- Row Values
 - Checksum of All Query-able Columns
- Column Value
 - ex: `DATE_UPDATED`

Lost Updates, Lock Row

- Lock Row Adds

FOR UPDATE NOWAIT

See Reference ~ Matt Mulvaney

<http://www.explorer.uk.com/apex-interactive-grid-preventing-lost-updates-record-locking/>

Who Can Edit (Authorizations)

Edit Authorization	
Add	- Select -
Update	- Select -
Delete	- Select - DEMO_USER IS_ADMIN Must Not Be Public User {Not DEMO_USER} {Not IS_ADMIN}
Performance	

If Editable

- Set a PK
- Set Reload on Submit
 - Only On Success



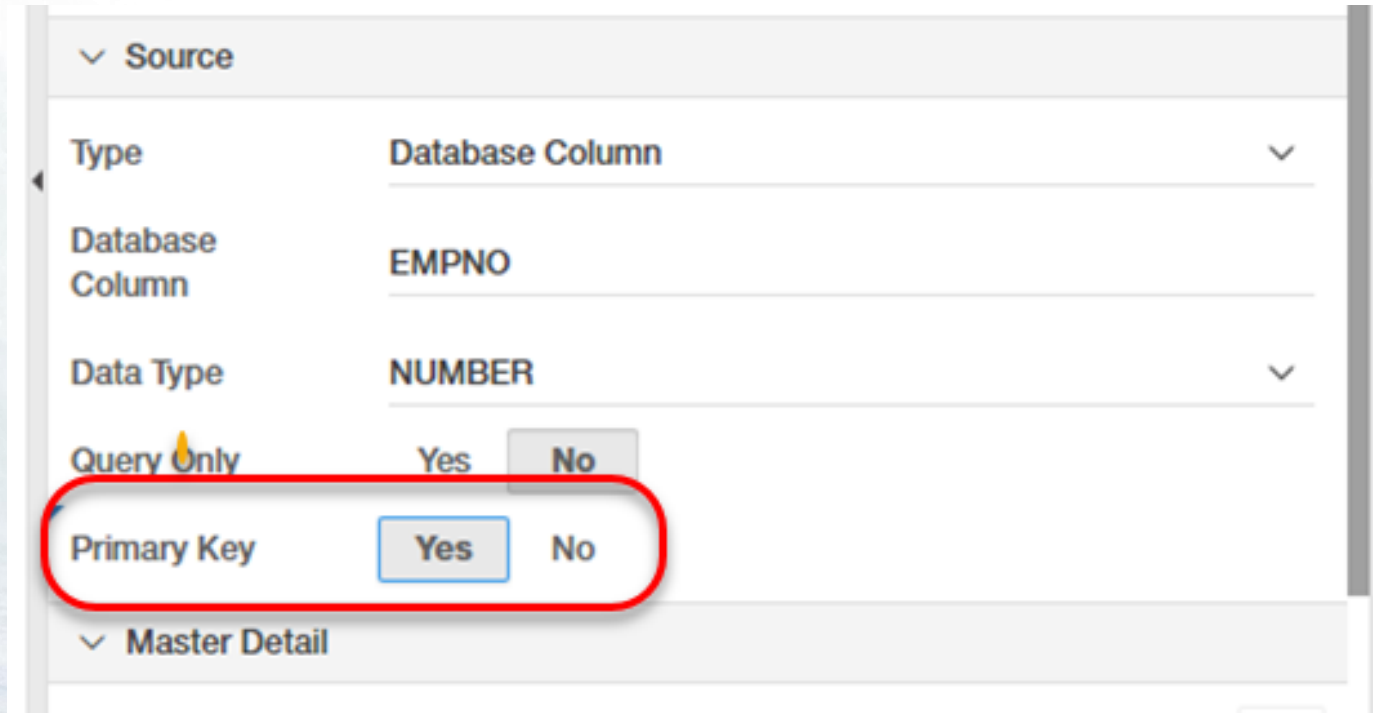
Interactive Grid 'E-Grid Casc LOV' doesn't have a primary key column defined which is required for editing or in a master detail relationship.

Contact your application administrator.

Technical Info (only visible for developers)

1. `is_internal_error`: true
2. `apex_error_code`: APEXDEV.IG.NO_PK
3. `component.type`:
APEX_APPLICATION_PAGE_REGIONS
4. `component.id`: 17111353091365601
5. `component.name`: E-Grid Casc LOV
6. `error_backtrace`:

Edit: Set One Column as a Primary Key



The screenshot shows a configuration window for a database column. The 'Source' section is expanded, showing the following details:

- Type: Database Column
- Database Column: EMPNO
- Data Type: NUMBER
- Query Only: Yes (selected), No
- Primary Key: Yes (checked), No

The 'Primary Key' section is highlighted with a red circle, indicating the current step in the process.



For an Editable Interactive Grid, the Page attribute 'Reload on Submit' must be set to 'Only for Success'.

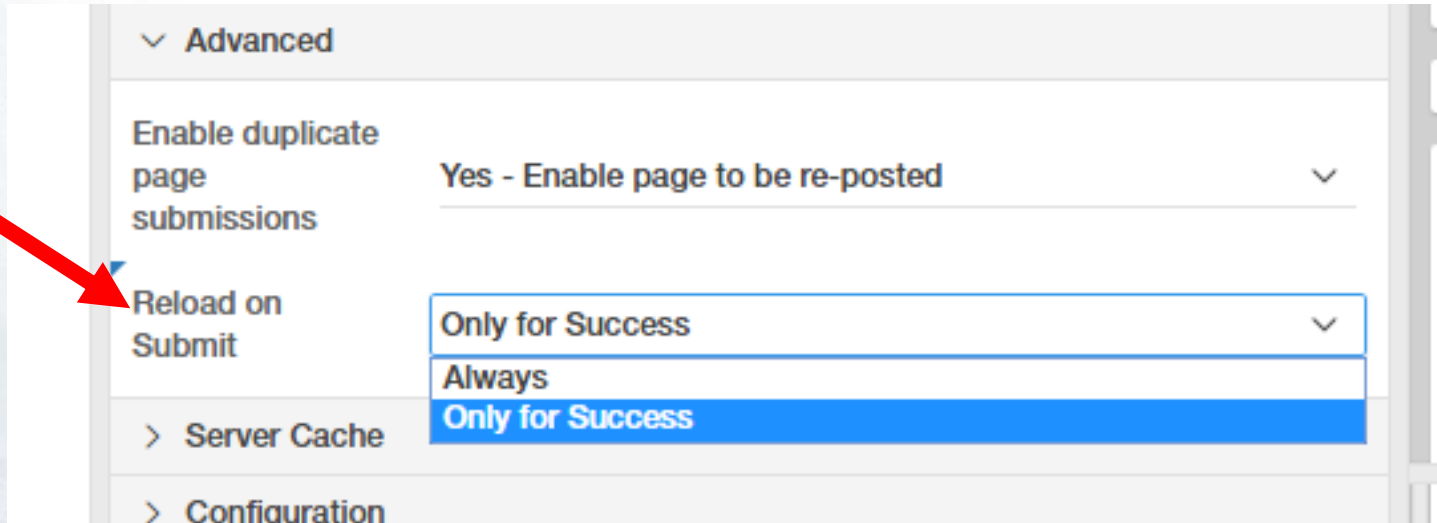
Contact your application administrator.

Technical Info (only visible for developers)

1. `is_internal_error`: true
2. `apex_error_code`:
APEXDEV.IG.RELOAD_ON_SUBMIT_EDITABLE
3. `component.type`:
APEX_APPLICATION_PAGE_REGIONS
4. `component.id`: 17111353091365601
5. `component.name`: E-Grid Casc LOV
6. `error_backtrace`:

```
----- PL/SQL Call Stack -----
   object      line object
   handle      number name
000007FFC480F540    935 package body APEX_050100.WWV_FLOW_ERROR
000007FFC480F540   1003 package body APEX_050100.WWV_FLOW_ERROR
000007FFC480F540   1395 package body APEX_050100.WWV_FLOW_ERROR
```

Edit: Page Reload on Submit



Advanced

Enable duplicate page submissions Yes - Enable page to be re-posted

Reload on Submit

- Only for Success
- Always
- Only for Success

Server Cache

Configuration

A red arrow points to the 'Reload on Submit' dropdown menu, which is currently open and showing three options: 'Only for Success', 'Always', and 'Only for Success'. The 'Only for Success' option at the bottom is highlighted in blue.

Row Selector

The screenshot shows the Oracle APEX Column Properties dialog for a column named APEX\$ROW_SELECTOR. The column is of type Row Selector. The settings are as follows:

Property	Value
Column Name	APEX\$ROW_SELECTOR
Type	Row Selector
Enable Multi Select	Yes
Show Select All	Yes
Hide Control	No

Three red arrows point to the 'Yes' buttons for 'Enable Multi Select', 'Show Select All', and 'No' for 'Hide Control'.

APEX\$ROW_SELECTOR – PL/SQL

- Same as Tab Form:

```
BEGIN
```

```
IF :APEX$ROW_SELECTOR = 'X' THEN
```

```
-- your processing goes here
```

```
DELETE FROM EMP WHERE EMPNO = :ID;
```

```
END IF;
```

```
END;
```


Working w the Row Selector

- JavaScript


```
menu$.menu("option").items.push({
  type:"action",
  label:"Total Salary",
  action: function(menu, element) {
    var i, records = view.getSelectedRecords(),
        total = 0;
    for ( i = 0; i < records.length; i++) {
      total += parseInt(view.model.getValue(records[i], "SAL"), 10);
    }
    alert("Total Salary for selected employees is " + total);
  }
});
```

Row Action Menu


▼  New Interactive Grid - Editable


▼  Columns

APEX\$ROW_SELECTOR

 APEX\$ROW_ACTION

 ROWID


 EMPNO

 ENAME


 JOB

 MGR

 HIREDATE

 SAL

 COMM

 DEPTNO

Add Row

▼ Layout

Sequence 20

▼ Advanced

Static ID

JavaScript Code

▼ Server-side Condition

Type - Select -

Master Detail

▼ Overview: Master Detail

There are wizards to help create master detail Interactive Grids but it is easy enough to set up manually was done for this example.

- Add the master Interactive Grid and the detail Interactive Grid.
- Configure the columns and set the primary key(s) for each and in the case of this example enable
- Then on region attributes for the detail grid set the Master Detail: Master Region attribute to the master table so set the Master Detail: Master Column to DEPTNO (the primary key of the master table).
- Make sure the detail grid toolbar save button is disabled.

• Disable Detail Grid Save Button

In General

- Each Column Is An Item
- Casc. LOV, Validations and Dynamic Actions Work the Same as for Page Items
- :COLUMN_NAME Bind Reference

The Same Stuff Applies!

Cascading LOV

- Just Like Page Item to Page Item

Pick Cascading LOV Parent Column(s)

Search

Name	Label
APEX\$ROW_SELECTOR	
APEX\$ROW_ACTION	
EMPNO	
ENAME	Ename
JOB	Job
MGR	Manager
HIREDATE	Hiredate
SAL	Sal
COMM	Comm
DEPTNO	Dept
COMM_ELIG	Comm?
BASE_ELIG	Base?

Type: List of Values

Shared Component: MANAGER_NAMES

Display Extra Values: Yes No

Display Null Value: Yes No

Null Display Value:

Null Return Value:

Cascading LOV Parent Column(s):

Link

Target:

Source

Type: Database Column

Database Column: MGR

Data Type: NUMBER

Query Only: Yes No


```

SELECT
  apex_item.checkbox (30,
    seq_id,
    'onclick="highlight_row(this, ' || seq_id || ')"',
    NULL,
    ':',
    'f30_' || LPAD (seq_id, 4, '0') ) delete_checkbox,
  apex_item.hidden (31, seq_id)
  || apex_item.select_list_from_query
    (32,
    c001,
    'SELECT dname d, ' || 'deptno r FROM
eba_demo_tf_dept',
    'style="width:220px" '
  || 'onchange="f_set_casc_sel_list_item(this, '
  || 'f33_'
  || LPAD (seq_id, 4, '0')
  || ')"',
  'YES',
  '0',
  '- Select Department -',
  'f32_' || LPAD (seq_id, 4, '0'),
  NULL,
  'NO' ) deptno,
  apex_item.select_list_from_query (33,
    c002,
    'SELECT ename d, '
  || 'empno r FROM eba_demo_tf_emp '
  || 'WHERE deptno = '
  || c001,
    'style="width:220px"',
    'YES',
    '0',
    '- Select Employee ('
  || (SELECT COUNT (*)
    FROM eba_demo_tf_emp
    WHERE deptno = c001)
  || ' ) -',
    'f33_' || LPAD (seq_id, 4, '0'),
    NULL,
    'NO' ) employee
FROM apex_collections
WHERE collection_name = 'DEPT_EMP'
UNION ALL

```

```

SELECT  apex_item.checkbox
        (30,
        NULL,
        'onclick="highlight_row(this, '
        || TO NUMBER (9900 + LEVEL)
        || ')"',
        NULL,
        ':',
        'f30_' || LPAD (9900 + LEVEL, 4, '0') ) delete_checkbox,
        apex_item.hidden (31, NULL)
        || apex_item.select_list_from_query
          (32,
          0,
          'SELECT dname d, ' || 'deptno r FROM eba_demo_tf_dept',
          'style="width:220px" '
        || 'onchange="f_set_casc_sel_list_item(this, '
        || 'f33_'
        || LPAD (9900 + LEVEL, 4, '0')
        || ')"',
        'YES',
        '0',
        '- Select Department -',
        'f32_' || LPAD (9900 + LEVEL, 4, '0'),
        NULL,
        'NO' ) deptno,
        apex_item.select_list_from_query
          (33,
          NULL,
          'SELECT ename d, '
          || 'empno r FROM eba_demo_tf_emp '
          || 'WHERE deptno = '
          || 0,
          'style="width:220px"',
          'YES',
          0,
          '- Select Employee (0) -',
          'f33_'
          || LPAD (9900 + LEVEL, 4, '0'),
          NULL,
          'NO'
          ) employee
FROM DUAL
WHERE :request = 'ADD_ROWS'
CONNECT BY LEVEL = 2

```

Before (Code for Tabular Form Cascading LOV)

After (Code for Grid Cascading LOV)

Dynamic Action on Grid Column

- Just Like On a Page Item
- Conditions/Action Reference Any Column In Row
- Cannot Reference Element in Other Rows

Edit Processing

Save Interactive Grid Data

Type Interactive Grid - Automatic Row Processing (DML) ▼ ☰

▼ Settings

Target Type Region Source
Table / View
PL/SQL Code

Prevent Lost Updates Yes No

Lock Row Yes ▼

Return Primary Key(s) after Insert Yes No

Target Type

Select which row processing technique is performed to save data changes made to the Interactive Grid.

Available options include:

Region Source

Inserts, updates, and deletes utilize the region source SQL Query .

Note - Ensure columns that are not insertable or updatable have **Query Only** set to Yes, to exclude them from processing.

Table / View

Inserts, updates, and deletes utilize a different table or view to the one defined in the region source SQL Query. For example, the SQL Query retrieves data from a table-join or non-updatable view and the DML processing needs to be performed against a single base table or updatable view.

Note - Ensure columns that are not insertable or updatable have **Query Only** set to Yes, to exclude them from processing.

PL/SQL Code

Processing is based on custom PL/SQL code. This option allows you to execute any PL/SQL code to implement any data manipulation in the underlying database.

This option should be utilized when using database packages, such as table APIs or calls to RESTful Web Services, to implement business logic.

Note - The PL/SQL code will be called for each row that has been inserted, updated or deleted.

Edit Processing/Target Type PL/SQL

Code Editor - PL/SQL Code to Insert/Update/Delete

```
1 BEGIN
2 CASE WHEN (:APEX$ROW_STATUS = 'C' ) THEN
3
4     INSERT INTO emp( empno, ename, deptno )
5     VALUES ( :EMPNO,:ENAME,:DEPTNO)
6     RETURNING empno INTO :EMPNO;
7
8     WHEN (:APEX$ROW_STATUS = 'U') THEN
9         UPDATE emp
10            SET ename = :ENAME,
11                deptno = :DEPTNO
12            WHERE empno = :EMPID;
13
14     WHEN (:APEX$ROW_STATUS = 'D') THEN
15         DELETE emp WHERE empno = :EMPNO;
16     END CASE;
17 END;
```

I

Edit Processing – Custom – Anything!

Code Editor - PL/SQL Code

```
1 begin
2   if not apex_collection.collection_exists( p_collection_name => 'LOG_EMPLOYEES' ) then
3     apex_collection.create_collection( p_collection_name => 'LOG_EMPLOYEES' );
4   end if;
5
6   apex_collection.add_member(
7     p_collection_name => 'LOG_EMPLOYEES',
8     p_d001 => sysdate,
9     p_c001 => :EMPNO,
10    p_c002 => :ENAME,
11    p_c003 => :JOB,
12    p_c004 => :HIREDATE,
13    p_c005 => :SAL,
14    p_c006 => :COMM,
15    p_c012 => :APEX$ROW_STATUS
16  );
17 end;
```

Grid Events

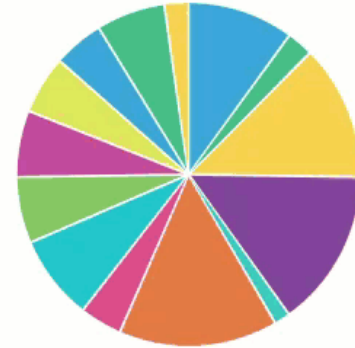
- Use Them!

The screenshot displays the configuration interface for APEX Grids, specifically the 'When' section. The 'Event' dropdown is set to 'Save [Interactive Grid]'. The 'Client-side' section is expanded, showing the 'Type' dropdown set to 'True'. The 'Advanced' section is also expanded, showing the 'Event Scope' dropdown set to 'True'. The 'Server-side' section is partially visible. A modal window is open, showing the configuration for the 'Refresh Chart on Grid Save Events' event. The modal is titled 'Save [Interactive Grid]' and has a dropdown menu with the following options: 'Refresh Chart on Grid Save Events' (highlighted in green), 'True', and 'False'. Under 'True', there are three sub-options: 'Show', 'Refresh', and 'Alert', each with a gear icon. Under 'False', there are no sub-options. The modal also shows the 'Row Initialization [Interactive Grid]' section, which is expanded to show the 'Show Directions Hide Chart' event, which is set to 'True' and has a 'Show' sub-option with a gear icon.

Grid Events

<input type="text" value="Search: All Text Columns"/> <input type="button" value="Go"/> <input type="button" value="Actions"/> <input type="button" value="Edit"/> <input type="button" value="Save"/> <input type="button" value="Reset"/> <input type="button" value="APEX World Help"/>									
<input type="button" value="Add Row"/>									
<input type="checkbox"/>	<input type="checkbox"/>	Empno	Ename	Job	Mgr	Hiredate	Sal	Comm	Dept
<input type="checkbox"/>	<input type="checkbox"/>	16734	DENTON	ANALYST	BLAKE	12-NOV-81	2000	222	SALES
<input type="checkbox"/>	<input type="checkbox"/>	7839	ALLEN	SALESMAN	JONES	17-NOV-81	501	1	SALES
<input type="checkbox"/>	<input type="checkbox"/>	7698	BLAKE	MANAGER	BLAKE	01-MAY-81	2555	5	SALES
<input type="checkbox"/>	<input type="checkbox"/>	7782	CLARKSON	ANALYST	ALLEN	09-JUN-81	1	-	ACCOUNTING
<input type="checkbox"/>	<input type="checkbox"/>	7566	JONES	MANAGER	ALLEN	02-APR-81	2975	-	RESEARCH
<input type="checkbox"/>	<input type="checkbox"/>	7788	SCOTT	ANALYST	JONES	09-DEC-82	300	-	RESEARCH
<input type="checkbox"/>	<input type="checkbox"/>	7902	TURNER	SALESMAN	JONES	03-DEC-81	3000	300	SALES
<input checked="" type="checkbox"/>	<input type="checkbox"/>	7369	SMITH	SALESMAN	BLAKE	17-DEC-80	800	80	OPERATIONS
<input type="checkbox"/>	<input type="checkbox"/>	7499	SMITHSO...	CLERK	JONES	20-FEB-81	1600	300	OPERATIONS

Pie Chart



Validations

**Server-side, Client-side, HTML5,
be aware of what fires when, and
on what rows**

Validation Strategy ~ Have One!

- Be Consistent
- Declarative Options First
 - Use Conditions
 - Use Attributes
- Data Model, Server Side, Client Side
 - Cover All the Bases

Validations on Grid Column

- Just Like On a Page Item
- :COLUMN_NAME References
- HTML5 Form Validation
 - data_valid_message
 - Required, Min, Max (Numbers)
 - Pattern
 - In Custom Attributes
- Dynamic Actions
 - Column Option

HTML5 Grid Validation – Ename

- EName Must Be Alpha

Static ID

CSS Classes

Custom Attributes

`pattern="^[A-Z a-z]*$" data-valid-message="Employee name is required, and must be all alpha`

Use As Row
Header



- Custom Message

Server-Side Grid Validation

Salary Between
100 and 2000

The screenshot shows the configuration for a server-side grid validation. It includes a 'Validation' section with a dropdown for 'Validations' and a 'PL/SQL Expression' field containing the text ':SAL BETWEEN 100 AND 2000'. Below this is an 'Always Execute' toggle switch. An 'Error' section is visible, with an 'Error Message' field containing the text '#COLUMN_HEADER# must be between 100 and 2000.'. At the bottom, there are dropdowns for 'Display Location' (set to 'Inline with Field and in No') and 'Associated Column' (set to 'SAL').

Validation

Editable Region Validations

Type PL/SQL Expression

PL/SQL Expression

:SAL BETWEEN 100 AND 2000

Always Execute

Error

Error Message

#COLUMN_HEADER# must be between 100 and 2000.

Display Location Inline with Field and in No

Associated Column SAL

Client-Side Validations - Use the Documented JS APIs

- apex.oracle.com/jsapi → JavaScript APIs
 - `apex.item` and `item`
 - `apex.message`
 - `apex.model`
 - Interactive Grid APIs

Note on JS API Syntax as of APEX 21.2

- As of APEX 21.2, Moving Away from JQuery UI Widget Syntax to Simpler, Intuitive JS API Syntax...

Future Direction

You may be wondering what will happen to the existing widget style APIs such as `grid` and `menu` that we have already documented. They are not deprecated and will likely remain supported for a long time. There may be a time when essentially the same functionality is available as a widget style API and a region interface style API. (You can still use the jQuery UI style API with the `mapRegion` or `facetedSearch` region as long as the method and property names are the same. This is good for backward compatibility but I expect everyone will want to switch to the newer, friendlier, documented API.)

Note on JS API Syntax as of APEX 21.2

If you wanted to get the currently selected records from IG you would do this:

```
var records = apex.region("regionid").call("getSelectedRecords");
```

Or using the older and more verbose syntax:

```
var records = apex.region("regionid").widget().interactiveGrid("getSelectedRecords");
```


Client-Side Validations - Validity JS APIs

- `apex.page.validate`
- `apex.page.submit({validate:true;});`
- `apex.page.confirm({validate:true;})`

- `apex.item getValidity`
- `apex.item.setCustomValidity`
- `apex.message`

Client-Side Grid Validation - Item

- Dynamic Action
 - On Change

The screenshot shows the 'Dynamic Actions' section of the 'SAL' item properties. A dynamic action named 'Client - Dynamic - Validate Salary' is selected. It is configured to trigger on the 'True' condition. The action is set to 'Execute JavaScript Code'.

- ✓ SAL
 - ✓ Validations
 - ✓ Validate Salary
 - ✓ Dynamic Actions
 - ✓ Client - Dynamic - Validate Salary
 - ✓ True
 - Execute JavaScript Code
 - False

The screenshot shows the 'When' configuration for the dynamic action. The event is set to 'Change', the selection type is 'Column(s)', and the region is 'Validations'. The column(s) to be validated is 'SAL'.

Sequence	10
When	
Event	Change
Selection Type	Column(s)
Region	Validations
Column(s)	SAL

Grid Validation - Salary

- Client-Side Salary Betw 100 and 3000

Code Editor - Code


```
1  var sal = apex.item('salary'),
2      num = sal.getNativeValue();
3  if ( num !== "" && ( num < 100 || num > 3000)) {
4      // this msg only used if there is no HTML data-valid-message attribute on the column
5      sal.node.setCustomValidity("Salary must be between 100 and 3000");
6  } else {
7      sal.node.setCustomValidity(""); // clear the error
8  }
```

Grid Validation – Dyn Action Salary

- Salary Betw 100 and 3000

Code Editor - Code

```
1  var sal = apex.item('salary'),
2      num = sal.getNativeValue();
3  if ( num !== "" && ( num < 100 || num > 3000)) {
4      // this msg only used if there is no HTML data-valid-message attribute on the column
5      sal.node.setCustomValidity("Salary must be between 100 and 3000");
6  } else {
7      sal.node.setCustomValidity(""); // clear the error
8  }
```

`getNativeValue()` → {number} 

Return the current value of the item as a JavaScript number. If the value is not a valid number returns NaN. This is useful when working with numbers because the `item#getValue` method always returns a string.

Returns:

Type
number

Example

In this example, page items P1_COST and P1_TAX are added together and then formatted and displayed in an alert dialog

```
var total = apex.item( "P1_COST" ).getNativeValue() + apex.item( "P1_TAX" ).getNativeValue();
apex.message.alert( "Total is: " + apex.locale.formatNumber( total, "L999G999G99" ) );
```

Grid Validation - Comm

- Server-Side - Row

At Least 10% of Salary
and Not > Salary

The screenshot shows the configuration for a grid validation rule. The 'Validation' section is active, showing the 'PL/SQL Expression' as `TO_NUMBER(:COMM) BETWEEN 0.1*TO_NUMBER(:SAL) AND TO_NUMBER(:SAL)`. The 'Always Execute' toggle is turned off. The 'Error' section is also active, showing the 'Error Message' as `#COLUMN_HEADER# must be at least 10% of Salary, and cannot be greater than the employee's salary.`. The 'Display Location' is set to 'Inline with Field and in Notification' and the 'Associated Column' is 'COMM'.

Validation

Editable Region Validations

Type PL/SQL Expression

PL/SQL Expression

```
TO_NUMBER(:COMM) BETWEEN 0.1*TO_NUMBER(:SAL) AND  
TO_NUMBER(:SAL)
```

Always Execute

Error

Error Message

```
#COLUMN_HEADER# must be at least 10% of Salary, and cannot  
be greater than the employee's salary.
```

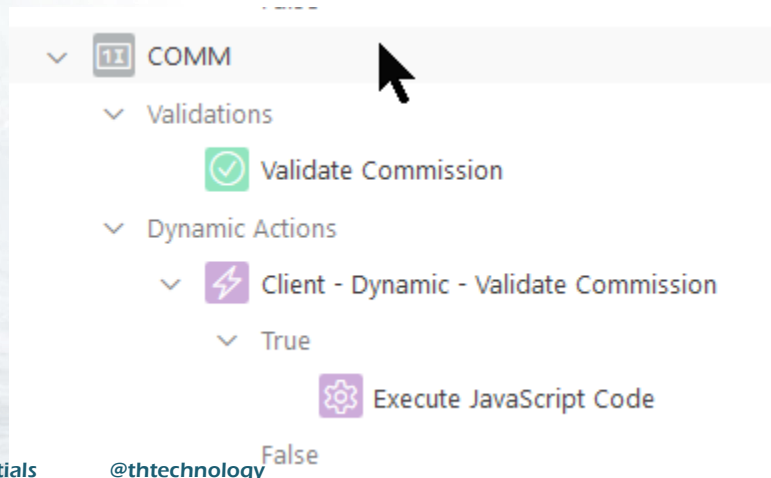
Display Location Inline with Field and in Notification

Associated Column COMM

10/27/2022 64

Grid Validation – Client Side -Row

- Dynamic Action
 - On Change
 - Execute JavaScript



Grid Validation – DA - Commission

- 10% Sal < Commission < Salary

```
1 var comm = apex.item("commission"),
2     sal = apex.item("salary"),
3     numComm = comm.getValue(), // same as $v("static id")
4     numSal = sal.getValue();
5
6 if ( numComm !== "" &&
7     (parseFloat(numComm) !== numComm || numComm < 0.1*numSal || numComm > numSal)) {
8     // this message used iff no data-valid-message attribute on the column item
9     comm.node.setCustomValidity("Commission must be at least 10% of Salary, and cannot be more than
10 } else {
11     comm.node.setCustomValidity(""); // clear the error
12 }
```

Grid Validation – DA - Commission

- $10\% \text{ Sal} < \text{Commission} < \text{Salary}$

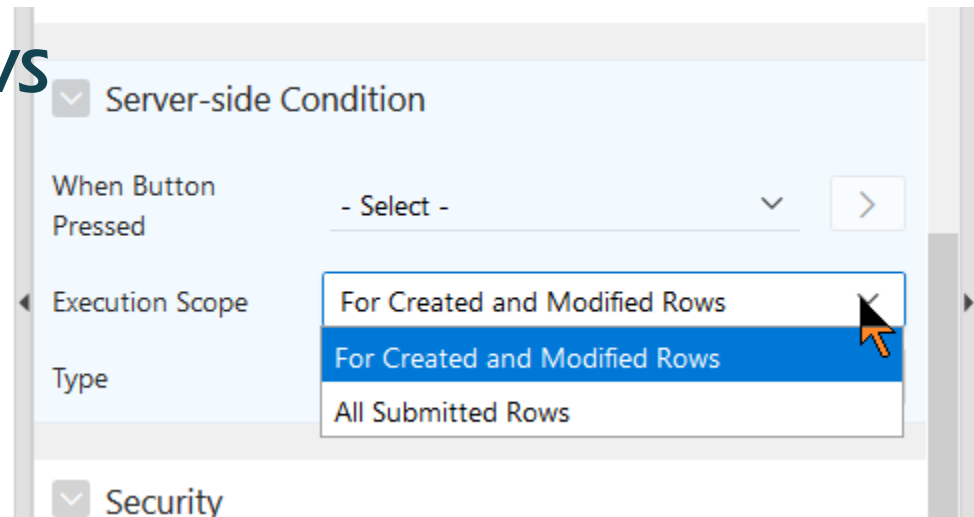
```
1 var comm = apex.item("commission"),
2     sal = apex.item("salary"),
3     numComm = comm.getNativeValue(), // get native JS number, the parseFloat stuff not needed
4     numSal = sal.getNativeValue();
5 //alert( numComm );
6 //alert ( 0.1*numSal);
7 if ( numComm !== "" && (numComm < 0.1*numSal )) {
8     // (parseFloat(numComm) !== numComm || numComm < 0.1*numSal )) {
9     // this message used iff no data-valid-message attribute on the column item
10    comm.node.setCustomValidity("Commission must be at least 10% of Salary.");
11 } else {
12    comm.node.setCustomValidity(""); // clear the error
13 }
```

Why Did My Grid Validation Not Fire?

- Grid Validation Settings
 - Created and Modified Rows
 - All Submitted Rows

“Submitted Rows”

... Not All Rows
Get Submitted!



Validations

Grids Are Designed to Validate Submitted Rows

What Is Your Use Case?

- Do you really want validations to fire on *All* rows?
- How Many is *All*?

<https://hardlikesoftware.com/weblog/2022/04/25/interactive-grid-validation/>

All Submitted Rows ?

- Do you really **NEED** validations to fire on *all* rows?
- “Touch” Each Row
 - Add a Dummy Column, Touch, Hide That
- Right Before the “Save” – Best Time ?

Always Edit

- Eliminate the Edit Button and/or Double-Click in the Grid
- Not Necessary All the Time!

```
ig$.interactiveGrid("getActions").set("edit", true);
```


In Page Function and Global Variable Declaration :

```
(function($) {  
  $(function() {  
    $("#IG_STATIC_ID").on("interactivegridviewchange", function(event, data) {  
      if ( data.view === "grid" && data.created ) {  
        setTimeout(function() {  
          var ig$ = apex.region("IG_STATIC_ID").widget();  
          ig$.interactiveGrid("getActions").set("edit", true);  
        }, 1);  
      }  
    });  
  });  
})(apex.jQuery);
```

Replace **IG_STATIC_ID** with your IG **static id**.

Grid Custom Configuration

**Customize the Toolbar, Row Actions, Actions
to suit your Business Requirements
Standardize for Consistency and Sanity**

Grid Customization

- Lots of Options
- JavaScript ~ Documented JS APIs
- Configure What YOU Need
- Standardize ~ Make it Easy on Yourself

Tweak Toolbar Buttons/Names

- **Guide to Simple Button Settings:**
 - `button.iconOnly = true/false;`
 - `button.hot = true/false;`
 - `button.label = "Your Label";`
 - `button.title="Button tooltip goes here";`
 - `button.iconBeforeLabel = true/false;`

Code Editor - JavaScript Initialization Code



```
1 function(config) {  
2     let $ = apex.jQuery,  
3         toolbarData = $.apex.interactiveGrid.copyDefaultToolbar(),  
4         saveAction = toolbarData.toolbarFind("save"),  
5         group = toolbarData.toolbarFind("actions2");  
6     // Edit is the 1st control in the actions2 group, remove it.  
7     group.controls.shift();  
8  
9     // rename the Save button  
10    saveAction.label = "Apply Changes";  
11    saveAction.iconBeforeLabel = true;  
12    saveAction.icon = "icon-check";  
13    saveAction.hot = true;  
14    saveAction.title = "Apply Changes to the database. Your tooltip message goes here."  
15  
16    config.toolbarData = toolbarData;  
17    return config;  
18 }
```

Change Toolbar Buttons

Delete Button Toolbar

Search: All Text Columns		Go	+ Add Employee		Delete Selected Rows		Save	Refresh/Cancel				
<input type="checkbox"/>		Empno	Ename	Job	Mgr	Hiredate	Sal	Comm	Dept	Comm Elig	Raise Elig	Riff El
<input checked="" type="checkbox"/>		16734	DENTON	ANALYST	BLAKE	12-NOV-81	2000	-	SALES	Yes	No	No
<input type="checkbox"/>		7839	ALLEN	SALESMAN	BLAKE	17-NOV-81	5001	-	SALES	Yes	Yes	No
<input type="checkbox"/>		7698	BLAKE	MANAGER	ALLEN	01-MAY-81	2851	-	SALES	No	Yes	No
<input type="checkbox"/>		7782	CLARK-SMITH	MANAGER	ALLEN	09-JUN-81	2450	-	ACCOUNTING	No	Yes	No
<input type="checkbox"/>		7566	JONES	MANAGER	ALLEN	02-APR-81	2975	-	RESEARCH	No	Yes	No
<input type="checkbox"/>		7788	SCOTT	ANALYST	JONES	09-DEC-82	300	-	RESEARCH	Yes	Yes	Yes
<input type="checkbox"/>		7902	TURNER	SALESMAN	JONES	03-DEC-81	3000	-	SALES	No	Yes	Yes
<input type="checkbox"/>		7369	SMITH	SALESMAN	TURNER	17-DEC-80	800	5	OPERATIONS	No	Yes	No
<input type="checkbox"/>		7499	SMITH-CLARK	CLERK	BLAKE	20-FEB-81	1600	300	SALES	Yes	Yes	Yes
<input type="checkbox"/>		7521	WARD	SALESMAN	BLAKE	20-FEB-81	1250	500	SALES	Yes	Yes	Yes

Search: All Text Columns

Go

Filter

Sort

Apply Changes

Refresh (Cancel)

State	Year	We...	Wave	FHTS ID	Vsl Num	Vs ID	Vsl Name	Error Desc	Send to State?	ACCSP Comments	Confirm?	
09	2021	23	3	130504	1149901	0900560	TEASER	STATE CHANGE REQUEST	No	-	No	test mak not
09	2021	23	3	130504	1149901	0900560	TEASER	STATE CHANGE REQUEST	No	-	No	-
09	2021	23	3	130504	1149901	0900560	TEASER	STATE CHANGE REQUEST	No	-	No	test TRA rec

Standardize via “Global” Settings

- JS Function in JS File
- Include File
- Reference JS Function in Advanced
→ JavaScript Code
- Add Classes on Grids

Custom Toolbar

- Toolbar
 - Search, Search Btn
 - Add Employee
 - Delete Row
 - Save
 - Refresh/Cancel

```
1 function(config) {
2     config.toolbarData = [
3         {
4             groupTogether: true,
5             controls: [
6                 {
7                     type: "TEXT",
8                     id: "search_field",
9                     enterAction: "search"
10                },
11                {
12                    type: "BUTTON",
13                    action: "search"
14                }
15            ]
16        },
17        {
18            controls: [
19                {
20                    type: "BUTTON",
21                    action: "row-add-row",
22                    label: "Add Employee",
23                    iconBeforeLabel: true
24                }
25            ]
26        },
27        {
28            align: "end",
29            controls: [
30                {
31                    type: "BUTTON",
32                    label: "Delete Selected Rows",
33                    action: "selection-delete",
34                    iconBeforeLabel: true
35                },
36                {
37                    type: "BUTTON",
38                    action: "save",
39                    hot: "true",
40                    iconBeforeLabel: true
41                },
42                {
43                    type: "BUTTON",
44                    label: "Refresh/Cancel",
45                    action: "refresh",
46                    iconBeforeLabel: true
47                },
48                {
49                    type: "BUTTON",
```

Standardize !

- Consistent Across Grids, Across Apps
- Use Common IG JS Config File
- Apply Classes to Grid
- Simple, Consistent
- Apply JS Best Practices (Files, MIN)





```
((function($){  
window.thtechGridApp = {  
  commonIGConfig: function(config) {  
    config.defaultGridViewOptions = {  
      stickyFooter: false  
    };  
    if ($("#" + config.regionStaticId ).hasClass("seqNbr")) {  
      config.defaultGridViewOptions.rowHeader = "sequence";  
    }  
    if ( config.pagination.type === "set" ) {  
      config.defaultGridViewOptions.pagination = {  
        firstAndLastButtons: false,  
        maxLinks: 12  
      };  
    }  
    if ( config.pagination.type === "scroll" ) {  
      config.defaultGridViewOptions.pagination = {  
        showtotalRowCount: false  
      };  
    }  
  }  
});
```

JavaScript

Content Delivery Network

None (use Web Server)  

File URLs

#APP_IMAGES#THTech_My_Grid_Standards_KC2.js Include Deprecated or Disupported
Javascript Functions Pre 18.1  18.xInclude jQuery Migrate  

Global

- Add JavaScript

Appearance

Template: Standard

Template Options: Use Template Defaults

CSS Classes: `seqNbr filterOnly`

Icon:

Item Display Position: Above Content

Advanced

Oracle Template Column

Email From Address

JavaScript Initialization Code

```
thtechIGridApp.commonIGConfig
```



Putting it all Together ...

- Select -

- ▶ This month, is the primary site where you dock or launch this vessel still located at CAPE ANN MARINA in , MA?
- ▶ During the last week (Monday through Sunday), how many saltwater fishing trips targeting finfish did the TUNA HUNTER take? Please remember to include ALL trips, not just those where you fished for large pelagic species.
- ▶ How many of these trips were with paying customers?
- ▶ How many of these trips consisted of more than one day of fishing?
- ▶ On how many of these trips did you target large pelagic species such as tunas, sharks, billfish, dolphin, wahoo, amberjack or similar offshore species?
- ▶ During the last week (Monday through Sunday) how many additional non-fishing trips did your boat make? Please include any trips taken for fuel, bait or other recreational activities.
- ▶ On what days did each of the saltwater fishing trips taken by the TUNA HUNTER occur? (Multiday Trips: Record date the trip ended)

▼ How many separate fishing trips did you take on each day?

Day	# Fishing Trips	# Non-Fishing Trips
02-Oct-2022 - Sunday	1	NA
30-Sep-2022 - Friday	1	NA
29-Sep-2022 - Thursday	1	NA

✓ Save # Trips/Day



Introduction



FHHS Logsheet



LPS Logsheet



Follow-Up

FHHS-CATI Survey - Logsheet for TUNA HUNTER Week 39

Enter responses in every non-shaded column. The full question for each column displays in the Questions region on hover over the column.
 Click Save Logsheet to validate and save your responses.
 When done, click NEXT to advance to the next page.

Questions

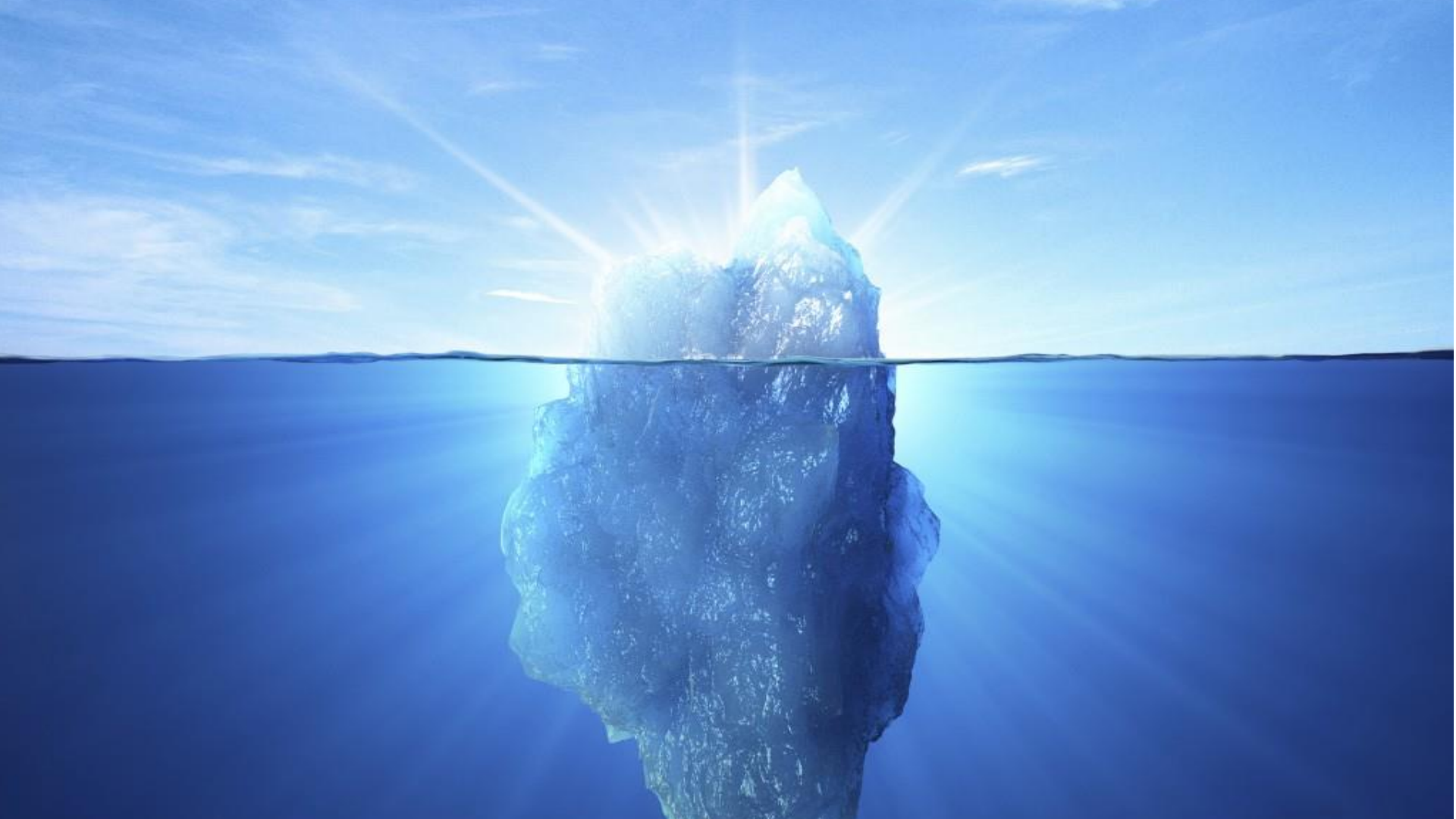


Was most of your fishing effort on that trip in the ocean, a sound, a river, or a bay?

	Trip Specifics						Site Trip Returned				Effort				
	For Hire?	LPS Trip?	Days Fished	Total Anglers	Trip Day	# Crew Fished	Trip State	Trip County	Trip Site	Trip Start Time	Trip End Time	Hours Fished	Method 1	Method 2	Fishin Area
1	Yes	Yes			02-Oct-2022 - Sunday	0	MASSACHUSETTS								
2	Yes	Yes			30-Sep-2022 - Friday	0	MASSACHUSETTS								
3	Yes				28-Sep-2022 - Wednesday		MASSACHUSETTS								

	Tournament		Fishing Specifics				Fishing Location				
	Tournament?	Tournament Name	Rod & Reel?	Other Gear	# Lines	Bait	Fishing Grounds	Miles	Depth (ft.)	Water Temp (°F)	
1								-	-		
2								-	-		
3								-	-		

Standardize
Stay Sane
Keep Your Users Happy



Challengework

- Sample Interactive Grid App
- IG Cookbook
<http://www.hardlikesoftware.com> get the latest
- IG Validations
<http://www.hardlikesoftware.com/weblog/2022/04/25/interactiv-e-grid-validation/>
- Read the JS Widget Code
`<apex_install_dir>/images/libraries/apex`

PLAY

Know Your Tool

Use Your Tool





Questions? Thank You

Karen Cannell

kcannell@thtechnology.com

[@thtechnology](#)

Email for Demo App & Slides



References

- [Hardlikesoftware.com](https://hardlikesoftware.com)
- IG Cookbook – latest version
- Grid Button Changes
<https://www.stinolez.com/2018/10/19/interactive-grids-buttons-customization/>
- Grid Validations – IG Validations
<https://hardlikesoftware.com/weblog/2022/04/25/interactive-grid-validation/>
- APEX 21.2 New JS Syntax
[https://hardlikesoftware.com/weblog/2021/10/11/apex-21-2-new-javascript-apis-joelkallmanday/APEX 21.1 New Features](https://hardlikesoftware.com/weblog/2021/10/11/apex-21-2-new-javascript-apis-joelkallmanday/APEX%2021.1%20New%20Features)
- Locks and Lost Update
<http://www.explorer.uk.com/apex-interactive-grid-preventing-lost-updates-record-locking>

Where to Put JS Files

- <https://www.talkapex.com/2017/01/how-to-reference-javascript-and-css-files-for-entire-application/js-storage.gif>

For Demo Application, Questions or Challenges:

kcannell@thtechnology.com
[@thtechnology](https://www.instagram.com/thtechnology)

`getNativeValue()` → {number} 📌

Return the current value of the item as a JavaScript number. If the value is not a valid number returns NaN. This is useful when working with numbers because the `item#getValue` method always returns a string.

Returns:

Type

number

Example

In this example, page items P1_COST and P1_TAX are added together and then formatted and displayed in an alert dialog

```
var total = apex.item( "P1_COST" ).getNativeValue() + apex.item( "P1_TAX" ).getNativeValue();
apex.message.alert( "Total is: " + apex.locale.formatNumber( total, "L999G999G99" );
```

Code Editor - Code

```
1 | var comm = apex.item("commission"),
2 |   sal = apex.item("salary"),
3 |   // use getNativeValue cuz getValue always returns a string ... we need nbrs here
4 |   numComm = comm.getNativeValue(),
5 |   numSal = sal.getNativeValue();
6 | //alert('Sal ' + numSal + ' Comm ' + numComm);
7 | if ( (numComm == "" ||
8 |     | (numComm < (0.1*numSal) || numSal < numComm) ) {
9 |     // this message only shown if there is no data-valid-msg
10 |    comm.node.setCustomValidity("Commission must be at least 10% of salary, and not > salary.");
11 | } else {
12 |    comm.node.setCustomValidity(""); // clear the error
13 | }
```


Code Editor - Code

```
1  var sal = apex.item('salary'),
2      num = sal.getNativeValue();
3  if ( num !== "" && ( num < 100 || num > 3000)) {
4      // this msg only used if there is no HTML data-valid-message attribute on the column
5      sal.node.setCustomValidity("Salary must be between 100 and 3000");
6  } else {
7      sal.node.setCustomValidity(""); // clear the error
8  }
```

I