TW Search Script

Description :

- In this concept we can retrieve data from process and task instances by using a JavaScript TWSearch object .
- Using TWSearch we can define which columns to retrieve, what filters to apply, how to sort and organize the results. You can also parse the results into a list of variables.
- To implement the custom portal we use TWSearch script .In order to get instance details,process details,task details ..etc.

Structure of a TWSearch query :

First we need to create TWSearch javascript object using below code.

```
var search = new TWSearch();
```

TWSearch query includes the following steps.

1.Define the columns of data that you want to retrieve.Each column that you want to return is defined as a TWSearchColumn object. The following code sample defines a column that returns the identifier of a process instance.

```
var colInstanceId = new TWSearchColumn();
colInstanceId.type = TWSearchColumn.Types.ProcessInstance;
colInstanceId.name = TWSearchColumn.ProcessInstanceColumns.ID;
```

2.Define any filters that you want to apply to a column. You can apply filters to any column by defining conditions. You define each condition as a TWSearchCondition object. For example, the following code sample defines a column that retrieves the name of a process instance and then narrows down the results to process instance names that start with Service Order Fulfillment.

```
var colInstanceName = new TWSearchColumn();
colInstanceName.type = TWSearchColumn.Types.ProcessInstance;
colInstanceName.name = TWSearchColumn.ProcessInstanceColumns.Name;

var condInstanceName = new TWSearchCondition;
condInstanceName.column = colInstanceName;
condInstanceName.operator = TWSearchCondition.Operations.StartsWith;
condInstanceName.value = "Account:";
```

3.Define the sort order for the returned records.You can define ordering criteria for any column. Each ordering is defined as a TWSearchOrdering object. For example, the following code sample lists the retrieved process instance identifiers in ascending order.

```
var orderInstanceId = new TWSearchOrdering();
orderInstanceId .column = colInstanceId;
orderInstanceId .order = TWSearchOrdering.Orders.Ascending;
```

4.Define how results are organized. You must also specify whether the results should be organized by process instance or by task. To do so, you use the TWSearch.organizedBy object. The behavior is the same as that of saved searches in Process Portal. If you choose to organize by process instance, you get only one result per process instance. The following code sample organizes search results by task.

```
search.organizedBy = TWSearch.OrganizeByTypes.Task;
```

5.Execute the search and parse the results into a list of complex variables.After you have specified the columns, conditions, ordering, and organization for the search, you can execute it to retrieve an array of JavaScript rows.

```
var results = search.execute();
```

The execute method returns TWSearchResults objects.

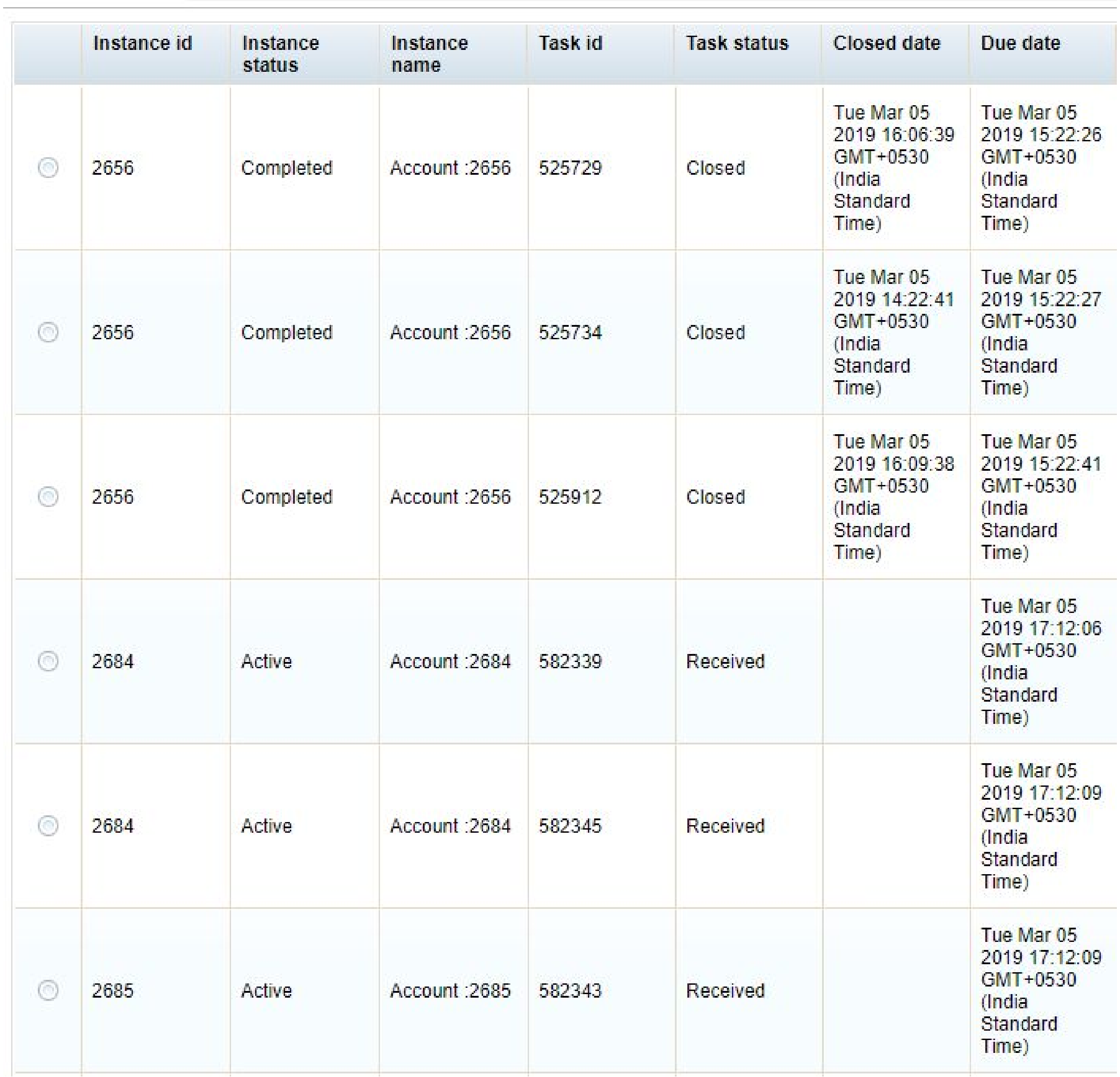
If you want to use these results outside of your script block, you must parse them and initialize equivalent variables.

The following code parses the native JavaScript array to create an array of custom variables.

```
tw.local.serviceOrderTasks = new tw.object.listOf.ServiceOrderTask();
 for(var i = 0; i < results.rows.length; i++) {
   var row = results.rows[i];
   tw.local.serviceOrderTasks[i] = new tw.object.ServiceOrderTask();

   tw.local.serviceOrderTasks[i].processInstanceId = row.values[0].toString();
   tw.local.serviceOrderTasks[i].processInstanceName = row.values[1].toString();
   tw.local.serviceOrderTasks[i].taskId = row.values[2].toString();
   tw.local.serviceOrderTasks[i].taskStatus = row.values[3].toString();
   if(row.values[4] != null)
   {
     tw.local.serviceOrderTasks[i].closedDate = row.values[4].toString();
   }
   if(row.values[5] != null)
   {
     tw.local.serviceOrderTasks[i].dueDate = row.values[5].toString();
   }
 }
```

When you run above script , you will get the following screen as shown below.



	Instance id	Instance status	Instance name	Task id	Task status	Closed date	Due date
○	2656	Completed	Account :2656	525729	Closed	Tue Mar 05 2019 16:06:39 GMT+0530 (India Standard Time)	Tue Mar 05 2019 15:22:26 GMT+0530 (India Standard Time)
○	2656	Completed	Account :2656	525734	Closed	Tue Mar 05 2019 14:22:41 GMT+0530 (India Standard Time)	Tue Mar 05 2019 15:22:27 GMT+0530 (India Standard Time)
○	2656	Completed	Account :2656	525912	Closed	Tue Mar 05 2019 16:09:38 GMT+0530 (India Standard Time)	Tue Mar 05 2019 15:22:41 GMT+0530 (India Standard Time)
○	2684	Active	Account :2684	582339	Received		Tue Mar 05 2019 17:12:06 GMT+0530 (India Standard Time)
○	2684	Active	Account :2684	582345	Received		Tue Mar 05 2019 17:12:09 GMT+0530 (India Standard Time)
○	2685	Active	Account :2685	582343	Received		Tue Mar 05 2019 17:12:09 GMT+0530 (India Standard Time)

Fig: TWSearch showing processInstances ,Task details.

Sample TWSearch script :

```
var search = new TWSearch();
var colInstanceId = new TWSearchColumn();
colInstanceId.type = TWSearchColumn.Types.ProcessInstance;
colInstanceId.name = TWSearchColumn.ProcessInstanceColumns.ID;

var colInstanceStatus = new TWSearchColumn();
```

```
colInstanceStatus.type = TWSearchColumn.Types.ProcessInstance;
colInstanceStatus.name = TWSearchColumn.ProcessInstanceColumns.Status;


var colInstanceName = new TWSearchColumn();
colInstanceName.type = TWSearchColumn.Types.ProcessInstance;
colInstanceName.name = TWSearchColumn.ProcessInstanceColumns.Name;

var colTaskId = new TWSearchColumn();
colTaskId.type = TWSearchColumn.Types.Task;
colTaskId.name = TWSearchColumn.TaskColumns.ID;

var colTaskStatus = new TWSearchColumn();
colTaskStatus.type = TWSearchColumn.Types.Task;
colTaskStatus.name = TWSearchColumn.TaskColumns.Status;

var colClosedDateTime = new TWSearchColumn();
colClosedDateTime.type = TWSearchColumn.Types.Task;
colClosedDateTime.name = TWSearchColumn.TaskColumns.ClosedDate;

var colDueDateTime = new TWSearchColumn();
colDueDateTime.type = TWSearchColumn.Types.Task;
colDueDateTime.name = TWSearchColumn.TaskColumns.DueDate;

search.columns=new
Array(colInstanceId,colInstanceStatus,colInstanceName,colTaskId,
colTaskStatus,colClosedDateTime,colDueDateTime);

var condInstanceName = new TWSearchCondition;
condInstanceName.column = colInstanceName;
condInstanceName.operator = TWSearchCondition.Operations.StartsWith;
condInstanceName.value = "Account :";

var conditions = new Array(condInstanceName);
search.conditions = conditions;

var orderInstanceId = new TWSearchOrdering();
orderInstanceId .column = colInstanceId;
orderInstanceId .order = TWSearchOrdering.Orders.Ascending;
```

```
search.orderBy = new Array(orderInstanceId);

search.organizedBy = TWSearch.OrganizeByTypes.Task;

var results = search.execute();

tw.local.instanceDetails=new tw.object.listOf.InstanceDetailsBO();
for ( var i = 0 ; i < results.rows.length ; i++ ) {
var row=results.rows[i];
tw.local.instanceDetails[i]=new tw.object.InstanceDetailsBO();
tw.local.instanceDetails[i].instanceId=row.values[0].toString();
tw.local.instanceDetails[i].instanceStatus=row.values[1].toString();
tw.local.instanceDetails[i].instanceName=row.values[2].toString();
tw.local.instanceDetails[i].taskId=row.values[3].toString();
tw.local.instanceDetails[i].taskStatus=row.values[4].toString();
tw.local.instanceDetails[i].closedDate=row.values[5];
tw.local.instanceDetails[i].dueDate=row.values[6];

}
```