

## Portal to SAIL Plugin

This plugin provides functions to retrieve portal report data for use in Tempo Records, Tempo Reports and processes.

### getPortalReportDataSubset

Retrieves a sorted page of data from a portal report based on the provided paging, context and filters. This function can be used in expressions in a similar manner to a Query Rule or the result of the toDataSubset() function. The data structure returned is documented below and contains many attributes for creating rich interfaces.

#### Syntax

```
getPortalReportDataSubset(reportId, [pagingInfo], [contextIds], [filters])
```

- » reportId: (Integer) The Appian portal report id.
- » pagingInfo: (PagingInfo) The paging and sorting configuration. The default portal report configuration will be used if not provided.
- » contextIds: (Integer Array) An array of Group Ids, Process Ids or Process Model Ids (depending on the report type) to use instead of the defaults.
- » filters: (PortalReportFilter Array) An array of additional filters to apply to the portal report.

#### PortalReportFilter

| Attribute | Data Type | Multiple | Description   |
|-----------|-----------|----------|---|
| field     | Text      | No       | The field name to apply the filter to.  |
| operator  | Text      | No       | GREATER_THAN, LESS_THAN, GREATER_THAN_OR_EQUAL, LESS_THAN_OR_EQUAL, EQUAL, NOT_EQUAL, IN, NOT_IN, BETWEEN, LIKE, NOT_LIKE |
| value     | Any Type  | No       | The filter value to apply.  |

**Returns****PortalReportDataSubset**

| Attribute          | Data Type              | Multiple | Description   |
|--------------------|------------------------|----------|---|
| <b>startIndex</b>  | Integer                | No       | The first record that was set to be retrieved.  |
| <b>batchSize</b>   | Integer                | No       | The maximum number of records to retrieve.  |
| <b>sort</b>        | SortInfo               | Yes      | The sorting information used to retrieve the data.                                    |
| <b>totalCount</b>  | Integer                | No       | The total number of data records available.   |
| <b>data</b>        | PortalReportColumnData | Yes      | The PortalReportColumnData records that contain the column data (one per column).     |
| <b>identifiers</b> | Integer                | Yes      | This field holds the row identifiers based on the report type (Process Id or Task Id) |
| <b>report</b>      | PortalReport           | No       | Contains attributes about the report.   |
| <b>columns</b>     | PortalReportColumn     | Yes      | This PortalReportColumn records that contain the column definitions (one per column). |

**PortalReport**

| Attribute          | Data Type | Multiple | Description             |
|--------------------|-----------|----------|-------------------------|
| <b>name</b>        | Text      | No       | The report name.        |
| <b>description</b> | Text      | No       | The report description. |

**PortalReportColumn (one per column)**

| Attribute | Data Type | Multiple | Description                |
|-----------|-----------|----------|----------------------------|
| label     | Text      | No       | The name of the column.    |
| field     | Text      | No       | The column identifier.     |
| alignment | Text      | No       | The column text alignment. |

**PortalReportColumnData (one per column)**

| Attribute | Data Type        | Multiple | Description   |
|-----------|------------------|----------|---|
| cells     | PortalReportCell | Yes      | The PortalReportCell records that contain the data associated with each cell. |

**PortalReportCell (one per row)**

| Attribute | Data Type | Multiple | Description   |
|-----------|-----------|----------|---|
| value     | Any Type  | No       | The cell value.   |
| drilldown | Any Type  | No       | The cell drilldown value (e.g task id, process id, page, URL, etc). |

**Examples****Retrieve Data Using Report Defaults**

```
=getPortalReportDataSubset(cons!MY_REPORT_DOC)
```

**Retrieve Filtered Data Subset**

```
=getPortalReportDataSubset(
  reportId: cons!MY_REPORT_DOC,
  pagingInfo: topaginginfo(1, 10),
  contextIds: {},
  filters: {
    { field: "0", operator: "NOT_LIKE", value: "* Region" },
    { field: "1", operator: "LIKE", value: "*a*" }
  }
)
```

### Access Data Values in Column 1

This example shows how to access all the value's for column 1 as an array. The same approach can be used to access the drilldown attribute too.

```
=with(  
  local!result: getPortalReportDataSubset(cons!MY_REPORT_DOC),  
  local!col1name: local!result.columns[1].label,  
  local!col1data: local!result.data[1].cells.value,  
  
  count(local!col1data) & " records were retrieved from " & local!col1name  
)
```

### Display Data as SAIL Paging Grid

This example uses rules from the [Appian Common Objects](#) and Portal Reports applications. Both must be imported for this example to work.

```
=load(  
  local!pagingInfo: type!PagingInfo(startIndex: 1, batchSize: 10),  
  with(  
    local!dataSubset: getPortalReportDataSubset(cons!MY_REPORT_DOC, local!pagingInfo),  
    rule!APN_uiPagingGrid(  
      label: local!dataSubset.report.name,  
      instructions: local!dataSubset.report.description,  
      columns: {  
        apply(  
          rule!APN_uiPortalReportGridColumn,  
          merge(  
            local!dataSubset.columns.label,  
            local!dataSubset.columns.field,  
            local!dataSubset.data,  
            local!dataSubset.columns.alignment  
          )  
        )  
      },  
      pagingInfo: local!pagingInfo,  
      totalCount: local!dataSubset.totalCount  
    )  
  )  
)
```

## executePortalReportForSail

Executes a portal report and return a graphical report for use directly in a Tempo Record or Report. The "View" configuration of the report will be used to determine the type of chart to display.

### Syntax

```
executePortalReportForSail(reportId)
```

>> reportId: (Integer) The Appian portal report id.

### Examples

```
=executePortalReportForSail(cons!MY_REPORT_DOC)
```

#### About Appian

Appian's market leading Business Process Management (BPM) Suite seamlessly integrates work automation with native mobile and social capabilities. All the data, all the processes, all the documents and all the collaborations – in one environment, on any device, through a simple social interface. The Appian BPM Suite is available on-premise and in the cloud, with complete portability. More than 3.5 million users, from Fortune 100 companies to the mid-market and small businesses worldwide, trust Appian to power their critical business processes. For more information, visit [www.appian.com](http://www.appian.com).